

Nos. 2016-2430, 2016-2431, 2016-2445, 2016-2446, 2016-2447, 2016-2448

**UNITED STATES COURT OF APPEALS
FOR THE FEDERAL CIRCUIT**

CONTENTGUARD HOLDINGS, INC.,

Plaintiff-Appellant,

v.

GOOGLE INC., HTC AMERICA, INC., HTC CORPORATION,
HUAWEI DEVICE USA, INC., HUAWEI TECHNOLOGIES CO.,
LTD., MOTOROLA MOBILITY LLC, SAMSUNG ELECTRONICS
AMERICA, INC., SAMSUNG ELECTRONICS CO., LTD.,

Defendants-Cross-Appellants,

**Appeal from the United States Court for the Eastern District of
Texas in Nos. 2:14-cv-00061-JRG and 2:16-cv-00176-JRG,
Honorable Judge Rodney Gilstrap**

**CORRECTED BRIEF FOR PLAINTIFF-APPELLANT
CONTENTGUARD HOLDINGS, INC.**

Dirk D. Thomas
MCKOOL SMITH P.C.
1999 K Street, Suite 600
Washington, DC 20006
Tel: (202) 370-8302
Fax: (202) 370-8344

Robert A. Cote
MCKOOL SMITH P.C.
One Bryant Park, 47th Floor
New York, New York 10036
Tel: (212) 402-9400
Fax: (212) 402-9444

*Attorneys for Plaintiff-Appellant
ContentGuard Holdings, Inc.*

October 13, 2016

CERTIFICATE OF INTEREST

Counsel for ContentGuard Holdings, Inc. certifies the following:

1. The full name of every party represented by me is:

ContentGuard Holdings, Inc.

2. The name of the real party in interest represented by me is:

ContentGuard Holdings, Inc.

3. All parent corporations and any publicly held companies that own 10 percent or more of the stock of the party represented by me is:

Pendrell Corporation

4. The names of all law firms and the partners or associates that appeared for the party represented by me in the trial court or are expected to appear in this Court are:

McKool Smith P.C.: Samuel F. Baxter, John C. Briody, R. Darryl Burke, Robert A. Cote, David R. Dehoney, Holly E. Engelmann, Daniel Hendler, Laura Handley*, Eric S. Hansen, Shahar Harel*, Seth Hasenour, Radu A. Lelutiu, Christopher J. Mierzejewski, Rosemary T. Snider, Dirk D. Thomas, Jennifer Truelove, Jonathan R. Yim, Karla Y. Valenzuela.

* No longer associated with McKool Smith P.C.

TABLE OF CONTENTS

CERTIFICATE OF INTEREST	i
TABLE OF AUTHORITIES	v
STATEMENT OF RELATED CASES	viii
I. JURISDICTIONAL STATEMENT	1
II. STATEMENT OF THE ISSUES PRESENTED FOR REVIEW	1
III. STATEMENT OF THE CASE	2
A. Preliminary statement.....	2
B. Relevant facts	9
1. The background of Dr. Stefik’s inventions.....	9
2. Dr. Stefik’s solution	10
3. The ContentGuard spin-off	13
4. The ’053 Patent	14
5. ContentGuard’s recent history	15
C. The District Court litigation	16
1. The District Court’s <i>Markman</i> and <i>Daubert</i> orders.....	17
2. Defendants’ improper “practicing the prior art”/prosecution disclaimer arguments	18
3. The jury verdict and the District Court’s denial of ContentGuard’s post-trial motions	20
IV. SUMMARY OF ARGUMENT	20
V. ARGUMENT	22

A.	The Court should overturn the District Court’s construction for usage rights and order a new trial	22
1.	Standard of review	22
2.	The District Court’s construction for usage rights was error because it is inconsistent with the claims, the specification, the prosecution history, and the commercial embodiment ContentGuard built based on Dr. Stefik’s inventions	24
3.	Once the District Court’s construction for usage rights is corrected, a new trial should be ordered.....	32
a.	The District Court’s claim construction error impacted all of Defendants’ non-infringement arguments.....	32
b.	The District Court’s claim construction error was not harmless	33
c.	The jury was not required to accept Defendants’ alleged non-infringement evidence	35
4.	The District Court’s evidentiary error that permitted Defendants to advance a “practicing the prior art”/prosecution disclaimer argument warrants a new trial	38
VI.	CONCLUSION AND RELIEF REQUESTED	44

