

1E



**Industria y Comercio**

S U P E R I N T E N D E N C I A

Delegatura de Propiedad Industrial

División de Nuevas Creaciones

SOLICITUD

# PATENTE DE INVENCION

21. EXPEDIENTE No.

04-16529

54. TÍTULO MATRICULAR / SUB-MATRICULAR UN SERVIDOR DE  
ADMINISTRACION DE DERECHOS DIGITALES (DDM) EN UNA  
ARQUITECTURA DDM

51. CLASIFICACIÓN INTERNACIONAL

<sup>6</sup> G06F 77/60

71. SOLICITANTE MICROSOFT CORPORATION

DOMICILIO REDMOND, WASHINGTON, ESTADOS UNIDOS DE  
AMERICA

74. APODERADO DARIO CORDERAS NAUAS

22. BOGOTÁ, D. C., 2004-II-25

202038

# PETITORIO

**Industria y Comercio**  
SUPERINTENDENTE

04 016529 00

SUPERINTENDENCIA Y COMERCIO  
Radicación : 04016529 000000000  
Folios: 353  
Fecha (AMD): 2004-02-25 14:39:30  
Trámite : 002 PATENTES O REGISTROS sin PRESENTAR  
Dependencia: 2000 DIVISION DE NUEVAS CREACIONES

## FORMULARIO UNICO DE SOLICITUD DE PATENTE 2000-3

### SOLICITUD DE:

☒ Patente de Invención

☐ Patente de Modelo de Utilidad

|                                                                                                                                                                                                                   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                                                                                                                                                                                                               |                                    |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------|
| <b>2</b><br>SOLICITANTE<br>(71)                                                                                                                                                                                   | Nombre: MICROSOFT CORPORATION<br><br>Dirección: One Microsoft Way Redmond, Washington 98052, Estados Unidos de America<br><br>Teléfono: 3123600 Fax: 3122410<br>E-Mail: <a href="mailto:general@cardenasvcardenas.com">general@cardenasvcardenas.com</a><br><br>Domicilio: REDMOND, WASHINGTON, ESTADOS UNIDOS DE AMERICA                                                                                                                                                                                                                                 | <b>IDENTIFICACION</b><br><br>C.C. <input type="checkbox"/> NIT <input type="checkbox"/><br>C.E. <input type="checkbox"/> Otro <input checked="" type="checkbox"/><br>Número _____                             |                                    |
| <b>3</b><br>REPRESENTANTE<br>O APODERADO<br>(74)                                                                                                                                                                  | Nombre: DARIO CARDENAS NAVAS<br><br>Dirección: Carrera 7 No 71-52 Torre B Piso 9<br><br>Teléfono: 3123600 Fax: 3122410<br>E-Mail: <a href="mailto:general@cardenasvcardenas.com">general@cardenasvcardenas.com</a>                                                                                                                                                                                                                                                                                                                                        | <b>IDENTIFICACION</b><br><br>C.C. <input checked="" type="checkbox"/> NIT <input type="checkbox"/><br>C.E. <input type="checkbox"/> Otro <input type="checkbox"/><br>Número 17.066.679 BTA<br>TP 1.250 C.S.J. |                                    |
| <b>4</b><br>INVENTOR(ES)<br>(72)                                                                                                                                                                                  | Nombre: 1. Gregory Kostal<br>2. Steve Bourne<br>3. Vinay Krishnaswamy<br><br>Dirección: 1. 425 10 th Avenue Kirkland, Washington 98033<br>2. 303 E. Pike Street # 602 Seattle, Washington 98122<br>3. 23319 NE 142 <sup>nd</sup> Place Woodinville, Washington 98072<br><br>Teléfono: 3123600 Fax: 3123600<br>E-Mail: <a href="mailto:general@cardenasvcardenas.com">general@cardenasvcardenas.com</a><br><br>Domicilio: 1. WASHINGTON, ESTADOS UNIDOS DE AMERICA<br>2. WASHINGTON, ESTADOS UNIDOS DE AMERICA<br>3. WASHINGTON, ESTADOS UNIDOS DE AMERICA | <b>IDENTIFICACION</b><br><br>C.C. <input type="checkbox"/> NIT <input type="checkbox"/><br>C.E. <input type="checkbox"/> Otro <input checked="" type="checkbox"/><br>Número _____                             |                                    |
| <b>5</b> Título(54):<br>MATRICULAR / SUB-MATRICULAR UN SERVIDOR DE ADMINISTRACION DE DERECHOS DIGITALES (DRM) EN UNA ARQUITECTURA DRM                                                                             |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                                                                                                                                                                                                               |                                    |
| <b>6</b> Prioridad<br><br>SI <input checked="" type="checkbox"/> NO <input type="checkbox"/>                                                                                                                      | (33) País de Origen<br>ESTADOS UNIDOS DE AMERICA                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | (32) Fecha<br>Febrero 25, 2003                                                                                                                                                                                | (31) N° de Solicitud<br>10/373,458 |
| <b>7</b> Para publicar a partir de la fecha de la presente solicitud a los:<br>6 meses <input type="checkbox"/> 12 meses <input type="checkbox"/> 18 meses <input type="checkbox"/> Otro <input type="checkbox"/> |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                                                                                                                                                                                                               |                                    |
| <b>8</b> Comprobante de Pago No.: 04-10.508 / 04-10.510 Fecha: Febrero 10, 2004                                                                                                                                   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                                                                                                                                                                                                               |                                    |

9

## ANEXOS

- ☒ Comprobante de pago de la tasa de presentación de la solicitud.
- ☐ Comprobante de pago de la tasa por concepto de excedente de palabras en la publicación.
- ☒ Comprobante de pago de la tasa por concepto de reivindicación de prioridad. (No 04-10.510)
- ☒ Poderes, si fuere el caso. (Protocolo No 16.069)
- ☒ Certificado de existencia y representación legal cuando el solicitante sea persona jurídica. Protocolo No 16.069)
- ☐ Copia certificada de la primera solicitud, si se reivindica prioridad.
- ☒ Traducción simple de la primera solicitud, si se reivindica prioridad.
- ☒ Documento de cesión del inventor al solicitante o a su causante.
- ☒ Descripción de la invención.
- ☒ Una o más reivindicaciones.
- ☒ Dibujos y/o planos necesarios.
- ☐ De ser el caso, copia del contrato de acceso.
- ☐ De ser el caso, documento que acredite la licencia o autorización de uso de conocimientos tradicionales de la comunidad de las comunidades indígenas.
- ☐ De ser el caso, certificado de depósito del material biológico.
- ☒ Arte final 12x12 y 6x6 cm.
- ☒ Resumen y extracto para publicación.

10

NOMBRE: DARIO CARDENAS NAVAS

FIRMA:

C.C. 17.066.629 BTA

TP. 1.250  
C.SJ.

**TÍTULO DE LA INVENCION**

Matricular / Sub-matricular un Servidor de Administración de Derechos Digitales (DRM) en una arquitectura DRM

**REFERENCIA CRUZADA A SOLICITUD RELACIONADA**

Las siguientes solicitudes de patente Estadounidenses describen la materia objeto que está relaciona con la materia objeto de la presente solicitud, y se incorporan aquí como referencia en su totalidad:

La solicitud de patente Estadounidense No. 10/185,527, presentada en junio 28, 2002 bajo el número de documento de abogado MSFT-1330 y titulada "Obtención de un Marcador de Derechos Firmados (SRL) para Contenido Digital y Obtener una Correspondiente Licencia Digital al Contenido con Base en el SRL en un Sistema de Administración de Derechos Digitales";

La solicitud de patente Estadounidense No. 10/185,278, presentada en junio 28, 2002 bajo el número de documento de abogado MSFT-1333 y titulada "Usar una Plantilla de Derechos para Obtener una Tarjeta de Derechos Firmados

2

(SRL) para el Contenido Digital en un Sistema de Administración de Derechos Digitales";

La solicitud de patente Estadounidense No. 10/185,511, presentada en junio 28, 2002 bajo el número de documento de abogado MSFT-1343 y titulada "Sistemas y Métodos para Emitir Licencias de Uso para el Contenido Digital y Servicios";

La solicitud de patente Estadounidense No. \_\_\_\_\_, presentada \_\_\_\_\_ bajo el número de documento de abogado MSFT-1498 y titulada "Publicar el Contenido Digital dentro de una Organización de acuerdo con un Sistema de Administración de Derechos Digitales (DRM);

La solicitud de patente Estadounidense No. \_\_\_\_\_, presentada \_\_\_\_\_ bajo el número de documento de abogado MSFT-1569 y titulada "Publicar el Contenido Digital dentro de una Organización de acuerdo con un Sistema de Administración de Derechos Digitales (DRM);

La solicitud de patente Estadounidense No. \_\_\_\_\_, presentada \_\_\_\_\_ concurrentemente con la presente solicitud bajo el número de documento de abogado

3  
5

MSFT-1537 y titulada "Emitir una Licencia de Uso de Publicador fuera de Línea en un Sistema de Administración de Derechos Digitales (DMR)".

#### CAMPO TÉCNICO

Esta invención se relaciona con un sistema de manejo de derechos digitales (DRM). Más particularmente, la invención se relaciona con el empleo de un sistema de DRM para publicar contenido digital en una organización tal como una oficina o una corporación o similar de modo que el suministro y el uso del contenido dentro de la organización puede estar restringido de acuerdo con el uso correspondiente y los términos de la licencia. Aun más particularmente, la presente invención se relaciona con una red de servidores de DRM que efectúan tal sistema DRM y un método de matricular o submatricular un servidor DRM dentro de la red.

#### ANTECEDENTES DE LA INVENCION

La administración y ejercicio de los derechos digitales es altamente deseable en conexión con el contenido digital tal como audio digital, video digital,

1

texto digital datos digitales, multimedia digital, etc., en donde tal contenido digital debe ser distribuido a uno o más usuarios. El contenido digital pudiera ser estático tal como un documento de texto, por ejemplo, o pudiera ser dinámico o en corriente, tal como un audio/video que corre de un evento en vivo. Los modos de distribución típicos incluyen dispositivos tangibles tal como un disco magnético (disquete) una cinta magnética, un disco óptico (compacto) (CD), etc., y medios intangibles tal como una tarjeta de boletín electrónico, una red electrónica, el Internet, etc. Luego de haber sido recibido por el usuario, tal usuario usa "juega" con el contenido digital con la ayuda de un dispositivo de uso apropiado tal como un dispositivo para ejecutar medios en un computador personal o similar.

En un escenario, un propietario del contenido o propietario de los derechos tal como un autor, un publicador, un difundidor, etc., desea distribuir tal contenido digital a cada uno de sus muchos usuarios o receptores en intercambio por una tarifa bajo una licencia o alguna otra consideración. En tal escenario, entonces, el contenido puede ser una canción, un álbum de canciones, una película, etc., y el propósito de la

5

distribución es general las tarifas bajo licencia. Tal propietario del contenido, dada la escogencia, le gustaría restringir lo que el usuario pueda hacer con tal contenido digital distribuido. Por ejemplo, el propietario del contenido le gustaría restringir al usuario de hacer copias y redistribuir tal contenido a un segundo usuario, por lo menos en una manera que se le niegue al propietario del contenido una tarifa bajo licencia por parte del segundo usuario.

C

Adicionalmente, el propietario del contenido puede desear suministrar al usuario con alguna flexibilidad para comprar diferente tipos de licencias de uso y diferentes tarifas bajo licencia, mientras que al mismo tiempo se mantenga al usuario dentro de los términos de cualquiera que sea el tipo de licencia es de hecho comprado. Por ejemplo, el propietario del contenido puede desear permitir el que el contenido digital distribuido sea ejecutado solamente un numero de veces limitado, solamente durante un cierto tiempo total, solamente en un cierto tipo de maquina, solamente en un cierto tipo de aparato para ejecutar medios, solamente por un cierto tipo de usuario, etc.

9

9

En otro escenario, un desarrollador de contenido, tal como un empleado en o miembro de una organización, desea distribuir tal contenido digital a uno o mas de los otros empleados o miembros en la organización o a otros individuos por fuera de la organización, pero le gustaría mantener a esos otros para que no divulguen el contenido. Aquí, la distribución del contenido es mas a un contenido con base en una organización compartido de una manera confidencial o restringida, en oposición a una distribución con base en difusión en intercambio por medio de una tarifa bajo licencia o alguna otra consideración.

En tal escenario, entonces, el contenido puede ser una presentación de un documento, una hoja de calculo, una base de datos, un correo electrónico, o similar, de modo que puede ser intercambiado dentro de un establecimiento de oficina, y el desarrollador del contenido puede desear asegurarse de que el contenido permanezca dentro de la organización o establecimiento de oficina y no sea distribuido y no sea usado por individuos no autorizados, tal como por ejemplo competidores o adversarios. Nuevamente, tal desarrollador de contenido desea restringir lo que un receptor pueda hacer con tal

contenido digital distribuido. Por ejemplo, el propietario del contenido le gustaría restringir al usuario de copiar y redistribuir tal contenido a un segundo usuario, por lo menos en una manera que exponga el contenido por fuera de los límites de los individuos que se les debería permitir el contenido.

Adicionalmente, el desarrollador del contenido puede desear suministrar a varios receptores diferentes niveles de derecho de uso. Por ejemplo, el desarrollador del contenido puede desear permitir que el contenido digital protegido sea visible pero no imprimible con respecto a una clase de individuo, y visible e imprimible con respecto a otra clase de individuo.

Sin embargo, en cualquiera que sea el escenario, luego que la distribución ha ocurrido, tal propietario de contenido/desarrollador tiene muy poco si es que tiene algún control sobre el contenido digital. Esto es especialmente problemático en vista del hecho de que prácticamente cada computador personal incluye el software y el hardware necesario para hacer una copia digital exacta de tal contenido digital, y para bajar tal copia digital exacta a un disco magnético describible u

óptico describible, o enviar tal copia digital exacta por una red tal como el Internet a cualquier destino.

Por su puesto, como parte de una transacción en donde el contenido es distribuido, el propietario del contenido/desarrollador, puede requerir que el usuario/receptor del contenido digital prometa no redistribuir tal contenido digital de una manera no deseada. Sin embargo, tal promesa es hecha fácilmente y fácilmente rota. Un propietario de contenido/desarrollador puede intentar impedir tal redistribución a través de cualquiera de los varios dispositivos de seguridad conocidos, que usualmente involucran la encriptación y desencriptación. Sin embargo, existe muy poco para impedir que un usuario en un medio determinado desencripte el contenido digital encriptado, guardando tal contenido digital en una forma no encriptada, y luego redistribuyendo el mismo.

Existe una necesidad, entonces, de suministrar una arquitectura para exigir y administrar derechos digitales (DRM) y un método que permita el uso o reproducción controlada de formas arbitrarias de contenido digital, en donde tal control sea flexible y definible por el

propietario del contenido/desarrollador de tal contenido digital. Mas específicamente, existe una necesidad para dar una arquitectura que permita y facilite tal uso controlado, especialmente en un ambiente de oficina y organización o similar en donde los documentos deben ser compartidos entre un grupo definido de individuos o clases de individuos. A un mas específicamente, existe una necesidad de un método de matricular servidores que garantice la aprobación dentro de la arquitectura.

#### RESUMEN DE LA INVENCION

Las necesidades anteriormente mencionadas son satisfechas por lo menos en parte por la presente invención en la cual un sistema de administración de derechos digitales (DRM) tiene una pluralidad de servidores de DRM que llevan a cabo una funcionalidad de DRM y un servidor de ingreso de DRM-E es matriculado dentro del sistema por un servidor matriculador de DRM-R tal que el servidor DRM-E que ingresa debe ser abalado dentro del sistema. En la invención, el servidor DRM-E procura un par de claves publica/privada (PU-E, PR-E) para identificar tal servidor de DRM-E dentro del sistema DRM, procura una identificación de propuesta del mismo, y envía una

solicitud de matricula al servidor de DRM-R incluyendo la identificación de propuesta y (PU-E).

El servidor DRM-R valida la identificación de propuesta, y, si la solicitud es aceptada, genera un certificado de matricula digital para el servidor DRM-E para matricular tal servidor de DRM-E en el sistema DRM. El certificado de matricula generado esta basado en por lo menos en parte en (PU-E). El servidor de DRM-R regresa el certificado de matricula generado al servidor de DRM-E solicitante y el servidor DRM-E ahora matriculado almacena el certificado de matricula que le llega en una localización apropiada para un uso futuro. El servidor de DRM-E con el certificado de matricula es capaz de emplearlo para emitir documentos de DRM dentro del sistema DRM.

#### **BREVE DESCRIPCIÓN DE LOS DIBUJOS**

El resumen anterior, lo mismo que en la descripción detallada siguiente de las modalidades de la presente invención, serán mejor entendidas cuando se lean en conjunto con los dibujos adjuntos. Para el propósito de ilustrar la invención, se han mostrado en los dibujos las

modalidades que son actualmente preferidas. Como debe entenderse, sin embargo, la invención no esta limitada a las disposiciones e instrumentaciones precisas mostradas. En los dibujos:

La figura 1 es un diagrama de bloques que representa un ambiente de cómputo ilustrativo no limitativo en el cual la presente invención puede ser implementada;

La figura 2 es un diagrama de bloques que representa un ambiente de red ilustrativo que tiene una variedad de dispositivos de cómputo en los cuales la presente invención puede ser implementada;

La figura 3 es un diagrama de bloques funcionales de una modalidad preferida de un sistema y método de acuerdo con la invención para publicar contenido digital;

La figura 4 es una tabla de flujo de una modalidad preferida de un método de acuerdo con la invención para publicar contenido de derechos digitales administrados;

La figura 4A es un diagrama de bloque que muestra la estructura de una tarjeta de derechos firmada como se ha producido por el método de la figura 4;

La figura 5 es un diagrama de bloque de una modalidad preferida de un sistema y método de acuerdo con la invención para licenciar el contenido de derechos digitales administrados;

Las figuras 6A y 6B son diagramas de flujo de una modalidad preferida de un método de acuerdo con la invención para licenciar contenidos de derechos digitales administrados;

La figura 7 es un diagrama de bloques que muestra un certificado emitido por un servidor DRM a un usuario para permitirle al usuario llevar a cabo publicación fuera de línea de acuerdo con una modalidad de la presente invención;

La figura 8 es un diagrama de bloques que muestra el certificado de la figura 7 junto con una licencia de publicador que le permite a un usuario publicador usar el

contenido fuera de línea publicado así de acuerdo con una modalidad de la presente invención;

La figura 9 es una carta de flujo que muestra las etapas clave llevadas a cabo por el usuario publicador para obtener la licencia de publicación de la figura 8 de acuerdo con una modalidad de la presente invención;

La figura 10 es una carta de flujo que muestra las etapas clave llevadas a cabo por el usuario publicador llevadas a cabo para emplear la licencia de publicación obtenida de la figura 9 para usar el contenido correspondiente de acuerdo con una modalidad de la presente invención;

La figura 11 es un diagrama de bloques que muestra una arquitectura para hacer cumplir de un ejemplo de un sistema con base en la confianza;

La figura 12 es un diagrama de bloques que muestra una pluralidad de servidores de DRM tal como puede existir en la arquitectura de la presente invención, en donde cada uno de los servidores DRM (que ingresan) es matriculado o submatriculado dentro de la arquitectura por otro

14

servidor de DRM (matriculador) que emite a ellos un certificado de matricula;

La figura 13 es un diagrama de bloques que muestra el certificado de matricula de la figura 12 junto con un certificado de garantía presentado en por lo menos algunas veces por un servidor de DRM que ingresa a un servidor de DRM que matricula;

Las figuras 14 y 15 son diagramas de flujo que muestran las etapas claves llevadas a cabo por los servidores DRM que matriculan e ingresan de la figura 13 y 14 para matricular (figura 14) o submatricular (figura 15) el servidor DRM que ingresa.

## DESCRIPCIÓN DETALLADA DE LA INVENCION

### AMBIENTE DE COMPUTADOR

La figura 1 y al siguiente discusión tienen el propósito de suministrar una descripción general breve de un ambiente de computo adecuado en el cual la invención puede ser implementada. Debe entenderse sin embargo, que dispositivos de mano, portátiles, y otros dispositivos de

computo de cualquier clase están contemplados para ser usados en conexión con la presente invención. Aunque un computador de propósito general se describe enseguida, este es solo un ejemplo, y la presente invención requiere solamente un solo cliente que tenga interoperabilidad e interacción de servidor de red. Así la presente invención puede ser implementada en un ambiente de servicios de red auspiciados en el cual muy poco de los recursos de clientes están implicados, por ejemplo, un ambiente en red en el cual el dispositivo del cliente sirve meramente como un buscador o interface de la red amplia mundial (World Wide Web).

Aunque no es requerido, la invención puede ser implementada por medio de una interface de programación de aplicación (API), para uso por un desarrollador, y/o incluida dentro del software de búsqueda de red que será descrito en el contexto general de instrucciones ejecutables por computador, tal como módulos de programa, que son ejecutados por uno o mas computadores tal como estaciones de trabajo de cliente, servidores, u otros dispositivos. Generalmente, los módulos de programa incluyen rutinas, programas, objetos, componentes, estructuras de datos y similares que llevan a cabo tareas

particulares o implementan tipos de datos abstractos particulares. Típicamente la funcionalidad de los módulos de programa pueden ser combinados o distribuidos como se desea en varias modalidades. Más aun, aquellos versados en la técnica podrán visualizar que la invención puede ser practicada con otras configuraciones de sistema de computador. Otros sistemas de computador bien conocidos, ambientes, y/o configuraciones de computador que puedan ser adecuado para uso con la invención incluyen pero no se limitan a, computadores personales (PCs), maquinas descriptoras automatizadas, computadores servidores, dispositivos portátiles o manuales, sistemas multiprocesadores, sistemas con base en microprocesador, dispositivos electrónicos de consumidor programables, PCs de red, minicomputadoras, computadoras de tamaño medio, y similares. La invención también puede ser practicada en ambientes de cómputo distribuidos en donde las tarea son llevadas al cabo por dispositivos de procesamiento remoto que están enlazados por medio de red de comunicaciones u otros medios de transmisión de datos. En un ambiente de cómputo distribuido, los módulos de programas pueden estar localizados tanto en medios de almacenamiento de computadoras locales y remotas incluyendo dispositivos de almacenamiento de memoria.

La figura 1 ilustra así un ejemplo de un ambiente sistema de computo adecuado 100 en el cual la invención puede ser implementada, aunque para hacerlo mas claro, el ambiente de sistema de computo 100 es solamente un ejemplo de un ambiente de computo adecuado y no tiene el propósito de sugerir ninguna limitación en cuanto al alcance del uso o funcionalidad de la invención. Tampoco debería el ambiente de computo 100 ser interpretado como que tiene una dependencia o requerimiento relacionada con una cualquiera o combinaciones de componentes ilustrados en el ambiente operativo e ilustrativo 100.

Con referencia a la figura 1, un sistema ilustrativo para implementar la invención incluye un dispositivo de cómputo de propósito general en la forma de un computador 110. Los componentes del computador 110 pueden incluir pero no se limitan a, una unidad de procesamiento 120, una memoria del sistema 130, y un bus del sistema 121 que acopla los diversos componentes del sistema incluyendo la memoria del sistema a la unidad de procesamiento 120. El bus del sistema 121 puede ser cualquiera de los tipos diversos de estructura de bus incluyendo un bus de memoria o controlador de memoria, un bus de periféricos y

un bus local que usa cualquiera de una variedad de arquitectura de bus. Como vía de ejemplo, y no como limitación, tales arquitecturas incluyen el bus en arquitectura en estándar de industria (ISA), el bus de arquitectura de microcanal (MCA), el bus de ISA mejorado (EISA), el bus local de la Asociación Estándar de Electrónicos de video (VESA), y el bus de interconexión de componentes periféricos (PCI) (también conocido como bus Mezzanine)

El computador 110 típicamente incluye una variedad de medios legibles por computador. Los medios legibles por computador pueden ser cualquiera de los medios disponibles que puedan ser accedidos por el computador 110 e incluye tanto medios volátiles como no volátiles, medios removibles y no removibles. Como vía de ejemplo, y no como limitación, los medios legibles por computador pueden comprender medios de almacenamiento en computador, y medios de comunicación. Los medios de almacenamiento en computadora incluyen tanto medio volátiles y no volátiles, removible y no removibles implementados en cualquier método o tecnología para el almacenamiento de información tal como las instrucciones legibles en computador, estructuras de datos, módulos de programa y

otros datos. Los medios de almacenamiento por computadora incluyen, pero no se limitan a, RAM, ROM, EEPROM, memoria flash y otras tecnologías de memoria, CDROM, discos digitales versátiles (DVD) u otros almacenamientos en disco óptico, casetes magnéticos, cintas magnéticas, almacenamiento en disco magnético u otros dispositivos de almacenamiento magnético, o cualquier otro medio que pueda ser usado para almacenar la información deseada y que pueda ser accedido por la computadora 110. los medios de comunicación típicamente involucran instrucciones legibles por computador, estructuras de datos, modelos de programas u otros datos en una señal de datos modulada tal como una onda portadora u otro mecanismo de transporte he incluye cualquier medio de suministro de información. El termino "señal de datos modulada" significa una señal que tiene una o mas de sus de características establecidas o cambiadas de tal manera como para codificar la información en la señal. Como vía de ejemplo, y no como limitación, los medios de comunicación incluyen medios alambrados tales como una red alambrada o una conexión directamente alambrada, y medios inalámbricos tales como acústicos, RF, infrarrojos y otros medios inalámbricos. Las combinaciones de

cualquiera de los anteriores deberían estar incluidas dentro del alcance de los medios legibles por computador.

La memoria del sistema 130 incluye medios de almacenamiento en computadora en la forma de memorias volátiles y/o no volátiles tales como memoria de solo lectura (ROM) 131 y memoria de acceso aleatorio (RAM) 132. Un sistema de entrada/salida base con 133(BIOS), que contiene las rutinas básicas que ayudan a transferir la información entre los elementos dentro de la computadora 110, tal como durante el inicio o el arranque, es típicamente almacenado en la ROM 131. La RAM 132 típicamente contiene datos y/o módulos de programa que son accesibles inmediatamente a y/o actualmente son operados en o por la unidad de procesamiento 120. como vía de ejemplo, y no como limitación, la figura 1 ilustra el sistema operativo 134, los programas de aplicaciones 135, otros módulos de programa 136, y datos del programa 137.

La computadora 110 puede también incluir otros medios de almacenamiento en computadora removibles/no removibles, volátiles/no volátiles. Como vía de ejemplo solamente, la figura 1 ilustra un reproductor de disco duro 141 que lee

de o escribe en medios magnéticos no removibles, no volátiles, un reproductor de disco magnético 151 que lee de o escribe en un disco magnético removible, no volátil 152, y un reproductor de disco óptico 155 que lee de o escribe en discos ópticos removibles no volátiles 156, tal como un CDROM u otros medios ópticos. Otros medios de almacenamiento en computadora removibles/no removibles, volátiles/no volátiles que pueden ser usados en el ambiente operativo ilustrativo incluyen, pero no se limitan a, casetes de cintas magnéticas, tarjeta de memoria flash, discos versátiles digitales, cintas digitales de video, RAM de estado sólido, ROM de estado sólido, y similares. El reproductor de disco duro 141 está típicamente conectado al bus del sistema 121 por medio de una interface de memoria no removible tal como la interface 140, y el reproductor de disco magnético 151 y el reproductor de disco óptico 155 están típicamente conectados al bus del sistema 121 por medio de una interface de memoria removible, tal como la interface 150.

Los reproductores y sus medios de almacenamiento por computadora asociados discutidos anteriormente y ilustrados en la figura 1 suministran el almacenamiento

de las instrucciones, estructura de dato, módulos de programas, y otros datos leíbles por computadora para la computadora 110. en la figura 1, por ejemplo, el reproductor de disco dura 141 esta ilustrado como el que almacena el sistema operativo 144, los programas de aplicaciones 145, otros módulos de programa 146 y datos de programa 147. Ninguno de estos componentes puede ser el mismo o diferente de el sistema operativo 134, los programas de aplicación 135, otros módulos de programa 136 y datos de programa 137. El sistema operativo 144, los programas de aplicación 145, otros módulos de programa 146, y los datos de programa 147 se les asignan diferentes numeras aquí para ilustrar que, como mínimo, ellos son copias diferentes. Un usuario puede ingresar comandos e información dentro de la computadora 110 por medio de los dispositivos de entrada tal como un teclado 162 o un dispositivo de apuntamiento 161, comúnmente referido como un ratón, una bola de rastreo, o una almohadilla de toque. Otros dispositivos de entrada (no mostrados) pueden incluir un micrófono, un joystick, una almohadilla de juego, un plato de satélite, un escáner, o similares. Estos y otros dispositivos de entrada están frecuentemente conectados a la unidad de procesamiento 120 por medio de una interface de entrada de usuario 160

que esta acoplada al bus del sistema 121, pero que puede estar conectada por medio de otra interface y estructura de bus, tal como un puerto paralelo, un puerto de juegos, o un bus serial universal (USB).

Un monitor 191 u otro tipo de dispositivo de visualización esta también conectado al bus del sistema 121 por medio de una interface, tal como una interface de videos 190. Una interface de grafico 182, tal como Northbridge, puede también estar conectada al bus del sistema 121. El Northbridge es un conjunto de circuito integrado que comunica con la CPU, o la unidad de procesamiento anfitrión 120, y asume la responsabilidad para el puerto de grafico acelerado (AGP) de las comunicaciones. Uno o mas unidades de procesamiento grafico (GPU) 184 puede comunicarse con la interface de gráficos 182. a este respecto, las GPU 184 generalmente incluyen un almacenamiento de memoria en circuito integrado, tal como un almacenamiento de registro y las GPU 184 se comunican con una memoria de video 186. las GPU 184, sin embargo, son ecepto por un ejemplo de un coprocesador y por lo tanto una variedad de dispositivos del coprocesamiento pueden estar incluidos en la computadora 110. un monitor 191 u otro tipo de

dispositivo de visualización también está conectado al bus del sistema 121 por medio de una interface, tal como una interface de video 190, a cual puede a su vez comunicarse con una memoria de video 186. Adicionalmente al monitor 191, las computadoras pueden incluir otros dispositivos de salida periféricos tales como parlantes 197 e impresora 196, y las cuales pueden estar conectadas por medio de una interface periférica de salida 195.

La computadora 110 puede operar en un ambiente en red usando conexiones lógicas a uno o más computadores remotos tal como una computadora remota 180. La computadora remota 180 puede ser un computador personal, un servidor, un enrutador, un PC de red, un dispositivo paralelo, u otro nodo de red común, y típicamente incluye muchos o todos los elementos descritos anteriormente en relación a la computadora 110, aunque solamente un dispositivo de almacenamiento de memoria 181 ha sido ilustrado en la figura 1. Las conexiones lógicas ilustradas en la figura 1 incluyen una red de área local (LAN) 171 y una red de área amplia (WAN) 173, pero también pueden incluir otras redes. Tales ambientes de red son lugares comunes en las oficinas, en las redes de computadora de empresas grandes, Intranet y el Internet.

Cuando es usado en un ambiente de red LAN, la computadora 110 esta conectada a la LAN 171 por medio de una interface o adaptador de red 170. Cuando es usado en un ambiente de red WAN, la computadora 110 típicamente incluye un módem 172 u otros medios para establecer comunicaciones sobre la WAN 173, tal como el Internet. El módem 172, el cual puede ser interno o externo, puede estar conectado al bus del sistema 121 por vía de la interface de entrada de usuario 160, u otro mecanismo apropiado. En un ambiente en red, los módulos de programas ilustrados con relación a la computador 110, o porciones de ellos, pueden estar almacenados en dispositivos de almacenamiento de memoria remotos. Como vía de ejemplo y no como limitación, la figura 1 ilustra programas de aplicación remotos 185 los cuales residen en un dispositivo de memoria 181. Se podrá apreciar que la conexione de red mostradas son ilustrativas y que otros medios de establecer enlaces de comunicaciones entre computadoras pueden ser usados.

Una persona versada en la técnica puede apreciar o visualizar que una computadora 110 y otro dispositivo de cliente pueden ser desplegados como parte de una red de

computadores. A este respecto, la presente invención pertenece a un sistema de computadores que tiene cualquier número de unidades de almacenamiento o de memoria y cualquier número de aplicaciones y procesos que ocurren por cualquier número de unidades de almacenamiento o volúmenes. La presente invención puede aplicarse a un ambiente con computadores servidores y computadores de cliente que se despliegan en un ambiente de red, que tienen almacenamiento remoto o local. La presente invención puede también aplicarse a un dispositivo de cómputo único, que tiene una funcionalidad de programación de lenguaje, interpretación y capacidades de ejecución.

Las facilidades de cómputo distribuidas que comparten los recursos de computador y los servicios por intercambio directo entre los dispositivos y sistema de computador. Estos recursos y servicios incluyen el intercambio de información, el almacenamiento cache y el almacenamiento en disco para los archivos. La computación distribuida toma ventaja de la conectividad de red, permitiéndole a los clientes nivelar sus potencias colectivas para beneficio de la empresa completa. A este respecto, una variedad de dispositivos pueden tener aplicaciones,

objetos o recursos que pueden interactuar para implicar técnicas de autenticación de la presente invención para líneas de grafico confiables.

La figura 2 suministra un diagrama esquemático de un ambiente de computo distribuido o en red ilustrativo. El ambiente de computo distribuido comprende objeto de computo 10a, 10b, etc. y objetos de computo o dispositivos 110a, 110b, 110c, etc. estos objetos pueden comprender programas, métodos, almacenamientos de datos, lógica programable, etc. esto objetos pueden comprender porciones de los mismo o diferentes dispositivos tales como PDA, televisiones, reproductores de MP3, televisiones, computadoras personales, etc. cada objeto puede comunicarse con otro objeto por vía de la red de comunicaciones 14. Esta red puede ella misma comprender otros objetos de cómputo y dispositivos de cómputo que suministren servicios al sistema de la figura 2. De acuerdo con un aspecto de la invención, cada objeto 10 110 puede contener una aplicación que pueda solicitar las técnicas de autenticación de la presente invención para líneas de gráficos confiable.

También puede visualizarse que un objeto, tal como 110c, puede ser hospedado en otro dispositivo de cómputo 10 o 110. así, aunque el ambiente físico ilustrado puede mostrar los dispositivos conectados como computadoras, tal ilustración es meramente ilustrativa y el ambiente físico puede alternativamente ser ilustrado o descrito como que comprende diversos dispositivos digitales tales como PDAs, televisiones, reproductores de MP3, etc., objetos de software tales como interfaces, objetos COM y similares.

Hay una variedad de sistemas, componentes, y configuraciones de red que soportan ambientes de cómputo distribuidos. Por ejemplo, los sistemas de cómputo pueden estar conectados juntos por sistemas alambrados o inalámbricos, por redes locales o redes de distribución amplia. Actualmente, muchas de las redes están acopladas al Internet, lo que suministra la infraestructura para cómputo distribuido amplio e involucra muchas redes diferentes.

En ambientes de red domésticos, hay por lo menos cuatro medios de transporte de red dispares que pueden cada uno soportar un protocolo único tal como una línea de poder,

datos (a tanto inalámbricos como alambrados), voz (por ejemplo., teléfono) y medios de entretenimiento. La mayoría de los dispositivos de control domésticos tales como interruptores de luz y electrodomésticos pueden usar la línea de potencia para la conectividad. Los servicios de datos pueden ingresar al teléfono como en banda ancha (por ejemplo, se DSL o un módem de cable) y ser accesibles dentro del hogar usando conectividad inalámbrica (por ejemplo, HomeRF o 802.11b) o alambrada (por ejemplo, Home PNA, Cat 5, y aun la línea de potencia). El tráfico de voz puede ingresar al hogar se por un sistema alambrado (por ejemplo, Cat 3) o inalámbrico (por ejemplo, teléfonos celulares) y puede estar distribuido dentro del hogar usando un alambrado Cat 3. Los medios de entretenimiento pueden entrar al hogar sea por medio de satélite o cable y típicamente ser distribuido en el hogar usando cable coaxial. El IEEE 1394 y el DVI son también interconexiones emergentes como digitales para sectores de los dispositivos de medios. Todos estos ambientes de red y otros que puedan emerger como estándares de protocolo pueden ser interconectado para formar una Intranet que pueda ser conectada al mundo exterior por vía del Internet. En resumen, una variedad de fuentes dispares existe para el almacenamiento y

transmisión de datos, y consecuentemente, para moverse adelante dispositivos de cómputo requerirán maneras de proteger los contenidos de todas las porciones de las líneas de procesamiento de datos.