TABLE OF AUTHORITIES

	Page(s)
Cases	
<i>Accent Packaging, Inc. v. Leggett & Platt, Inc.</i> , 707 F.3d 1318 (Fed. Cir. 2013)	26
<i>Amgen Inc. v. Hoechst Marion Roussel, Inc.</i> , 457 F.3d 1293 (Fed. Cir. 2006)	24
<i>In re Apple</i> , 2015 U.S. App. LEXIS 23101 (Fed. Cir. June 9, 2015) (unpublished)	2, 16, 31
<i>Apple Inc. v. Motorola, Inc.</i> , 757 F.3d 1286 (Fed. Cir. 2014)	24
<i>Aventis Pharm. Inc. v. Amino Chemicals Ltd.</i> , 715 F.3d 1363 (Fed. Cir. 2013)	26
<i>Avid Tech., Inc. v. Harmonic, Inc.</i> , 812 F.3d 1040 (Fed. Cir. 2016)	8, 22, 37
<i>Baisden v. I'm Ready Prods., Inc.</i> , 693 F.3d 491 (5th Cir. 2012)	23
<i>Baxter Healthcare Corp. v. Spectramed, Inc.</i> , 49 F.3d 1575 (Fed. Cir. 1995)	41, 42
<i>Bettcher Indus. v. Bunzl USA, Inc.</i> , 661 F.3d 629 (Fed. Cir. 2011)	23
<i>Ecolab Inc. v. Paraclipse, Inc.</i> , 285 F.3d 1362 (Fed. Cir. 2002)	<i>passim</i>
<i>In re Hiniker Co.</i> , 150 F.3d 1362 (Fed. Cir. 1998)	24
<i>Invitrogen Corp. v. Biocrest Mfg., L.P.</i> , 327 F.3d 1364 (Fed. Cir. 2003)	27

<i>Kinetic Concepts, Inc. v. Blue Sky Med. Group, Inc.</i> , 554 F.3d 1010 (Fed. Cir. 2009)	9
<i>Lava Trading, Inc. v. Sonic Trading Mgmt., LLC</i> , 445 F.3d 1348 (Fed. Cir. 2006)	27
<i>MBO Labs., Inc. v. Becton, Dickinson & Co.</i> , 474 F.3d 1323 (Fed. Cir. 2007)	27
<i>Oatey Co. v. IPS Corp.</i> , 514 F. 3d 1271 (Fed. Cir. 2008)	27
<i>In re Omeprazole Patent Litig. v. Apotex Corp.</i> , 536 F.3d 1361 (Fed. Cir. 2008)	41
<i>Phillips v. AWH Corp.</i> , 415 F. 3d 1303 (Fed. Cir. 2005)	24, 31
<i>Tate Access Floors v. Interface Architectural Res.</i> , 279 F.3d 1357 (Fed. Cir. 2002)	41
<i>U.S. Bank Nat'l Ass'n v. Verizon Commcns, Inc.</i> , 761 F.3d 409 (5th Cir. 2014)	23
<i>United States v. Riddle</i> , 103 F.3d 423 (5th Cir. 1997)	23
<i>Vanderlande Indust. Nederland BV v. U.S. Int'l Trade Comm'n</i> , 366 F.3d 1311 (Fed. Cir. 2004)	27
<i>Verizon Servs. Corp. v. Vonage Holdings Corp.</i> , 503 F.3d 1295 (Fed. Cir. 2007)	27

STATEMENT OF RELATED CASES

Pursuant to FED. CIR. R. 47.5, Plaintiff-Appellant ContentGuard Holdings, Inc. (“ContentGuard”) states that an appeal and cross-appeal pending under the caption *ContentGuard Holdings, Inc. v. Apple, Inc.*, Nos. 2016-1916, 2016-2007, have been designated companion appeals to this matter. In addition, the appellee in the *Apple* matter previously sought *mandamus* relief from this Court with respect to the District Court’s denial of Apple’s motion to transfer venue. *See In re Apple*, No. 2015-136, 2015 U.S. App. LEXIS 23101 (Fed. Cir. June 9, 2015) (unpublished). The relief requested by Apple was denied. *Id.*