La "Internet" comúnmente se refiere a la colección de redes y puertos de salidas que utilizan los protocolos TCP/IP, que son bien conocidos en la técnica de redes por computadora. El TCP/IP es un acrónimo para "Transport Control Protocol/interface Program". La Internet puede ser descrita como un sistema de redes de computadores remotos distribuidos geográficamente interconectados por computadoras que ejecutan protocolos de red que le permite a los usuarios interactuar y compartir información sobre la red. Debido a tales redes remotas que comparten información de una manera tan amplia tal como el Internet por lo tanto se evoluciona generalmente a un sistema abierto para el cual los desarrolladores pueden diseñar aplicaciones de software para llevar a cabo operaciones y servicios especializados, esencialmente sin restricción.

Así, la infraestructura de red le permite aun anfitrión de topologías de red tal como una arquitectura cliente-

servidor, paralela o híbrida. El "clientes" es un miembro de una clase o grupo que usa los servicios de otra clase o grupo con el cual no está relacionado. Así, en cómputo un cliente es un proceso, por ejemplo, un conjunto de instrucciones o tareas, que requiere un servicio suministrado por otro programa. El proceso del cliente utiliza el servicio solicitado sin tener que "conocer" ninguno de los detalles de trabajo acerca del otro programa o del servicio mismo. En una arquitectura cliente-servidor, particularmente un sistema en red, una computadora que acceda a recursos de red compartidos suministrados por otra computadora por ejemplo, un servidor. En el ejemplo de la figura 2, los computadores 110a, 110b, etc., pueden pensarse como clientes y las computadora 10a, 10b, etc., pueden pensarse como servidores en donde los servidores 10a, 10b etc., mantiene los datos que son luego replicados en los computadores clientes 110a, 110b, etc.

Un servidor es típicamente un sistema de cómputo remoto accesible por una red remota tal como el Internet. El proceso del cliente puede ser activo en un primer sistema de cómputo, y el proceso del servidor puede ser activo en un segundo sistema de cómputo, y comunicarse

uno con otro por un medio de comunicaciones, suministrando así una funcionalidad distribuida y permitiendo que múltiples cliente tomen ventajas de las capacidades de conseguir información del servidor.

El cliente y el servidor se comunican uno al otro utilizando la funcionalidad suministrada por una capa de protocolo. Por ejemplo el protocolo de transferencia de hipertexto (HTTP) es un protocolo común que es usado en conjunto con la red amplia mundial (WWW). Típicamente, una dirección de red de computador tal como un localizador universal de recursos (URL) o un protocolo de Internet (IP) es usado para identificar el servidor o los computadores cliente uno con el otro. La dirección de red puede referirse como la dirección de localización de recursos universal. Por ejemplo, la comunicación puede ser suministrada sobre un medio de comunicaciones. En particular, el cliente y el servidor pueden estar acoplados uno al otro por medio de conexiones TCP/IP para comunicaciones de alta capacidad.

Así, la figura 2 ilustra un ambiente distribuido en red ejemplar, con un servidor en comunicación con computadoras cliente por medio de una red/bus, en el cual

la presente invención puede ser empleada. En mas detalle, un numero de servidores 10a, 10b, etc., están interconectados por una red/bus de comunicaciones 14, la cual puede ser una LAN, WAN, Intranet, Internet, etc., con un numero de clientes o dispositivos de cómputos remotos 110a, 110b, 110c, 110d, 110e, etc., tal como un computador portátil, un computador de mano, un cliente fino, un electrodoméstico en red, u. otro dispositivo tal como un VCT, TV, horno, luz, calentador y similares de acuerdo con la presente invención. Se tiene en cuenta que la presente invención puede aplicar a cualquiera de los dispositivos de computo en conexión con los cuales es deseable el proceso, almacenamiento o volver seguro el contenido de una fuente confiable.

En un ambiente de red en el cual las comunicaciones red/bus 14 es el Internet, por ejemplo los servidores 10 pueden ser servidores Web con los cuales los cliente 110a, 110b, 110c, 110d, 110e, etc. se comunican por vía de cualquiera de un numero de protocolos conocidos tales como HTTP. Los servidores 10 también pueden ser tanto servidores como clientes 110 ya que pueden tener características de un ambiente de cómputo distribuido. Las comunicaciones pueden ser alambradas o inalámbricas,

en donde se ha apropiado. Los dispositivos de clientes 110 pueden o no comunicarse por vía de comunicaciones de red/bus 14 y pueden tener comunicaciones independientes asociadas con ellas. Por ejemplo, en el caso de un TV o VCR, puede haber un aspecto en red para el control de ellos. Cada computadora de cliente 110 y computadora de servidor 10 puede estar equipada con varios módulos de programas de aplicación u objetos 135 y con conexiones o accesos a varios tipos de elementos de almacenamiento, a través de los cuales los archivos pueden ser almacenados o en los cuales

Porciones de archivos que pueden ser bajados o migrados. Así, la presente invención puede ser utilizada en un ambiente de red de computadoras que tienen computadoras clientes 110a, 110b, etc. que pueden acceder o interactuar con una red/bus de computadora 14 y computadoras servidor 10a, 10b, etc. que pueden interactuar con las computadoras clientes 110a, 110b, etc. y otros dispositivos 111 y bases de datos 20.

## Generalidades de la Administración de Derechos Digitales (DRM)

Como es conocido, y nos referimos ahora a la figura 11, la administración de derechos digitales (DRM) y el cumplimiento es altamente deseable en conexión con el contenido digital 12 tal como un audio digital, video digital, texto digital, datos digitales, multimedia digital, etc., en donde tal contenido digital 12 va a ser distribuido a usuarios. Luego de ser recibido por el usuario, tal usuario utiliza o "ejecuta" el contenido digital con la ayuda de un dispositivo de ejecución apropiado tal como un dispositivo reproductor de medios en una computadora personal 14 o similar.

Típicamente, un propietario o desarrollador de contenido (de aquí en adelante "propietario") que distribuye tal contenido digital 12 desea restringir lo que el usuario pueda hacer con tal contenido digital distribuido 12. Por ejemplo, el propietario del contenido puede desear restringir al usuario que haga copias o que redistribuya tal contenido 12 a un segundo usuario, o puede desear permitir que el contenido digital distribuido 12 sea ejecutado sólo un número de veces limitado, solamente

durante un tiempo total limitado, o solamente en un cierto tipo de máquina, o solamente en un cierto tipo de reproductor de medios o solamente por un cierto tipo de usuario, etc.

Sin embargo, luego de que la distribución ha ocurrido, tal propietario del contenido tiene muy poco si es que tiene algún control sobre el contenido digital 12. Un sistema DRM 10 entonces, permite la ejecución controlada de formas arbitrarias del contenido digital 12, en donde tal control es flexible y definible por el propietario del contenido de tal contenido digital. Típicamente, el contenido 12 es distribuido al usuario en la forma de un paquete 13 por vía de un canal de distribución apropiado. El paquete de contenido digital 13 tal como está distribuido puede incluir el contenido digital 12 encriptado con una clave simétrica de encriptación/desencriptación (KD), (es decir, (KD(CONTENT))), lo mismo que otra información que identifique el contenido, como se adquiere una licencia para tal contenido, etc.

El sistema DRM con base en la confianza 10 permite a un propietario del contenido digital 12 especificar las

reglas de licencia que deben satisfacerse antes de que tal contenido digital 12 se le permita ser ejecutado en un dispositivo de cómputo del usuario 14. Tales reglas de licencia pueden incluir los requerimientos temporales anteriormente mencionados, y puede también involucrar dentro de una licencia digital o un documento de uso (de aquí en adelante "licencia") 16 que el dispositivo de cómputo del usuario/usuarios 14 (de aquí en adelante, tales términos son intercambiables a menos que las circunstancias lo requieran de otra manera) pueda obtener del propietario del contenido o de un agente del mismo. Tal licencia 16 también incluye la clave de descriptación (KD) para descriptar el contenido digital, quizás encriptado de acuerdo con una clave descriptable por el dispositivo de cómputo del usuario.

El propietario del contenido para una pieza de contenido digital 12 puede confiar que el dispositivo de cómputo del usuario 14 estará rigiéndose por las reglas y requerimientos especificados por tal propietario de contenido en la licencia 16, por ejemplo que el contenido digital 12 no será ejecutado a menos que las reglas y requerimientos dentro de la licencia 16 sean satisfechos. Preferiblemente, entonces, el dispositivo de cómputo del

usuario 14 está provisto con un componente o mecanismo confiado 18 que no ejecuta el contenido digital 12 excepto de acuerdo con las reglas de licencia involucradas en la licencia 16 asociadas con el contenido digital 12 y obtenidas por el usuario.

El componente confiado 18 típicamente tiene un evaluador de licencia 20 que determina si la licencia 16 es válida, revisa las reglas y requerimientos de la licencia en tal licencia válida 16, y determina con base en las reglas y requerimientos de la licencia revisados si el usuario que solicita tiene el derecho de ejecutar el contenido digital solicitado 12 en la manera vista, entre otras cosas. Como debe entenderse, el evaluador de licencia 20 es confiado en el sistema DRM 10 para llevar a cabo los deseos del propietario del contenido digital 12 de acuerdo con las reglas y requerimientos en la licencia 16, y el usuario no debe ser capaz de alterar de manera fácil tal elemento confiado para cualquier propósito, infracción o similar.

Como debe ser entendido, las reglas y requerimientos en la licencia 16 pueden especificar si el usuario tiene derechos de ejecutar el contenido digital 12 con base en

cualquiera de los factores diversos, que incluyen quien es el usuario, en donde esta el usuario localizado, que tipo de dispositivo de cómputo usa el usuario, que aplicación de ejecución está llamando al sistema DRM, la fecha, el tiempo, etc. En adición, las reglas y requerimientos de la licencia 16 pueden limitar la licencia 16 a un número de ejecuciones predeterminado, o a un número de juegos predeterminado, por ejemplo.

Las reglas y requerimientos pueden ser especificadas en la licencia 16 de acuerdo con cualquier lenguaje o sintaxis apropiada. Por ejemplo, el lenguaje puede simplemente especificar atributos y valores que pueden ser satisfechos (DATE puede ser posterior que X, p.e.), o puede requerir el comportamiento o el llevar a cabo funciones de acuerdo con una lista de comandos (IF DATE mayor que X, THEN DO..., p.e.).

Luego de que el evaluador de licencia 20 determina que la licencia 16 es válida y que el usuario satisface las reglas y requerimientos en ella, el contenido digital 12 puede entonces ser ejecutado. En particular, para ejecutar el contenido 12, la clave de descricción (KD) es obtenida de la licencia 12 y es aplicada al

(KD(CONTENT)) desde el paquete de contenido 13 que resulta en el contenido actual 12, y el contenido real 12 es entonces de hecho ejecutado.

#### Contenido Digital Publicado

La figura 3 es un diagrama de bloques funcional de un sistema y método para publicar contenido digital. "Publicar", es aquel término usado aquí que se refiere a un proceso que una aplicación o servicio sigue para establecer con una entidad confiada un conjunto de derechos y condiciones que la entidad puede emitir para ese contenido, lo mismo que de aquellos cuyos derechos y condiciones pueden ser otorgados. De acuerdo con la invención, el proceso de publicar incluye encriptar el contenido digital y asociar una lista de derechos exigibles persistentes que el autor del contenido tiene el propósito para los posibles usuarios del contenido. Este proceso puede llevarse a cabo de una manera segura prohibiendo el acceso a cualquiera de los derechos o a los contenidos a menos que sea el propósito de los autores del contenido.

Tres entidades en particular son empleadas para publicar contenido digital seguro: una solicitud de preparación de contenido 302 que ejecuta sobre el cliente 300 y prepara el contenido para publicación, unas aplicaciones de administración de derechos digitales (DRM) como interface de programa (API) 306 que también reside en el dispositivo del cliente 300, y un servidor DRM 320 que está acoplado de manera comunicada al cliente 300 por medio de una red de comunicación 330 tal como la Internet, una red local o de área amplia, o una combinación de ellos.

La aplicación de preparación de contenido 302 puede ser cualquier aplicación que produzca contenido digital. Por ejemplo, la aplicación 302 puede ser un procesador de palabras u otro publicador que produzca archivos de texto digital, música digital, video, u otro de tales contenidos. El contenido también podría incluir contenido en corriente, tal como el audio/video en corriente de un evento en vivo o grabado en cinta, o un ejemplo. La aplicación 302 está provista con una clave criptográfica para encriptar el contenido digital, formando así un archivo de contenido digital encriptado 304, y el usuario suministra los datos de derechos que están asociados muy

cercanamente con el contenido encriptado en el archivo de contenido digital 304. Los datos de derechos incluyen una identidad para cada entidad que tiene derechos en el contenido digital, y un conjunto de derechos y condiciones para cada entidad identificada.

Tal una entidad puede ser, por ejemplo, un individuo, una clase de individuos, o un dispositivo. Tales derechos pueden incluir el derecho a leer, editar, copiar, imprimir, etc., el contenido digital. Las condiciones pueden incluir requerimientos mínimos del sistema, limitaciones en cuanto a fechas y tiempos, veces de ejecución, y similares.

El cliente API 306 pasa el contenido digital encriptado y los datos de derechos al servidor DRM 320. Usando un proceso que está descrito en detalle más adelante, el servidor DRM 320 determina si puede hacer cumplir los datos de derechos y si es así el servidor DRM 320 firma los datos de derechos para formar una etiqueta de derechos firmada (SRL) 308. En general, sin embargo, cualquier entidad confiada puede firmar los datos de derechos, preferiblemente usando una clave confiada por el servidor DRM 320. Por ejemplo, un cliente puede firmar

los datos de derechos usando una clave suministrada a éste por el servidor DRM 320.

La etiqueta de derechos 308 puede incluir datos que representan la descripción de los derechos, en la clave del contenido encriptado, y la firma digital sobre la descripción de los derechos y la clave de contenido encriptado. Si el servidor DRM 320 está firmando la etiqueta de derechos, éste pasa la etiqueta de derechos firmada 308 de nuevo al cliente por el API del cliente 306, el cual almacena la etiqueta de derechos firmada 308 en el dispositivo del cliente 300. La aplicación de la preparación del contenido 302 entonces asocia la etiqueta de derechos firmada 308 con el archivo de contenido digital encriptado 304, de modo que por ejemplo por concatenación para formar un archivo de contenido administrado de derechos 310. Note, sin embargo, que el SRL 308 podría estar almacenado en una localización separada conocida del archivo de contenido 304 con una referencia al SRL 308 concatenada al archivo de contenido 304 para formar el archivo de contenido 310.

Haciendo referencia ahora a la figura 4, un método para publicar el contenido digital administrado por derechos

se muestra. En la etapa 402, la aplicación 302 genera una clave de contenido (CK) que es usada para encriptar el contenido digital. La clave del contenido (CK) es típicamente una clave simétrica aunque cualquier clave puede ser usada para encriptar el contenido digital. Como es bien conocido, una clave simétrica es utilizada por un algoritmo de clave simétrica tanto para encriptar como desencriptar. De acuerdo a lo anterior, (CK) debería estar bien escondida cuando es compartida entre el que envía y un receptor.

En la etapa 404, la aplicación 302 encripta el contenido digital con (CK) para formar el contenido digital encriptado 304 (p.e., (CK(content))). Adicionalmente, los datos de derechos correspondientes al (CK(content)) son generados, sea por el publicador del contenido o por otra entidad. Note que tales datos de derechos pueden ser datos de derechos adaptados o datos de derechos según se obtienen de una plantilla predefinida. Como se discutió anteriormente, los datos de derechos pueden incluir una lista de entidades las cuales tendrán derecho a consumir el contenido, los derechos específicos que cada una de las entidades posee con respecto al contenido, y

cualesquier condiciones que puedan ser impuestas sobre aquellos derechos.

En la etapa 406, el API 306 genera una segunda clave de encriptación (K2), que es usada para encriptar la clave del contenido (CK). Preferiblemente, (K2) es también una clave simétrica. En la etapa 408, el API 306 encripta (CK) con (K2) resultando en (K2(CK)). En la etapa 410, el API 306 descarta (CK), con el resultado siendo que (CK) puede ahora ser obtenida solamente desenscriptando (K2(CK)). Para asegurarse que (CK(content)) está protegido en un servidor DRM central 320 y que todas "las solicitudes de licencia" para el contenido se hacen centralizadamente de acuerdo con los datos de derechos, el API 306, en la etapa 412, hace contacto con el servidor DRM 320 suministrado y recupera la clave pública (PU-DRM) del mismo. En la etapa 414, el API 306 encripta (K2) con (PU-DRM) resultando en (PU-DRM(K2)). Así, (CK) puede estar protegida en (PU-DRM) para asegurar que el servidor DRM 320 es la única entidad que será capaz de tener acceso a (CK), como es requerido para desenscriptar (CK(content)). En la etapa 416, el API 306 encripta los datos de derechos (p.e., la lista de entidades autorizadas y los derechos respectivos y condiciones

asociadas con cada una de las entidades autorizadas en la lista) con (K2) resultando en (K2(rightsdata)).

En una modalidad alternativa, (CK) puede ser usada para encriptar directamente los datos de derechos para resultar en (CK(rightsdata)), y (PU-DRM) puede ser usada para encriptar directamente (CK) resultando (PU-DRM(CK)), por lo tanto el uso anterior de (K2) completamente. Sin embargo, usando (K2) para encriptar los datos de derechos y (CK) permite que tal (K2) se conforme a un algoritmo para el particular que pueda ser llevable al servidor DRM, mientras que (CK) puede ser especificado por una entidad independiente del servidor DRM y puede no ser llevable al mismo.

En la etapa 418, la aplicación de protección de contenido 302 submite (PU-DRM(K2)) y (K2(rightsdata)) al servidor DRM 320 como una etiqueta de derechos para ser firmada. Alternativamente, el cliente mismo puede firmar los datos de derechos en la manera establecida como sigue. Si los datos de derechos han sido presentados al servidor para firma, entonces, en la etapa 420 el servidor DRM 320 accesa los datos de derechos y verifica que pueda hacer cumplir los derechos y condiciones en la etiqueta de

derechos presentada. Para verificar que éste pueda hacer cumplir los datos de derechos, el servidor DRM 320 aplica la clave privada (PR-DRM) correspondiente a (PU-DRM) a (PU-DRM(K2)) que resulta en (K2), y luego aplica (K2) a (K2(rightsdata)) para resultar en los datos de derechos desenscriptados. El servidor 320 puede entonces hacer cualquier verificación de la política para verificar que los usuarios, derechos y condiciones especificadas en los datos de derechos están dentro de cualquier política que se debe hacer cumplir por el servidor 320. EL servidor 320 firma la etiqueta de derechos presentada originalmente incluyendo (PU-DRM(K2)) y (K2(rightsdata)) para resultar en la etiqueta de derechos firmada (SRL) 308, en donde la firma está basada en la clave privada del servidor DRM 320 (PR-DRM), y regresa el SRL 308 de nuevo al API 306, el cual entonces presenta el SRL 308 regresado a la aplicación del cliente 302.

El SRL 308 es un documento firmado digitalmente, el cual lo hace inviolable. Adicionalmente, el SRL 308 es independiente del tipo de clave real y algoritmo usado para encriptar el contenido sino que mantiene la relación fuerte 1-1 al contenido que está protegiendo. Haciendo referencia ahora a la figura 4A, en una modalidad de la

presente invención, el SRL 308 puede incluir información sobre el contenido que es la base del SRL 308, incluyendo quizás un ID del contenido; información sobre el servidor DRM que firma el SRL 308, incluyendo  $(PU-DRM(K2))$  e información de referencia tal como un URL para localizar el servidor DRM en una red e información de respaldo si el URL falla; información que describa el SRL 308 en sí mismo;  $(K(rightsdata)):(K2(CK))$ ; y una firma digital  $(S(PR-DRM))$ , entre otras cosas.

Por medio de asegurar que una entidad confiada firma los datos de derechos para crear una etiqueta de derechos firmada 308, el servidor DRM 320 está afirmando que éste otorgará o emitirá licencias para el contenido de acuerdo con los términos establecidos por el publicador como se describió en los datos de derechos en la etiqueta de derechos 308. Como podría ser apreciado, un usuario se le ha requerido obtener una licencia para visualizar el contenido, especialmente en tanto que la licencia contiene la clave del contenido (CK). Cuando un usuario quiere obtener una licencia para el contenido encriptado, el usuario puede presentar una solicitud de licencia que incluye el SRL 308 para el contenido y un certificado de verificación de las credenciales del usuario al servidor

DRM 320 u otra entidad otorgadora de licencias. La entidad otorgadora de licencias puede entonces desenscriptar  $(PU-DRM(K2))$  y  $(K2(rightsdata))$  para producir los datos de derechos, la lista de todos los derechos otorgados por el autor (si hay alguno) a la entidad solicitadora de la licencia, y construir una licencia con solamente aquellos derechos específicos.

Como se ha establecido anteriormente, luego de que la aplicación 302 recibe el SRL 308, tal aplicación 302 concatena la etiqueta de derechos firmada 308 con  $(CK(content))$  304 correspondiente para formar el contenido digital administrado por derechos. Alternativamente, los datos de derechos son almacenados en una localización conocida con una referencia a aquella localización suministrada con el contenido digital encriptado. Así, una aplicación que se ejecuta que es autorizada por el DRM puede descubrir la etiqueta de derechos firmada 308 por medio de la pieza de contenido de la aplicación que se ejecuta al intentar ejecutarla. Este descubrimiento dispara la aplicación que se ejecuta para iniciar una solicitud de licencia contra el servidor de licencias DRM 320. La aplicación de publicación 302 puede almacenar un URL en el servidor de licencia DRM

320, por ejemplo, o el servidor de licencias DRM 320 puede involucrar su propio URL como una pieza de metadatos en la etiqueta de derechos antes de firmarla digitalmente, de modo que el API 306 del cliente DRM llamado por la aplicación que se ejecuta puede identificar el servidor de licencias DRM 320 correcto.

#### Obtener una Licencia para el Contenido Publicado

Haciendo referencia ahora a la figura 5, un sistema y un método para licenciar el contenido digital administrado por derechos se muestra. "licenciar" es aquel término usado aquí, que se refiere a un proceso en que una aplicación o servicio luego de la solicitud y recepción de una licencia permitirá a una entidad llamada en la licencia consumir el contenido de acuerdo con los términos especificados en la licencia. Las entradas al proceso de licenciamiento pueden incluir la etiqueta de derechos firmada (SRL) 308 asociada con el contenido para el cual una licencia está siendo solicitada, y el certificado(s) de clave pública de la entidad(s) para las cuales la licencia está siendo solicitada. Note que la entidad que solicita una licencia no necesariamente es la entidad para la cual la licencia está siendo solicitada.

Típicamente, una licencia incluye la descripción de derechos desde el SRL 308, una clave encriptada que puede desencriptar el contenido encriptado, y una firma digital sobre la descripción de derechos y la clave encriptada para asegurar la legitimidad e impedir la violación.

Preliminarmente, el cliente API 306 dirige la etiqueta de derechos firmada 308 del contenido administrado por derechos 310 al servidor DRM 320 por medio de la red de comunicaciones 330. Como se describió anteriormente, la etiqueta de derechos 308 contiene la clave del contenido (CK) encriptada de acuerdo con la clave pública del servidor DRM 320 (PU-DRM) (es decir, (PU-DRM(CK))). En el proceso de emisión de una licencia, entonces, el servidor DRM 320 aplica (PR-DRM) al (PU-DRM(CK)) para obtener (CK). Este entonces usa la clave pública (PU-ENTITY) en el certificado de clave pública que es pasado en la solicitud de licencia para reencriptar (CK) (es decir, (PU-ENTITY(CK))). La nuevamente encriptada (PU-ENTITY(CK)) es entonces colocada dentro de la licencia. Así, la licencia puede ser regresada al que llama sin riesgo de exponer (CK), ya que solamente el que posee la clave privada (PR-ENTITY) correspondiente a (PU-ENTITY) puede recuperar (CK) de (PU-ENTITY(CK)). El cliente API

306 entonces usa (CK) para desenscriptar el contenido encriptado para formar el contenido digital desenscriptado 312. La aplicación de cliente 302 puede entonces usar el contenido digital desenscriptado 312 de acuerdo con los derechos que son suministrados en la licencia.

Alternativamente, y como se establece enseguida en más detalle, un cliente tal como el cliente que publica puede por ejemplo otorgar una licencia de uso así mismo para consumir el contenido.

Volviendo ahora a las figuras 6A y 6B, un método para licenciar el contenido digital administrado por derechos se muestra. En la etapa 602, una entidad emisora de licencia tal como el servidor DRM 320 recibe una solicitud de licencia incluyendo sea un certificado de clave pública o una identidad para cada uno o más de los licenciatarios solicitantes. Presumiblemente, si una entidad es especificada, el servidor DRM 320 puede procurar un certificado de clave pública correspondiente de un directorio, una base de datos, o similar. Si una licencia es solicitada para solamente un licenciatario, solamente un certificado o identidad es llamado. Si una licencia es solicitada para una pluralidad de

licenciarios, un certificado o una identidad puede ser llamada para cada licenciario potencial. En la etapa 604, la entidad solicitante (es decir, la entidad que hace la solicitud de licencia) es autenticada, si se desea. En la etapa 606, se ha determinado si la entidad se le permite solicitar una licencia nuevamente si es deseado.

Si, en la etapa 608, la entidad que otorga determina que el certificado de clave pública no está incluido en la solicitud de licencia, entonces la entidad que otorga usa la identidad especificada para llevar a cabo una búsqueda en un servicio de directorio o base de datos por el certificado de clave pública apropiado. Si, en la etapa 610, la entidad que otorga determina que el certificado está en el directorio, entonces, en la etapa 612, el certificado es recuperado. Si un certificado no puede ser encontrado para un licenciario potencial dado, sea en la solicitud o en el directorio, entonces el servidor de licencias no generará una licencia para ese licenciario potencial y, en la etapa 614, un error es regresado a la entidad solicitante.

Asumiendo que el servidor DRM 320 tiene un certificado de clave pública para por lo menos un licenciatario potencial, entonces, en la etapa 616, tal servidor DRM 320 valida la verdad de cada certificado de licenciatario. Si no es validado, el servidor DRM 320 determina que el otorgador de un certificado de licenciatario no está en la lista de otorgadores verdaderos, entonces la solicitud falla para ese licenciatario, y un error es generado en la etapa 614. Así, cualquier licenciatario potencial cuyo certificado no sea otorgado por un otorgador verdadero no recibiría una licencia.

Adicionalmente, el servidor DRM 320 preferiblemente lleva a cabo la validación de la firma digital en todas las entidades en la cadena de certificados que va desde los certificados verdaderamente otorgados a los certificados de clave pública del licenciatario individual. El proceso de validar las firmas digitales en una cadena es un algoritmo bien conocido. Si el certificado de clave pública para un licenciatario dado potencial no se valida, o un certificado en la cadena no se valida, el licenciatario potencial no se le da confianza, y una licencia, por lo tanto, no es otorgada al licenciatario

potencial. De otra manera, en la etapa 618, una licencia puede ser otorgada. El proceso repite en la etapa 620 hasta que todas las entidades para las cuales una licencia ha sido solicitada han sido procesadas.

Como se muestra en la figura 6B, el servidor DRM 320 procede a validar la etiqueta de derechos firmada 308 que es recibida en la solicitud de licencia. En una modalidad, el servidor DRM 320 tiene una copia maestra de todas las etiquetas de derechos firmadas allí. Al momento de la licencia (en la etapa 622), entonces, el servidor DRM 320 puede recuperar una copia de la etiqueta de derechos maestra. La etiqueta de derechos maestra puede estar mas a la fecha que la copia de la etiqueta de derechos enviada en la solicitud de licencia, y por lo tanto la etiqueta de derechos será empleada para producir la licencia requerida. Si no se encuentra una etiqueta de derechos maestra el servidor DRM 320, en la etapa 624, determina de acuerdo con una política predefinida si se otorga una licencia con base en la etiqueta de derechos en la solicitud. Si la política no lo permite, la solicitud de licencia falla en la etapa 626, y un error se regresa al API 306 en la etapa 628.

En la etapa 630, el servidor DRM 320 valida el SRL 308 y específicamente la firma digital del mismo. Si el SRL 308 no se valida, la solicitud de licencia falla en la etapa 626 y un error es regresado al API 306 en la etapa 628.

Después de que todas las validaciones han ocurrido, el servidor DRM construye una licencia para cada licencia aprobada con base en el SRL 308. En la etapa 632, el servidor DRM 320 genera una descripción de derechos respectiva para la licencia que va a ser otorgada a cada licenciatario. Para cada licenciatario, el servidor DRM 320 evalúa la identidad llamada en el certificado de clave pública de ese licenciatario contra las identidades llamadas en la descripción de derechos en la etiqueta de derechos. En la etapa 636, el servidor DRM 320 obtiene  $(PU-DRM(K2))$  y  $(K2(CK))$  del SRL 308 y aplica  $(PR-DRM)$  para obtener  $(CK)$ . La entidad otorgante entonces reencrypta  $(CK)$  usando  $(PU-ENTITY)$  del certificado de clave pública del licenciatario para resultar en  $(PU-ENTITY(CK))$ . En la etapa 638, el servidor DRM 320 concatena la descripción de derechos generada con  $(PU-ENTITY(CK))$  y firma digitalmente la estructura de datos resultante usando  $(PR-DRM)$  (es decir,  $S(PR-DRM)$ ). La

estructura de datos firmada es entonces la licencia para este licenciatario particular.

En la etapa 640, el servidor DRM 320 determina que no hay más licencias para generar en la solicitud particular. Las licencias generadas son entonces regresadas a la entidad solicitante, en la etapa 642, junto con una cadena de certificado apropiada que liga las licencias de nuevo a una autoridad confiada.

#### Autopublicar la Etiqueta de Derechos Firmada 308

En una modalidad de la presente invención, el SRL 308 puede ser firmado por el solicitante/usuario publicador mismo. De acuerdo con lo anterior, tal usuario no requiere contactar el servidor DRM 320 para obtener un SRL 308 para una pieza asociada de contenido. Como resultado, autopublicar puede también referirse a publicar fuera de línea. En tal modalidad, el usuario publicador debería ser capaz de emitir el mismo una licencia de publicador, especialmente siempre que el contenido autopublicado esté ahora protegido por el DRM y tal licencia de publicador se es requerida para permitir que el usuario publicador utilice el contenido ahora

protegido. Debería también entenderse que un usuario publicador se le puede permitir emitir licencias a otros usuarios.

En particular, y haciendo referencia ahora a la figura 7, en la modalidad, un usuario publicador de fuera de línea está primero aprovisionado para publicar fuera de línea por medio de recibir de un servidor DRM 320 un certificado de publicación fuera de línea (OLP) 810 incluyendo una clave pública (PU-OLP) y una clave privada correspondiente (PR-OLP) encriptada de acuerdo con una clave pública directa o indirectamente accesible al componente confiado 18 (Figura 11) del usuario (PU-ENTITY) para resultar en (PU-ENTITY(PR-CERT)). Note que (PU-ENTITY) puede por ejemplo ser la clave pública del componente confiado 18, o puede ser una clave pública del usuario que es accesible por vía de la clave pública del componente confiado 18. El certificado OLP 810 debería estar firmado por la clave privada del servidor DRM 320 (PR-DRM) de modo que tal servidor DRM 320 pueda verificar tal certificado OLP, como será discutido en más detalle enseguida.

Adicionalmente, el certificado OLP 810 debería incluir una cadena de certificado desde (PU-DRM) hacia atrás hasta la autoridad confiada que se ha confiado por el componente confiado 18 del usuario publicador o de otro usuario de modo que el componente confiado 18 pueda verificar tal certificado OLP 810 y cualquier otro certificado o licencia que esté asociado con tal certificado OLP 810, como será discutido enseguida. Brevemente, como debería entenderse, una cadena de certificados comienza con un certificado raíz firmado por la clave privada de una autoridad confiada y que tiene la clave pública del siguiente certificado en la cadena. Cada certificado intermedio en la cadena, entonces, es firmado por la clave privada correspondiente a la clave pública del certificado previo en la cadena, y tiene la clave pública del siguiente certificado en la cadena. Finalmente, el certificado o licencia al cual la cadena está ligada es firmado por la clave privada correspondiente a la clave pública del último certificado en la cadena.

Así, para verificar el certificado o licencia al cual la cadena está ligada, el conocimiento de la clave pública correspondiente a la clave privada de la autoridad

confiada es ganado, y tal clave pública de la autoridad confiada es empleada para verificar la firma del certificado raíz en la cadena. Presumiendo que la firma del certificado raíz verifica, entonces la clave pública del certificado raíz es obtenido y empleado para verificar la firma del primer certificado intermedio en la cadena. El proceso se repite seriadamente a través de la cadena hasta que cada firma del mismo sea verificada, y entonces la clave pública del último certificado intermedio en la cadena es obtenido y empleada para verificar la firma del certificado o licencia al cual la cadena está ligada.

Como debería apreciarse, el certificado OLP 810 crea un enlace en la cadena de verdad entre el contenido 304 que va a ser publicado fuera de línea y el servidor DRM 320 que otorgará una licencia para el contenido 304. El certificado OLP 810 puede ser creado con base en un lenguaje XML/XrML o en cualquier otro lenguaje apropiado.

Como debería apreciarse, el certificado OLP 810 y la cadena de certificado ligada autoriza al usuario publicador para autopublicar. Como puede apreciarse adicionalmente, el par de claves (PU-OLP, PR-OLP) están

separadas de (PU-ENTITY, PR-ENTITY), y son empleadas específicamente para autopublicación. Note que el par de claves (PU-OLP, PR-OLP) pueden dispensarse con, en cuyo caso el certificado DRM 810 incluye solamente la clave pública del usuario (PU-ENTITY) y es firmado por la clave privada del servidor, DRM 320 (PR-DRM) de modo que el servidor DRM 320 pueda verificar la misma.

La autopublicación difiere de la publicación como se muestra en la figura 4 en que el usuario esencialmente toma el lugar del servidor DRM 320 con respecto a las etapas llevadas a cabo allí. Significativamente, el usuario firma la etiqueta de derechos presentada incluyendo (PU-DRM(K2)) y (K2(rightsdata)) o incluye (PU-DRM(CK)) y (CK(rightsdata)) (la última estando mostrada en las figuras 7 y 8) con (PR-OLP) según se obtuvo del certificado DRM 810 (es decir, S(PR-OLP)) para resultar en la etiqueta de derechos firmada (SRL) 308. El componente confiado 18 del cliente en uso del certificado OLP 810 verifica típicamente al mismo con base en la cadena de certificado ligada. Como puede apreciarse, el componente confiado 18 del usuario obtiene (PR-OLP) del certificado OLP 810 obteniendo (PU-ENTITY(PR-OLP)) de tal certificado OLP 810 y aplica el (PR-ENTITY). Note, Sin

embargo, que el usuario publicador no puede verificar que el servidor DRM 320 pueda hacer cumplir los derechos en un SRL 308 autopublicado. De acuerdo a lo anterior, el servidor DRM 320 mismo debería llevar a cabo la verificación en el momento en que una licencia es solicitada con base en el SRL 308 de autopublicado.

Una vez que el usuario publicador autopublica el SRL 308, el usuario concatena tal SRL 308 autopublicado y el certificado OLP 810 empleado para producir el mismo al contenido 304, y tal contenido 304 con el SRL 308 y el certificado DRM 810 es distribuido como el contenido administrado por derechos 310 a otro usuario. De ahí en adelante, el otro usuario solicita y obtiene una licencia para el contenido 304/310 del servidor DRM 320 de sustancialmente la misma manera como se mostró en las figuras 6A y 6B. Aquí, sin embargo, el usuario que solicita la licencia presente al servidor DRM 320 tanto el SRL 308 autopublicado y el certificado OLP 810 como concatenado al contenido 304. El servidor DRM 320 entonces verifica el S(PR-DRM) en el certificado OLP 810 con base en el (PU-DRM) correspondiente, y obtiene el (PU-OLP) del certificado DRM 810. El servidor DRM 320 entonces verifica el S(PR-OLP) en el SRL 308 con base el

(PU-CERT) obtenido, y continua como antes. Note, sin embargo, que ya que el usuario publicador no verifico que el servidor DRM 320 pueda hacer cumplir los derechos en el SRL 308, y como estaba establecido anteriormente el servidor DRM 320 mismo debería llevar a cabo la verificación en este momento.