I. JURISDICTIONAL STATEMENT

The U.S. District Court for the Eastern District of Texas had jurisdiction under 28 U.S.C. §§ 1331 and 1338(a) over the action giving rise to this appeal. This Court has jurisdiction over this appeal under 28 U.S.C. § 1295(a). Notice of appeal was timely filed under FED. R. APP. P. 4(a) and 28 U.S.C. § 2107(a) on August 4, 2016 from the final judgment entered on October 13, 2015, which the District Court declined to set aside on July 8, 2016. Appx1; Appx161; Appx452 (D.I. 489). The final judgment disposes of all claims and counterclaims at issue in this case.

II. STATEMENT OF THE ISSUES PRESENTED FOR REVIEW

Issue 1: Whether the District Court’s *Markman* and *Daubert* claim construction rulings concerning a key claim term—“usage rights”—were erroneous given that they directly contradict the claim language, the specification, and the prosecution history.

Issue 2: Whether ContentGuard is entitled to a new trial based on the District Court’s erroneous claim construction rulings for usage rights, which directly impacted all of Defendants’ non-infringement arguments and were not harmless.

Issue 3: Whether Defendants’ improper “practicing the prior art”/prosecution disclaimer arguments entitle ContentGuard to a new trial.

III. STATEMENT OF THE CASE

A. Preliminary statement

This appeal concerns five patents owned by ContentGuard, a “small company . . . incorporated and headquartered in Plano, Texas.” *In re Apple*, 2015 U.S. App. LEXIS 23101, at *1-2. Although in recent years ContentGuard has been developing mobile apps, it “was originally a partnership created by Xerox Corporation and Microsoft Corporation to continue work done at Xerox’s Palo Alto Research Center [‘Xerox PARC’] on ‘digital rights management’.” *Id.* at *2. All of the inventions at issue stem from research undertaken by Xerox PARC and ContentGuard scientists to find practical solutions to a problem that vexed the scientific community throughout the 1990s: how to safely distribute digital content over the Internet.

The first-named and principal inventor of four of the patents at issue is Dr. Mark J. Stefik, one of Xerox PARC’s foremost scientists. Conceived in the face of enormous skepticism, Dr. Stefik’s digital rights management (“DRM”) and “trusted computing” inventions were singled-out as “pioneering” in a 2003 report prepared by the U.S. Patent and Trademark Office (“PTO”) for the U.S. Congress. Appx10679:11-25. Dr. Stefik is now widely praised as the “father” of DRM.

Appx10675-10676 (lines 22-7)¹; Appx10677:21-25. The fifth patent at issue, U.S. Patent No. 8,001,053 (the “’053 Patent”) reflects inventions that build upon Dr. Stefik’s pioneering work. Appx349.

Dr. Stefik’s DRM-related research focused on the creation of trusted computing systems (referred to in his patents as “repositories”) that allow consumers to buy and rent digital content (*e.g.*, an e-book) and use it in accordance with rules and restrictions imposed by content owners (*e.g.*, the book’s author) or distributors (*e.g.*, the Google Play store). Dr. Stefik’s patents refer to these rules and restrictions as “usage rights.” Appx194 at col. 6:5-7. Examples of usage rights include the right to read an e-book once, or for a limited period of time, or on a particular device.

The relationship between usage rights and content is discussed at length in Dr. Stefik’s patents and, critically, addressed in some of the claims themselves. Two of the asserted claims expressly recite that “usage rights” and content are “associated with” one another. *See* ’859 Patent Claim 1, Appx217 at col. 51:16-38 (reciting “usage rights *associated with* the content”); U.S. Patent No. 7,523,072 (the “’072 Patent”) Claim 1, Appx264 at col. 52:7-22 (reciting “at least one usage right *associated with* the digital document”) (emphasis added).