Note, también, que el servidor DRM 320 necesita solamente verificar el S(PR-DRM) en el certificado OLP 810, ya que presumiblemente éste confía en sí mismo. De acuerdo a lo anterior, la cadena de certificado asociada del certificado OLP 810 no necesariamente requiere ser enviado al servidor DRM 320 junto con tal certificado OLP 810, a menos por supuesto que en la cadena sea necesario de otra manera, tal como por ejemplo si la cadena misma es por lo menos parcialmente la base para el S(PR-DRM).

De manera muy importante, sin embargo, el usuario publicador debería ser capaz de ejecutar el contenido ahora protegido 304/310 sin tener que ir al servidor DRM 320 para una licencia. Puesto de otra manera, un usuario publicador que publica fuera de línea el contenido 304/310 sin ir al servidor DRM 320 con base en un certificado OLP 810 debería ser capaz de también otorgar

el mismo una licencia, de una manera fuera de línea sin tener que ir al servidor DRM 320 de modo que el usuario pueda ejecutar el contenido publicado fuera de línea 304/310. De acuerdo a lo anterior, un usuario publicador puede continuar trabajando con el contenido autopublicado 310 sin ninguna conectividad a un servidor DRM 320.

En una modalidad de la presente invención, entonces, y volviendo ahora a la figura 8, un usuario publicador emite el mismo una licencia de publicador fuera de línea 820 firmada por (PR-OLP) y con base en el SRL 308 autopublicado, incluye el certificado OLP 810 y la cadena de certificado del mismo. Presumiblemente, la licencia de publicador 820 otorga al usuario publicador acceso completo al contenido autopublicado 310, aunque en una cantidad menor de acceso pudiera ser otorgada. La licencia de publicador 820 puede estar escrita en un lenguaje XML/XrML o en otro lenguaje, como es el caso con otras licencias DRM. Como pudiera apreciarse, la licencia de publicador 820 incluye la clave de contenido (CK) encriptada de acuerdo con (PU-ENTITY), la cual puede ser obtenida por un componente confiado 18 del dispositivo de cómputo del usuario 14, para formar (PU-ENTITY(CK)).

La cadena para la licencia de publicador 820 por lo tanto va desde tal licencia 820 al certificado OLP 810 y luego de nuevo a un certificado raíz de una autoridad confiada, quizás por vía de uno o más certificados intermedios. Ya que el componente confiado 18 del usuario presumiblemente puede ganar la clave pública correspondiente a la clave privada de la autoridad confiada que fue empleada para firmar el certificado raíz, el componente confiado 18 puede en sí mismo verificar la licencia de publicador 820 por vía de la cadena de certificado del mismo, y luego de la verificación puede entonces obtener el (PU-ENTITY(CK)) de la misma, aplicar (PR-ENTITY) a ella y obtener (CK), y aplicar (CK) a (CK(content)) para resultar en el contenido 304 para propósitos de su ejecución. Como un resultado, el usuario publicador puede continuar trabajando con el contenido 310 publicado fuera de línea de esa manera mientras que permanece fuera de línea.

De acuerdo con lo anterior, entonces, y refiriéndose ahora a la figura 9, un usuario publicador que publica fuera de línea el contenido 304/310 y emite el mismo una licencia de publicador fuera de línea 820 para tal contenido 304/310 en la siguiente manera.

Preliminarmente, y como debía apreciarse, el contenido 304 es desarrollado de una manera apropiada y encriptado de acuerdo con una clave de contenido (CK)(etapa 901), y el usuario publicador crea una etiqueta de derechos para el contenido 304 con la información apropiada {(PU-DRM(CK)) y (CK(rightsdata)), p.e.}(etapa 903). De ahí en adelante, el usuario publicador, que presumiblemente está ya en posesión de un certificado OLP 810 del servidor DRM 320, obtiene tal certificado OLP 810 (etapa 905) y verifica el mismo con base en la firma de él y la cadena de certificados que lleva de nuevo a la autoridad de raíz (etapa 907). Como pudiera apreciarse, tal verificación se lleva a cabo realmente por un componente confiado 18 en un dispositivo de cómputo 14 del usuario publicador. Asumiendo que la verificación es exitosa, entonces, el usuario publicador/componente confiado 18 (de aquí en adelante "el usuario publicador") recupera el (PU-ENTITY(PR-OLP)) del certificado OLP 810 (etapa 909), aplica el (PR-ENTITY) al (PU-ENTITY(PR-OLP)) para obtener el(PR-OLP)(etapa 911), y luego firma la etiqueta de derechos creada con tal (PR-OLP) para crear un SRL 308 (etapa 913).

Después, el usuario publicador concatena tal SRL 308 y el certificado OLP 810 empleado para producirlo al contenido 304 para formar el contenido autopublicado 310 (etapa 915), y tal contenido administrado por derechos 310 es por lo tanto capaz de ser distribuido a otro usuario. Para que el usuario publicador continúe usando o ejecutando el contenido 310, sin embargo, tal usuario publicador debe emitir él mismo una licencia de publicador fuera de línea 820 correspondiente.

Así, el usuario publicador crea una licencia de publicador 820 definiendo los datos de derechos apropiados para él mismo y encriptando los mismos de acuerdo con la clave de contenido (CK) para resultar en (CK(rightsdata))(etapa 917). Note aquí que tales datos de derechos pueden ser obtenidos del SRL 308 del contenido 310, puede ser algún conjunto original de datos de derechos que otorgan al usuario publicador un acceso parcial o completo al contenido autopublicado 310, o puede ser derivado de otra fuente. En adición, el usuario publicador encripta la clave de contenido (CK) de acuerdo con el (PU-ENTITY) para formar (PU-ENTITY(CK))(etapa 919). Tal (CK(rightsdata)) y (PU-ENTITY(CK)) son entonces formateados en la licencia de publicador 820 (etapa 921),

el certificado OLP 810 y la cadena certificados del mismo quedan sujetas (etapa 923) y tal licencia de publicador 820 es firmada con base en el (PR-OLP) como fue obtenido en la etapa 911 (etapa 925). Note aquí que el contenido 304 (es decir, (CK(content))), la licencia de publicación 820, y el certificado OLP en combinación forman una cadena 830 de elementos digitales de nuevo a la autoridad confiada.

Para que el usuario publicador ejecute el contenido publicado 310, entonces, y volviendo ahora a la figura 10, tal usuario publicador no requiere hacer contacto con el servidor DRM 320, sino en su lugar gana la clave pública correspondiente a la clave privada de la autoridad confiada que fue empleada para firmar el certificado raíz (1001), verifica el certificado raíz (etapa 1003), y luego verifica cada certificado intermedio en la cadena (etapa 1005) por, para cada uno de tales certificados intermedios, obtener la clave pública del certificado previo y emplear la misma para verificar la firma de tal certificado. De ahí en adelante, el (PU-DRM) del último certificado en la cadena es empleado para verificar la firma del certificado OLP 810 (es decir, S(PR-DRM))(etapa 1007), el (PU-OLP) es

obtenido del certificado OLP 810 (etapa 1009), y tal (PU-OLP) es empleado para verificar la firma de la licencia de publicador 820 (es decir, S(PR-OLP)) (etapa 1010).

Una vez que la licencia de publicador 820 ha sido verificada, entonces, el (CK(rightsdata)) y el (PU-ENTITY(CK)) son recuperados de la misma (etapa 1011), el (PR-ENTITY) es aplicado al (PU-ENTITY(CK)) para resultar en el (CK) (etapa 1013), y el (CK) es aplicado al (CK(rightsdata)) para resultar en los datos de derechos (etapa 1015). Como se puede apreciar ahora, los datos de derechos son revisados por el componente confiado 18 del dispositivo de cómputo 14 del usuario publicador para determinar que tales datos de derechos permiten ejecutar de la manera vista (etapa 1017), tal componente confiado 18 entonces aplica el (CK) al (CK(content)) del contenido 310 para resultar en el contenido (etapa 1019), y tal contenido es entonces enviado a una aplicación de ejecución apropiada para ejecutarlo realmente (etapa 1021). Así, las etapas de la figura 10 en efecto atraviesan la cadena 830 de elementos digitales desde la autoridad confiada hasta el contenido 304.

Note que el componente confiado 18 podría aplicar concebiblemente el (CK) al (CK(content)) para resultar en el contenido sin primero revisar los datos de derechos y sin importar de que datos de derechos estaba permitido o no permitido, pero es confiado y ha sido construido para de hecho producir el contenido solamente si después de revisar los datos de derechos y satisfacer él mismo que los datos de derechos permiten la ejecución de tal contenido.

Una vez nuevamente, como un resultado de tener la licencia de publicador §20, el usuario publicador puede continuar trabajando con el contenido 310 publicado fuera de línea así mientras permanezca fuera de línea mientras que el servidor DRM 320 no requiere ser contactado.

#### Matriculado y Submatriculado de los Servidores DRM

En la arquitectura como se ve en la figura 3, solamente un servidor DRM único 320 se muestra. Sin embargo, como puede ser apreciado, tal arquitectura puede y parece incluir múltiples servidores DRM 320. En particular, y en una modalidad de la presente invención, tal arquitectura incluye una red distribuida de servidores DRM 320. Cada

uno de tales servidores DRM 320 puede tener cualquier función particular y todos los servidores DRM 320 pueden estar organizados de una manera apropiada sin salirse del espíritu y alcance de la presente invención.

Por ejemplo, y volviendo ahora a la figura 12, una organización particular puede tener uno o más servidores DRM de nivel de usuario 320 para el propósito de firmar las etiquetas de derechos para producir SRL 308, otorgar licencias 16, otorgar licencias de publicación 320, otorgar certificados a los usuarios, emitir certificados a dispositivos de cómputo 14, y similares. Cada uno de tales servidores DRM de nivel de usuario 320 pueden estar asignados geográficamente o pueden ser asignados con base en función o carga, por ejemplo. Similarmente, para sobremirar los múltiples servidores DRM de nivel de usuario 320, una organización puede tener uno o más servidores DRM de administración 320. Tales servidores DRM con base en organización 320 pueden estar localizados detrás de una barrera de organización si así se desea.

En adición a los servidores DRM con base en la organización 320, puede haber DRM trans-organización 320 que facilitan la funcionalidad DRM inter-organización.

Por ejemplo, tales servidores DRM trans-organización 320 pueden permitir a un par de organizaciones compartir cierto contenido DRM 12. También, puede haber una red de servidores DRM veedores 320 que puedan supervisar y mantener todos los otros servidores DRM 320 y suministrar enlaces apropiados para todos los otros servidores DRM 320 de nuevo a la raíz o a la autoridad confiada que es la base para la cadena del conjunto de certificados establecidos previamente. Tales servidores DRM con base en no organizaciones 320 no son fáciles de localizar detrás de cualquier barrera de organización.

Críticamente, cada servidor DRM 320 en la arquitectura de la figura 12 debe ser capaz de probar que puede ser confiado. Así, como debería ahora apreciarse a partir de la cadena anterior de discusión de certificados, cada servidor DRM 320 luego de entrar la arquitectura es provisto con un certificado de matrícula 1310, como se ve en la figura 13. Significativamente, y en una modalidad de la presente invención, el certificado de matrícula 1310 está provisto al ingresar el servidor DRM 320 (de aquí en adelante "el servidor DRM-E 320") por otro servidor DRM "de matriculado" 320 ya existente en la arquitectura (de aquí en adelante "el servidor DRM-R

320"). También significativamente, el sujeto al certificado de matriculado provisto 1310 del servidor DRM-R de matriculado 320 está una cadena de certificados 1320 que incluyen el certificado de matrícula 1310 del servidor DRM 320 matriculado, el certificado 1310 de matrícula del servidor DRM 320 que matriculó el servidor DRM-R matriculado 320, y así en adelante todo el trayecto hacia atrás hasta el servidor DRM raíz 320. Tal servidor DRM de raíz 320 puede representar la raíz o la autoridad confiada, o la cadena de certificados 1320 puede extenderse más allá para alcanzar la raíz o la autoridad confiada. Como puede apreciarse ahora, tal certificado de matrícula 1310 y la cadena de certificados 1320 en combinación forman la cadena de certificados que están ligados o sujetos a un certificado OLP 810 provisto por un servidor DRM-E que ha ingresado o matriculado 320 a un usuario de publicación tal como fue mostrado en la figura 8.

En una modalidad de la presente invención, el certificado de matriculado 1310 provisto a un servidor DRM-E 320 por un servidor DRM-R 320 está en una forma tal como un certificado con base en XrML 1.2. Como puede ser apreciado, tal tipo de certificado 1310 no está propuesto

por ningún tercer partido, y por lo tanto tal tipo de certificado 1310 no representa ninguna clase de garantía independiente por un tercer partido para el poseedor de tal certificado 1310.

En una modalidad de la presente invención, el método con el cual un servidor DRM-E particular 320 se ha matriculado en la arquitectura depende de si el servidor DRM-R de matrícula 320 conoce o tiene razón para confiar en el ingreso del servidor DRM-E 320. Si no, el servidor DRM-E 320 debería ser requerido para que pruebe al servidor DRM-R 320 que es confiable y que cumplirá con la arquitectura DRM. Si es así, el servidor DRM-E 320 no debería ser requerido para probar al servidor DRM-R 320 de que es confiable, al menos no hasta el mismo grado. Así un servidor DRM-R 320 no confiado/sin conocimiento "matricula" un servidor DRM-E 320, mientras que un servidor DRM-R que conoce/confía 320 "submatricula" a un servidor DRM-E 320.

Típicamente, un servidor DRM-R 320 conoce/confía en un servidor DRM-E 320 si ambos están operados por o para el beneficio de la misma organización, aunque el conocimiento/confianza también puede surgir de otras

situaciones sin salirse del espíritu y alcance de la presente invención. Así, el método con el cual un servidor DRM-E particular 320 que es matriculado en la arquitectura típicamente depende en si el servidor DRM-R 320 de matriculado está basado organizacionalmente o basado no organizacionalmente. Como un resultado, un servidor DRM-R 320 basado en no-organizacionalmente "matricula" un servidor DRM-E 320, mientras que un servidor DRM-R 320 basado organizacionalmente "submatricula" un servidor DRM-E 320.

#### Matriculado

En una modalidad de la presente invención, y volviendo ahora a la figura 14, un servidor DRM-R 320 sin conocimiento/sin confiar matricula un servidor DRM-E 320 de la siguiente manera.

Preliminarmente, se podrá apreciar que el servidor DRM-e 320 que desea ser matriculado por el servidor DRM-R 320 que no tiene conocimiento/ni confianza es probable que conozca a tal servidor DRM-R 320. De acuerdo a lo anterior, y en una modalidad de la presente invención, el servidor DRM-E 320 debe procurar un certificado de

garantía 1330 de un tercer partido que desee dar garantía para tal servidor DRM-E 320 (etapa 1401). Típicamente, tal tercer partido es un agente otorgador de certificados independiente que es confiado por el servidor DRM-R 320 para llevar a cabo tal garantía, tal como por ejemplo VERISIGN Corporation of Mountain view, California. Tal certificado de garantía 1330 puede por ejemplo estar en una forma tal como un certificado X.509. Note que en el servidor DRM-R 320 que descansa en la confianza de la tercera parte para presentar garantía frente al servidor DRM-E 320, la responsabilidad de tal servidor DRM-R 320 por cualquier acto malo del servidor DRM-E 320 se mitiga.

Como debería apreciarse y como es típico, y como también se puede ver en la figura 13, el certificado de garantía 1330 incorpora allí una clave pública (PU-V) y una clave privada correspondiente (PR-V), es firmada por la tercera parte confiada, y puede ser acompañado por una cadena de certificados que llevan a una raíz de conocimiento para propósitos de validación. Como también es típico, el (PR-V) dentro del certificado de garantía 1330 es protegido de una manera accesible al servidor DRM-E 320 para garantía que es la base del certificado de garantía 1330. Por ejemplo, y como puede verse en la figura 13, el (PR-

V) podría estar encriptado de acuerdo con una clave pública apropiada.

Dentro de la arquitectura DRM, el servidor DRM-E 320 que ingresa debe tener una única identidad. Aquí debe entenderse que la identidad del DRM es muy diferente de (PU-V, PR-V), aunque la identidad del DRM puede también coincidir con tal (PU-V, PR-V) sin salirse del espíritu y el alcance de la presente invención. De acuerdo a lo anterior, para establecer tal identidad, tal servidor DRM-E 320 genera u obtiene un nuevo par de claves pública/privada (PU-E, PR-E) (etapa 1403). También, dentro de la arquitectura DRM, el servidor DRM-E 320 que se matricula debe decidir que entidades pueden revocar la autoridad del mismo para participar. De acuerdo a lo anterior, tal servidor DRM-E 320 identifica a cada una de tales entidades de revocación en una lista, quizás por vía de una clave pública del mismo (etapa 1405).

El servidor DRM-E 320 debería ser capaz de establecer al servidor DRM-R 320 que matricula que tal servidor DRM-E de hecho conoce el certificado de garantía 1330 que fue obtenido en la etapa 1401. De acuerdo a lo anterior, el servidor DRM-E 320 sea que emplee el (PR-V) del

certificado de garantía 1330 para encriptar el (PU-E) para resultar en el (PR-V(PU-E)) como un indicativo de propiedad, o firma el (PU-E) con el (PR-V) para resultar en el (PU-E)S(PR-V) como un indicio de su propiedad (etapa 1407). En cualquier caso, aplicar el (PU-V) para desencriptar el (PU-E) o para verificar la firma que establece la posición del (PR-V) y por lo tanto el certificado de garantía 1330.

Hasta ahora, el servidor DRM-E 320 tiene el certificado de garantía 1330, el (PU-E) y el (PR-E), la lista de autoridad de revocación, y el (PR-V(PU-E)) o el (PU-E)S(PR-V) como indicios de su propiedad. Para solicitar la matrícula, entonces, tal servidor DRM-E 320 envía el certificado de garantía 1330, el (PU-E), la lista de autoridad de revocación, y el (PR-V(PU-E)) o el (PU-E)S(PR-V) como el indicio de propiedad al servidor DRM-R 320 (etapa 1409), y el servidor DRM-R 320 procede a matricular tal servidor DRM-E 320 que lo solicita. Note que la solicitud o parte de la misma puede estar en la forma de un certificado firmado por (PR-E).

En particular, el servidor DRM-R 320 valida el certificado de garantía 1330 con base en la firma del

mismo por la tercera parte confiada y la cadena de certificados que llevan a la raíz conocida (etapa 1411). Así, el servidor DRM-R 320 establece que el servidor DRM-E 320 ha sido garantizado para esto. También, el servidor DRM-R 320 verifica los indicios de propiedad aplicando el (PU-v) de la solicitud sea para descryptar el (PU-E) o para verificar la firma y así establecer la posición del (PR-V) y por lo tanto el certificado de garantía 1330 en la solicitud (1410). En adición, y significativamente, el servidor DRM-R 320 lleva a cabo cualquier lógica adaptada necesaria para decidir si acepta la solicitud (etapa 1413). Tal lógica adaptada puede ser cualquier lógica apropiada sin salirse del espíritu y alcance de la presente invención, y puede por ejemplo incluir una verificación de antecedentes sobre el servidor DRM-E 320 y/o su operador, una determinación de si el servidor DRM-E 320 tiene un componente confiado actual 18 y/o un sistema operativo o similar, una determinación de si el servidor DRM-E 320 está en una lista de revocación u otra lista de vigilancia, y similares.

Asumiendo que la lógica adaptada permite que la solicitud sea aceptada, entonces, y en una modalidad de la presente invención, el servidor DRM-R 320 genera el certificado de

matrícula 1310 para el servidor DRM-E 320 (etapa 1415). En particular, y como se ve en la figura 13, el servidor DRM-R 320 incorpora dentro del certificado de matrícula 1310:

- un identificador del servidor DRM-R 320, tal como una clave pública del mismo (PU-R);
- un identificador del servidor DRM-E 320, tal como el (PU-E);
- indicios de identificación del certificado de garantía 1330 que incluye la tercera parte confiada que otorgó el mismo, un número serie del certificado de garantía 1330, y la otorgación como identificado dentro del certificado de garantía 1330;
- cualquier información de rango de validez que especifique un rango durante el cual el certificado de matrícula 1310 es válido, tal como por ejemplo un rango de fecha;
- la lista de autoridad de revocación;
- una firma con base en una clave privada del servidor DRM-R 320 (PR-R) correspondiente a el (PU-R);

- y cualquier otra información apropiada.

Tal otra información apropiada puede incluir pero no estar limitada a: el tiempo en que el certificado fue otorgado o emitido; una indicación de que clase de actividades DRM un servidor matriculado se le permite llevar a cabo tal como por ejemplo todas las actividades, activación de contabilidad solamente, firma de etiquetas de derechos solamente, emitir licencias de contenido solamente, y combinaciones de ellas; y un rango de tiempo permitido para llevar a cabo las actividades DRM. Note que el rango de tiempo permitido es diferente del rango de validez en que el tiempo actual está dentro del rango de validez para aceptar cualquier certificado que incluya el certificado de matriculado 1310 en la cadena de certificado. En contraste, el tiempo permitido de certificados niño debe caer dentro del rango del tiempo permitido del certificado pariente para llevar a cabo las actividades DRM.

Como puede ser apreciado, en la generación del certificado de matrícula 1310, el servidor DRM-R 320 puede generar inicialmente información de certificado y

entonces permitir que la lógica adaptada para generar información adicional o modificar la información existente. Tal lógica adaptada puede por ejemplo asegurar que el servidor DRM-R 320 incluya la información apropiada, o puede hacer cumplir la política de arquitectura DRM predefinida. Por supuesto, la firma del certificado de matrícula 1310 está creada después de que cualquier de tal lógica adaptada se lleve a cabo. Como podrá apreciarse, el servidor DRM-R 320 sujeta la cadena de certificados 1320 que lleva de nuevo a la autoridad de raíz confiada al certificado de matrícula generado 1310 de modo que el certificado de matrícula 1310 generado pueda ser validado con base en tal cadena de certificados 1320.

Note en particular que las indicaciones de identificación del certificado de garantía 1330 tal como se colocaron dentro del certificado de matrícula 1310 siempre viajan con tal certificado de matrícula 1310 y actúan como un puente al certificado de garantía 1330. Así, y nuevamente, tales indicios de identificación muestran al mundo que el servidor DRM-R 320 descansa en una tercera parte confiada otorgadora del certificado de garantía 1330 para proponer el servidor DRM-E 320, y la

responsabilidad de tal servidor DRM-R 320 por cualquier acto malo del servidor DRM-E 320 es mitigado.

Una vez que el servidor DRM-r 320 ha generado de manera exitosa el certificado de matrícula 1310 con la cadena de certificados 1320 sujeta, el servidor DRM-R 320 entonces regresa el mismo al servidor DRM-E 320 que lo solicita (etapa 1417), y el servidor DRM-E 320 nuevo matriculado almacena el mismo en una localización apropiada para un uso futuro (etapa 1419). Como fue aludido anteriormente, el (PU-E) en el certificado de matrícula 1310 y el (PR-E) correspondiente son el par de claves pública/privada que el servidor DRM-E 320 usará como (PU-DRM) y (PR-DRM) cuando firma etiquetas de derechos para producir un SRL 308, emitiendo un certificado OLP 810, o de otra manera participando dentro de la arquitectura DRM. De acuerdo a lo anterior, tal certificado de matrícula 1310 y la cadena de certificados 1320 en combinación forman la cadena de certificados que está sujeta a tal certificado OLP 810 y similar.

SubMatrícula

En una modalidad de la presente invención, y volviendo ahora a la figura 15, un servidor DRM-R 320 que conoce/confía submatricula un servidor DRM-E 320 de la siguiente manera.

Preliminarmente, debe apreciarse que el servidor DRM-E 320 que desea ser submatriculado por el servidor DRM-R 320 que conoce/confía debería ser todavía requerido para que se identifique el mismo frente a tal servidor DRM-R 320 ya que tal conocimiento o confianza puede no ser completa. Sin embargo, tal requerimiento de identificación no requiere que surja al nivel de una propuesta por una tercera parte confiada ni tampoco que el servidor DRM-R 320 tenga conocimiento/confianza del servidor DRM.-E. De acuerdo a lo anterior, y en una modalidad de la presente invención, el servidor DRM-E 320 obtiene o es suministrado con alguna clase de credenciales 1340 (figura 13) que son reconocibles por y se espera que sean aceptadas por el servidor DRM-R 320, y que identifique el servidor DRM-E 320 a satisfacción del servidor DRM-R 320(etapa 1501).

Si ambos los servidores DRM-R y DRM-E 320 están dentro de la misma organización, tales credenciales 1340 pueden ser

las credenciales con base en la organización, tal como por ejemplo un ID de red si ambos servidores 320 están en una red común, un ID de dominio si ambos servidores 320 comparten un dominio común, o similar. Si ambos servidores DRM-R y DRM-E 320 no están dentro de la misma organización, tales credenciales 1340 pueden todavía ser un ID de red si ambos servidores 320 están sobre una red común, un ID de dominio si ambos servidores 320 comparten un dominio común, o similar, o pueden ser otras credenciales tales como por ejemplo credenciales emitidas por un tercer partido y reconocidas por el servidor DRM-R 320.

Note que en la presente situación, el servidor DRM-R 320 no descansa en un tercer partido confiado para proponer para el servidor DRM-E 32, y por lo tanto la responsabilidad de tal servidor DRM-R 320 por cualquier mal acto del servidor DRM-E 320 no se mitiga. Sin embargo, el servidor DRM-R 320 está deseando llevar a cabo o tomar un riesgo con base en el conocimiento de o la confianza en el servidor DRM-E 320 con respecto a que se lleven a cabo hechos o malos actos.

Tal como se mencionó anteriormente, dentro de la arquitectura DRM, el servidor DRM-E 320 que ingresa debe tener una identidad única. Aquí debe entenderse que la identidad DRM está un poco separada de las credenciales 1340, aunque la identidad DRM puede también coincidir con las credenciales 1340 sin salirse del espíritu y alcance de la presente invención. De acuerdo a lo anterior, para establecer tal identidad, tal servidor DRM-E 320 genera u obtiene un par de claves nuevas pública/privada (PU-E, PR-E) (etapa 1503). También como se menciona anteriormente, dentro de la arquitectura DRM, el servidor DRM-E 320 submatriculado debe decidir que entidades pueden revocar la autoridad del mismo para participar. De acuerdo a lo anterior, tal servidor DRM-E 320 identifica cada uno de tales entidades de revocación en una lista, quizás por vía de una clave pública del mismo (etapa 1505).

Hasta ahora, el servidor DRM-E 320 tiene las credenciales 1340, el (PU-E) y el (PR-E), y la lista de autoridad de revocación. Para solicitar la submatrícula, entonces, tal servidor DRM-E 320 envía las credenciales 1340, el (PU-E), y la lista de autoridades de revocación al servidor DRM-R 320 (etapa 1507), y el servidor DRM-R 320 procede a submatricular tal servidor DRM-E 320 que lo solicita.

Note como se dijo anteriormente, la solicitud o parte de la misma puede estar en la forma de un certificado firmado por el (PR-E).

En particular, el servidor DRM-r 320 valida las credenciales 1340 con base en cualesquiera lógica o recursos que sean necesarios y estén disponibles para de esa manera validar (etapa 1509). Así, el servidor DRM-R 320 establece con base en las credenciales validadas 1340 que el servidor DRM-E 320 debe ser confiado para aceptar y obedecer la arquitectura DRM. En adición, y como se mencionó anteriormente, el servidor DRM-R 320 lleva a cabo cualquier lógica adaptada necesaria para decidir si acepta la solicitud (etapa 1411).

Asumiendo que la lógica adaptada permite que la solicitud sea aceptada, entonces, y en una modalidad de la presente invención, el servidor DRM-R 320 genera un certificado de submatrícula 1310 para el servidor DRM-E 320 (etapa 1513). En particular, y como se ve en la figura 13, el servidor DRM-R 320 incorpora dentro del certificado de submatrícula 1310:

- un identificador del servidor DRM-R 320 tal como una clave pública del mismo (PU-R);
- un identificador del servidor DRM-E 320, tal como (PU-E);
- las credenciales 1340 o una referencia a ellas;
- cualquier información de rango de validez que especifique un rango durante el cual el certificado de submatrícula 1310 es válido, tal como por ejemplo un rango de fechas;
- la lista de autoridades de revocación;
- una firma con base en una clave privada del servidor DRM-R 320 (PR-R) correspondiente a un (PU-R);
- y cualquier otra información apropiada.

Como se mencionó, al generar el certificado de submatrícula 1310, el servidor DRM-R 320 puede inicialmente generar información del certificado y entonces permitir que a la lógica adaptada generar información adicional o modificar la información existente. Nuevamente, la firma del certificado de submatrícula 1310 es creada después de que cualquiera de tal lógica adaptada se lleve a cabo. Como se mencionó, el

servidor DRM-R 320 sujeta la cadena de certificados 1320 que lleva de nuevo a la autoridad de raíz confiada al certificado de submatrícula generado 1310 de modo que el certificado de submatrícula generado 1310 puede ser validado con base en tal cadena de certificados 1320.

Note aquí que las credenciales 1340 o la referencia a ellas no se cree que sean necesarias especialmente, pero pueden sin embargo ser incluidas para completar la información. Note también que el certificado de submatrícula 1310 no contiene ningún indicio de identificación de un certificado de garantía 1330 ya que como no se requirió un certificado de garantía en el presente escenario de submatrícula.

Una vez que el servidor DRM-R 320 ha generado exitosamente el certificado de submatrícula 1310 con la cadena de certificados 1320 sujeta, el servidor DRM-R 320 regresa entonces el mismo al servidor DRM-E 320 que lo solicitó(etapa 1515), y el servidor DRM-E 320 ahora nuevo submatriculado almacena el mismo en una localización apropiada para su uso futuro (etapa 1517). Como se dijo, el (PU-E) en el certificado de submatrícula 1310 y el (PR-E) correspondiente son el par de claves

pública/privada que el servidor DRM-E 320 usará como (PU-DRM) y el (PR-DRM) cuando firme etiquetas de derechos para producir un SRL 308, emita un certificado OLP 810, y de otra manera participe dentro de la arquitectura DRM. De acuerdo a lo anterior, tal certificado de submatrícula 1310 y la cadena de certificados 1320 en combinación forman la cadena de certificados que está sujeta a tal certificado OLP 810 y similares.

### Conclusión

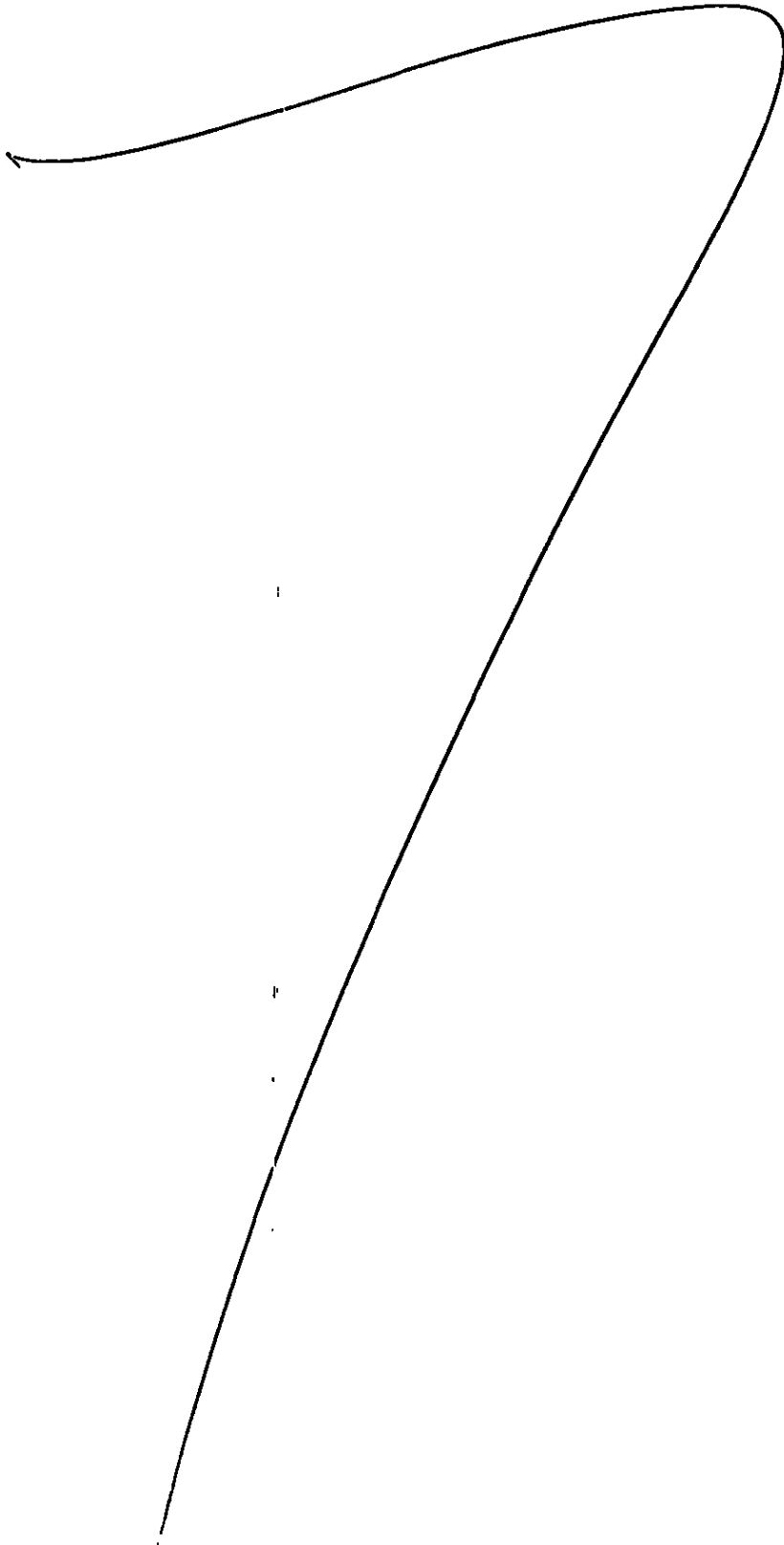
La programación necesaria para efectuar los procesos llevados a cabo en conexión con la presente invención es relativamente directa y debería ser evidente al público de programación relevante. De acuerdo a lo anterior, tal programación no se adjunta aquí. Cualquier programación particular, entonces, puede ser empleada para efectuar la presente invención sin salirse del espíritu y alcance de la misma.

En la presente invención, una arquitectura para administración de derechos digitales (DRM) y hacerlos cumplir y el método que permita la visualización controlada de las formas arbitrarias de contenido

digital, en donde tal control es flexible y definible por el propietario del contenido/desarrollador de tal contenido digital. La arquitectura permite y facilita que tal ejecución controlada, especialmente en una organización u oficina, y similar en donde los documentos van a ser compartidos entre un grupo definido de individuos o clases de individuos. Tal arquitectura incluye un mecanismo para matricular/submatricular servidores DRM 320 que se les otorga aprobación dentro de la arquitectura.

Debería apreciarse que varios cambios pueden hechos a las modalidades descritas anteriormente sin salirse de los conceptos inventivos de la misma. Por ejemplo, si una licencia o una etiqueta de derechos es firmada con base en los datos de derechos allí indicados, tales datos de derechos no requieren necesariamente estar encriptados. Similarmente, en la solicitud y construcción de un certificado de matrícula o submatrícula 1310, la lista de autoridades de revocación y otra información similar no necesariamente requiere ser empleada. Debe entenderse, por lo tanto, que esta invención no está limitada a las modalidades particulares descritas, si no que se tiene el propósito de cubrir tales modificaciones dentro del

espíritu y alcance de la presente invención según se definió en las reivindicaciones adjuntas.