¹ Citations to line numbers of multi-page trial transcripts follow the format suggested by the Clerk of the Court, wherein the first line reference corresponds to the first cited transcript page and the second line reference corresponds to the last cited transcript page.

Similarly, the specification repeatedly describes the relationship between usage rights and digital content as one of “associat[ion].” By way of example, beginning with the “Abstract of the Invention” of the ’859 Patent, Dr. Stefik notes that his inventions include a “repository . . . operative to enforce usage rights *associated with* the content.” Appx176 (emphasis added). Elsewhere, Dr. Stefik writes that “[t]he present invention uses . . . rights *associated with* digital works and their parts” and that one role of the “repository” is to “check usage rights *associated with* the digital work.” Appx199 at col. 16:64-66 (emphasis added); Appx194 at 6:52-53 (emphasis added).

In certain places, Dr. Stefik’s patents also refer to usage rights as “attached” to content. But “attachment” is just another term for “association,” as the specification and prosecution history both demonstrate. The first time the specification mentions “attachment,” there are quotes around the term (Appx194 at col. 6:11-12), thereby indicating that Dr. Stefik was not referring to the physical “attachment” of data bits or blocks—an impossibility in the digital world—but rather used the term metaphorically. For good reason. The data bits or blocks that represent usage rights and content, respectively, are stored across hundreds of millions of transistors that comprise the memory of a computing device. The only way to connect usage rights to content, therefore, is through a logical connection—a broad relationship best captured in common parlance by the term “association.”

“Attachment” and “association” thus mean one and the same thing in Dr. Stefik’s patents. Indeed, in disposing of one of the dozens of validity attacks launched against Dr. Stefik’s patents after this litigation began, the Patent Trial and Appeal Board (“PTAB”) expressly held that both terms describe the same relationship between usage rights and content. Appx10215-10216 (PTAB decision denying IPR institution and holding that “‘associated with’ and ‘attached to’ refer to the *same relationship* between usage rights and a digital work”) (emphasis added).

The parties vigorously disputed the meaning of usage rights at the claim construction stage. ContentGuard argued that, in light of the claims’ syntax, it was unnecessary to specify the relationship between usage rights and content. Appx25; Appx28. In contrast, Defendants asked the District Court to find that usage rights must be “permanently attached” to content. *Id.* Although the District Court correctly rejected Defendants’ “permanent attachment” limitation, it also rejected ContentGuard’s arguments and instead held that usage rights must be “*attached, or treated as attached*” to content. Appx35 (emphasis added).

This overly-narrow construction was error, which the District Court further compounded in a *Daubert* ruling that expressly prohibited ContentGuard from proffering evidence or argument that an “association between the content and the

usage rights is enough” (Appx154)—*even though an association is all that the claim language requires.*

Defendants pounced on the District Court’s claim construction error at trial. Although *Defendants stipulated that in the accused Google DRM system usage rights are in fact “associated with the content files”* (Appx12663:19-20), Defendants stressed repeatedly that, under the District Court’s construction, “association” was “not enough.” Appx10584:17-21; Appx11041-11042 (lines 25-4); Appx 11099:11-17; Appx12663:20-22; Appx12678:18-20. Instead, relying upon the District Court’s error, Defendants argued that the patents-in-suit require that usage rights and content come from the same server, “travel together,” and remain inseparable—like “two pieces of paper . . . stapled together” or like “tags on a shirt.” Appx12663:6-7; Appx12659:15-18.

The District Court’s claim construction error impacted all of Defendants’ non-infringement arguments and entitles ContentGuard to a new trial. Defendants made two overarching non-infringement arguments: (1) that the Google DRM scheme does not rely on usage rights; and (2) that the accused Google and Samsung customer devices are not trusted repositories. Appx12655:14-18; Appx12669:12-16; Appx9746. The construction for “repository,” however, expressly references “usage rights”—and each of the repository’s security features acts “in the support of usage rights.” Appx17 (construing “repository” as “a trusted

system in that it maintains physical, communications, and behavioral integrity *in the support of usage rights*”) (emphasis added). Thus, the jury’s evaluation of Defendants’ two interlocking non-infringement arguments was necessarily swayed by the erroneous construction for usage rights.