## REIVINDICACIONES

1. Un método en combinación con un sistema de administración de derechos digitales (DRM) que tiene una pluralidad de servidores DRM que llevan a cabo la funcionalidad DRM, el método para que un servidor DRM-E que ingresa sea matriculado en el sistema por un servidor DRM-R que matricula tal que el servidor DRM-E que ingresa sea confiado dentro del sistema y comprende:

- el servidor DRM-E procura un par de claves pública/privada (PU-E, PR-E) para identificar tal servidor DRM-E dentro del sistema DRM;
- el servidor DRM-E procura una identificación de propuesta del mismo;
- el servidor DRM-E envía una solicitud de matrícula al servidor DRM-R, la solicitud incluye la identificación de propuesta y el (PU-E);
- el servidor DRM-R valida la identificación de propuesta;

no es  
patente

no es el  
dato  
se  
relaciona

X

X

X

X

X

2

- el servidor DRM-R, si la solicitud ha sido aceptada, genera un certificado de matrícula digital para el servidor DRM-E para matricular tal servidor DRM-E dentro del sistema DRM, el certificado de matrícula generado está basado en por lo menos una parte en el (PU-E);
- el servidor DRM-R regresa el certificado de matrícula generado al servidor DRM-E solicitante; y
- el ahora servidor DRM-E matriculado almacena el certificado de matrícula regresado en una localización apropiada para su uso futuro, el servidor DRM-E con el certificado de matrícula es capaz de emplear el mismo para emitir documentos de DRM dentro del sistema DRM.

2. El método de la reivindicación 1, en donde el servidor DRM-R no tiene base existente para confiar en el servidor DRM-E, el método comprende:

- El servidor DRM-E procura una identificación de propuesta del mismo que comprende un certificado de garantía de una parte que desea garantizar tal

servidor DRM-E, el certificado de garantía incorpora en él una clave pública (PU-V) y una clave privada correspondiente (PR-V);

- El servidor DRM-E emplea el (PU-E) y el (PR-V) para formular indicios de propiedad para mostrar que el servidor DRM-E posee el certificado de garantía;

- El servidor DRM-E envía una solicitud de matrícula al servidor DRM-R, la solicitud incluye el certificado de garantía, el (PU-E) y los indicios de propiedad;

- El servidor DRM-R valida el certificado de garantía;

- El servidor DRM-R verifica los indicios de propiedad; y

- El servidor DRM-R si la solicitud va a ser aceptada, genera un certificado de matrícula digital para el servidor DRM-E para matricular tal servidor DRM-E dentro del sistema DRM, el certificado de matricula se ha generado en base por lo menos en parte sobre el significado de garantía y el (PU-E).

un 98  
no  
com  
4/15

3. El método de la reivindicación 2 que comprende el servidor DRM-E que procura el certificado de garantía de un agente otorgador de certificados independiente que es confiable y descansa en el servidor DRM-R para llevar a cabo tal garantía.

4. El método de la reivindicación 2 que comprende el servidor DRM-E que procura un certificado de garantía tipo X.509.

5. El método de la reivindicación 2 que comprende:

- El servidor DRM-E procura el certificado de garantía firmado por la parte garantizadora y acompañado por una cadena de certificados que llevan a una raíz conocida para propósitos de validación;

- El servidor DRM-R valida el certificado de garantía con base en la firma en el mismo por la parte garantizadora y la cadena de certificados para establecer que el servidor DRM-E ha sido garantizado.

6. El método de la reivindicación 2 comprende:

- El servidor DRM-E desempeña uno de empleo (PR-V) para encriptar (PU-E) para resultar en (PR-V(PU-E)) como lo indica el propietario, o firmar (PU-E) con (PR-V) para resultar en (PU-E)S(PR-V) como lo indica el propietario; y
- El servidor DRM-R verifica lo que indica el propietario mediante la aplicación de (PU-V) de la solicitud para desencriptar (PU-E) o verificar la firma para establecer que el propietario del servidor DRM-E (PR-V) y por lo tanto el certificado de garantía.

7. El método de la reivindicación 2 comprende generar el certificado de matrícula del servidor DRM-R para incluir (PU-E) como un identificador del servidor DRM-E, identificar el indicio para identificar el certificado de garantía, y una firma basada en una clave privada del servidor DRM-R, por lo tanto el indicio de identificación para el certificado de garantía en el certificado de matrícula actúa como

sin  
antes  
de  
log

un puente al certificado de garantía y muestra que el servidor DRM-R es auténtico y confiable en la parte de garantía para garantizar al servidor DRM-E.

8. El método de la reivindicación 7 comprende el servidor DRM-R que genera el certificado de matrícula que incluye adicionalmente una clave pública del servidor DRM-R como un identificador de éste.

9. El método de la reivindicación 7 comprende el servidor DRM-R que genera el certificado de matrícula para incluir adicionalmente un rango válido de información que especifica un rango durante el cual el certificado de matrícula se valida.

10. El método de la reivindicación 1 en donde el servidor DRM-R tiene una base existente para confiar el servidor DRM-E, el método comprende:

- el servidor DRM-E que obtiene una identificación que ofrece de ésta comprende credenciales reconocibles a y

se espera sean estimadas por el servidor DRM-R;

- el servidor DRM-E envía una solicitud de matrícula al servidor DRM-R, la solicitud incluye las credenciales y (PU-E);
- el servidor DRM-R valida las credenciales; y
- el servidor DRM-R si la solicitud no se estima, genera un certificado de matrícula digital para el servidor DRM-E para matrícula tal como el servidor DRM-E en el sistema DRM, el certificado de matrícula generado se basa al menos en parte en las credenciales y (PU-E).

11. El método de la reivindicación 10 comprende el servidor DRM-E que procura credenciales seleccionadas de un grupo que consiste de un ID de red o un ID de dominio, y credenciales publicadas por una tercera parte.

12. El método de la reivindicación 10 comprende el servidor DRM-R que genera el certificado de

matrícula para incluir (PU-E) como un identificador del servidor DRM-E, que identifica el indicio para identificar las credenciales, y una firma basada en una clave privada del servidor DRM-R.

13. El método de la reivindicación 12 comprende el servidor DRM-R que genera el certificado de matrícula para incluir además una clave pública del servidor DRM-R como un identificador de ésta.

14. El método de la reivindicación 12 comprende el servidor DRM-R que genera el certificado de matrícula para incluir adicionalmente un rango de validez de información que especifica un rango durante el cual el certificado de matrícula se valida.

15. El método de la reivindicación 1 comprende además el servidor DRM-R que desempeña una lógica adecuada para decidir si estima lo solicitado.

16. El método de la reivindicación 15 comprende el servidor DRM-R que desempeña una lógica adecuada seleccionada de un grupo que consiste de desempeñar

una revisión de antecedente sobre el servidor DRM-E y/o su operador, determinar si el servidor DRM-E y/o una porción de éste es actual, determinar si el servidor DRM-E está en una lista de revocación o mirar la lista, y combinaciones de esto.

17. El método de la reivindicación 1 comprende el servidor DRM-R que genera el certificado de matrícula para incluir (PU-E) como un identificador del servidor DRM-E, y una firma basada en una clave privada del servidor DRM-R.

18. El método de la reivindicación 17 comprende el servidor DRM-R que genera el certificado de matrícula para incluir adicionalmente una clave pública del servidor DRM-R como un identificador de ésta.

19. El método de la reivindicación 17 comprende el servidor DRM-R que genera el certificado de matrícula para incluir adicionalmente un rango de validez de información que especifica un rango durante el cual el certificado de matrícula se valida.

20. El método de la reivindicación 17 comprende el servidor DRM-R que genera el certificado de matrícula para incluir adicionalmente un indicio de identificador para identificar la identificación que se ofrece.

21. El método de la reivindicación 1 comprende el servidor DRM-R que genera el certificado de matrícula mediante el empleo de una lógica adecuada para generar al menos una porción de la información en el certificado de matrícula.

22. El método de la reivindicación 1 comprende además el servidor DRM-R que adhiere al certificado de matrícula generado una cadena de certificados que se vuelve a una autoridad de raíz confiada de tal manera que el certificado de matrícula generado puede ser validado basado en tal cadena de certificados.

23. El método de la reivindicación 1 comprende además el servidor DRM-E que identifica una lista de autoridad de revocación al menos una entidad con una

autoridad para revocar la matrícula de tal servidor DRM-E dentro del sistema DRM, que comprende el servidor DRM-E que envía una solicitud de matrícula al servidor DRM-R, la solicitud incluye la identificación que ofrece, (PU-E), y la lista de autoridad de revocación, y comprende el servidor DRM-R, si la solicitud se estima, generar un certificado de matrícula para el servidor DRM-E para matricular tal servidor DRM-E en el sistema de servidor DRM, el certificado de matrícula generado se basa en al menos una parte del (PU-E) y la lista de autoridad de revocación.

24. El método de la reivindicación 23 que comprende el servidor DRM-E que identifica cada entidad en la lista de autoridad de revocación por medio de una clave pública de ésta.

25. El método de la reivindicación 23 que comprende el servidor DRM-R que genera el certificado de matrícula para incluir (PU-E) como un identificador del servidor DRM-E, la lista de autoridad de revocación de la solicitud, y la firma basada en una clave privada del servidor DRM-R.

26. El método de la reivindicación 1 comprende el servidor DRM-R que genera un certificado de matrícula XrML.

27. Un método en combinación con un sistema de Administración de Derechos Digitales (DRM) que tiene una pluralidad de servidores DRM que desempeñan una funcionalidad DRM, el método para entrar el servidor DRM-E para ser matriculado en el sistema mediante un servidor DRM-R de matrícula tal como el servidor DRM-E entrante se confía dentro del sistema y comprende:

- el servidor DRM-E que consigue un par de claves pública/privada (PU-E, PR-E) para identificar tal servidor DRM-E dentro del sistema DRM;
- el servidor DRM-E que consigue una identificación que ofrece de ésta;
- el servidor DRM-E envía una solicitud de matrícula al servidor DRM-R, la solicitud incluye la identificación que ofrece y (PU-E), el servidor DRM-R que

valida la identificación que ofrece, si la solicitud se estima, que genera un certificado de matrícula digital para el servidor DRM-E para matricular tal servidor DRM-E en el sistema DRM, el certificado de matrícula generado se basa al menos en parte en (PU-E), y regresa el certificado de matrícula generado al servidor DRM-E solicitante; y

el servidor DRM-E ahora matriculado almacena el certificado de matrícula devuelto en una ubicación apropiada para uso futuro, el servidor DRM-E con el certificado de matrícula capaz de emplear el mismo para publicar documentos DRM dentro del sistema DRM.

28. El método de la reivindicación 27 en donde el servidor DRM-R no tiene una base existente para el servidor DRM-E confiado, el método comprende:

el servidor DRM-E que busca una identificación que ofrece de ésta

comprende un certificado de garantía de una parte que quier garantizar para tal servidor DRM-E, el certificado de garantía incorpora en éste una clave pública (PU-V) y una clave privada correspondiente (PR-V);

- el servidor DRM-E que emplea (PU-E) y (PR-V) para formular indicios de propiedad para mostrar que el servidor DRM-E pertenece al certificado de garantía;

- el servidor DRM-E envía una solicitud de matrícula al servidor DRM-R, la solicitud incluye el certificado de garantía, (PU-E), y el indicio de propietario, el servidor DRM-R valida el certificado de garantía, verificando el indicio de propietario; y, si la solicitud se estima , generar un certificado de matrícula para el servidor DRM-E para matricular tal servidor DRM-E dentro del sistema DRM, el certificado de matrícula generado se

basa al menos en parte en el certificado de garantía y (PU-E).

29. El método de la reivindicación 28 comprende el servidor DRM-E que busca el certificado de garantía de un agente de publicación de certificado independiente que se confía mediante y confiable para el servidor DRM-R para que desarrolle tal garantía.

30. El método de la reivindicación 28 que comprende el servidor DRM-E que procura un certificado de garantía X.509.

31. El método de la reivindicación 28 comprende el servidor DRM-E que procura que el certificado de garantía firmado por la parte garantizadora y acompañado por una cadena de certificados hacia una raíz conocida para propósitos de validación, el servidor DRM-R que valida el certificado de garantía basado en la firma de éste mediante una parte garantizadora y la cadena de certificados para establecer que el servidor DRM-E está garantizando.

32. El método de la reivindicación 28 comprende el servidor DRM-E que desempeña uno de emplear (PR-V) para encriptar (PU-E) para resultar (PR-V(PU-E)) como el indicio del propietario, o firmar (PU-E) con (PR-V) para resultar en (PU-E)S(PR-V) como lo indica el propietario, el servidor DRM-R verifica el indicio del propietario mediante aplicar (PU-V) desde la solicitud para desencriptar (PU-E) o verificar la firma para establecer que el servidor DRM-E pertenece (PR-V) y por lo tanto certifica la garantía.

33. El método de la reivindicación 28 comprende el servidor DRM-E ahora matriculado que almacena el certificado de matrícula devuelto que incluye (PU-E) como un identificador del servidor DRM-E, que identifica el indicio para identificar el certificado de garantía, y una firma basada en una clave privada del servidor DRM-R mediante la identificación del indicio para el certificado de garantía en los actos de certificado de garantía como un puente para certificar la garantía y mostrar que el servidor DRM-R es confiable y descansa en la

parte de garantía para garantizar para el servidor DRM-E.

34. El método de la reivindicación 33 comprende el servidor DRM-E ahora matriculado que almacena el certificado de matrícula devuelto que además incluye una clave pública del servidor DRM-R como un identificador de éste.

35. El método de la reivindicación 33 que comprende el servidor DRM-E ahora matriculado que almacena el certificado de matrícula devuelto que incluye además una información de rango válida que especifica un rango durante el cual el certificado de matrícula se valida.

36. El método de la reivindicación 27 en donde el servidor DRM-R tiene una base existente para confiar el servidor DRM-E, el método comprende:

- el servidor DRM-E busca una identificación de oferta de éste que comprende credenciales reconocibles a y

se espera sea estimado por el servidor

DRM-R;

- el servidor DRM-E que envía una solicitud de matrícula al servidor DRM-R, la solicitud incluye las credenciales y (PU-E), el servidor DRM-R que valida las credenciales, y, si la solicitud se estima, generar un certificado de matrícula digital para el servidor DRM-E para matricular tal servidor DRM-E en el sistema DRM, el certificado de matrícula generado se basa al menos en parte en las credenciales y (PU-E).

37. El método de la reivindicación 36 que comprende el servidor DRM-E que busca credenciales seleccionadas de un grupo que consiste de una ID de red o un ID de dominio, y credenciales publicadas por una tercera parte.

38. El método de la reivindicación 36 que comprende el servidor DRM-E ahora matriculado que almacena el certificado de matrícula devuelto que incluye (PU-E) como un identificador del servidor DRM-E, que

identifica el indicio para identificar las credenciales, y una firma basada en una clave privada del servidor DRM-R.

39. El método de la reivindicación 38 que comprende el servidor DRM-E que almacena el certificado de matrícula devuelto que incluye además una clave pública del servidor DRM-R como un identificador de éste.

40. El método de la reivindicación 38 que comprende el servidor DRM-E ahora matriculado que almacena el certificado de matrícula devuelto que incluye además una información de rango válido que especifica un rango durante el cual el certificado de matrícula se valida.

41. El método de la reivindicación 27 que comprende el servidor DRM-E ahora matriculado que almacena el certificado de matrícula devuelto que incluye (PU-E) como un identificador del servidor DRM-E, y una firma basada en una clave privada del servidor DRM-R.

42. El método de la reivindicación 41 que comprende el servidor DRM-E ahora matriculado que almacena un certificado de matrícula devuelto incluye además una clave pública de un servidor DRM-R como un identificador de éste.

43. El método de la reivindicación 41 que comprende el servidor DRM-E ahora matriculado que almacena el certificado de matrícula devuelto que incluye además información de un rango válido que especifica un rango durante el cual el certificado de matrícula se valida.

44. El método de la reivindicación 41 que comprende el servidor DRM-E ahora matriculado que almacena el certificado de matrícula devuelto que incluye además identificar el indicio para identificar la identificación que se ofrece.

45. El método de la reivindicación 27 que comprende el servidor DRM-E ahora matriculado que almacena el certificado de matrícula devuelto que incluye una cadena de certificados que se vuelve a una autoridad de raíz confiada de tal manera que el certificado de

matrícula generado puede ser validado sobre tal cadena de certificados.

46. El método de la reivindicación 27 que comprende además el servidor DRM-E que identifica una lista de autoridad de revocación en al menos una entidad con la autoridad para revocar la matrícula de tal servidor DRM-E dentro del sistema DRM, que comprende el servidor DRM-E que envía una solicitud de matrícula al servidor DRM-R, la solicitud incluye una identificación que ofrece, (PU-E), y la lista de autoridad de revocación, el servidor DRM-R, si la solicitud se estima, genera un certificado de matrícula digital para el servidor DRM-E tal como el servidor DRM-E dentro del sistema DRM, el certificado de matrícula generado se basa al menos en parte en (PU-E) y la lista de autoridad de revocación.

47. El método de la reivindicación 46 que comprende el servidor DRM-E que identifica cada entidad en la lista de autoridad de revocación por vía de una clave pública de éste.

48. El método de la reivindicación 46 que comprende el servidor DRM-E ahora matriculado que almacena el certificado de matrícula devuelto que incluye (PU-E) como un identificador del servidor DRM-E, la lista de autoridad de revocación de la solicitud, y una firma basada en una clave privada del servidor DRM-R.

49. El método de la reivindicación 27 que comprende el servidor DRM-E ahora matriculado que almacena el certificado de matrícula devuelto en un certificado de matrícula XrML.

50. Un método en combinación con un sistema Administrador de Derechos Digitales (DRM) que tiene una pluralidad de servidores DRM que desempeñan funcionalidad DRM, el método para entrada al servidor DRM-E a ser matriculado en el sistema por un servidor DRM-R de matrícula tal como el servidor DRM-E de entrada se confía dentro del sistema y comprende:

- el servidor DRM-R recibe una solicitud de matrícula del servidor DRM-E que

incluye una identificación que ofrece y una clave pública del servidor DRM-E (PU-E) para identificar tal servidor DRM-E con el sistema DRM;

- el servidor DRM-R valida la identificación que ofrece;
- el servidor DRM-R si la solicitud se estima, genera un certificado de matrícula digital para el servidor DRM-E para matricular tal servidor DRM-E dentro del sistema DRM, el certificado de matrícula generado se basa al menos en parte en (PU-E); y
- el servidor DRM-R retorna el certificado de matrícula generado al servidor DRM-E solicitante, el servidor DRM-E ahora matriculado almacena el certificado de matrícula devuelto en una ubicación apropiada para futuro uso, el servidor DRM-E con el certificado de matrícula capaz de emplea el mismo para publicar documentos DRM dentro del sistema DRM.

51. El método de la reivindicación 50 en donde el servidor DRM-R no tiene una base existente para confiar el servidor DRM-E, el método comprende:

- el servidor DRM-R recibe una solicitud de matrícula del servidor DRM-E que incluye (PU-E) y una identificación que ofrece que comprende un certificado de garantía de una parte que desea garantizar para tal servidor DRM-E, el certificado de garantía incorpora en éste una clave pública (PU-V) y una clave privada correspondiente (PR-V), el servidor DRM-E que emplea (PU-E) y (PR-V) para formular un indicio de propiedad muestra que el servidor DRM-E es propietario del certificado de garantía, la solicitud de matrícula además incluye los indicios de propietario;

- el servidor DRM-R valida el certificado de garantía;

- el servidor DRM-R verifica el indicio de propiedad;

117

119

- el servidor DRM-R, si la solicitud se estima, genera un certificado de matrícula digital para el servidor DRM-E para matricular tal servidor DRM-E en el sistema DRM, el certificado de matrícula generado se basa al menos en parte en un certificado de garantía y (PU-E).

52. El método de la reivindicación 51 que comprende el servidor DRM-R que recibe una solicitud de matrícula del servidor DRM-E que incluye un certificado de garantía de un agente publicador de certificado independiente que es confiable y descansa en el servidor DRM-R para llevar a cabo tal garantía.

53. El método de la reivindicación 51 que comprende el servidor DRM-R que recibe una solicitud de matrícula del servidor DRM-E que incluye un certificado de garantía X.509.

54. El método de la reivindicación 51 comprende:

1

- el servidor DRM-R que recibe una solicitud de matrícula del servidor DRM-E que incluye un certificado de matrícula firmado por la parte garantizadora y se acompaña por una cadena de certificados dirigidos a la raíz conocida para propósitos de validación; y

- el servidor DRM-R que valida el certificado de garantía basado en la firma de éste mediante una parte garantizadora y la cadena de certificados para establecer que el servidor DRM-E ha sido garantizador.

55. El método de la reivindicación 51 en donde el servidor DRM-E desarrolla uno de emplear (PR-V) para encriptar (PU-E) para resultar en (PR-V(PU-E)) como lo indica el propietario, o firmando (PU-E) con (PR-V) para resultar en (PU-E)S(PR-V) como lo indica el propietario, el método comprende el servidor DRM-R que verifica el indicio del propietario mediante la aplicación (PU-V) de la solicitud para desencriptar (PU-E) o verificar la firma para establecer que el

propietario del servidor DRM-E (PR-V) y por lo tanto el certificado de garantía.

56. El método de la reivindicación 51 que comprende el servidor DRM-R que genera el certificado de matrícula para incluir (PU-E) como un identificador del servidor DRM-E, que identifica el indicio para identificar el certificado de garantía, y una firma basada en una clave privada del servidor DRM-R, por lo tanto el indicio que identifica para el certificado de garantía en los actos del certificado de matrícula como un puente al certificado de garantía y muestra que el servidor DRM-R es confiable y reposa en la parte de garantía para garantizar el servidor DRM-E.

57. El método de la reivindicación 56 que comprende el servidor DRM-R que genera el certificado de matrícula para incluir adicionalmente una clave pública del servidor DRM-R como un identificador de éste.

58. El método de la reivindicación 56 que comprende el servidor DRM-R que genera el certificado de

matrícula para incluir adicionalmente información de rango válido que especifica un rango durante el cual el certificado de matrícula se valida.

59. El método de la reivindicación 50 en donde el servidor DRM-R tiene una base existente para confiar el servidor DRM-E, el método comprende:

- el servidor DRM-R recibe una solicitud de matrícula del servidor DRM-E que incluye (PU-E) y una indicación que ofrece que comprende credenciales reconocibles a y que se espera sean estimadas por el servidor DRM-R;
- el servidor DRM-R valida las credenciales;
- el servidor DRM-R, si la solicitud se estima, generará un certificado de matrícula digital para el servidor DRM-E para matricular tal servidor DRM-E dentro del sistema DRM, el certificado de matrícula generado se basa al menos en parte en las credenciales y (PU-E).

60. El método de la reivindicación 59 que comprende el servidor DRM-R recibe una solicitud de matrícula del servidor DRM-E que incluye credenciales seleccionadas de un grupo que consiste de un ID de red o un ID de dominio, y credenciales publicadas por una tercera parte.

61. El método de la reivindicación 59 que comprende el servidor DRM-R genera el certificado de matrícula para incluir (PU-E) como un identificador del servidor DRM-E, que identifica el indicio para identificar las credenciales, y una firma basada en una clave privada del servidor DRM-R.

62. El método de la reivindicación 61 que comprende el servidor DRM-R genera el certificado de matrícula para incluir adicionalmente una clave pública del servidor DRM-R como un identificador de éste.

63. El método de la reivindicación 61 que comprende el servidor DRM-R que genera el certificado de matrícula para incluir adicionalmente información en un rango válido especifica un rango durante el cual el certificado de matrícula se valida.

64. El método de la reivindicación 50 comprende además el servidor DRM-R que desarrolla la lógica adecuada para decidir si estima la solicitud.

65. El método de la reivindicación 64 que comprende el servidor DRM-R que desarrolla la lógica acostumbrada seleccionada de un grupo que consiste de desarrollar una revisión de fondo sobre el servidor DRM-E y/o su operador, determinar si el servidor DRM-E y/o una porción de éste es actual, determinar si el servidor DRM-E es una lista de revocación o mirar la lista, y combinaciones de ésta.

66. El método de la reivindicación 50 que comprende el servidor DRM-R genera el certificado de matrícula para incluir (PU-E) como un identificador del servidor DRM-E, y una firma basada en una clave privada del servidor DRM-R.

67. El método de la reivindicación 66 que comprende el servidor DRM-R genera el certificado de matrícula

para incluir adicionalmente una clave pública del servidor DRM-R como un identificador de éste.

68. El método de la reivindicación 66 comprende el servidor DRM-R que genera el certificado de matrícula para incluir adicionalmente información en un rango válido que especifica un rango durante el cual el certificado de matrícula se valida.

69. El método de la reivindicación 66 que comprende el servidor DRM-R que genera el certificado de matrícula para incluir adicionalmente el indicio de identificación para identificar la identificación que ofrece.

70. El método de la reivindicación 50 que comprende el servidor DRM-R que genera el certificado de matrícula mediante el empleo de la lógica adecuada para generar al menos una porción de la información en el certificado de matrícula.

71. El método de la reivindicación 50 que comprende además el servidor DRM-R que adjunta al certificado de matrícula generado una cadena de certificados que

vuelve a la autoridad de raíz confiada de tal manera que genera el certificado de matrícula generado que puede ser validado basado en tal cadena de certificados.

72. El método de la reivindicación 50 que comprende además el servidor DRM-R que recibe una solicitud de ~~matrícula del servidor DRM-E~~ que incluye además una lista de autoridad de revocación que identifica al menos una entidad con autoridad para revocar la matrícula de tal servidor DRM-E dentro del sistema DRM, y comprende el servidor DRM-R, si la solicitud se estima, genera un certificado de matrícula digital para el servidor DRM-E para matricular tal servidor DRM-E en el sistema DRM, el certificado de matrícula generado se basa al menos en parte en la lista de autoridad de revocación.

73. El método de la reivindicación 72 que comprende el servidor DRM-R genera el certificado de matrícula para ~~incluir~~ (PU-E) como un identificador del servidor DRM-E, la lista de autoridad de revocación, y una firma basada en una clave privada del servidor DRM-R.

74. El método de la reivindicación 50 que comprende  
el servidor DRM-R que genera un certificado de  
matrícula XrML.

---

**RESUMEN DE LA DESCRIPCIÓN**

Un sistema de Administración de Derechos Digitales (DRM) tiene una pluralidad de servidores DRM que desempeñan funcionalidad DRM y un servidor DRM-E de entrada se matricula en el sistema mediante un servidor DRM-R de matrícula tal como el servidor DRM-E de entrada se confía dentro de el sistema. El servidor DRM-E envía una solicitud de matrícula al servidor DRM-R que incluye una identificación que ofrece y una clave pública (PU-E). El servidor DRM-R valida la identificación que ofrece, y, si la solicitud se estima, genera un certificado de matrícula digital con (PU-E) para el servidor DRM-E para matricular tal servidor DRM-E en el sistema DRM. El servidor DRM-E ahora matriculado con el certificado de matrícula generado es capaz de emplear los mismos documentos DRM publicados dentro del sistema DRM.

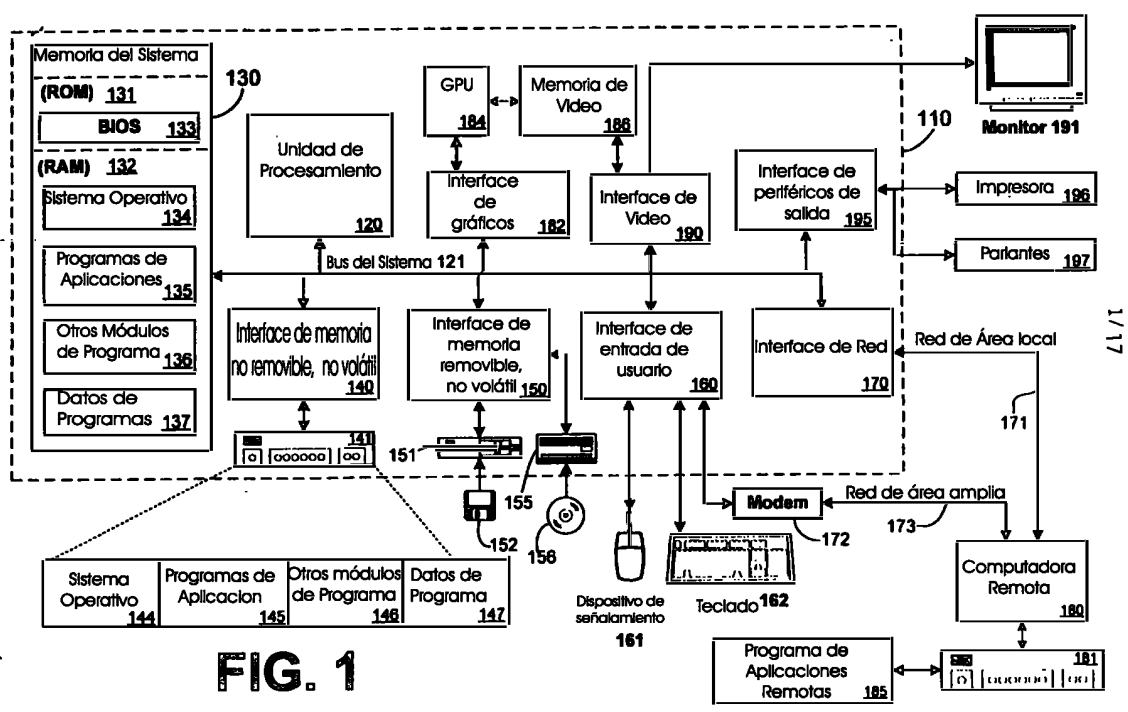
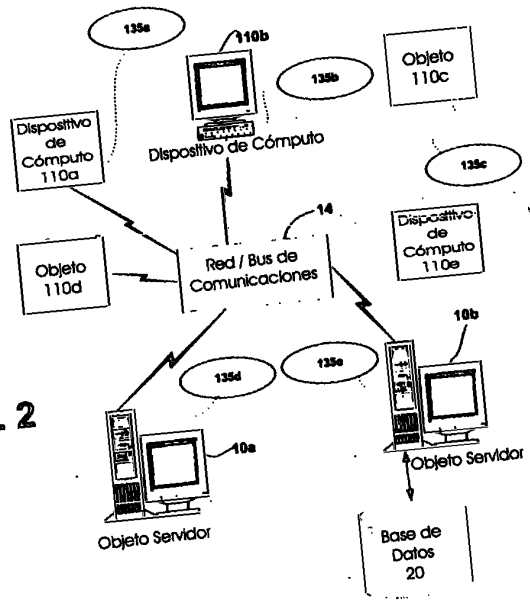


FIG. 1

100

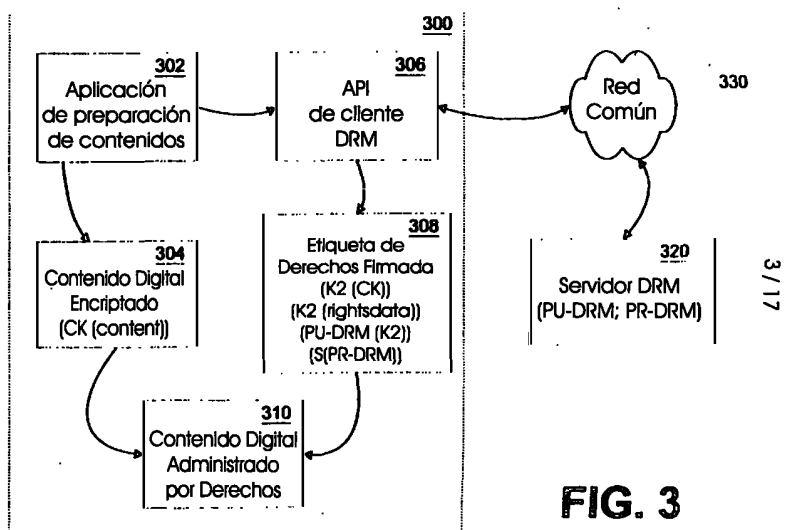
730

FIG. 2



2 / 17

130



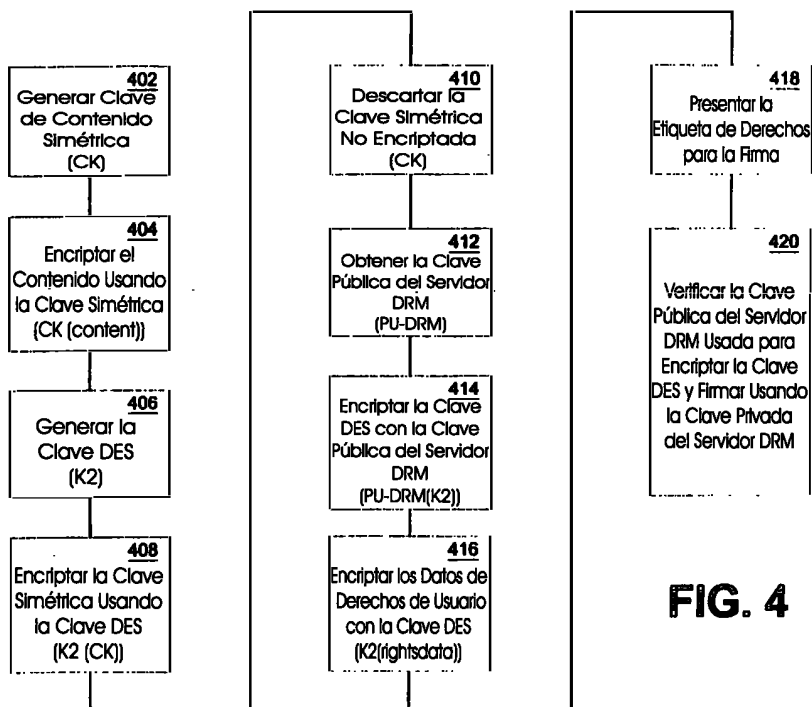


FIG. 4

|                             |
|-----------------------------|
| SRL 308                     |
| INFORMACION DE CONTENIDO    |
| INFORMACION DE SERVIDOR DRM |
| - (PU-DRM(K2))              |
| INFORMACION DE REFERENCIA   |
| --URL                       |
| --TRAER DE NUEVO            |
| INFO. ETIQUETA DE DERECHOS  |
| (K2(RIGHTSDATA))            |
| (K2(CK))                    |
| S (PR-DRM)                  |

5 / 17

FIG. 4A

188

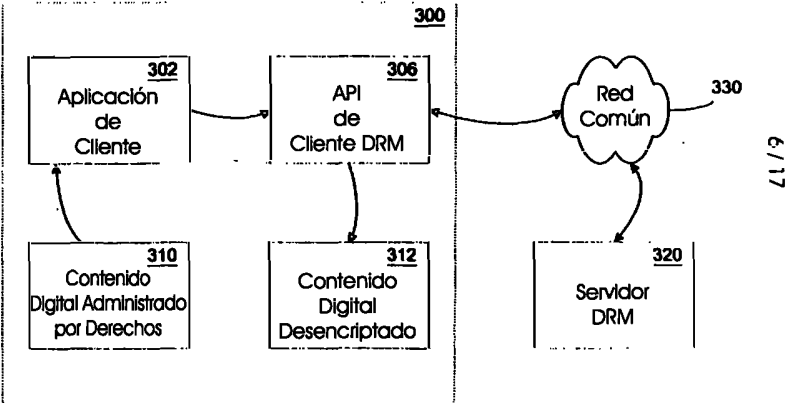
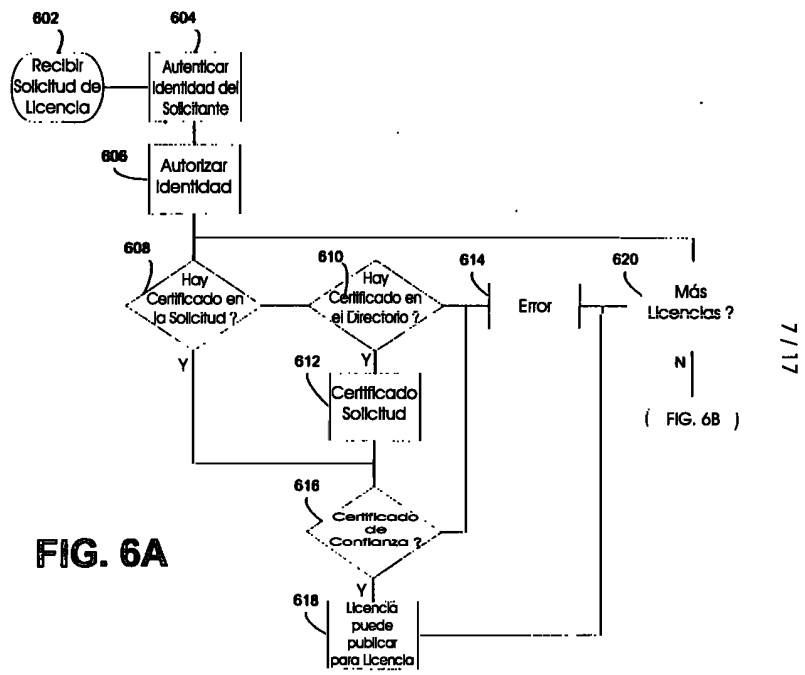
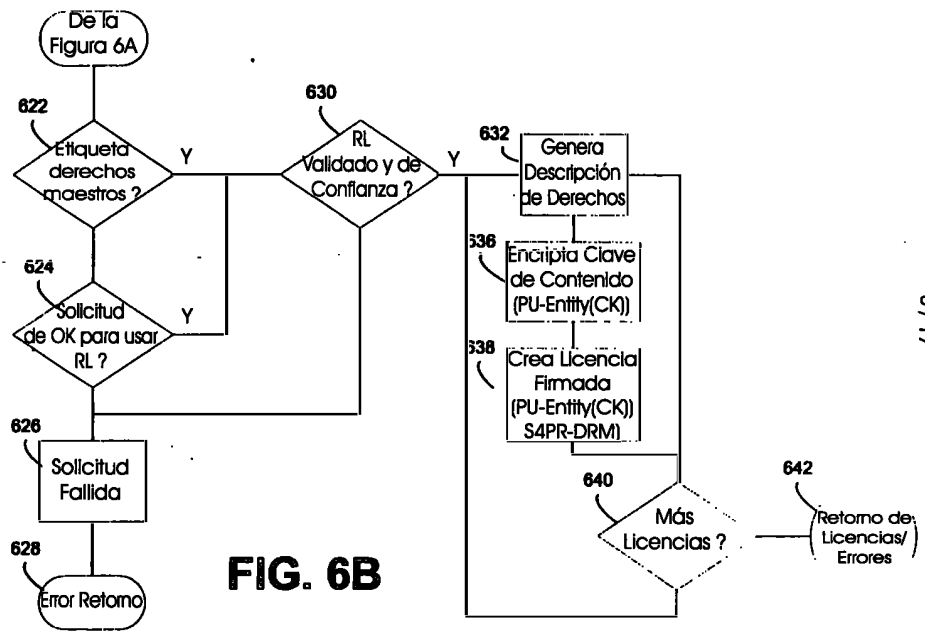