It is impossible to overstate the extent to which the District Court’s claim construction error skewed the trial. It enabled Defendants’ entire non-infringement defense by facilitating both of Defendants’ non-infringement arguments. It required ContentGuard’s technical experts to agree with Defendants on cross-examination that “association” was not enough—notwithstanding the language of the claims themselves. It allowed Defendants’ own expert to mischaracterize the scope of ContentGuard’s inventions. It anchored Defendants’ oft-repeated claims that the Google DRM scheme was “different” from that claimed in ContentGuard’s patents. And it became the centerpiece of Defendants’ closing argument. Because it permeated the entire trial, the District Court’s claim construction error, standing on its own, compels a new trial.

And even if this error had not provided the keystone of Defendants’ defense, a new trial is still warranted because the error was not harmless. There is ample record evidence to support a verdict of infringement by a properly instructed jury. Indeed, in post-trial briefing, Defendants acknowledged that the jury was not required to accept any of Defendants’ non-infringement defenses. Rather, in urging

the District Court not to delve into the merits of ContentGuard's JMOL motion, Defendants argued that their non-infringement defenses raised "fact question[s]" that were "properly submitted to the jury." Appx9748; Appx9768; Appx9771; Appx9776. Defendants have thus conceded that, if this Court finds error in the District Court's construction for usage rights, a new trial is warranted both because the claim construction error permeated all of Defendants' non-infringement arguments and because the error was not harmless. *Ecolab Inc. v. Paraclipse, Inc.*, 285 F.3d 1362, 1358 (Fed. Cir. 2002) ("Because we find sufficient evidence to support a jury verdict of infringement under the correct interpretation of [the] claim . . ., we conclude that [the plaintiff] suffered prejudice by the erroneous jury instruction. We therefore vacate the verdict of noninfringement . . . and remand for a new trial."); *Avid Tech., Inc. v. Harmonic, Inc.*, 812 F.3d 1040, 1047 (Fed. Cir. 2016) ("the error in the instruction governing this central dispute at trial would be harmless only if a reasonable jury would have been required by the evidence to find non-infringement even without the error").

A new trial is also warranted because the District Court made a significant evidentiary ruling that affected ContentGuard's substantial rights. Specifically, over ContentGuard's objection, the District Court allowed Defendants to argue that they do not infringe ContentGuard's patents because Google's DRM scheme allegedly embodies a prior-art DRM "license server" system that had allegedly

been disclaimed in the specification of Dr. Stefik’s patents. But “‘practicing the prior art’ is not a defense to patent infringement,”² and the question of whether the scope of ContentGuard’s patents was limited by an alleged prosecution disclaimer was an issue for the District Court. Allowing a lay jury to decide the scope of the asserted claims while evaluating the question of infringement was an error that warrants a new trial.

B. Relevant facts

1. The background of Dr. Stefik’s inventions

Hailed by the *New York Times* as a “pioneer in th[e] field of digital-rights management” (Appx457), ContentGuard traces its origins to an ambitious research project that began in the early 1990s within the Xerox PARC organization. The project was spearheaded by one of Xerox PARC’s top scientists—Dr. Mark J. Stefik. Appx10605:18-20. Dr. Stefik has been a member of the Xerox PARC organization since the early 1980s. Appx10604-10605 (lines 24-1). He holds a Ph.D. in Computer Science from Stanford, is a named inventor on approximately 100 U.S. patents, and has authored numerous books. Appx10604-10605 (lines 15-25); Appx10608:8-10.

Around 1993, after completing his work on a textbook on artificial

² *Kinetic Concepts, Inc. v. Blue Sky Med. Group, Inc.*, 554 F.3d 1010, 1025 (Fed. Cir. 2009); *Ecolab Inc.*, 285 F.3d at 1377.

intelligence, Dr. Stefik was looking for a new challenge. Appx10613:9-25. After some investigation, Dr. Stefik focused on what he perceived to be a very complex problem—how to safely distribute valuable digital content across the Internet. Appx10614:1-16. While this new medium offered exciting new ways to reach customers, content creators and distributors were “panicked” and “terrified” that their valuable digital content could forever “get away” with “the click of a button” and “they’d lose their livelihood.” Appx10614-10615 (lines 9-19).

Dr. Stefik quickly concluded that “no one had solved th[is] problem.” Appx10615:20-23. The problem was sufficiently challenging that some commentators even characterized it as insoluble, stating that content creators wishing to make their work available over the Internet were “sailing into the future on a sinking ship.” Appx10615-10618 (lines 24-1).

2. Dr. Stefik’s solution

Dr. Stefik concluded that the solution required “trusted systems,” that is, networks of computing devices capable of reliably enforcing the rights and restrictions that content owners or distributors may wish to impose on their distributed content. The solution Dr. Stefik conceived thus has two main components: (1) computing devices with special security characteristics that render them so-called “repositories”; and (2) “usage rights” that are assigned to content and persistently enforced by the repositories. Appx194 at col. 6:22-23.

The “repository.” “Repositories” are computers capable of “reliably carrying out the commercial transactions [entrusted to them]. That the systems can be responsible . . . is fundamentally an issue of integrity. The integrity of repositories has three parts: physical integrity, communications integrity, and behavioral integrity.” Appx197 at col. 11:52-61.

“Physical integrity” refers to security characteristics on the computing device that “protect[] access to the content of digital works when the content resides on the repository device.” Appx197 at col. 12:1-2.

“Communications integrity” refers to “the integrity of the communications channels between repositories,” which requires security measures (*e.g.*, encryption, an exchange of digital certificates, or nonces) that prevent pirates or eavesdroppers from “listening in” or intercepting communications between repositories. Appx197 at col. 12:21-34.

“Behavioral integrity” refers to the manner in which the repository ensures that it is executing trustworthy software. “The integrity of the software is generally assured only by knowledge of its source. . . . Behavioral integrity is maintained by requiring that repository software be certified and be distributed with proof of such certification, *i.e.*, a digital certificate.” Appx197 at col. 12:34-43 (emphasis added).

“Usage rights.” Usage rights are “rights granted to a recipient of a digital work” that “define how a digital work can be used and if it can be further

distributed.” Appx194 at col. 6:3-7. “Each usage right may have one or more specified conditions which must be satisfied before the right may be exercised.” Appx194 at col. 6:7-10. Usage rights are assigned and enforced by “repositories.”

Because enforcement of usage rights is one of the cornerstones of Dr. Stefik’s inventions, the relationship between content and usage rights is described at length in Dr. Stefik’s patents. The specification uses two terms to describe this relationship—“associat[ion]” and “attach[ment].” The terms are used interchangeably. Because the bits of data that comprise the digital content and the usage rights are merely the charged or discharged state of a transistor (or other two-way device), “attachment” of data bits to one another is a literal impossibility in the digital world. As such, because data bits (or data blocks) can be (and almost always are) associated via logical links with other data, “attachment” is merely a metaphor for a logical “association.”

This conclusion is underscored by one of several dozen PTAB decisions denying numerous validity challenges that were lodged against Dr. Stefik’s patents after this litigation began. Appx7652-7655. Specifically, in a June 20, 2015 decision, the PTAB expressly held that the terms “associated with” and “attached to” describe the same relationship between usage rights and content. Appx10215-10216 (holding that “‘associated with’ and ‘attached to’ refer to *the same relationship* between usage rights and a digital work”) (emphasis added).

Over the years, Dr. Stefik has received extraordinary acclaim for solving what market commentators considered to be an “immense, unsolved conundrum” and overcoming a daunting challenge some likened to “building fences around tornadoes.” Appx455; Appx10669-10670 (lines 21-2). Xerox Corporation conferred upon him the President’s Award (Appx10678-10679 (lines 24-3)); Microsoft and other research institutions praised him as the “father” of DRM (Appx10675-10676 (lines 22-7); Appx10677:21-25); and the PTO singled-out Dr. Stefik’s research as “pioneering” in a report prepared for the U.S. Congress. Appx10679:11-25.