FIG. 5

107





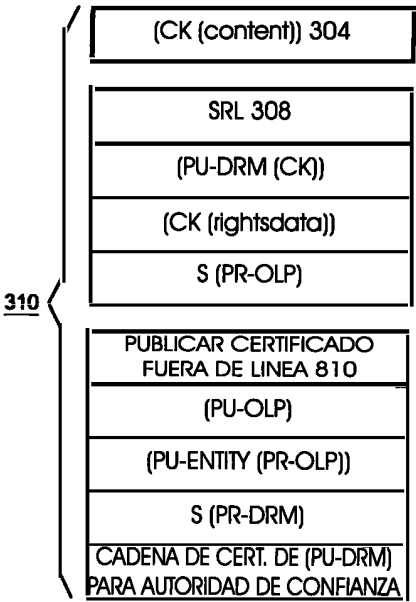
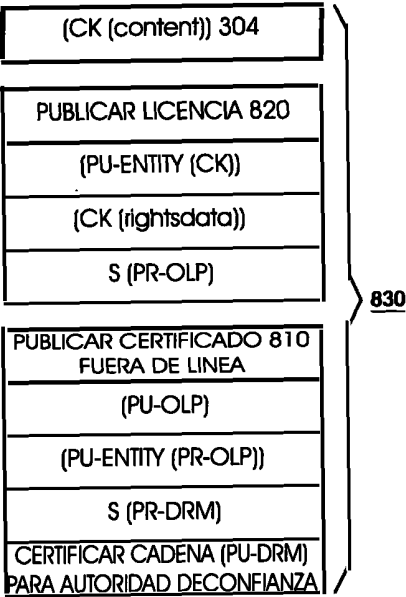


FIG. 7

187



10 / 17

FIG. 8

738

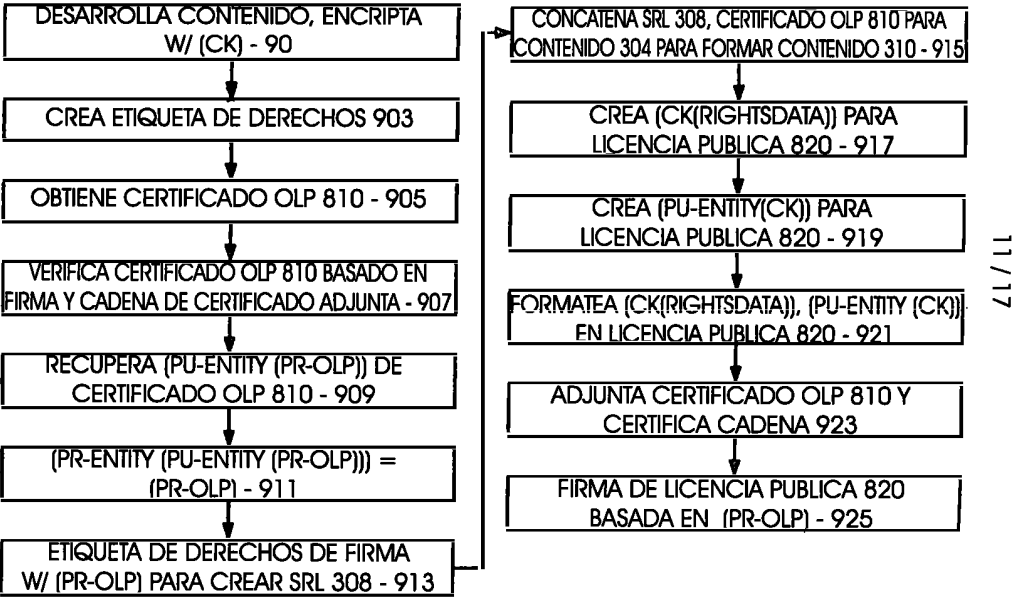
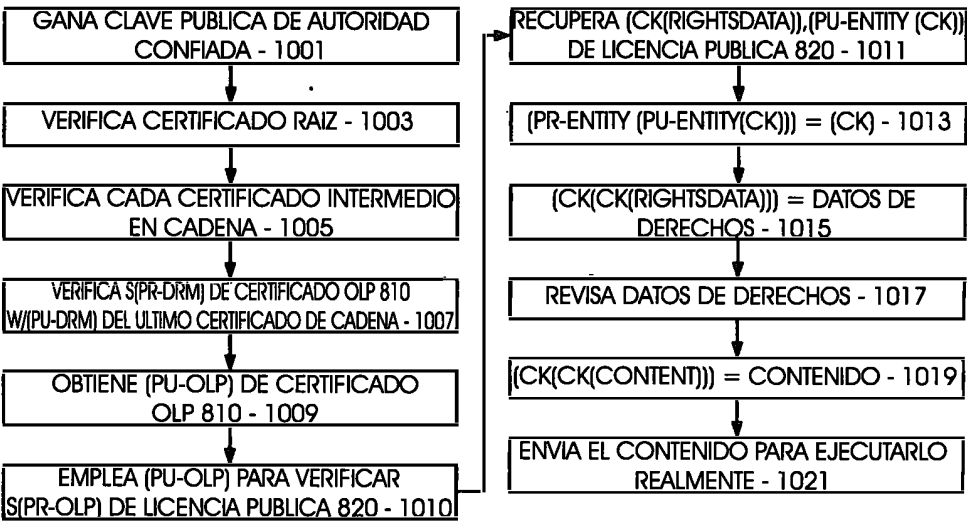


Fig. 9

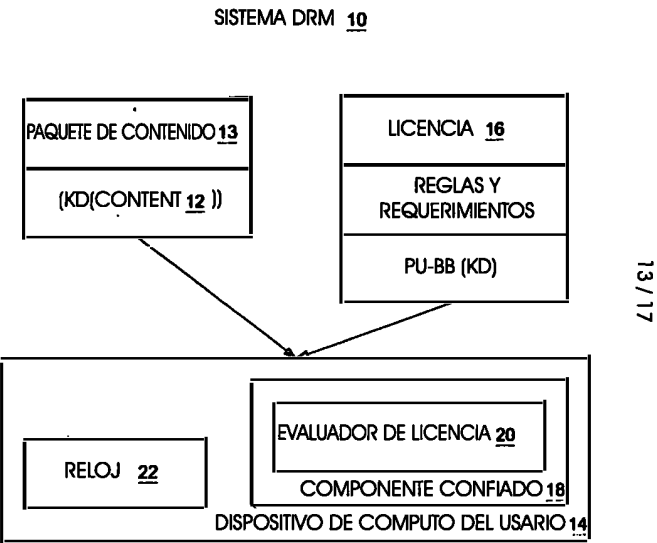


12 / 17

Fig. 10

140

747



**Fig. 11**

147

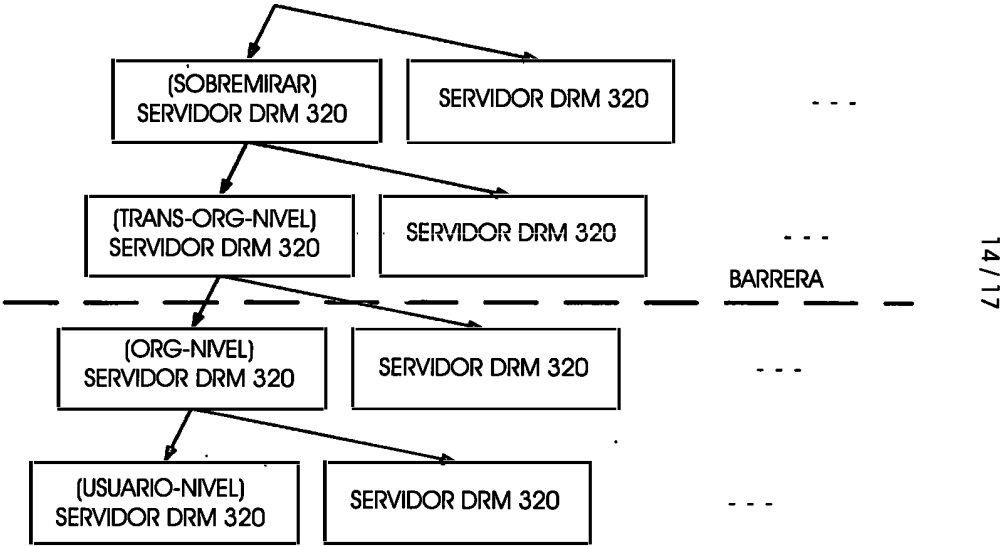


Fig. 12

14/17

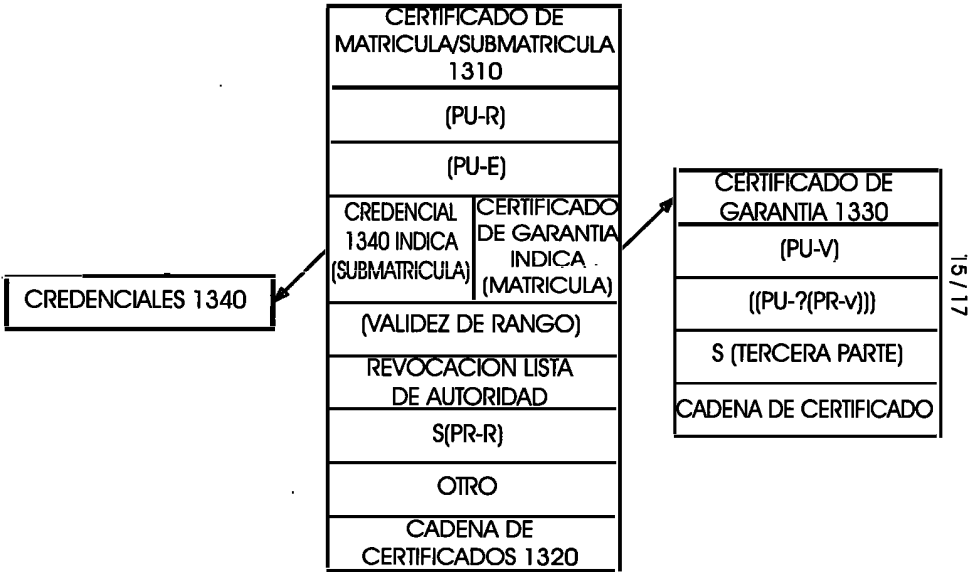


FIG. 13

143

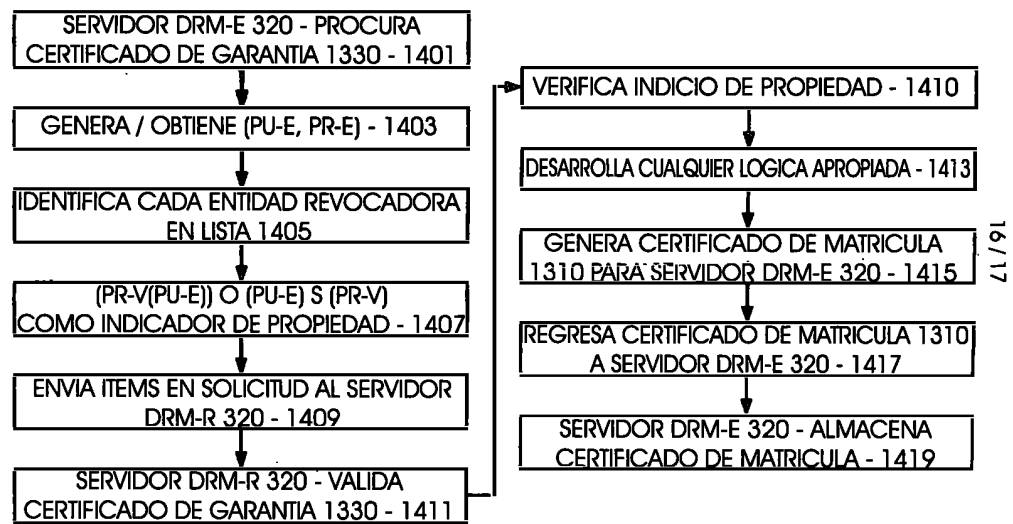


Fig. 14

Handwritten signature or mark.

279/273

745

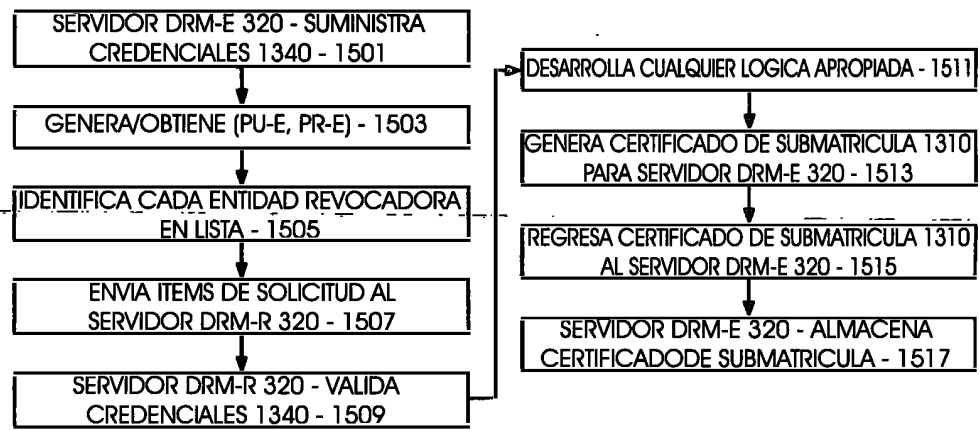


Fig. 15

1/5



INDUSTRIA Y COMERCIO  
Radicación : 04016523 00000000 Folios: 223  
Fecha (AM): 2004-02-25 14:39:10  
Tramite : 002 PATENTES D I REGISTRO/ 411 ASSENIAC  
Dependencia: 2020 DIVISION DE NUEVAS CREACIONES

NIT 800176089-2

RECIBO OFICIAL DE CAJA : 04 - 10,510  
FECHA : FEBRERO 10 DE 2004

\*\*\*\*\* CONSIGNACION \*\*\*\*\*

| DEPOSITANTE              | TIPO PAGO    | BANCO         | CUENTA      | No. PAGO | FECHA PAGO | Vr. PAGO     |
|--------------------------|--------------|---------------|-------------|----------|------------|--------------|
| CARDENAS Y CARDENAS P.I. | CONSIGNACION | BANCO POPULAR | 050-00110-6 | 21077195 | 10/02/2004 | 3,451,000.00 |

\*\*\*\*\* CONCEPTO \*\*\*\*\*

| CANT. RENTISTICO | CONCEPTO | TOTAL CONCEPTO |
|------------------|----------|----------------|
|------------------|----------|----------------|

1 50005-01-01 SOLICITUDES 82 INVOCACION DE UNA PRIORIDAD EN NU 118,000.00  
TOTAL : 118,000.00

SON: CIENTO DIEZ Y OCHO MIL PESOS

Matricular/Sub-Matricular  
un servidor de Administración  
de derechos digitales  
(DEM) en una Arquitectura  
DRM

RESPONSABLE : \_\_\_\_\_

RECIBO DE CAJA APLICADO AL EXPEDIENTE No. \_\_\_\_\_

RECEIVED  
11 FEB 25 2004  
INDUSTRIA Y COMERCIO

**TITLE OF THE INVENTION**

5     Enrolling / Sub-Enrolling a Digital Rights Management (DRM) Server Into a DRM  
Architecture

**CROSS-REFERENCE TO RELATED APPLICATIONS**

The following U.S. Patent Applications disclose subject matter that is  
related to the subject matter of the present application, and are hereby

10    incorporated herein by reference in their entirety:

U.S. Patent Application No. 10/185,527, filed June 28, 2002 under  
attorney docket number MSFT-1330 and entitled "Obtaining a Signed Rights Label  
(SRL) for Digital Content and Obtaining a Digital License Corresponding to the  
Content Based on the SRL in a Digital Rights Management System";

15     U.S. Patent Application No. 10/185,278, filed June 28, 2002 under  
attorney docket number MSFT-1333 and entitled "Using a Rights Template to  
Obtain a Signed Rights Label (SRL) for Digital Content in a Digital Rights  
Management System";

20     U.S. Patent Application No. 10/185,511, filed June 28, 2002 under  
attorney docket number MSFT-1343 and entitled "Systems And Methods For  
Issuing Usage Licenses For Digital Content And Services";

U.S. Patent Application No. \_\_\_\_\_, filed \_\_\_\_\_ under attorney docket number MSFT-1498; and entitled "Publishing Digital Content Within an Organization in Accordance with a Digital Rights Management (DRM) System;

U.S. Patent Application No. \_\_\_\_\_, filed \_\_\_\_\_ under attorney  
5 docket number MSFT-1569; and entitled "Publishing Digital Content Within an Organization in Accordance with a Digital Rights Management (DRM) System; and

U.S. Patent Application No. \_\_\_\_\_, filed \_\_\_\_\_ concurrently with the present application under attorney docket number MSFT-1537 and  
10 entitled "Issuing a Publisher Use License Off-Line in a Digital Rights Management (DRM) System".

**TECHNICAL FIELD**

This invention relates to a digital rights management (DRM) system.  
15 More particularly, the invention relates to employing a DRM system to publish digital content in an organization such as an office or corporation or the like such that rendering and use of the content within the organization may be constrained according to corresponding use or license terms. Even more particularly, the present invention relates to a network of DRM servers that effectuate such DRM  
20 system and a method of enrolling or sub-enrolling a DRM server into the network.

**BACKGROUND OF THE INVENTION**

Digital rights management and enforcement is highly desirable in connection with digital content such as digital audio, digital video, digital text,  
25 digital data, digital multimedia, etc., where such digital content is to be distributed to one or more users. Digital content could be static, such as a text document, for example, or it could be streamed, such as the streamed audio/ video of a live event. Typical modes of distribution include tangible devices such as a magnetic (floppy) disk, a magnetic tape, an optical (compact) disk (CD), etc., and intangible  
30 media such as an electronic bulletin board, an electronic network, the Internet, etc. Upon being received by the user, such user renders or 'plays' the digital content

with the aid of an appropriate rendering device such as a media player on a personal computer or the like.

In one scenario, a content owner or rights-owner such as an author, a publisher, a broadcaster, etc., wishes to distribute such digital content to each of  
5 many users or recipients in exchange for a license fee or some other consideration. In such scenario, then, the content may be a song, an album of songs, a movie, etc., and the purpose of the distribution is to generate the license fees. Such content owner, given the choice, would likely wish to restrict what the user can do with such distributed digital content. For example, the content owner  
10 would like to restrict the user from copying and re-distributing such content to a second user, at least in a manner that denies the content owner a license fee from such second user.

In addition, the content owner may wish to provide the user with the flexibility to purchase different types of use licenses at different license fees, while  
15 at the same time holding the user to the terms of whatever type of license is in fact purchased. For example, the content owner may wish to allow distributed digital content to be played only a limited number of times, only for a certain total time, only on a certain type of machine, only on a certain type of media player, only by a certain type of user, etc.

20 In another scenario, a content developer, such as an employee in or member of an organization, wishes to distribute such digital content to one or more other employees or members in the organization or to other individuals outside the organization, but would like to keep others from rendering the content. Here, the distribution of the content is more akin to organization-based content  
25 sharing in a confidential or restricted manner, as opposed to broad-based distribution in exchange for a license fee or some other consideration.

In such scenario, then, the content may be a document presentation, spreadsheet, database, email, or the like, such as may be exchanged within an office setting, and the content developer may wish to ensure that the content stays  
30 within the organization or office setting and is not rendered by non-authorized individuals, such as for example competitors or adversaries. Again, such content developer wishes to restrict what a recipient can do with such distributed digital

content. For example, the content owner would like to restrict the user from copying and re-distributing such content to a second user, at least in a manner that exposes the content outside the bounds of individuals who should be allowed to render the content.

5                   In addition, the content developer may wish to provide various recipients with different levels of rendering rights. For example, the content developer may wish to allow protected digital content to be viewable and not printable with respect to one class of individual, and viewable and printable with respect to another class of individual.

10                   However, and in either scenario, after distribution has occurred, such content owner / developer has very little if any control over the digital content. This is especially problematic in view of the fact that practically every personal computer includes the software and hardware necessary to make an exact digital copy of such digital content, and to download such exact digital copy to a write-  
15                   able magnetic or optical disk, or to send such exact digital copy over a network such as the Internet to any destination.

Of course, as part of a transaction wherein the content is distributed, the content owner / developer may require the user / recipient of the digital content to promise not to re-distribute such digital content in an unwelcome manner.

20                   However, such a promise is easily made and easily broken. A content owner / developer may attempt to prevent such re-distribution through any of several known security devices, usually involving encryption and decryption. However, there is likely very little that prevents a mildly determined user from decrypting encrypted digital content, saving such digital content in an un-encrypted form, and  
25                   then re-distributing same.

A need exists, then, for providing a digital rights management (DRM) and enforcement architecture and method that allows the controlled rendering or playing of arbitrary forms of digital content, where such control is flexible and definable by the content owner / developer of such digital content. More  
30                   specifically, a need exists for such an architecture that allows and facilitates such controlled rendering, especially in an office or organization environment or the like where documents are to be shared amongst a defined group of individuals or

classes of individuals. Even more specifically, a need exists for a method of enrolling approval-granting servers into the architecture.

### SUMMARY OF THE INVENTION

5           The aforementioned needs are satisfied at least in part by the present invention in which a Digital Rights Management (DRM) system has a plurality of DRM servers performing DRM functionality and an entering DRM-E server is enrolled into the system by an enrolling DRM-R server such that the entering DRM-E server is to be trusted within the system. In the invention, the  
10   DRM-E server procures a public / private key pair (PU-E, PR-E) for identifying such DRM-E server within the DRM system, procures a proffering identification thereof, and sends an enrollment request to the DRM-R server including the proffering identification and (PU-E).

          The DRM-R server validates the proffering identification, and, if the  
15   request is to be honored, generates a digital enrollment certificate for the DRM-E server to enroll such DRM-E server into the DRM system. The generated enrollment certificate is based at least in part on (PU-E). The DRM-R server returns the generated enrollment certificate to the requesting DRM-E server and the now-enrolled DRM-E server stores the returned enrollment certificate in an  
20   appropriate location for future use. The DRM-E server with the enrollment certificate is able to employ same to issue DRM documents within the DRM system.

### BRIEF DESCRIPTION OF THE DRAWINGS

25           The foregoing summary, as well as the following detailed description of the embodiments of the present invention, will be better understood when read in conjunction with the appended drawings. For the purpose of illustrating the invention, there are shown in the drawings embodiments which are presently preferred. As should be understood, however, the invention is not limited to the  
30   precise arrangements and instrumentalities shown. In the drawings:

          Fig. 1 is a block diagram representing an exemplary non-limiting computing environment in which the present invention may be implemented;

Fig. 2 is a block diagram representing an exemplary network environment having a variety of computing devices in which the present invention may be implemented;

Fig. 3 is a functional block diagram of a preferred embodiment of a system and method according to the invention for publishing digital content;

Fig. 4 is a flowchart of a preferred embodiment of a method according to the invention for publishing rights managed digital content;

Fig. 4A is a block diagram showing the structure of a signed rights label as produced by the method of Fig. 4;

Fig. 5 is a block diagram of a preferred embodiment of a system and method according to the invention for licensing rights managed digital content;

Figs. 6A and 6B are flowcharts of a preferred embodiment of a method according to the invention for licensing rights managed digital content;

Fig. 7 is a block diagram showing a certificate issued by a DRM server to a user to allow the user to perform off-line publishing in accordance with one embodiment of the present invention;

Fig. 8 is a block diagram showing the certificate of Fig. 7 along with a publisher license that allows a publishing user to render content off-line published thereby in accordance with one embodiment of the present invention;

Fig. 9 is a flow chart showing key steps performed by the publishing user to obtain the publishing license of Fig. 8 in accordance with one embodiment of the present invention;

Fig. 10 is a flow chart showing key steps performed by the publishing user to employ the obtained publishing license of Fig. 9 to render the corresponding content in accordance with one embodiment of the present invention;

Fig. 11 is a block diagram showing an enforcement architecture of an example of a trust-based system;

Fig. 12 is a block diagram showing a plurality of DRM servers such as may exist in the architecture of the present invention, where each (entering) DRM server is enrolled or sub-enrolled into the architecture by another (enrolling) DRM server issuing same an enrollment certificate;

Fig. 13 is a block diagram showing the enrollment certificate of Fig. 12 along with a vouching certificate presented in at least some instances by an entering DRM server to an enrolling DRM server; and

5 Figs. 14 and 15 are flow diagrams showing key steps performed by the enrolling and entering DRM servers of Figs. 13 and 14 to enroll (Fig. 14) or sub-enroll (Fig. 15) the entering DRM server.

## DETAILED DESCRIPTION OF THE INVENTION

### COMPUTER ENVIRONMENT

10 Fig. 1 and the following discussion are intended to provide a brief general description of a suitable computing environment in which the invention may be implemented. It should be understood, however, that handheld, portable, and other computing devices of all kinds are contemplated for use in connection with the present invention. While a general purpose computer is described below, 15 this is but one example, and the present invention requires only a thin client having network server interoperability and interaction. Thus, the present invention may be implemented in an environment of networked hosted services in which very little or minimal client resources are implicated, e.g., a networked environment in which the client device serves merely as a browser or interface to 20 the World Wide Web.

Although not required, the invention can be implemented via an application programming interface (API), for use by a developer, and/or included within the network browsing software which will be described in the general context of computer-executable instructions, such as program modules, being 25 executed by one or more computers, such as client workstations, servers, or other devices. Generally, program modules include routines, programs, objects, components, data structures and the like that perform particular tasks or implement particular abstract data types. Typically, the functionality of the program modules may be combined or distributed as desired in various 30 embodiments. Moreover, those skilled in the art will appreciate that the invention may be practiced with other computer system configurations. Other well known

computing systems, environments, and/or configurations that may be suitable for use with the invention include, but are not limited to, personal computers (PCs), automated teller machines, server computers, hand-held or laptop devices, multi-processor systems, microprocessor-based systems, programmable consumer electronics, network PCs, minicomputers, mainframe computers, and the like. The invention may also be practiced in distributed computing environments where tasks are performed by remote processing devices that are linked through a communications network or other data transmission medium. In a distributed computing environment, program modules may be located in both local and remote computer storage media including memory storage devices.

Fig. 1 thus illustrates an example of a suitable computing system environment 100 in which the invention may be implemented, although as made clear above, the computing system environment 100 is only one example of a suitable computing environment and is not intended to suggest any limitation as to the scope of use or functionality of the invention. Neither should the computing environment 100 be interpreted as having any dependency or requirement relating to any one or combination of components illustrated in the exemplary operating environment 100.

With reference to Fig. 1, an exemplary system for implementing the invention includes a general purpose computing device in the form of a computer 110. Components of computer 110 may include, but are not limited to, a processing unit 120, a system memory 130, and a system bus 121 that couples various system components including the system memory to the processing unit 120. The system bus 121 may be any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using any of a variety of bus architectures. By way of example, and not limitation, such architectures include Industry Standard Architecture (ISA) bus, Micro Channel Architecture (MCA) bus, Enhanced ISA (EISA) bus, Video Electronics Standards Association (VESA) local bus, and Peripheral Component Interconnect (PCI) bus (also known as Mezzanine bus).

Computer 110 typically includes a variety of computer readable media. Computer readable media can be any available media that can be

accessed by computer 110 and includes both volatile and nonvolatile media, removable and non-removable media. By way of example, and not limitation, computer readable media may comprise computer storage media and communication media. Computer storage media includes both volatile and nonvolatile, removable and non-removable media implemented in any method or technology for storage of information such as computer readable instructions, data structures, program modules or other data. Computer storage media includes, but is not limited to, RAM, ROM, EEPROM, flash memory or other memory technology, CDROM, digital versatile disks (DVD) or other optical disk storage, magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices, or any other medium which can be used to store the desired information and which can be accessed by computer 110. Communication media typically embodies computer readable instructions, data structures, program modules or other data in a modulated data signal such as a carrier wave or other transport mechanism and includes any information delivery media. The term "modulated data signal" means a signal that has one or more of its characteristics set or changed in such a manner as to encode information in the signal. By way of example, and not limitation, communication media includes wired media such as a wired network or direct-wired connection, and wireless media such as acoustic, RF, infrared, and other wireless media. Combinations of any of the above should also be included within the scope of computer readable media.

The system memory 130 includes computer storage media in the form of volatile and/or nonvolatile memory such as read only memory (ROM) 131 and random access memory (RAM) 132. A basic input/output system 133 (BIOS), containing the basic routines that help to transfer information between elements within computer 110, such as during start-up, is typically stored in ROM 131. RAM 132 typically contains data and/or program modules that are immediately accessible to and/or presently being operated on by processing unit 120. By way of example, and not limitation, Fig. 1 illustrates operating system 134, application programs 135, other program modules 136, and program data 137.

The computer 110 may also include other removable/non-removable, volatile/nonvolatile computer storage media. By way of example only,

Fig. 1 illustrates a hard disk drive 141 that reads from or writes to non-removable, nonvolatile magnetic media, a magnetic disk drive 151 that reads from or writes to a removable, nonvolatile magnetic disk 152, and an optical disk drive 155 that reads from or writes to a removable, nonvolatile optical disk 156, such as a CD ROM or other optical media. Other removable/non-removable, volatile/nonvolatile computer storage media that can be used in the exemplary operating environment include, but are not limited to, magnetic tape cassettes, flash memory cards, digital versatile disks, digital video tape, solid state RAM, solid state ROM, and the like. The hard disk drive 141 is typically connected to the system bus 121 through a non-removable memory interface such as interface 140, and magnetic disk drive 151 and optical disk drive 155 are typically connected to the system bus 121 by a removable memory interface, such as interface 150.

The drives and their associated computer storage media discussed above and illustrated in Fig. 1 provide storage of computer readable instructions, data structures, program modules and other data for the computer 110. In Fig. 1, for example, hard disk drive 141 is illustrated as storing operating system 144, application programs 145, other program modules 146, and program data 147. Note that these components can either be the same as or different from operating system 134, application programs 135, other program modules 136, and program data 137. Operating system 144, application programs 145, other program modules 146, and program data 147 are given different numbers here to illustrate that, at a minimum, they are different copies. A user may enter commands and information into the computer 110 through input devices such as a keyboard 162 and pointing device 161, commonly referred to as a mouse, trackball or touch pad. Other input devices (not shown) may include a microphone, joystick, game pad, satellite dish, scanner, or the like. These and other input devices are often connected to the processing unit 120 through a user input interface 160 that is coupled to the system bus 121, but may be connected by other interface and bus structures, such as a parallel port, game port or a universal serial bus (USB).

A monitor 191 or other type of display device is also connected to the system bus 121 via an interface, such as a video interface 190. A graphics interface 182, such as Northbridge, may also be connected to the system bus 121.

Northbridge is a chipset that communicates with the CPU, or host processing unit 120, and assumes responsibility for accelerated graphics port (AGP) communications. One or more graphics processing units (GPUs) 184 may communicate with graphics interface 182. In this regard, GPUs 184 generally  
5 include on-chip memory storage, such as register storage and GPUs 184 communicate with a video memory 186. GPUs 184, however, are but one example of a coprocessor and thus a variety of co-processing devices may be included in computer 110. A monitor 191 or other type of display device is also connected to the system bus 121 via an interface, such as a video interface 190,  
10 which may in turn communicate with video memory 186. In addition to monitor 191, computers may also include other peripheral output devices such as speakers 197 and printer 196, which may be connected through an output peripheral interface 195.

The computer 110 may operate in a networked environment using  
15 logical connections to one or more remote computers, such as a remote computer 180. The remote computer 180 may be a personal computer, a server, a router, a network PC, a peer device or other common network node, and typically includes many or all of the elements described above relative to the computer 110, although only a memory storage device 181 has been illustrated in Fig. 1. The  
20 logical connections depicted in Fig. 1 include a local area network (LAN) 171 and a wide area network (WAN) 173, but may also include other networks. Such networking environments are commonplace in offices, enterprise-wide computer networks, intranets and the Internet.

When used in a LAN networking environment, the computer 110 is  
25 connected to the LAN 171 through a network interface or adapter 170. When used in a WAN networking environment, the computer 110 typically includes a modem 172 or other means for establishing communications over the WAN 173, such as the Internet. The modem 172, which may be internal or external, may be connected to the system bus 121 via the user input interface 160, or other  
30 appropriate mechanism. In a networked environment, program modules depicted relative to the computer 110, or portions thereof, may be stored in the remote memory storage device. By way of example, and not limitation, Fig. 1 illustrates

remote application programs 135 as residing on memory device 181. It will be appreciated that the network connections shown are exemplary and other means of establishing a communications link between the computers may be used.

One of ordinary skill in the art can appreciate that a computer 110 or  
5 other client device can be deployed as part of a computer network. In this regard, the present invention pertains to any computer system having any number of memory or storage units, and any number of applications and processes occurring across any number of storage units or volumes. The present invention may apply to an environment with server computers and client computers deployed in a  
10 network environment, having remote or local storage. The present invention may also apply to a standalone computing device, having programming language functionality, interpretation and execution capabilities.

Distributed computing facilitates sharing of computer resources and services by direct exchange between computing devices and systems. These  
15 resources and services include the exchange of information, cache storage, and disk storage for files. Distributed computing takes advantage of network connectivity, allowing clients to leverage their collective power to benefit the entire enterprise. In this regard, a variety of devices may have applications, objects or resources that may interact to implicate authentication techniques of the present  
20 invention for trusted graphics pipeline(s).

Fig. 2 provides a schematic diagram of an exemplary networked or distributed computing environment. The distributed computing environment comprises computing objects 10a, 10b, etc. and computing objects or devices 110a, 110b, 110c, etc. These objects may comprise programs, methods, data  
25 stores, programmable logic, etc. The objects may comprise portions of the same or different devices such as PDAs, televisions, MP3 players, televisions, personal computers, etc. Each object can communicate with another object by way of the communications network 14. This network may itself comprise other computing objects and computing devices that provide services to the system of Fig. 2. In  
30 accordance with an aspect of the invention, each object 10 or 110 may contain an application that might request the authentication techniques of the present invention for trusted graphics pipeline(s).

It can also be appreciated that an object, such as 110c, may be hosted on another computing device 10 or 110. Thus, although the physical environment depicted may show the connected devices as computers, such illustration is merely exemplary and the physical environment may alternatively be depicted or described comprising various digital devices such as PDAs, televisions, MP3 players, etc., software objects such as interfaces, COM objects and the like.

There are a variety of systems, components, and network configurations that support distributed computing environments. For example, computing systems may be connected together by wireline or wireless systems, by local networks or widely distributed networks. Currently, many of the networks are coupled to the Internet, which provides the infrastructure for widely distributed computing and encompasses many different networks.

In home networking environments, there are at least four disparate network transport media that may each support a unique protocol such as Power line, data (both wireless and wired), voice (e.g., telephone) and entertainment media. Most home control devices such as light switches and appliances may use power line for connectivity. Data Services may enter the home as broadband (e.g., either DSL or Cable modem) and are accessible within the home using either wireless (e.g., HomeRF or 802.11b) or wired (e.g., Home PNA, Cat 5, even power line) connectivity. Voice traffic may enter the home either as wired (e.g., Cat 3) or wireless (e.g., cell phones) and may be distributed within the home using Cat 3 wiring. Entertainment media may enter the home either through satellite or cable and is typically distributed in the home using coaxial cable. IEEE 1394 and DVI are also emerging as digital interconnects for clusters of media devices. All of these network environments and others that may emerge as protocol standards may be interconnected to form an intranet that may be connected to the outside world by way of the Internet. In short, a variety of disparate sources exist for the storage and transmission of data, and consequently, moving forward, computing devices will require ways of protecting content at all portions of the data processing pipeline.

161

The 'Internet' commonly refers to the collection of networks and gateways that utilize the TCP/IP suite of protocols, which are well-known in the art of computer networking. TCP/IP is an acronym for "Transport Control Protocol/Interface Program." The Internet can be described as a system of  
5 geographically distributed remote computer networks interconnected by computers executing networking protocols that allow users to interact and share information over the networks. Because of such wide-spread information sharing, remote networks such as the Internet have thus far generally evolved into an open system for which developers can design software applications for performing  
10 specialized operations or services, essentially without restriction.

Thus, the network infrastructure enables a host of network topologies such as client/server, peer-to-peer, or hybrid architectures. The "client" is a member of a class or group that uses the services of another class or group to which it is not related. Thus, in computing, a client is a process, i.e., roughly a set  
15 of instructions or tasks, that requests a service provided by another program. The client process utilizes the requested service without having to "know" any working details about the other program or the service itself. In a client/server architecture, particularly a networked system, a client is usually a computer that accesses shared network resources provided by another computer e.g., a server. In the  
20 example of Fig. 2, computers 110a, 110b, etc. can be thought of as clients and computer 10a, 10b, etc. can be thought of as the server where server 10a, 10b, etc. maintains the data that is then replicated in the client computers 110a, 110b, etc.

A server is typically a remote computer system accessible over a  
25 remote network such as the Internet. The client process may be active in a first computer system, and the server process may be active in a second computer system, communicating with one another over a communications medium, thus providing distributed functionality and allowing multiple clients to take advantage of the information-gathering capabilities of the server.

30 Client and server communicate with one another utilizing the functionality provided by a protocol layer. For example, Hypertext-Transfer Protocol (HTTP) is a common protocol that is used in conjunction with the World

162

Wide Web (WWW). Typically, a computer network address such as a Universal Resource Locator (URL) or an Internet Protocol (IP) address is used to identify the server or client computers to each other. The network address can be referred to as a Universal Resource Locator address. For example, communication can be  
 5 provided over a communications medium. In particular, the client and server may be coupled to one another via TCP/IP connections for high-capacity communication.

Thus, Fig. 2 illustrates an exemplary networked or distributed environment, with a server in communication with client computers via a  
 10 network/bus, in which the present invention may be employed. In more detail, a number of servers 10a, 10b, etc., are interconnected via a communications network/bus 14, which may be a LAN, WAN, intranet, the Internet, etc., with a number of client or remote computing devices 110a, 110b, 110c, 110d, 110e, etc., such as a portable computer, handheld computer, thin client, networked  
 15 appliance, or other device, such as a VCR, TV, oven, light, heater and the like in accordance with the present invention. It is thus contemplated that the present invention may apply to any computing device in connection with which it is desirable to process, store or render secure content from a trusted source.

In a network environment in which the communications network/bus  
 20 14 is the Internet, for example, the servers 10 can be Web servers with which the clients 110a, 110b, 110c, 110d, 110e, etc. communicate via any of a number of known protocols such as HTTP. Servers 10 may also serve as clients 110, as may be characteristic of a distributed computing environment. Communications may be wired or wireless, where appropriate. Client devices 110 may or may not  
 25 communicate via communications network/bus 14, and may have independent communications associated therewith. For example, in the case of a TV or VCR, there may or may not be a networked aspect to the control thereof. Each client computer 110 and server computer 10 may be equipped with various application program modules or objects 135 and with connections or access to various types  
 30 of storage elements or objects, across which files may be stored or to which portion(s) of files may be downloaded or migrated. Thus, the present invention can be utilized in a computer network environment having client computers 110a,

110b, etc. that can access and interact with a computer network/bus 14 and server computers 10a, 10b, etc. that may interact with client computers 110a, 110b, etc. and other devices 111 and databases 20.

## 5 Digital Rights Management (DRM) Overview

As is known, and referring now to Fig. 11, digital rights management (DRM) and enforcement is highly desirable in connection with digital content 12 such as digital audio, digital video, digital text, digital data, digital multimedia, etc., where such digital content 12 is to be distributed to users. Upon being received  
10 by the user, such user renders or 'plays' the digital content with the aid of an appropriate rendering device such as a media player on a personal computer 14 or the like.

Typically, a content owner or developer (hereinafter 'owner') distributing such digital content 12 wishes to restrict what the user can do with  
15 such distributed digital content 12. For example, the content owner may wish to restrict the user from copying and re-distributing such content 12 to a second user, or may wish to allow distributed digital content 12 to be played only a limited number of times, only for a certain total time, only on a certain type of machine, only on a certain type of media player, only by a certain type of user, etc.

20 However, after distribution has occurred, such content owner has very little if any control over the digital content 12. A DRM system 10, then, allows the controlled rendering or playing of arbitrary forms of digital content 12, where such control is flexible and definable by the content owner of such digital content. Typically, content 12 is distributed to the user in the form of a package 13 by way  
25 of any appropriate distribution channel. The digital content package 13 as distributed may include the digital content 12 encrypted with a symmetric encryption / decryption key (KD), (i.e., (KD(CONTENT))), as well as other information identifying the content, how to acquire a license for such content, etc.

The trust-based DRM system 10 allows an owner of digital content  
30 12 to specify license rules that must be satisfied before such digital content 12 is allowed to be rendered on a user's computing device 14. Such license rules can

include the aforementioned temporal requirement, and may be embodied within a digital license or use document (hereinafter 'license') 16 that the user / user's computing device 14 (hereinafter, such terms are interchangeable unless circumstances require otherwise) must obtain from the content owner or an agent thereof. Such license 16 also includes the decryption key (KD) for decrypting the digital content, perhaps encrypted according to a key decryptable by the user's computing device.

The content owner for a piece of digital content 12 must trust that the user's computing device 14 will abide by the rules and requirements specified by such content owner in the license 16, i.e. that the digital content 12 will not be rendered unless the rules and requirements within the license 16 are satisfied. Preferably, then, the user's computing device 14 is provided with a trusted component or mechanism 18 that will not render the digital content 12 except according to the license rules embodied in the license 16 associated with the digital content 12 and obtained by the user.

The trusted component 18 typically has a license evaluator 20 that determines whether the license 16 is valid, reviews the license rules and requirements in such valid license 16, and determines based on the reviewed license rules and requirements whether the requesting user has the right to render the requested digital content 12 in the manner sought, among other things. As should be understood, the license evaluator 20 is trusted in the DRM system 10 to carry out the wishes of the owner of the digital content 12 according to the rules and requirements in the license 16, and the user should not be able to easily alter such trusted element for any purpose, nefarious or otherwise.

As should be understood, the rules and requirements in the license 16 can specify whether the user has rights to render the digital content 12 based on any of several factors, including who the user is, where the user is located, what type of computing device the user is using, what rendering application is calling the DRM system, the date, the time, etc. In addition, the rules and requirements of the license 16 may limit the license 16 to a pre-determined number of plays, or pre-determined play time, for example.

The rules and requirements may be specified in the license 16 according to any appropriate language and syntax. For example, the language may simply specify attributes and values that must be satisfied (DATE must be later than X, e.g.), or may require the performance of functions according to a  
5 specified script (IF DATE greater than X, THEN DO . . . , e.g.).

Upon the license evaluator 20 determining that the license 16 is valid and that the user satisfies the rules and requirements therein, the digital content 12 can then be rendered. In particular, to render the content 12, the decryption key (KD) is obtained from the license 12 and is applied to (KD(CONTENT)) from  
10 the content package 13 to result in the actual content 12, and the actual content 12 is then in fact rendered.

#### Publishing Digital Content

Fig. 3 is a functional block diagram of a system and method for  
15 publishing digital content. "Publishing," as that term is used herein, refers to a process that an application or service follows to establish with a trusted entity a set of rights and conditions that the entity can issue for that content, as well as to whom those rights and conditions can be issued. According to the invention, the publishing process includes encrypting the digital content and associating a list of  
20 persistent enforceable rights that the author of the content intended for all possible users of the content. This process can be carried out in a secure way to prohibit access to any of the rights or to the content unless intended by the author of the content.

Three entities in particular are employed to publish secure digital  
25 content: a content preparation application 302 that executes on the client 300 and prepares the content for publishing, a digital rights management (DRM) applications program interface (API) 306 that also resides on the client device 300, and a DRM server 320 that is communicatively coupled to the client 300 via a communication network 330 such as the Internet, a local or wide area network, or  
30 a combination thereof.

The content preparation application 302 can be any application that produces digital content. For example, the application 302 can be a word processor or other publisher that produces digital text files, digital music, video, or other such content. The content could also include streamed content, such as  
5 streamed audio/video of a live or taped event, or example. The application 302 is provided with a cryptographic key to encrypt the digital content, thus forming an encrypted digital content file 304, and the user provides rights data to be tightly associated with the encrypted content in the digital content file 304. The rights data includes an identity for each entity that has rights in the digital content, and a  
10 set of rights and conditions for each identified entity.

Such an entity can be, for example, an individual, a class of individuals, or a device. Such rights can include the right to read, edit, copy, print, etc, the digital content. Conditions may include minimum system requirements, date and time limitations, play counts, and the like.

15 The client API 306 passes the encrypted digital content and the rights data to the DRM server 320. Using a process that is described in detail below, the DRM server 320 determines whether it can enforce the rights data and if so the DRM server 320 signs the rights data to form a signed rights label (SRL) 308. In general, however, any trusted entity can sign the rights data, preferably  
20 using a key trusted by the DRM server 320. For example, a client can sign the rights data using a key provided to it by the DRM server 320.

The rights label 308 can include data representing the rights description, the encrypted content key, and the digital signature over the rights description and the encrypted content key. If the DRM server 320 is signing the  
25 right label, it passes the signed rights label 308 back to the client through the client API 306, which stores the signed rights label 308 on the client device 300. The content preparation application 302 then associates the signed rights label 308 with the encrypted digital content file 304, such as for example by concatenation to form a rights managed content file 310. Note, though, that the  
30 SRL 308 could be stored in a known location separate from the content file 304 with a reference to the SRL 308 concatenated to the content file 304 to form the content file 310.

Referring now to Fig. 4, one method for publishing rights managed digital content is shown. At step 402, the application 302 generates a content key (CK) that is used to encrypt the digital content. The content key (CK) is typically a symmetric key although any key can be used to encrypt the digital content. As is known, a symmetric key is employed by a symmetric key algorithm both to encrypt and decrypt. Accordingly, (CK) should be well-hidden when shared between a sender and a receiver.

At step 404, the application 302 encrypts the digital content with (CK) to form encrypted digital content 304 (i.e., (CK(content))). Additionally, rights data corresponding to (CK(content)) is generated, either by the publisher of the content or by another entity. Note that such rights data may be custom rights data or rights data as obtained from a pre-defined template. As was discussed above, the rights data can include a list of entities that will be entitled to consume the content, the specific rights that each of the entities possesses with respect to the content, and any conditions that may be imposed on those rights.

At step 406, the API 306 generates a second encryption key (K2), which is used to encrypt the content key (CK). Preferably, (K2) is also a symmetric key. At step 408, the API 306 encrypts (CK) with (K2) to result in (K2(CK)). At step 410, the API 306 discards (CK), with the result being that (CK) can now be obtained only by decrypting (K2(CK)). To ensure that (CK(content)) is protected to a central DRM server 320 and that all "license requests" for the content are done centrally in accordance with the rights data, the API 306, at step 412, contacts the provided DRM server 320 and retrieves the public key (PU-DRM) thereof. At step 414, the API 306 encrypts (K2) with (PU-DRM) to result in (PU-DRM (K2)). Thus, (CK) can be protected to (PU-DRM)) to ensure that the DRM server 320 is the only entity that will be able to get access to (CK), as is required to decrypt (CK(content)). At step 416, the API 306 encrypts the rights data (i.e., the list of authorized entities and the respective rights and conditions associated with each authorized entities in the list) with (K2) to result in (K2(rightsdata)).

In an alternative embodiment, (CK) can be used to directly encrypt the rights data to result in (CK(rightsdata)), and (PU-DRM) can be used to directly encrypt (CK) to result in (PU-DRM(CK)), thereby foregoing the use of (K2) completely. However, using (K2) to encrypt the rights data and (CK) allows such  
5 (K2) to conform to any particular algorithm that might be amenable to the DRM server, whereas (CK) might be specified by an entity independent from the DRM server and might not be as amenable thereto.

At step 418, the content protection application 302 submits (PU-DRM(K2)) and (K2(rightsdata)) to the DRM server 320 as a rights label for signing.  
10 Alternatively, the client itself can sign the rights data in the manner set forth below. If the rights data is being submitted to the server for signing, then, at step 420, the DRM server 320 accesses the rights data and verifies that it can enforce the rights and conditions in the submitted rights label. To verify that it can enforce the rights data, the DRM server 320 applies the private key (PR-DRM) corresponding to  
15 (PU-DRM) to (PU-DRM(K2)) to result in (K2), and then applies (K2) to (K2(rightsdata)) to result in the rights data in the clear. The server 320 can then do any policy checks to verify that the users, rights, and conditions specified in the rights data are within any policy enforced by the server 320. The server 320 signs the originally submitted rights label including (PU-DRM(K2)) and (K2(rightsdata))  
20 to result in the signed rights label (SRL) 308, where the signature is based on the private key of the DRM server 320 (PR-DRM), and returns the SRL 308 back to the API 306, which then presents the returned SRL 308 to the client application 302.

The SRL 308 is a digitally signed document, which makes it tamper-  
25 resistant. Additionally, the SRL 308 is independent of the actual key type and algorithm used to encrypt the content but maintains the strong 1-1 relation to the content it is protecting. Referring now to Fig. 4A, in one embodiment of the present invention, the SRL 308 may include information on the content that is the basis of the SRL 308, including perhaps an ID of the content; information on the  
30 DRM server that signs the SRL 308, including (PU-DRM(K2)) and referral information such as a URL for locating the DRM server on a network and fall-back information if the URL fails; information describing the SRL 308 itself;

(K2(rightsdata)): (K2(CK)); and a digital signature (S (PR-DRM)), among other things.

By ensuring that a trusted entity signs the rights data to create a signed rights label 308, the DRM server 320 is asserting that it will issue licenses for the content in accordance with the terms set forth by the publisher as described in the rights data of the rights label 308. As should be appreciated, a user is required to obtain a license to render the content, especially inasmuch as the license contains the content key (CK). When a user wants to obtain a license for the encrypted content, the user can present a license request including the SRL 308 for the content and a certificate verifying the user's credentials to the DRM server 320 or other license issuing entity. The license issuing entity can then decrypt (PU-DRM(K2)) and (K2(rightsdata)) to produce the rights data, list all the rights granted by the author (if any) to the license requesting entity, and construct a license with only those specific rights.

As set forth above, upon the application 302 receiving the SRL 308, such application 302 concatenates the signed rights label 308 with the corresponding (CK(content)) 304 to form rights managed digital content. Alternatively, the rights data is stored in a known location, with a reference to that location provided with the encrypted digital content. Thus, a rendering application that is DRM-enabled can discover the signed rights label 308 via the piece of content the rendering application is attempting to render. This discovery triggers the rendering application to initiate a license request against the DRM licensing server 320. Publishing application 302 can store a URL to the DRM licensing server 320, for example, or the DRM licensing server 320 can embed its own URL as a piece of metadata into the rights label before digitally signing it, so that the DRM client API 306 called by the rendering application can identify the correct DRM licensing server 320.

Obtaining a License for the Published Content

Referring now to Fig. 5, a system and method for licensing rights managed digital content is shown. "Licensing," as that term is used herein, refers

170

to a process that an application or service follows to request and receive a license that will enable an entity named in the license to consume the content in accordance with the terms specified in the license. Inputs to the licensing process can include the signed rights label (SRL) 308 associated with the content for  
5 which a license is being requested, and the public key certificate(s) of the entity(s) for which the license is being requested. Note that the entity requesting a license need not necessarily be the entity for which the license is being requested. Typically, a license includes the rights description from the SRL 308, an encrypted key that can decrypt the encrypted content, and a digital signature over the rights  
10 description and the encrypted key to assert legitimacy and prevent tampering.

Preliminarily, the client API 306 forwards the signed rights label 308 of the rights managed content 310 to the DRM server 320 via the communication network 330. As described above, the rights label 308 contains the content key (CK) encrypted according to the public key of the DRM server 320 (PU-DRM)  
15 (i.e., (PU-DRM(CK))). In the process of issuing a license, then, the DRM server 320 applies (PR-DRM) to (PU-DRM(CK)) to obtain (CK). It then uses the public key (PU-ENTITY) in the public key certificate that is passed up in the license request to re-encrypt (CK) (i.e., (PU-ENTITY(CK))). The newly encrypted (PU-ENTITY(CK)) is then placed into the license. Thus, the license can be returned to  
20 the caller without risk of exposing (CK), since only the holder of the private key (PR-ENTITY) corresponding to (PU-ENTITY) can recover (CK) from (PU-ENTITY(CK)). The client API 306 then uses (CK) to decrypt the encrypted content to form decrypted digital content 312. The client application 302 can then use the decrypted digital content 312 according to the rights that are provided in the  
25 license.

Alternatively, and as set forth below in more detail, a client such as the publishing client can for example issue a use license to itself to consume the content.

Turning now to Figs. 6A and 6B, a method for licensing rights  
30 managed digital content is shown. At step 602, a license issuing entity such as a DRM server 320 receives a license request including either a public key certificate or an identity for each of one or more requested licensees. Presumably, if am

identity is specified, the DRM server 320 can procure a corresponding public key certificate from a directory, a database, or the like. If a license is requested for only one licensee, only one certificate or identity is named. If a license is requested for a plurality of licensees, a certificate or an identity can be named for each potential licensee. At step 604, the requesting entity (i.e., the entity making the license request) is authenticated, if desired. At step 606, it is determined whether the entity is allowed to request a license, again if desired.

If, at step 608, the issuing entity determines that the public key certificate is not included in the license request, then the issuing entity uses the specified identity to perform a lookup in a directory service or database for the appropriate public key certificate. If, at step 610, the issuing entity determines that the certificate is in the directory, then, at step 612, the certificate is retrieved. If a certificate cannot be found for a given potential licensee, either in the request or in the directory, then the license server does not generate a license for that potential licensee and, at step 614, an error is returned to the requesting entity.

Assuming the DRM server 320 has a public key certificate for at least one potential licensee, then, at step 616, such DRM server 320 validates the trust of each licensee certificate. If not validated, the DRM server 320 determines that the issuer of the licensee certificate is not in the list of trusted issuers, then the request fails for that licensee, and an error is generated at step 614. Thus, any potential licensee whose certificate is not issued by a trusted issuer would not receive a license.

Additionally, the DRM server 320 preferably performs digital signature validation on all entities in the certificate chain going from the trusted issuer certificates to the individual licensee public key certificates. The process of validating the digital signatures in a chain is a well-known algorithm. If the public key certificate for a given potential licensee does not validate, or a certificate in the chain does not validate, the potential licensee is not trusted, and a license, therefore, is not issued to that potential licensee. Otherwise, at step 618, a license can issue. The process repeats at step 620 until all entities for which a license has been requested have been processed.

As shown in Fig. 6B, the DRM server 320 proceeds to validate the signed rights label 308 that is received in the license request. In one embodiment, the DRM server 320 has a master copy of every rights label signed thereby. At license time (at step 622), then, the DRM server 320 may retrieve a copy of the master rights label. The master rights label could be more up to date than the copy of the rights label sent in the license request, and therefore will be the rights label employed to produce the requested license. If no master rights label is found the DRM server 320, at step 624, determine according to pre-defined policy whether to issue a license based on the rights label in the request. If the policy does not allow, the license request fails at step 626, and an error is returned to the API 306 at step 628.

At step 630, the DRM server 320 validates the SRL 308 and specifically the digital signature thereof. If the SRL 308 does not validate, the license request fails at step 626, and an error is returned to the API 306 at step 628.

After all the validations have occurred, the DRM server constructs a license for each approved license based on the SRL 308. At step 632, the DRM server 320 generates a respective rights description for the license to be issued to each licensee. For each licensee, the DRM server 320 evaluates the identity named in the public key certificate of that licensee against the identities named in the rights description in the rights label. At step 636, the DRM server 320 obtains (PU-DRM(K2)) and (K2(CK)) from the SRL 308 and applies (PR-DRM) to obtain (CK). The issuing entity then re-encrypts (CK) using (PU-ENTITY) from the licensee's public key certificate to result in (PU-ENTITY(CK)). At step 638, the DRM server 320 concatenates the generated rights description with (PU-ENTITY(CK)) and digitally signs the resulting data structure using (PR-DRM) (i.e., S (PR-DRM)). The signed data structure is thus the license for this particular licensee.

At step 640, the DRM server 320 determines that there are no more licenses to generate for the particular request. The generated licenses are then returned to the requesting entity, at step 642, along with an appropriate certificate chain that ties the licenses back to a trusted authority.

### Self-Publishing the Signed Rights Label 308

In one embodiment of the present invention, the SRL 308 may be signed by the requesting / publishing user itself. Accordingly, such user need not  
5 contact the DRM server 320 to obtain an SRL 308 for an associated piece of content. As a result, self-publishing may also be referred to as off-line publishing. In such embodiment, the publishing user should also be able to issue itself a publisher license, especially inasmuch as the self-published content is now DRM-protected and such a publisher license is required to allow the publishing user to  
10 render the now-protected content. It should also be understood that a publishing user may be enabled to issue licenses to other users.

In particular, and referring now to Fig. 7, in the embodiment, an off-line publishing user is first provisioned to off-line publish by receiving from a DRM server 320 an off-line publishing (OLP) certificate 810 including a public key (PU-  
15 OLP) and a corresponding private key (PR-OLP) encrypted according to a public key directly or indirectly accessible to the trusted component 18 (Fig. 11) of the user (PU-ENTITY) to result in (PU-ENTITY(PR-CERT)). Note that (PU-ENTITY) may for example be the public key of the trusted component 18, or may be a public key of the user which is accessible by way of the public key of the trusted  
20 component 18. The OLP certificate 810 should be signed by the private key of the DRM server 320 (PR-DRM) so that such DRM server 320 can verify such OLP certificate, as will be discussed in more detail below.

In addition, the OLP certificate 810 should include a certificate chain from (PU-DRM) back to a trusted authority that is trusted by the trusted  
25 component 18 of the publishing user or of another user so that such trusted component 18 can verify such OLP certificate 810 and any other certificate or license that is associated with such OLP certificate 810, as will be discussed below. Briefly, and as should be understood, a chain of certificates begins with a root certificate signed by the private key of a trusted authority and having the  
30 public key of the next certificate in the chain. Each intermediate certificate in the chain, then, is signed by the private key corresponding to the public key of the

previous certificate in the chain, and has the public key of the next certificate in the chain. Finally, the certificate or license to which the chain is attached is signed by the private key corresponding to the public key of the last certificate in the chain.

Thus, to verify the certificate or license to which the chain is  
5 attached, knowledge of the public key corresponding to the private key of the trusted authority is gained, and such public key of the trusted authority is employed to verify the signature of the root certificate in the chain. Presuming the root certificate signature verifies, then, the public key from the root certificate is obtained and employed to verify the signature of the first intermediate certificate in  
10 the chain. The process repeats serially through the chain until every signature thereof is verified, and then the public key from the last intermediate certificate in the chain is obtained and employed to verify the signature of the certificate or license to which the chain is attached.

As should be appreciated, the OLP certificate 810 creates a link in  
15 the chain of trust between content 304 that is to be published off-line and the DRM server 320 that will issue a license for the content 304. The OLP certificate 810 may be created based on an XML/XrML language or any other appropriate language.

As should also be appreciated, the OLP certificate 810 and attached  
20 certificate chain authorizes the publishing user to self-publish. As may further be appreciated, the key pair (PU-OLP, PR-OLP) are separate from (PU-ENTITY, PR-ENTITY), and are employed specifically for self-publishing. Note that the key pair (PU-OLP, PR-OLP) may be dispensed with, in which case the DRM certificate 810 includes only the public key of the user (PU-ENTITY) and is signed by the private  
25 key of the DRM server 320 (PR-DRM) so that such DRM server 320 can verify same.

Self-publishing differs from publishing as shown in Fig. 4 in that the user essentially takes the place of the DRM server 320 with regard to steps performed thereby. Significantly, the user signs the submitted rights label  
30 including (PU-DRM(K2)) and (K2(rightsdata)) or including (PU-DRM(CK)) and (CK(rightsdata)) (the latter being shown in Figs. 7 and 8) with (PR-OLP) as obtained from the DRM certificate 810 (i.e., S (PR-OLP)) to result in the signed

rights label (SRL) 308. The trusted component 18 client in using the OLP certificate 810 typically verifies same based on the attached certificate chain. As should be appreciated, the trusted component 18 of the user obtains (PR-OLP) from the OLP certificate 810 by obtaining (PU-ENTITY(PR-OLP)) from such OLP certificate 810 and applying (PR-ENTITY) thereto. Note, though, that the publishing user cannot verify that the DRM server 320 can enforce the rights in a self-published SRL 308. Accordingly, the DRM server 320 itself should perform the verification at the time a license is requested based on the self-published SRL 308.

10               Once the publishing user self-publishes the SRL 308, the user concatenates such self-published SRL 308 and the OLP certificate 810 employed to produce same to the content 304, and such content 304 with SRL 308 and DRM certificate 810 is distributed as the rights managed content 310 to another user. Thereafter, the other user requests and obtains a license for the content 15 304/310 from the DRM server 320 in substantially the same manner as shown in Figs. 6A and 6B. Here, though, the license-requesting user submits to the DRM server 320 both the self-published SRL 308 and the OLP certificate 810 as concatenated to the content 304. The DRM server 320 then verifies S (PR-DRM) in the OLP certificate 810 based on the corresponding (PU-DRM), and obtains 20 (PU-OLP) from the DRM certificate 810. The DRM server 320 then verifies S (PR-OLP) in the SRL 308 based on the obtained (PU-CERT), and continues as before. Note, though, that since the publishing user did not verify that the DRM server 320 can enforce the rights in the SRL 308, and as was set forth above, the DRM server 320 itself should perform the verification at this time.

25               Note, too, that the DRM server 320 need only verify S (PR-DRM) in the OLP certificate 810, since presumably it trusts itself. Accordingly, the associated certificate chain from the OLP certificate 810 need not necessarily be sent to the DRM server 320 along with such OLP certificate 810, unless of course the chain is otherwise necessary, such as for example if the chain itself is at least 30 partially the basis for S (PR-DRM).

Importantly, though, the publishing user ought to be able to render the now-protected content 304/310 without having to go to the DRM server 320 for

a license. Put another way, a publishing user that off-line publishes content 304/310 without going to the DRM server 320 based on an OLP certificate 810 ought to be able to also issue itself a license in an off-line manner without going to the DRM server 320 so that such user can render the off-line published content 304/310. Accordingly, a publishing user can continue to work with self-published content 310 without any connectivity to a DRM server 320.

In one embodiment of the present invention, then, and turning now to Fig. 8, a publishing user issues itself an off-line publisher license 820 signed by (PR-OLP) and based on the self-published SRL 308, and including the OLP certificate 810 and certificate chain thereof. Presumably, the publisher license 820 grants the publishing user full access to the self-published content 310, although a lesser amount of access could also be granted. The publisher license 820 may be written in an XML/XrML language or another language, as is the case with other DRM licenses. As should be appreciated, the publisher license 820 includes the content key (CK) encrypted according to (PU-ENTITY), which may be obtained by the trusted component 18 of the user's computing device 14, to form (PU-ENTITY(CK)).

The chain for the publisher license 820 thus goes from such license 820 to the OLP certificate 810 and then back to a root certificate from a trusted authority, perhaps by way of one or more intermediate certificates. Since the trusted component 18 of the user presumably can gain the public key corresponding to the private key of the trusted authority that was employed to sign the root certificate, the trusted component 18 can itself verify the publisher license 820 by way of the certificate chain thereof, and upon verification can then obtain (PU-ENTITY(CK)) therefrom, apply (PR-ENTITY) thereto to obtain (CK), and apply (CK) to (CK(content)) to result in the content 304 for purposes of rendering thereof. As a result, the publishing user can continue to work with content 310 off-line published thereby while remaining offline.

In accordance with the above, then, and referring now to Fig. 9, a publishing user off-line publishes content 304/310 and issues itself an off-line publisher license 820 for such content 304/310 in the following manner.

‡ Preliminarily, and as should be appreciated, the content 304 is developed in an appropriate manner and encrypted according to a content key (CK) (step 901), and the publishing user creates a rights label for the content 304 with appropriate information {(PU-DRM(CK)) and (CK(rightsdata)), e.g.} (step 5 903). Thereafter, the publishing user, which presumably is already in possession of an OLP certificate 810 from the DRM server 320, obtains such OLP certificate 810 (step 905) and verifies same based on the signature thereof and the chain of certificates that lead back to the root authority (step 907). As should be appreciated, such verification is actually performed by a trusted component 18 on 10 a computing device 14 of the publishing user. Assuming the verification is successful, then, the publishing user / trusted component 18 (hereinafter 'the publishing user') retrieves (PU-ENTITY(PR-OLP)) from the OLP certificate 810 (step 909), applies (PR-ENTITY) to (PU-ENTITY(PR-OLP)) to obtain (PR-OLP) (step 911), and then signs the created rights label with such (PR-OLP) to create 15 an SRL 308 (step 913).

Thereafter, the publishing user concatenates such SRL 308 and the OLP certificate 810 employed to produce same to the content 304 to form the self-published content 310 (step 915), and such rights managed content 310 is therefore able to be distributed to another user. For the publishing user to 20 continue using or rendering the content 310, however, such publishing user must issue itself a corresponding off-line publisher license 820.

Thus, the publishing user creates a publisher license 820 by defining appropriate rights data for itself and encrypting same according to the content key (CK) to result in (CK(rightsdata)) (step 917). Note here that such rights data may 25 be obtained from the SRL 308 from the content 310, may be some default set of rights data granting the publishing user partial or full access to the self-published content 310, or may be derived from another source. In addition, the publishing user encrypts the content key (CK) according to (PU-ENTITY) to form (PU-ENTITY(CK)) (step 919). Such (CK(rightsdata)) and (PU-ENTITY(CK)) are then 30 formatted into the publisher license 820 (step 921), the OLP certificate 810 and chain of certificates thereof is attached (step 923) and such publisher license 820 is signed based on (PR-OLP) as was obtained in step 911 (step 925). Note here

that the content 304 (i.e., (CK(content))), the publishing license 820, and the OLP certificate in combination form a chain 830 of digital items back to the trusted authority.

For the publishing user to render the published content 310, then,  
5 and turning now to Fig. 10, such publishing user need not contact the DRM server 320, but instead gains the public key corresponding to the private key of the trusted authority that was employed to sign the root certificate (step 1001), verifies the root certificate (step 1003), and then verifies each intermediate certificate in the chain (step 1005) by, for each such intermediate certificate, obtaining the  
10 public key from the previous certificate and employing same to verify the signature of such certificate. Thereafter, (PU-DRM) from the last certificate in the chain is employed to verify the signature of the OLP certificate 810 (i.e., S (PR-DRM)) (step 1007), (PU-OLP) is obtained from the OLP certificate 810 (step 1009), and such (PU-OLP) is employed to verify the signature of the publisher license 820  
15 (i.e., S (PR-OLP)) (step 1010).

Once the publisher license 820 is verified, then, (CK(rightsdata)) and (PU-ENTITY(CK)) are retrieved from same (step 1011), (PR-ENTITY) is applied to (PU-ENTITY(CK)) to result in (CK) (step 1013), and (CK) is applied to (CK(rightsdata)) to result in the rights data (step 1015). As should now be  
20 appreciated, the rights data is reviewed by the trusted component 18 of the computing device 14 of the publishing user to determine that such rights data allows rendering in the manner sought (step 1017), such trusted component 18 thus applies (CK) to (CK(content)) from the content 310 to result in the content (step 1019), and such content is then forwarded to an appropriate rendering  
25 application for actual rendering (step 1021). Thus, the steps of Fig. 10 in effect traverse the chain 830 of digital items from the trusted authority to the content 304.

Note that the trusted component 18 could conceivably apply (CK) to (CK(content)) to result in the content without first reviewing the rights data and  
30 regardless of what the rights data may allow or disallow, but is trusted and has been constructed to in fact produce the content only after reviewing the rights data and satisfying itself that the rights data allows the rendering of such content.

Once again, as a result of having the publisher license 820, the publishing user can continue to work with content 310 off-line published thereby while remaining offline inasmuch as the DRM server 320 need not be contacted.

5 \*Enrollment and Sub-Enrollment of DRM Servers

In the architecture as seen in Fig. 3, only a single DRM server 320 is shown. However, and as may be appreciated, such architecture can and likely does include multiple DRM servers 320. In particular, and in one embodiment of the present invention, such architecture includes a distributed network of DRM  
10 servers 320. Each of such DRM servers 320 may have any particular function and all of the DRM servers 320 may be organized in any appropriate manner without departing from the spirit and scope of the present invention.

For example, and turning now to Fig. 12, a particular organization may have one or more user-level DRM servers 320 for the purpose of signing  
15 rights labels to produce SRLs 308, issuing licenses 16, granting publishing licenses 320, issuing certificates to users, issuing certificates to computing devices 14, and the like. Each such user-level DRM server 320 may be geographically assigned or may be assigned based on function or load, for example. Likewise, to oversee multiple user-level DRM servers 320, an  
20 organization may have one or more management DRM servers 320. Such organization-based DRM servers 320 may be located behind an organization firewall if so desired.

In addition to organization-based DRM servers 320, there may also be trans-organization DRM servers 320 that facilitate inter-organization DRM  
25 functionality. For example, such trans-organization DRM servers 320 may allow a pair of organizations to share certain DRM content 12. Also, there may be a network of overseer DRM servers 320 that enable all other DRM servers 320. For example, such overseer DRM servers 320 may oversee and maintain all other DRM servers 320 and provide appropriate linkage for all other DRM servers 320  
30 back to the root or trusted authority that is the basis for the chain of certificates set

forth previously. Such non-organization-based DRM servers 320 are likely not located behind any organization firewall.

Critically, each DRM server 320 in the architecture of Fig. 12 must be able to prove that it is to be trusted. Thus, and as should now be appreciated from the above chain of certificates discussion, each DRM server 320 upon entering the architecture is provided with an enrollment certificate 1310, as is seen from Fig. 13. Significantly, and in one embodiment of the present invention, the enrollment certificate 1310 is provided to the entering DRM server 320 (hereinafter "the DRM-E server 320") by another 'enrolling' DRM server 320 already in the architecture (hereinafter "the DRM-R server 320"). Also significantly, attached to the provided enrollment certificate 1310 from the enrolling DRM-R server 320 is a chain of certificates 1320 including the enrollment certificate 1310 of the enrolling DRM server 320, the enrollment certificate 1310 of the DRM server 320 that enrolled the enrolling DRM-R server 320, and so on all the way back to a root DRM server 320. Such root DRM server 320 may represent the root or trusted authority, or the chain of certificates 1320 may extend beyond to reach the root or trusted authority. As should now be appreciated, such enrollment certificate 1310 and the chain of certificates 1320 in combination form the chain of certificates that are attached to an OLP certificate 810 provided by an enrolled or entered DRM-E server 320 to a publishing user, such as that shown in Fig. 8.