3. The ContentGuard spin-off

In November 1994, recognizing the importance of Dr. Stefik’s inventions, Xerox PARC filed for patent protection. Four of the patents in suit—the ’859 and ’072 patents, and U.S. Patent Nos. 8,370,956 (the “’956 Patent”) and 8,393,007 (the “’007 Patent”)—all claim priority to Dr. Stefik’s November 1994 application.

After the patent application was filed, Xerox PARC commenced a public outreach campaign to raise awareness about Dr. Stefik’s inventions and build momentum in the marketplace for commercial embodiments. Appx10667-10668 (lines 19-3). Over the course of several years, Xerox PARC representatives met with numerous publishers and hardware companies. Appx10667-10668 (lines 19-3).

The once-skeptical industry expressed a great deal of interest in Dr. Stefik's inventions. Appx10670-10671 (lines 3-20). In the late 1990s, encouraged by this feedback, Xerox set up a new business unit to commercialize Dr. Stefik's inventions and ContentGuard was thus born. Appx11246:6-12.

In 2000, Xerox collaborated with Microsoft to spin-off ContentGuard and share ownership of ContentGuard with Microsoft. Appx10674:4-15. Xerox contributed the various patents and patent applications that reflected Dr. Stefik's inventions, as well as the engineering team that was tasked with commercializing and further developing those inventions. Appx456.

4. The '053 Patent

After the spin-off, ContentGuard maintained its focus on research, innovation, and building upon Dr. Stefik's seminal work. Appx11246:18-22. The '053 Patent reflects ContentGuard's continued efforts to innovate in the 2000s.

Recognizing that "it is difficult for a content owner to commercially exploit content unless the owner has a relationship with each party in the distribution chain" (Appx371 at col. 2:60-62), the inventors of the '053 Patent invented the concept of "meta-rights" enforceable by a repository (Appx372 at col. 4:14-19). The '053 Patent, which incorporates by reference the disclosure of Dr. Stefik's patents (Appx371 at col. 1:50-55), defines "meta-rights" as "the rights that one has to [be able to] generate, manipulate, modify, dispose of or otherwise derive other

rights.” Appx373 at col. 5:22-23. “Meta-rights can be thought of as usage rights to usage rights. Meta-rights can include rights to offer, grant, obtain, transfer, delegate, track, surrender, exchange, and revoke usage rights to/from others.” Appx373 at col. 5:24-28.

To ensure that the entirety of the content distribution chain is secure, the ’053 Patent requires that meta-rights be “enforceable by a repository.” ’053 Patent Claim 1, Appx380 at col. 20:56-57. The parties agree, and the District Court’s claim construction order makes clear, that the term “repository” carries the same meaning in Dr. Stefik’s patents and the ’053 Patent. Appx101.

ContentGuard spent significant resources to “productize and commercialize” Dr. Stefik’s inventions. Appx11246:18-22. Despite some early successes, however, ContentGuard’s product development efforts were discontinued when many of ContentGuard’s potential business partners were wiped out by the burst of the “Internet bubble.” Appx11246-11247 (lines 23-8).

5. ContentGuard’s recent history

Between 2000 and today, ContentGuard’s equity has been held by five companies—Xerox, Microsoft, Technicolor, Time Warner, and Pendrell. Appx11286:1-11. ContentGuard’s present owners are Time Warner and Pendrell. Appx11286:20-25.

Although ContentGuard has a history of widely licensing its patents,

ContentGuard is not a patent assertion entity. As of late 2015, ContentGuard's Plano-based employees were actively involved in developing mobile apps. Appx11249:15-19.

Since 2006, ContentGuard has completed ten licensing agreements. Appx11253:9-15. Many of Defendants' competitors—*e.g.*, Casio, Fujitsu, Hitachi, LG, Microsoft, NEC, Nokia, Panasonic, Sanyo, Sharp, Sony, and Toshiba, among others—have licensed ContentGuard's patents. Appx461-462.