In one embodiment of the present invention, the enrollment certificate 1310 provided to a DRM-E server 320 by a DRM-R server 320 is in a form such as an XrML 1.2 based certificate. As may be appreciated, such type of certificate 1310 is not independently proffered by any third party, and thus such type of certificate 1310 does not represent any sort of independent vouching by a third party for the holder of such certificate 1310.

In one embodiment of the present invention, the method with which a particular DRM-E server 320 is enrolled into the architecture depends on whether the enrolling DRM-R server 320 knows or has reason to trust the entering DRM-E server 320. If not, the DRM-E server 320 should be required to prove to the DRM-R server 320 that it is trustworthy and will enforce the DRM architecture. If so, the DRM-E server 320 should not be required to prove to the DRM-R server

181

320 that it is trustworthy, at least not to the same degree. Thus, a non-trusting / non-knowing DRM-R server 320 'enrolls' a DRM-E server 320, while a knowing / trusting DRM-R server 320 'sub-enrolls' a DRM-E server 320.

Typically, a DRM-R server 320 knows / trusts a DRM-E server 320 if  
 5 both are operated by or for the benefit of the same organization, although knowledge / trust may also arise from other situations without departing from the spirit and scope of the present invention. Thus, the method with which a particular DRM-E server 320 is enrolled into the architecture typically depends on whether the enrolling DRM-R server 320 is organizationally based or non-organizationally  
 10 based. As a result, a non-organizationally based DRM-R server 320 'enrolls' a DRM-E server 320, while an organizationally based DRM-R server 320 'sub-enrolls' a DRM-E server 320.

#### Enrollment

15 In one embodiment of the present invention, and turning now to Fig. 14, a non-knowing / non-trusting DRM-R server 320 enrolls a DRM-E server 320 in the following manner.

Preliminary, it is to be appreciated that the DRM-E server 320 wishing to be enrolled by the non-knowing / non-trusting DRM-R server 320 is  
 20 likely not known to such DRM-R server 320. Accordingly, and in one embodiment of the present invention, the DRM-E server 320 must procure a vouching certificate 1330 from a third party willing to vouch for such DRM-E server 320 (step 1401). Typically, such third party is an independent certificate-issuing agent that is trusted by the DRM-R server 320 to perform such vouching, such as for  
 25 example VERISIGN Corporation of Mountain view, California. Such vouching certificate 1330 may for example be in a form such as an X.509 certificate. Note that in the DRM-R server 320 relying on the trusted third party to vouch for the DRM-E server 320, the liability of such DRM-R server 320 for any bad acts of the DRM-E server 320 is mitigated.

30 As should be appreciated and as is typical, and as is also seen in Fig. 13, the vouching certificate 1330 incorporates therein a public key (PU-V) and

a corresponding private key (PR-V), is signed by the trusted third party, and may be accompanied by a chain of certificates leading to a known root for purposes of validation. As is also typical, (PR-V) within the vouching certificate 1330 is protected in a manner accessible to the vouched-for DRM-E server 320 that is the basis of the vouching certificate 1330. For example, and as seen in Fig. 13, (PR-V) could be encrypted according to an appropriate public key.

Within the DRM architecture, the entering DRM-E server 320 must have a unique identity. Here it is to be understood that the DRM identity is likely apart from (PU-V, PR-V), although the DRM identity may also coincide with such (PU-V, PR-V) without departing from the spirit and scope of the present invention. Accordingly, to establish such identity, such DRM-E server 320 generates or obtains a new public / private key pair (PU-E, PR-E) (step 1403). Also, within the DRM architecture, the enrolling DRM-E server 320 should decide which entities can revoke the authority thereof to participate. Accordingly, such DRM-E server 320 identifies each such revoking entity in a list, perhaps by way of a public key thereof (step 1405).

The DRM-E server 320 should be able to establish to the enrolling DRM-R server 320 that such DRM-E server in fact owns the vouching certificate 1330 that was obtained at step 1401. Accordingly, the DRM-E server 320 either employs (PR-V) from the vouching certificate 1330 to encrypt (PU-E) to result in (PR-V(PU-E)) as an ownership indicia, or signs (PU-E) with (PR-V) to result in (PU-E) S (PR-V) as the ownership indicia (step 1407). In either instance, applying (PU-V) to decrypt (PU-E) or verify the signature establishes possession of (PR-V) and therefore the vouching certificate 1330.

Thus far, the DRM-E server 320 has the vouching certificate 1330, (PU-E) and (PR-E), the revocation authority list, and (PR-V(PU-E)) or (PU-E) S (PR-V) as the ownership indicia. To request enrollment, then, such DRM-E server 320 sends the vouching certificate 1330, (PU-E), the revocation authority list, and (PR-V(PU-E)) or (PU-E) S (PR-V) as the ownership indicia to the DRM-R server 320 (step 1409), and the DRM-R server 320 proceeds to enroll such requesting DRM-E server 320. Note that the request or a part thereof may be in the form of a certificate signed by (PR-E).

In particular, the DRM-R server 320 validates the vouching certificate 1330 based on the signature thereof by the trusted third party and the chain of certificates leading to the known root (step 1411). Thus, the DRM-R server 320 establishes that the DRM-E server 320 has been vouched for. Also, 5 the DRM-R server 320 verifies the ownership indicia by applying (PU-V) from the request to either decrypt (PU-E) or verify the signature and thus establish possession of (PR-V) and therefore the vouching certificate 1330 in the request (step 1410). In addition, and significantly, the DRM-R server 320 performs any custom logic necessary to decide whether to honor the request (step 1413). Such 10 custom logic may be any appropriate logic without departing from the spirit and scope of the present invention, and may for example include a background check on the DRM-E server 320 and/or its operator, a determination of whether the DRM-E server 320 has a current trusted component 18 and/or operating system or the like, a determination of whether the DRM-E server 320 is on a revocation 15 list or other watch list, and the like.

Assuming the custom logic permits the request to be honored, then, and in one embodiment of the present invention, the DRM-R server 320 generates the enrollment certificate 1310 for the DRM-E server 320 (step 1415). In particular, and as seen in Fig. 13, the DRM-R server 320 incorporates within the 20 enrollment certificate 1310:

- an identifier of the DRM-R server 320, such as a public key thereof (PU-R);
- an identifier of the DRM-E server 320, such as (PU-E);
- identifying indicia from the vouching certificate 1330 including 25 the trusted third party that issued same, a serial number from the vouching certificate 1330, and the issuee as identified within the vouching certificate 1330;
- any validity range information specifying a range during which the enrollment certificate 1310 is valid, such as for example a 30 date range;
- the revocation authority list;

- a signature based on a private key of the DRM-R server 320 (PR-R) corresponding to (PU-R);
- and any other appropriate information.

5           ↘ Such other appropriate information may include but is not limited to: the time the certificate was issued; an indication of what sort of DRM activities an enrolled server is allowed to perform such as for example all activities, account activation only, sign rights labels only, issue content licenses only, and combinations thereof; and an allowed time-range for performing DRM activities.

10       Note that the allowed time-range is different from the validity range in that the current time must lie within the validity range to honor any certificate that includes the enrollment certificate 1310 in the certificate chain. In contrast, the issued time of child certificates must fall within allowed time-range of the parent certificate to perform DRM activities.

15           As should be appreciated, in generating the enrollment certificate 1310, the DRM-R server 320 may initially generate certificate information and then allow custom logic to generate additional information or modify existing information. Such custom logic may for example ensure that the DRM-R server 320 includes appropriate information, or may enforce pre-defined DRM  
20       architecture policy. Of course, the signature of the enrollment certificate 1310 is created after any such custom logic is performed. As should also be appreciated, the DRM-R server 320 attaches the chain of certificates 1320 that leads back to the trusted root authority to the generated enrollment certificate 1310 so that the generated enrollment certificate 1310 may be validated based on such chain of  
25       certificates 1320.

              Note in particular that the identifying indicia from the vouching certificate 1330 as placed within the enrollment certificate 1310 will always travel with such enrollment certificate 1310 and acts as a bridge to the vouching certificate 1330. Thus, and again, such identifying indicia shows to the world that  
30       the DRM-R server 320 is relying on the trusted third party issuer of the vouching certificate 1330 to vouch for the DRM-E server 320, and the liability of such DRM-R server 320 for any bad acts of the DRM-E server 320 is mitigated.

Once the DRM-R server 320 has successfully generated the enrollment certificate 1310 with the attached chain of certificates 1320, the DRM-R server 320 then returns same to the requesting DRM-E server 320 (step 1417), and the now-enrolled DRM-E server 320 stores same in an appropriate location for future use (step 1419). As was alluded to above, (PU-E) in the enrollment certificate 1310 and the corresponding (PR-E) are the public / private key pair that the DRM-E server 320 will use as (PU-DRM) and (PR-DRM) when signing a rights label to produce an SRL 308, issuing an OLP certificate 810, and otherwise participating within the DRM architecture. Accordingly, such enrollment certificate 1310 and the chain of certificates 1320 in combination form the chain of certificates that are attached to such OLP certificate 810 and the like.

#### Sub-Enrollment

In one embodiment of the present invention, and turning now to Fig. 15, a knowing / trusting DRM-R server 320 sub-enrolls a DRM-E server 320 in the following manner.

Preliminary, it is to be appreciated that the DRM-E server 320 wishing to be sub-enrolled by the knowing / trusting DRM-R server 320 should still be required to identify itself to such DRM-R server 320 inasmuch as such knowledge or trust may not be complete. However, such identification requirement need not rise to the level of a proffer by a trusted third party inasmuch as the DRM-R server 320 does have some knowledge / trust of the DRM-E server. Accordingly, and in one embodiment of the present invention, the DRM-E server 320 obtains or is provided with some sort of credentials 1340 (Fig. 13) that are recognizable by and expected to be honored by the DRM-R server 320, and that identify the DRM-E server 320 to the satisfaction of the DRM-R server 320 (step 1501).

If both the DRM-R and DRM-E servers 320 are within the same organization, such credentials 1340 may be organization-based credentials, such as for example a network ID if both servers 320 are on a common network, a domain ID if both servers 320 share a common domain, or the like. If both the

DRM-R and DRM-E servers 320 are not within the same organization, such credentials 1340 may still be a network ID if both servers 320 are on a common network, a domain ID if both servers 320 share a common domain, or the like, or may be other credentials such as for example credentials issued by a third party  
5 and recognized by the DRM-R server 320.

Note that in the present situation, the DRM-R server 320 is not relying on a trusted third party to vouch for the DRM-E server 320, and therefore the liability of such DRM-R server 320 for any bad acts of the DRM-E server 320 is not as mitigated. Nevertheless, the DRM-R server 320 is willing to take such a  
10 risk based on knowledge of or trust in the DRM-E server 320 to not in fact perform such bad acts.

As before, within the DRM architecture, the entering DRM-E server 320 must have a unique identity. Here it is to be understood that the DRM identity is likely apart from the credentials 1340, although the DRM identity may also  
15 coincide with the credentials 1340 without departing from the spirit and scope of the present invention. Accordingly, to establish such identity, such DRM-E server 320 generates or obtains a new:public / private key pair (PU-E, PR-E) (step 1503). Also as before, within the DRM architecture, the sub-enrolling DRM-E server 320 should decide which entities can revoke the authority thereof to participate.  
20 Accordingly, such DRM-E server 320 identifies each such revoking entity in a list, perhaps by way of a public key thereof (step 1505).

Thus far, the DRM-E server 320 has the credentials 1340, (PU-E) and (PR-E), and the revocation authority list. To request sub-enrollment, then, such DRM-E server 320 sends the credentials 1340, (PU-E), and the revocation  
25 authority list to the DRM-R server 320 (step 1507), and the DRM-R server 320 proceeds to sub-enroll such requesting DRM-E server 320. Note that as before, the request or a part thereof may be in the form of a certificate signed by (PR-E).

In particular, the DRM-R server 320 validates the credentials 1340 based on whatever logic or resources are necessary and available to so validate  
30 (step 1509). Thus, the DRM-R server 320 establishes based on the validated credentials 1340 that the DRM-E server 320 is to be trusted to honor and obey the

DRM architecture. In addition, and as before, the DRM-R server 320 performs any custom logic necessary to decide whether to honor the request (step 1511).

Assuming the custom logic permits the request to be honored, then, and in one embodiment of the present invention, the DRM-R server 320 generates  
5 a sub-enrollment certificate 1310 for the DRM-E server 320 (step 1513). In particular, and as seen in Fig. 13, the DRM-R server 320 incorporates within the sub-enrollment certificate 1310:

- an identifier of the DRM-R server 320, such as a public key thereof (PU-R);
- 10 - an identifier of the DRM-E server 320, such as (PU-E);
- the credentials 1340 or a reference thereto;
- any validity range information specifying a range during which the sub-enrollment certificate 1310 is valid, such as for example a date range;
- 15 - the revocation authority list;
- a signature based on a private key of the DRM-R server 320 (PR-R) corresponding to (PU-R);
- and any other appropriate information.

As before, in generating the sub-enrollment certificate 1310, the  
20 DRM-R server 320 may initially generate certificate information and then allow custom logic to generate additional information or modify existing information. Again, the signature of the sub-enrollment certificate 1310 is created after any such custom logic is performed. As before, the DRM-R server 320 attaches the chain of certificates 1320 that leads back to the trusted root authority to the  
25 generated sub-enrollment certificate 1310 so that the generated sub-enrollment certificate 1310 may be validated based on such chain of certificates 1320.

Note here that the credentials 1340 or reference thereto are not believed to be especially necessary, but may nevertheless be included for completeness. Note too that the sub-enrollment certificate 1310 contains no  
30 identifying indicia from a vouching certificate 1330 inasmuch as no vouching certificate was required in the present sub-enrollment scenario.

Once the DRM-R server 320 has successfully generated the sub-enrollment certificate 1310 with the attached chain of certificates 1320, the DRM-R server 320 then returns same to the requesting DRM-E server 320 (step 1515), and the now-sub-enrolled DRM-E server 320 stores same in an appropriate  
 5 location for future use (step 1517). As before, (PU-E) in the sub-enrollment certificate 1310 and the corresponding (PR-E) are the public / private key pair that the DRM-E server 320 will use as (PU-DRM) and (PR-DRM) when signing a rights label to produce an SRL 308, issuing an OLP certificate 810, and otherwise participating within the DRM architecture. Accordingly, such sub-enrollment  
 10 certificate 1310 and the chain of certificates 1320 in combination form the chain of certificates that are attached to such OLP certificate 810 and the like.

#### Conclusion

The programming necessary to effectuate the processes performed  
 15 in connection with the present invention is relatively straight-forward and should be apparent to the relevant programming public. Accordingly, such programming is not attached hereto. Any particular programming, then, may be employed to effectuate the present invention without departing from the spirit and scope thereof.

20 In the present invention, a digital rights management (DRM) and enforcement architecture and method allow the controlled rendering or playing of arbitrary forms of digital content, where such control is flexible and definable by the content owner / developer of such digital content. The architecture allows and facilitates such controlled rendering, especially in an office or organization  
 25 environment or the like where documents are to be shared amongst a defined group of individuals or classes of individuals. Such architecture includes a mechanism to enroll / sub-enroll approval-granting DRM servers 320 into the architecture.

It should be appreciated that changes could be made to the  
 30 embodiments described above without departing from the inventive concepts thereof. For example, if a license or rights label is signed based on rights data

therein, such rights data need not necessarily be encrypted. Likewise, in requesting and constructing an enrollment or sub-enrollment certificate 1310, the revocation authority list and other similar information need not necessarily be employed. It should be understood, therefore, that this invention is not limited to

5 the particular embodiments disclosed, but it is intended to cover modifications within the spirit and scope of the present invention as defined by the appended claims.

## CLAIMS

5

1. A method in combination with a Digital Rights Management (DRM) system having a plurality of DRM servers performing DRM functionality, the method for an entering DRM-E server to be enrolled into the system by an enrolling DRM-R server such that the entering DRM-E server is to be trusted  
10 within the system and comprising:
- the DRM-E server procuring a public / private key pair (PU-E, PR-E) for identifying such DRM-E server within the DRM system;
  - the DRM-E server procuring a proffering identification thereof;
  - the DRM-E server sending an enrollment request to the DRM-  
15 R server, the request including the proffering identification and (PU-E);
  - the DRM-R server validating the proffering identification;
  - the DRM-R server, if the request is to be honored, generating a digital enrollment certificate for the DRM-E server to enroll such DRM-E server into the DRM system, the generated enrollment certificate being based at least in  
20 part on (PU-E);
  - the DRM-R server returning the generated enrollment certificate to the requesting DRM-E server; and
  - the now-enrolled DRM-E server storing the returned enrollment certificate in an appropriate location for future use, the DRM-E server  
25 with the enrollment certificate able to employ same to issue DRM documents within the DRM system.

2. The method of claim 1 wherein the DRM-R server has no existing basis for trusting the DRM-E server, the method comprising:
- 30 the DRM-E server procuring a proffering identification thereof comprising a vouching certificate from a party willing to vouch for such DRM-E

server, the vouching certificate incorporating therein a public key (PU-V) and a corresponding private key (PR-V);

the DRM-E server employing (PU-E) and (PR-V) to formulate ownership indicia to show that the DRM-E server owns the vouching certificate;

5 the DRM-E server sending an enrollment request to the DRM-R server, the request including the vouching certificate, (PU-E), and the ownership indicia;

the DRM-R server validating the vouching certificate;

the DRM-R server verifying the ownership indicia; and

10 the DRM-R server, if the request is to be honored, generating a digital enrollment certificate for the DRM-E server to enroll such DRM-E server into the DRM system, the generated enrollment certificate being based at least in part on the vouching certificate and (PU-E).

15 3. The method of claim 2 comprising the DRM-E server procuring the vouching certificate from an independent certificate-issuing agent that is trusted by and relied upon by the DRM-R server to perform such vouching.

20 4. The method of claim 2 comprising the DRM-E server procuring an X.509 vouching certificate.

5. The method of claim 2 comprising:  
the DRM-E server procuring the vouching certificate signed by the vouching party and accompanied by a chain of certificates leading to a  
25 known root for purposes of validation; and  
the DRM-R server validating the vouching certificate based on the signature thereof by the vouching party and the chain of certificates to establish that the DRM-E server has been vouched for.

30 6. The method of claim 2 comprising:

the DRM-E server performing one of employing (PR-V) to encrypt (PU-E) to result in (PR-V(PU-E)) as the ownership indicia, or signing (PU-E) with (PR-V) to result in (PU-E) S (PR-V) as the ownership indicia; and

the DRM-R server verifying the ownership indicia by applying  
5 (PU-V) from the request to decrypt (PU-E) or verify the signature to establish that the DRM-E server owns (PR-V) and therefore the vouching certificate.

7. The method of claim 2 comprising the DRM-R server generating the enrollment certificate to include (PU-E) as an identifier of the DRM-  
10 E server, identifying indicia to identify the vouching certificate, and a signature based on a private key of the DRM-R server, whereby the identifying indicia for the vouching certificate in the enrollment certificate acts as a bridge to the vouching certificate and shows that the DRM-R server is trusting and relying on the vouching party to vouch for the DRM-E server.

15

8. The method of claim 7 comprising the DRM-R server generating the enrollment certificate to further include a public key of the DRM-R server as an identifier thereof.

20

9. The method of claim 7 comprising the DRM-R server generating the enrollment certificate to further include validity range information specifying a range during which the enrollment certificate is valid.

10. The method of claim 1 wherein the DRM-R server has an  
25 existing basis for trusting the DRM-E server, the method comprising:

the DRM-E server procuring a proffering identification thereof comprising credentials recognizable to and expected to be honored by the DRM-R server;

the DRM-E server sending an enrollment request to the DRM-  
30 R server, the request including the credentials and (PU-E);

the DRM-R server validating the credentials; and

the DRM-R server, if the request is to be honored, generating a digital enrollment certificate for the DRM-E server to enroll such DRM-E server into the DRM system, the generated enrollment certificate being based at least in part on the credentials and (PU-E).

5

11. The method of claim 10 comprising the DRM-E server procuring credentials selected from a group consisting of a network ID or a domain ID, and credentials issued by a third party.

10

12. The method of claim 10 comprising the DRM-R server generating the enrollment certificate to include (PU-E) as an identifier of the DRM-E server, identifying indicia to identify the credentials, and a signature based on a private key of the DRM-R server.

15

13. The method of claim 12 comprising the DRM-R server generating the enrollment certificate to further include a public key of the DRM-R server as an identifier thereof.

20

14. The method of claim 12 comprising the DRM-R server generating the enrollment certificate to further include validity range information specifying a range during which the enrollment certificate is valid.

25

15. The method of claim 1 further comprising the DRM-R server performing custom logic to decide whether to honor the request.

30

16. The method of claim 15 comprising the DRM-R server performing custom logic selected from a group consisting of performing a background check on the DRM-E server and/or its operator, determining whether the DRM-E server and/or a portion thereof is current, determining whether the DRM-E server is on a revocation list or watch list, and combinations thereof.

17. The method of claim 1 comprising the DRM-R server generating the enrollment certificate to include (PU-E) as an identifier of the DRM-E server, and a signature based on a private key of the DRM-R server.

5 18. The method of claim 17 comprising the DRM-R server generating the enrollment certificate to further include a public key of the DRM-R server as an identifier thereof.

10 19. The method of claim 17 comprising the DRM-R server generating the enrollment certificate to further include validity range information specifying a range during which the enrollment certificate is valid.

15 20. The method of claim 17 comprising the DRM-R server generating the enrollment certificate to further include identifying indicia to identify the proffering identification.

20 21. The method of claim 1 comprising the DRM-R server generating the enrollment certificate by employing custom logic to generate at least a portion of the information in the enrollment certificate.

22. The method of claim 1 further comprising the DRM-R server attaching to the generated enrollment certificate a chain of certificates that leads back to a trusted root authority so that the generated enrollment certificate may be validated based on such chain of certificates.

25 23. The method of claim 1 further comprising the DRM-E server identifying in a revoking authority list at least one entity with authority to revoke enrollment of such DRM-E server within the DRM system, comprising the DRM-E server sending an enrollment request to the DRM-R server, the request including  
30 the proffering identification, (PU-E), and the revocation authority list, and comprising the DRM-R server, if the request is to be honored, generating a digital enrollment certificate for the DRM-E server to enroll such DRM-E server into the

DRM system, the generated enrollment certificate being based at least in part on (PU-E) and the revocation authority list.

24. The method of claim 23 comprising the DRM-E server  
5 identifying each entity in the revocation authority list by way of a public key thereof.

25. The method of claim 23 comprising the DRM-R server  
generating the enrollment certificate to include (PU-E) as an identifier of the DRM-  
10 E server, the revocation authority list from the request, and a signature based on a private key of the DRM-R server.

26. The method of claim 1 comprising the DRM-R server  
generating an XrML enrollment certificate.

15

27. A method in combination with a Digital Rights Management (DRM) system having a plurality of DRM servers performing DRM functionality, the method for an entering DRM-E server to be enrolled into the system by an enrolling DRM-R server such that the entering DRM-E server is to be trusted  
20 within the system and comprising:

the DRM-E server procuring a public / private key pair (PU-E, PR-E) for identifying such DRM-E server within the DRM system;

the DRM-E server procuring a proffering identification thereof;

the DRM-E server sending an enrollment request to the DRM-  
25 R server, the request including the proffering identification and (PU-E), the DRM-R server validating the proffering identification, if the request is to be honored, generating a digital enrollment certificate for the DRM-E server to enroll such DRM-E server into the DRM system, the generated enrollment certificate being based at least in part on (PU-E), and returning the generated enrollment certificate  
30 to the requesting DRM-E server; and

the now-enrolled DRM-E server storing the returned enrollment certificate in an appropriate location for future use, the DRM-E server

with the enrollment certificate able to employ same to issue DRM documents within the DRM system.

28. The method of claim 27 wherein the DRM-R server has no  
5 existing basis for trusting the DRM-E server, the method comprising:  
the DRM-E server procuring a proffering identification thereof  
comprising a vouching certificate from a party willing to vouch for such DRM-E  
server, the vouching certificate incorporating therein a public key (PU-V) and a  
corresponding private key (PR-V);  
10 the DRM-E server employing (PU-E) and (PR-V) to formulate  
ownership indicia to show that the DRM-E server owns the vouching certificate;  
the DRM-E server sending an enrollment request to the DRM-  
R server, the request including the vouching certificate, (PU-E), and the ownership  
indicia, the DRM-R server validating the vouching certificate, verifying the  
15 ownership indicia; and, if the request is to be honored, generating a digital  
enrollment certificate for the DRM-E server to enroll such DRM-E server into the  
DRM system, the generated enrollment certificate being based at least in part on  
the vouching certificate and (PU-E).

- 20 29. The method of claim 28 comprising the DRM-E server  
procuring the vouching certificate from an independent certificate-issuing agent  
that is trusted by and relied upon by the DRM-R server to perform such vouching.

30. The method of claim 28 comprising the DRM-E server  
25 procuring an X.509 vouching certificate.

31. The method of claim 28 comprising the DRM-E server  
procuring the vouching certificate signed by the vouching party and accompanied  
by a chain of certificates leading to a known root for purposes of validation, the  
30 DRM-R server validating the vouching certificate based on the signature thereof  
by the vouching party and the chain of certificates to establish that the DRM-E  
server has been vouched for.

32. The method of claim 28 comprising the DRM-E server performing one of employing (PR-V) to encrypt (PU-E) to result in (PR-V(PU-E)) as the ownership indicia, or signing (PU-E) with (PR-V) to result in (PU-E) S (PR-V) as the ownership indicia, the DRM-R server verifying the ownership indicia by applying (PU-V) from the request to decrypt (PU-E) or verify the signature to establish that the DRM-E server owns (PR-V) and therefore the vouching certificate.

10 33. The method of claim 28 comprising the now-enrolled DRM-E server storing the returned enrollment certificate including (PU-E) as an identifier of the DRM-E server, identifying indicia to identify the vouching certificate, and a signature based on a private key of the DRM-R server, whereby the identifying indicia for the vouching certificate in the enrollment certificate acts as a bridge to  
15 the vouching certificate and shows that the DRM-R server is trusting and relying on the vouching party to vouch for the DRM-E server.

34. The method of claim 33 comprising the now-enrolled DRM-E server storing the returned enrollment certificate further including a public key of  
20 the DRM-R server as an identifier thereof.

35. The method of claim 33 comprising the now-enrolled DRM-E server storing the returned enrollment certificate further including validity range information specifying a range during which the enrollment certificate is valid.

25 36. The method of claim 27 wherein the DRM-R server has an existing basis for trusting the DRM-E server, the method comprising:  
the DRM-E server procuring a proffering identification thereof comprising credentials recognizable to and expected to be honored by the DRM-R  
30 server;  
the DRM-E server sending an enrollment request to the DRM-R server, the request including the credentials and (PU-E), the DRM-R server

validating the credentials, and, if the request is to be honored, generating a digital enrollment certificate for the DRM-E server to enroll such DRM-E server into the DRM system, the generated enrollment certificate being based at least in part on the credentials and (PU-E).

5

37. The method of claim 36 comprising the DRM-E server procuring credentials selected from a group consisting of a network ID or a domain ID, and credentials issued by a third party.

10

38. The method of claim 36 comprising the now-enrolled DRM-E server storing the returned enrollment certificate including (PU-E) as an identifier of the DRM-E server, identifying indicia to identify the credentials, and a signature based on a private key of the DRM-R server.

15

39. The method of claim 38 comprising the now-enrolled DRM-E server storing the returned enrollment certificate further including a public key of the DRM-R server as an identifier thereof.

20

40. The method of claim 38 comprising the now-enrolled DRM-E server storing the returned enrollment certificate further including validity range information specifying a range during which the enrollment certificate is valid.

25

41. The method of claim 27 comprising the now-enrolled DRM-E server storing the returned enrollment certificate including (PU-E) as an identifier of the DRM-E server, and a signature based on a private key of the DRM-R server.

30

42. The method of claim 41 comprising the now-enrolled DRM-E server storing the returned enrollment certificate further including a public key of the DRM-R server as an identifier thereof.

43. The method of claim 41 comprising the now-enrolled DRM-E server storing the returned enrollment certificate further including validity range information specifying a range during which the enrollment certificate is valid.

5           44. The method of claim 41 comprising the now-enrolled DRM-E server storing the returned enrollment certificate further including identifying indicia to identify the proffering identification.

45. The method of claim 27 comprising the now-enrolled DRM-E  
10 server storing the returned enrollment certificate including a chain of certificates that leads back to a trusted root authority so that the generated enrollment certificate may be validated based on such chain of certificates.

46. The method of claim 27 further comprising the DRM-E server  
15 identifying in a revoking authority list at least one entity with authority to revoke enrollment of such DRM-E server within the DRM system, comprising the DRM-E server sending an enrollment request to the DRM-R server, the request including the proffering identification, (PU-E), and the revocation authority list, the DRM-R server, if the request is to be honored, generating a digital enrollment certificate  
20 for the DRM-E server to enroll such DRM-E server into the DRM system, the generated enrollment certificate being based at least in part on (PU-E) and the revocation authority list.

47. The method of claim 46 comprising the DRM-E server  
25 identifying each entity in the revocation authority list by way of a public key thereof.

48. The method of claim 46 comprising the now-enrolled DRM-E server storing the returned enrollment certificate including (PU-E) as an identifier  
30 of the DRM-E server, the revocation authority list from the request, and a signature based on a private key of the DRM-R server.

49. The method of claim 27 comprising the now-enrolled DRM-E server storing the returned enrollment certificate storing an XrML enrollment certificate.

5 50. A method in combination with a Digital Rights Management (DRM) system having a plurality of DRM servers performing DRM functionality, the method for an entering DRM-E server to be enrolled into the system by an enrolling DRM-R server such that the entering DRM-E server is to be trusted within the system and comprising:

10 the DRM-R server receiving an enrollment request from the DRM-E server including a proffering identification and a public key of the DRM-E server (PU-E) for identifying such DRM-E server within the DRM system;

the DRM-R server validating the proffering identification;

the DRM-R server, if the request is to be honored, generating

15 a digital enrollment certificate for the DRM-E server to enroll such DRM-E server into the DRM system, the generated enrollment certificate being based at least in part on (PU-E); and

the DRM-R server returning the generated enrollment certificate to the requesting DRM-E server, the now-enrolled DRM-E server storing

20 the returned enrollment certificate in an appropriate location for future use, the DRM-E server with the enrollment certificate able to employ same to issue DRM documents within the DRM system.

51. The method of claim 50 wherein the DRM-R server has no

25 existing basis for trusting the DRM-E server, the method comprising:

the DRM-R server receiving an enrollment request from the DRM-E server including (PU-E) and a proffering identification comprising a vouching certificate from a party willing to vouch for such DRM-E server, the vouching certificate incorporating therein a public key (PU-V) and a corresponding

30 private key (PR-V), the DRM-E server having employed (PU-E) and (PR-V) to formulate ownership indicia to show that the DRM-E server owns the vouching certificate, the enrollment request further including the ownership indicia;

the DRM-R server validating the vouching certificate;  
the DRM-R server verifying the ownership indicia; and  
the DRM-R server, if the request is to be honored, generating  
a digital enrollment certificate for the DRM-E server to enroll such DRM-E server  
5 into the DRM system, the generated enrollment certificate being based at least in  
part on the vouching certificate and (PU-E).

52. The method of claim 51 comprising the DRM-R server  
receiving an enrollment request from the DRM-E server including a vouching  
10 certificate from an independent certificate-issuing agent that is trusted by and  
relied upon by the DRM-R server to perform such vouching.

53. The method of claim 51 comprising the DRM-R server  
receiving an enrollment request from the DRM-E server including an X.509  
15 vouching certificate.

54. The method of claim 51 comprising:  
the DRM-R server receiving an enrollment request from the  
DRM-E server including a vouching certificate signed by the vouching party and  
20 accompanied by a chain of certificates leading to a known root for purposes of  
validation; and  
the DRM-R server validating the vouching certificate based on  
the signature thereof by the vouching party and the chain of certificates to  
establish that the DRM-E server has been vouched for.

25  
55. The method of claim 51 wherein the DRM-E server performs  
one of employing (PR-V) to encrypt (PU-E) to result in (PR-V(PU-E)) as the  
ownership indicia, or signing (PU-E) with (PR-V) to result in (PU-E) S (PR-V) as  
the ownership indicia, the method comprising the DRM-R server verifying the  
30 ownership indicia by applying (PU-V) from the request to decrypt (PU-E) or verify  
the signature to establish that the DRM-E server owns (PR-V) and therefore the  
vouching certificate.

56. The method of claim 51 comprising the DRM-R server generating the enrollment certificate to include (PU-E) as an identifier of the DRM-E server, identifying indicia to identify the vouching certificate, and a signature  
5 based on a private key of the DRM-R server, whereby the identifying indicia for the vouching certificate in the enrollment certificate acts as a bridge to the vouching certificate and shows that the DRM-R server is trusting and relying on the vouching party to vouch for the DRM-E server.

10 57. The method of claim 56 comprising the DRM-R server generating the enrollment certificate to further include a public key of the DRM-R server as an identifier thereof.

58. The method of claim 56 comprising the DRM-R server  
15 generating the enrollment certificate to further include validity range information specifying a range during which the enrollment certificate is valid.

59. The method of claim 50 wherein the DRM-R server has an existing basis for trusting the DRM-E server, the method comprising:  
20 the DRM-R server receiving an enrollment request from the DRM-E server including (PU-E) and a proffering identification comprising credentials recognizable to and expected to be honored by the DRM-R server;  
the DRM-R server validating the credentials; and  
the DRM-R server, if the request is to be honored, generating  
25 a digital enrollment certificate for the DRM-E server to enroll such DRM-E server into the DRM system, the generated enrollment certificate being based at least in part on the credentials and (PU-E).

60. The method of claim 59 comprising the DRM-R server  
30 receiving an enrollment request from the DRM-E server including credentials selected from a group consisting of a network ID or a domain ID, and credentials issued by a third party.

61. The method of claim 59 comprising the DRM-R server generating the enrollment certificate to include (PU-E) as an identifier of the DRM-E server, identifying indicia to identify the credentials, and a signature based on a private key of the DRM-R server.

62. The method of claim 61 comprising the DRM-R server generating the enrollment certificate to further include a public key of the DRM-R server as an identifier thereof.

10

63. The method of claim 61 comprising the DRM-R server generating the enrollment certificate to further include validity range information specifying a range during which the enrollment certificate is valid.

64. The method of claim 50 further comprising the DRM-R server performing custom logic to decide whether to honor the request.

65. The method of claim 64 comprising the DRM-R server performing custom logic selected from a group consisting of performing a background check on the DRM-E server and/or its operator, determining whether the DRM-E server and/or a portion thereof is current, determining whether the DRM-E server is on a revocation list or watch list, and combinations thereof.

66. The method of claim 50 comprising the DRM-R server generating the enrollment certificate to include (PU-E) as an identifier of the DRM-E server, and a signature based on a private key of the DRM-R server.

67. The method of claim 66 comprising the DRM-R server generating the enrollment certificate to further include a public key of the DRM-R server as an identifier thereof.

30

68. The method of claim 66 comprising the DRM-R server generating the enrollment certificate to further include validity range information specifying a range during which the enrollment certificate is valid.

5 69. The method of claim 66 comprising the DRM-R server generating the enrollment certificate to further include identifying indicia to identify the proffering identification.

70. The method of claim 50 comprising the DRM-R server  
10 generating the enrollment certificate by employing custom logic to generate at least a portion of the information in the enrollment certificate.

71. The method of claim 50 further comprising the DRM-R server  
15 attaching to the generated enrollment certificate a chain of certificates that leads back to a trusted root authority so that the generated enrollment certificate may be validated based on such chain of certificates.

72. The method of claim 50 further comprising the DRM-R server  
20 receiving an enrollment request from the DRM-E server further including a revocation authority list identifying at least one entity with authority to revoke enrollment of such DRM-E server within the DRM system, and comprising the DRM-R server, if the request is to be honored, generating a digital enrollment certificate for the DRM-E server to enroll such DRM-E server into the DRM system, the generated enrollment certificate being based at least in part on the  
25 revocation authority list.

73. The method of claim 72 comprising the DRM-R server  
30 generating the enrollment certificate to include (PU-E) as an identifier of the DRM-E server, the revocation authority list from the request, and a signature based on a private key of the DRM-R server.

205

74. The method of claim 50 comprising the DRM-R server generating an XrML enrollment:certificate.

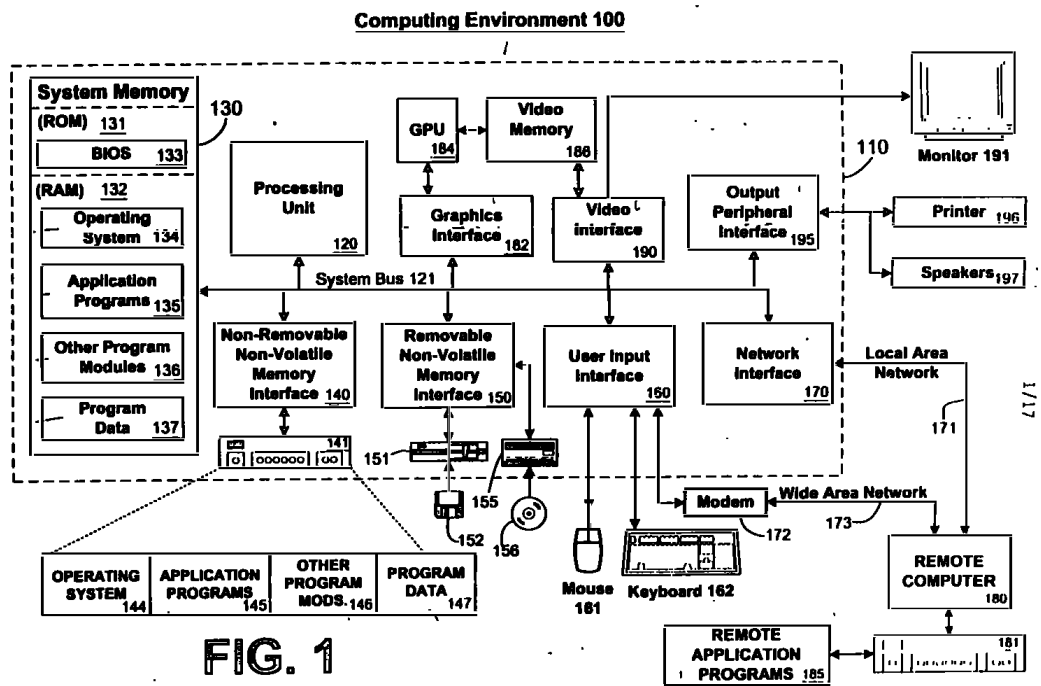
**ABSTRACT OF THE DISCLOSURE**

5

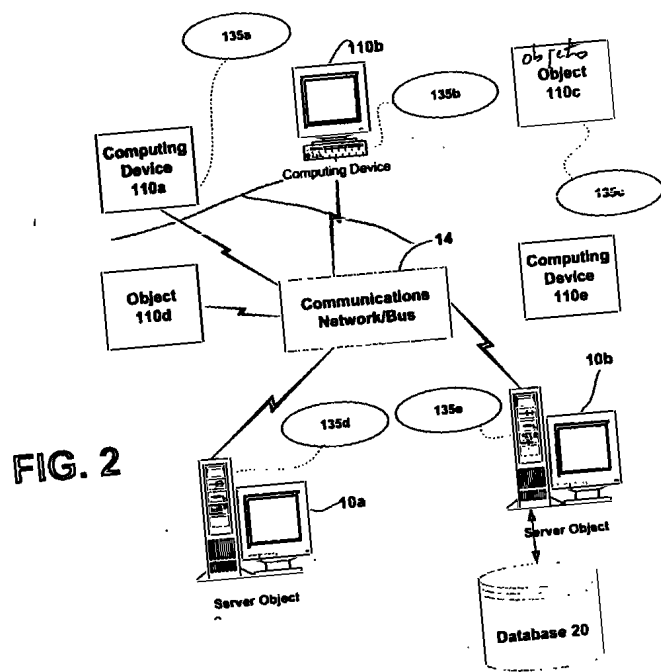
A Digital Rights Management (DRM) system has a plurality of DRM servers performing DRM functionality and an entering DRM-E server is enrolled into the system by an enrolling DRM-R server such that the entering DRM-E server is to be trusted within the system. The DRM-E server sends an enrollment request to the DRM-R server including a proffering identification and a public key (PU-E). The DRM-R server validates the proffering identification, and, if the request is to be honored, generates a digital enrollment certificate with (PU-E) for the DRM-E server to enroll such DRM-E server into the DRM system. The now-enrolled DRM-E server with the generated enrollment certificate is able to employ same to issue DRM documents within the DRM system.

10

15



208



2/17

208

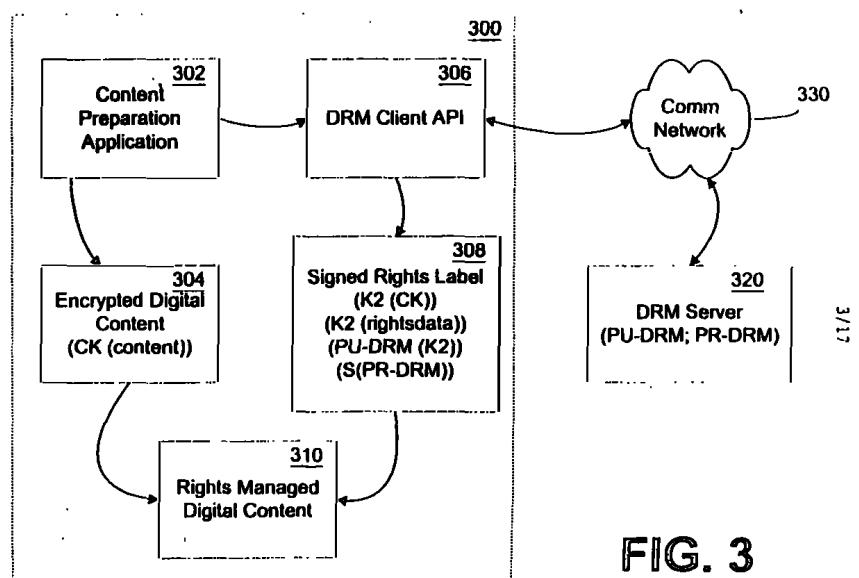


FIG. 3

209

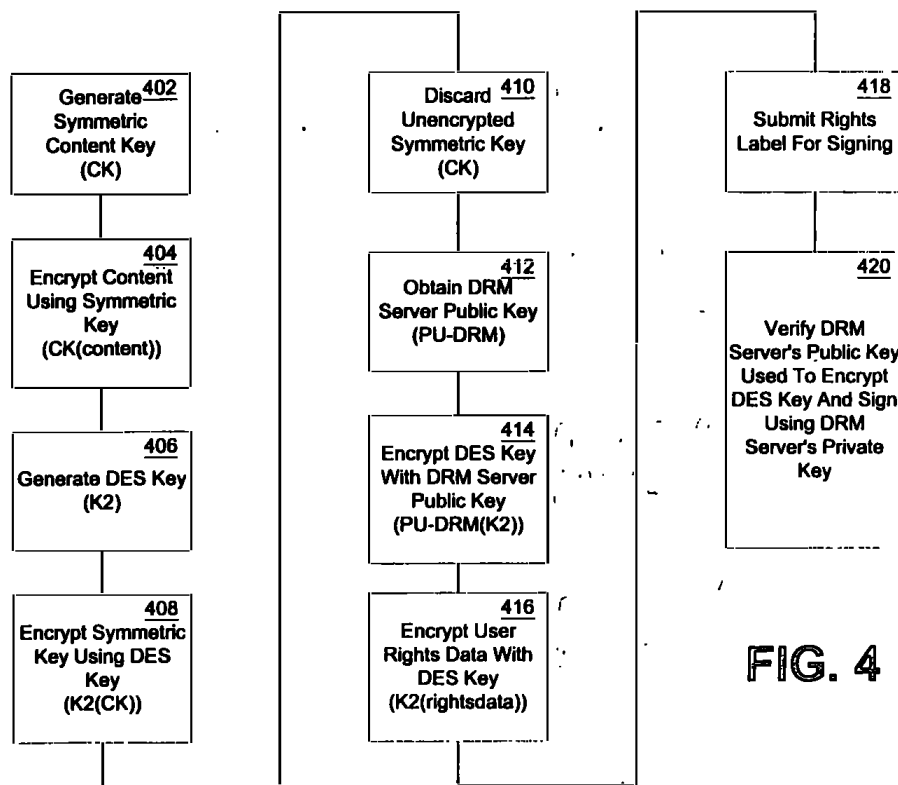


FIG. 4

4/17

210

|                   |
|-------------------|
| SRL 308           |
| CONTENT INFO      |
| DRM SERVER INFO   |
| - (PU-DRM(K2))    |
| - REFERRAL INFO   |
| -- URL            |
| -- FALL-BACK      |
| RIGHTS LABEL INFO |
| (K2(RIGHTSDATA))  |
| (K2(CK))          |
| S (PR-DRM)        |

5/17

FIG. 4A

211

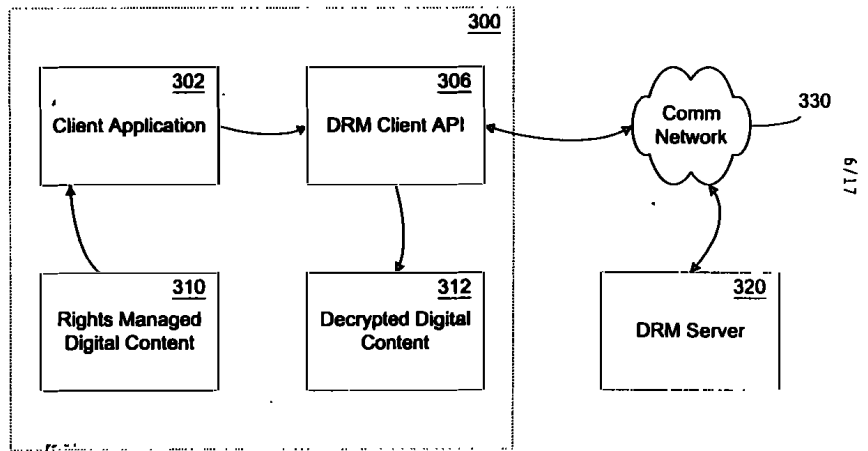
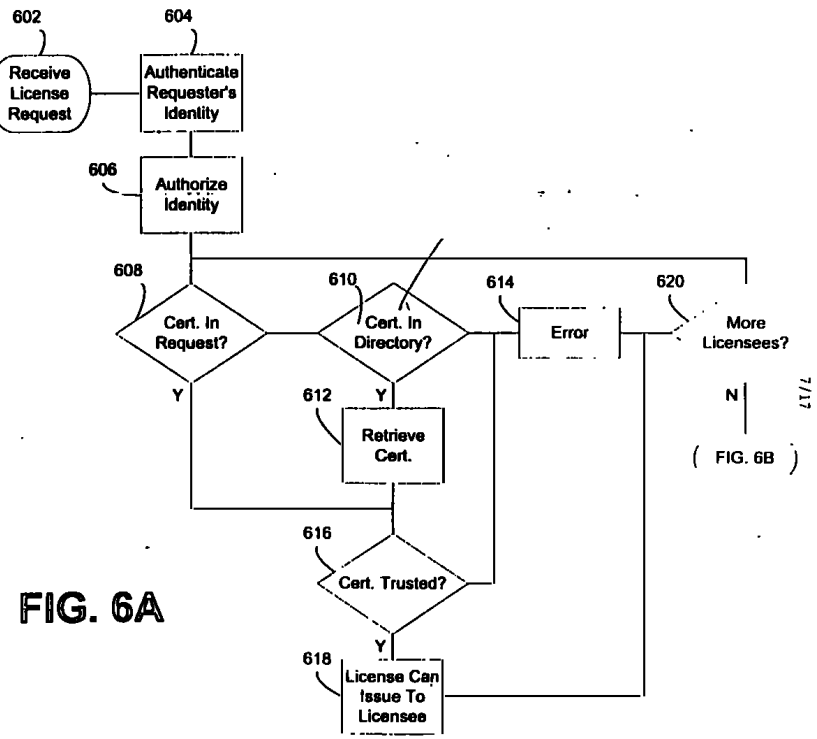


FIG. 5



( FIG. 6B )

2/17

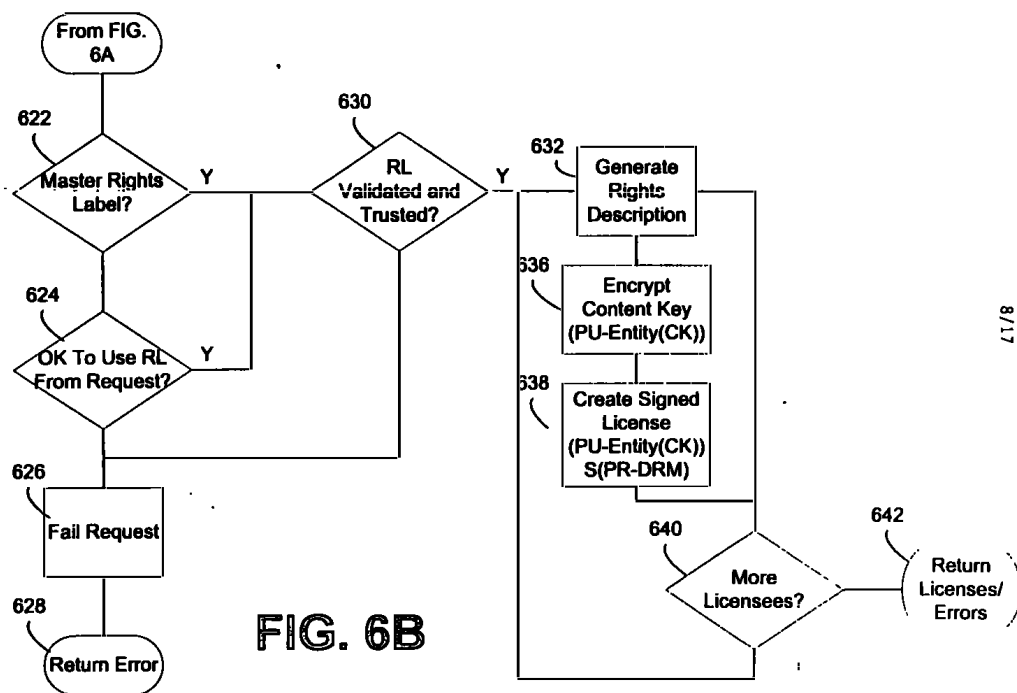
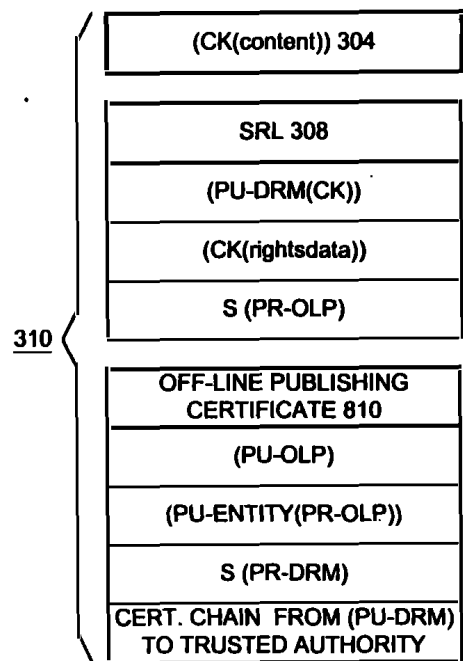


FIG. 6B

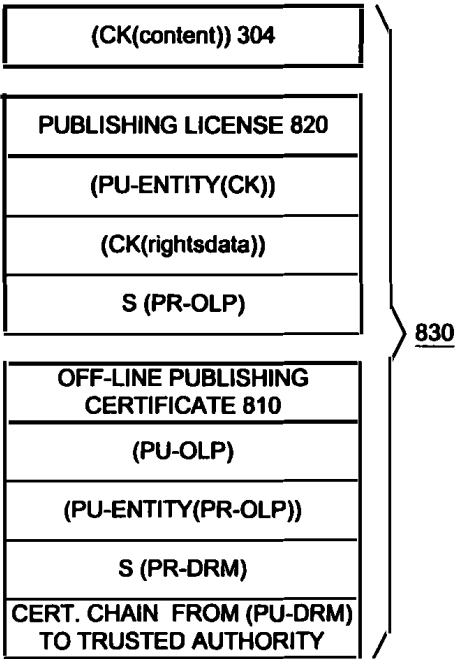
8/17

478



9/17

FIG. 7



10/17

FIG. 8

216

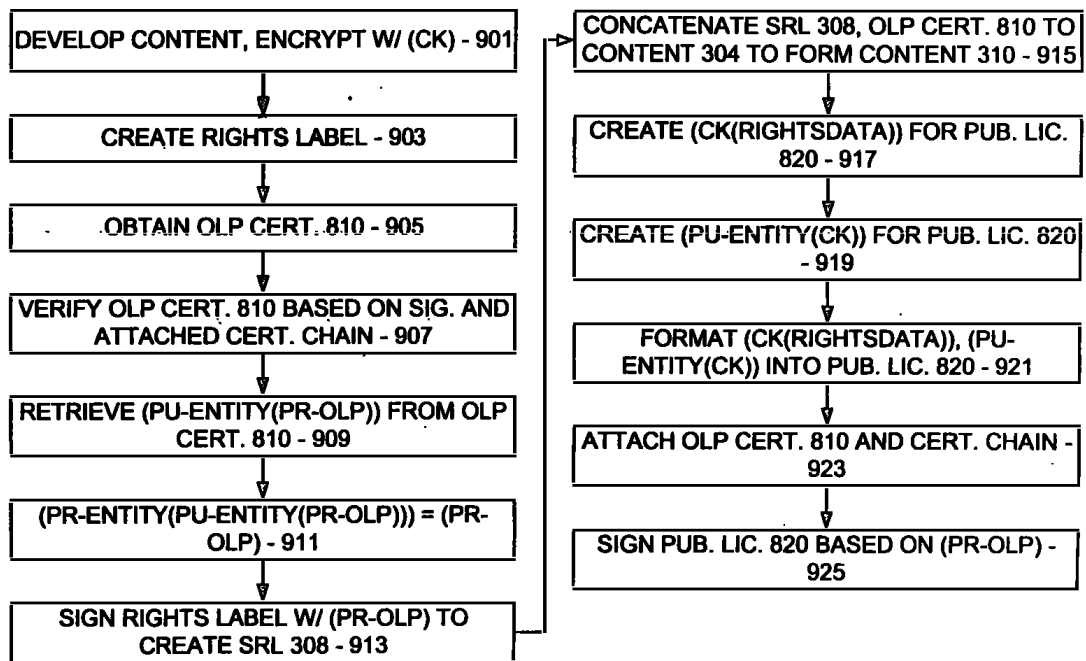


Fig. 9

217

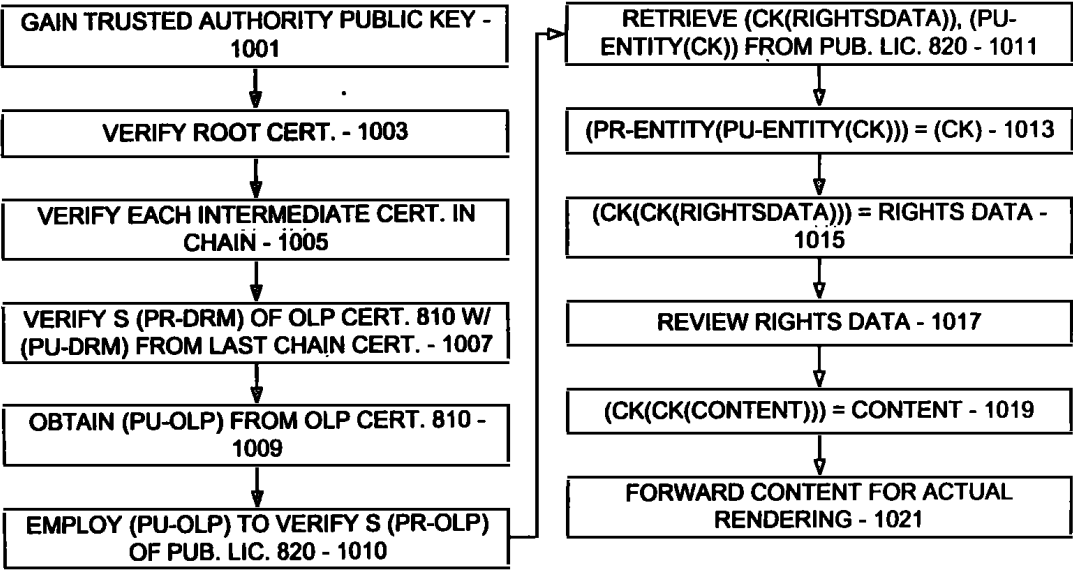


Fig. 10

218

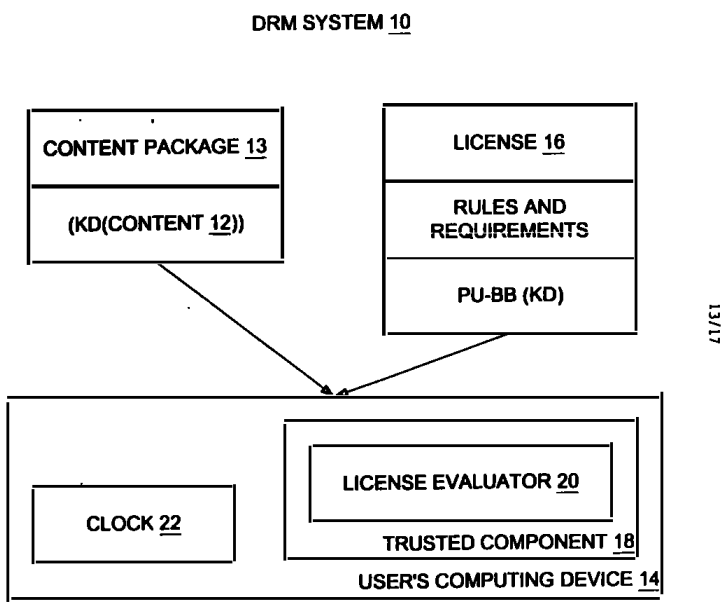


Fig. 11

219

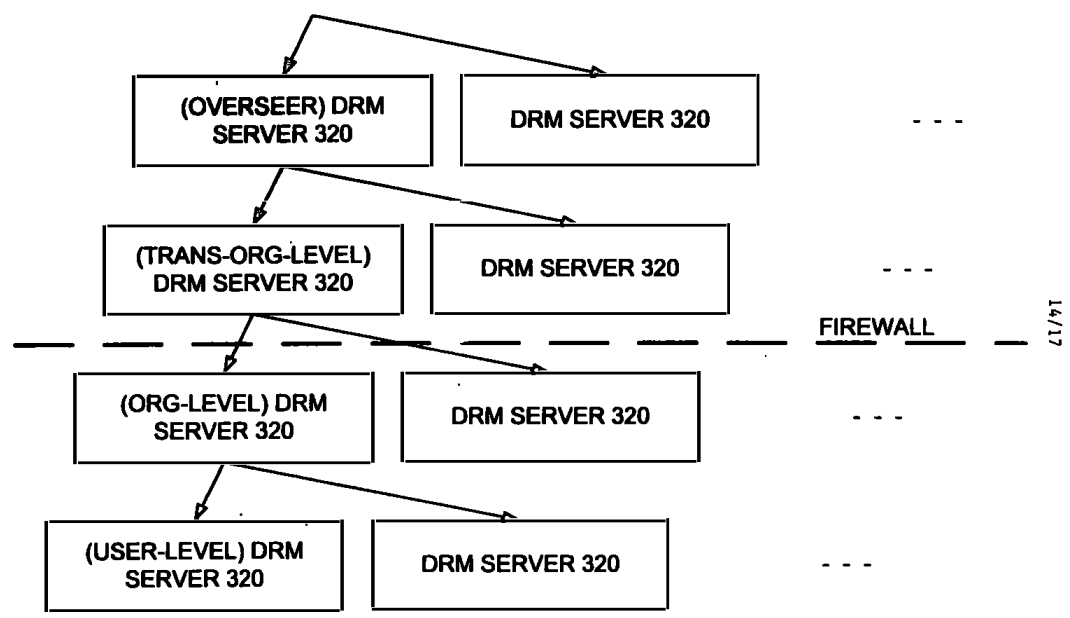


Fig. 12

288

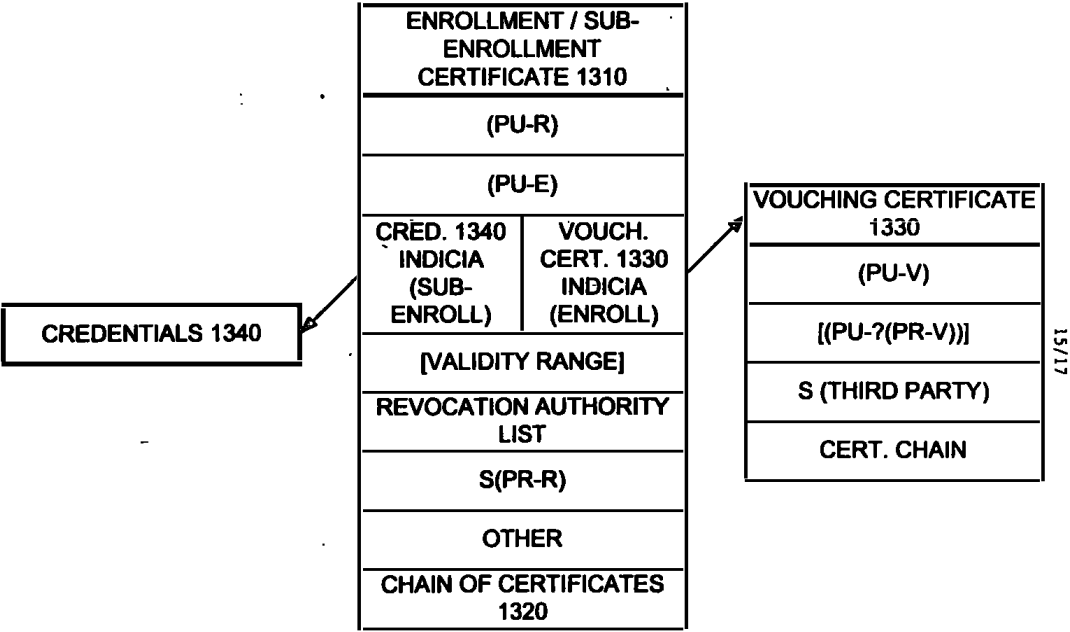


FIG. 13

227

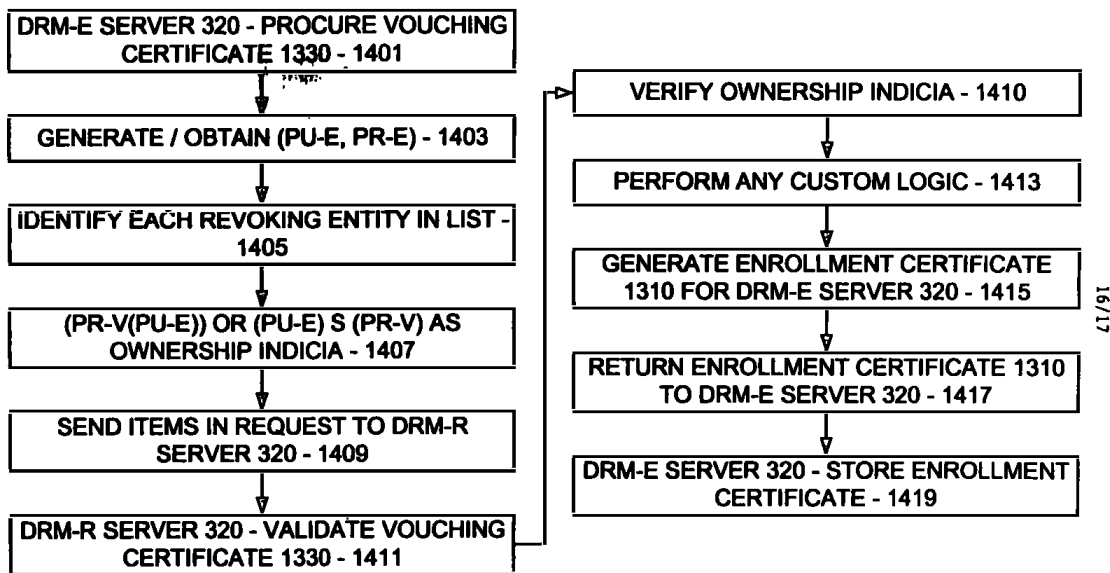


Fig. 14

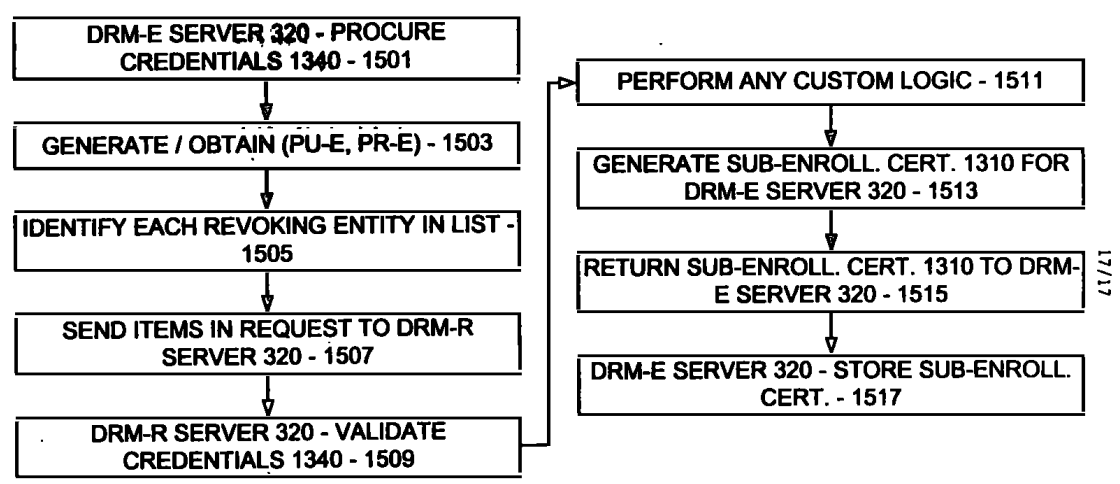


Fig. 15

9/25

224

Radicación SUPERINDUSTRIA Y COMERCIO  
04016529 000000000  
Fecha (AÑO) 2004-02-25 14 39 38 Folios 225  
Trámite 002 PATENTES D 1 REGISTRO/ 411 PRESENCIAL  
Dependencia 2020 DIVISION DE NUEVAS CREACIONES

REQUISITOS NECESARIOS PARA PRESENTACION Y AGILIZACION DEL TRAMITE

|                                                                                                           | USUARIO        | RADICADOR |
|-----------------------------------------------------------------------------------------------------------|----------------|-----------|
| PATENTE DE INVENCION                                                                                      | ( )            | ( / )     |
| MODELO DE UTILIDAD                                                                                        | ( )            | ( / )     |
| Indicacion de que se solicita la concesión de una patente                                                 | ( )            | ( / )     |
| Datos de identificación del solicitante o de la persona que se presenta la solicitud                      | ( )            | ( / )     |
| Descripción de la invención                                                                               | ( )            | ( / )     |
| Dibujos de ser estos pertinentes                                                                          | ( )            | ( / )     |
| Comprobante de Pago de las tasas establecidas                                                             |                |           |
| COMPLETA ( )                                                                                              | INCOMPLETA ( ) | ( )       |
| DISEÑO INDUSTRIAL ( )                                                                                     |                |           |
| Indicación de que se solicita el registro de Diseño Industrial                                            | ( )            | ( )       |
| Datos de identificación del solicitante o de la persona que presenta la solicitud                         | ( )            | ( )       |
| Representacion grafica y fotografica del Diseno Industrial o muestra del Material que incorpora el diseño | ( )            | ( )       |
| Comprobante de pago de las tasas establecidas                                                             | ( )            | ( )       |
| COMPLETA ( )                                                                                              | INCOMPLETA ( ) | ( )       |
| ESQUEMA DE TRAZADO ( )                                                                                    |                |           |
| Indicacion de que solicita el registro de un Esquema de trazado                                           | ( )            | ( )       |
| Datos de identificación del solicitante o de la persona que presenta La solicitud                         | ( )            | ( )       |
| Representacion gráfica del esquema de trazado                                                             | ( )            | ( )       |
| Comprobante de pago de las tasas establecidas                                                             | ( )            | ( )       |
| COMPLETA ( )                                                                                              | INCOMPLETA ( ) | ( )       |

Firma Responsable Claudia M Nueva

Fecha 1 -



## PROTOCOLO

225

Expediente 04-16529

### SUPERINDUSTRIA Y COMERCIO SISTEMA DE REGISTRO

#### MANTENIMIENTO DE PROTOCOLOS

Numero : 16069

Fecha : 04/08/1995

Conferido: MICROSOFT CORPORATION

Pais : US ESTADOS UNIDOS DE AMERICA

Ciudad : 234 REDMOND, WASHINGTON

Represent: S Sustituir: S Desistir: S

No. Radicacion : 95 34920 Clase: 0 Seev: 0 Seac: 0

| scncia | Codigo | Ctr | Nombre                     |
|--------|--------|-----|----------------------------|
| 1      | 77     | A   | CARDENAS NAVAS DARIO       |
| 2      | 3653   |     | CARDENAS CABALLERO EDUARDO |
| 3      | 5048   |     | CARDENAS MARTINEZ BERNARDO |

Encontrados (1/1) registro(s)

Carrera 13 No. 27-00 Pisos 5, 7 y 10

Conmutador: 382 0840

Fax: 350 5220

E-mail: info@sic.gov.co

www.sic.gov.co

Bogotá, D.C., Colombia

2020  
Bogotá, D. C.

Doctor(a)  
DARIO CARDENAS NAVAS  
Apoderado(a)

Oficio No. **63419**

**REFERENCIA:**

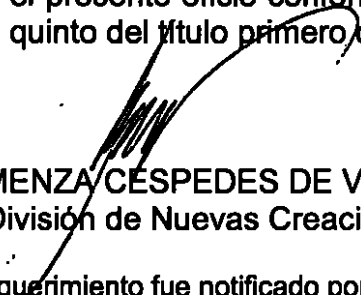
|            |          |
|------------|----------|
| Radicación | 04 16529 |
| Trámite    | 002      |
| Evento     | 001      |
| Actuación  | 430      |
| Folios     | 001      |

Como resultado del examen de forma, realizado con fundamento en el artículo 38 de la Decisión 486 de la Comisión de la Comunidad Andina, se le comunica que:

- Allegar copia del documento en que conste la cesión del derecho a la patente otorgado por el(los) inventor(es) al solicitante o a su causante. (art. 26-k, Dec 486/2000).

En cumplimiento del artículo 39 de la Decisión 486, se le requiere para que dentro del término de dos (2) meses siguientes a la fecha de notificación del presente oficio, complete los requisitos anteriormente enunciados. En caso de no dar cumplimiento al presente requerimiento, la solicitud se considerará abandonada y perderá su prelación.

Notifíquese el presente oficio conforme a lo dispuesto en el numeral 5.2, literal e), del capítulo quinto del título primero de la Circular Unica No.10 de 2001.

  
**ALIX CARMENZA CÉSPEDES DE VERGEL**  
Jefe de la División de Nuevas Creaciones

El anterior requerimiento fue notificado por fijación en lista de fecha

**01 ABR. 2004**

Se recuerda al interesado que debe presentar, dentro de los dieciséis meses siguientes contados a partir de la fecha de presentación de la solicitud cuya prioridad se invoca, copia de la misma certificada por la autoridad que la expidió y, en los casos en que haya lugar, la traducción simple del capítulo reivindicatorio. El incumplimiento de éste plazo, acarreará la pérdida de la prioridad invocada.

Carrera 13 No. 27-00 Pisos 5, 7 y 10

Conmutador: 382 0840

Fax: 350 5220

E-mail: [info@sic.gov.co](mailto:info@sic.gov.co)

[www.sic.gov.co](http://www.sic.gov.co)

Bogotá, D.C., Colombia