C. The District Court litigation

In December 2013, after eight years of unfruitful licensing negotiations, ContentGuard filed suit against Defendant Samsung in the U.S. District Court for the Eastern District of Texas. Appx462. ContentGuard's complaint also named several other mobile device manufacturers that, like Samsung, had declined to accept a license to ContentGuard's patents. After Google commenced a declaratory judgment action against ContentGuard in the U.S. District Court for the Northern District of California, ContentGuard commenced litigation against Google in the Eastern District of Texas. *See In re Apple*, 2015 U.S. App. LEXIS 23101, *3 n.1; Appx463. Relevant aspects of the proceeding are summarized below.

1. The District Court’s *Markman* and *Daubert* orders

The District Court construed several claim terms that are relevant here, including the critical “usage rights and “repository” terms.³ With respect to the term “usage rights,” the parties’ positions were as follows (*see* Appx25):

Plaintiff’s proposal	Defendants’ proposal
“an indication of the manner in which a [digital work / digital content / content / a digital document] may be used or distributed as well as any conditions on which use or distribution is premised”	“statements in a language for defining the manner in which a digital work may be used or distributed, as well as any conditions on which use or distribution is premised. Usage rights must be permanently attached to the digital work.”

Besides proposing an altogether different construction for this claim term, ContentGuard also objected to Defendants’ request that the District Court specify the manner in which usage rights must relate to content. ContentGuard argued that Defendants’ proposal (“permanent attachment”) was incorrect and, in light of the syntax of the claims themselves, unnecessary. Appx28.

The District Court’s construction partly tracked ContentGuard’s proposal, but nonetheless sought to specify the manner in which usage rights and content

³ In this appeal, ContentGuard only claims error with respect to the District Court’s construction of “usage rights.”

relate to each other. Appx35. Without accounting for the language of the claims (which refer to “usage rights” that are “*associated with*” content) and the specification’s teachings concerning this relationship (which is consistent with the claim language), the District Court held that usage rights must be “*attached, or treated as attached*” to content. Appx35 (emphasis added).

On its face, the *Markman* order did not prohibit ContentGuard from arguing that usage rights that are “associated with” content met the District Court’s construction. However, in a pre-trial *Daubert* ruling, the District Court compounded its *Markman* error by expressly forbidding ContentGuard from making that argument. Appx154 (“No expert may opine or insinuate that a mere association between the content and the usage rights is enough to meet the requirement that the usage rights be ‘attached’ to the content.”).

The District Court construed the term “repository” as “a trusted system in that it maintains physical, communications, and behavioral integrity *in the support of usage rights.*” Appx17 (emphasis added). By its terms, this construction references “*usage rights,*” and thus the question as to whether a computer is a repository *necessarily* depends on whether it supports usage rights.

2. Defendants’ improper “practicing the prior art”/prosecution disclaimer arguments

Beginning with their opening statement, Defendants argued that the jury should deliver a non-infringement verdict because the Google DRM scheme

allegedly embodies a so-called “license server” approach to DRM—*i.e.*, the Google Play store relies on separate servers to store and deliver usage rights and content, respectively—that Dr. Stefik “criticize[d]” and purportedly “walked away from” in his patents. More specifically, Defendants argued that:

Google uses the license server approach that ContentGuard and Dr. Stefik *walked away* from in their patent application. . .

ContentGuard can’t tell the Patent Office that it *doesn’t like, doesn’t want, and didn’t invent* the license server approach in order to get their patent and *then turn around and when the patents are about to expire, say, well, actually we invented the license server approach*

ContentGuard just can’t have it both ways. . . .

[Y]ou can’t get a patent for something that existed before. And Dr. Stefik explained, therefore, why he thought that what *he was suggesting with the trusted system approach was different and better* than what Mr. Griswold had done. . . .

In short, Dr. Stefik *teaches in his patent, don’t use a license server approach with licenses that are only associated with the content*, and do use the trusted system with usage rights that are attached or treated as attached to the content. ContentGuard *can’t walk away from those teachings now*.

And you’ll see over the next few days, there’s actually no doubt or dispute that *Google uses the license server approach*. . . .

This case will come down to whether you believe Google is using the license server approach or whether you believe they’re using Dr. Stefik’s trusted system

approach, plain and simple. And ContentGuard won't be able to dispute the evidence that *Google uses the license server approach*.

Appx10579-10586 (lines 4-9) (emphasis added).

This “practicing the prior art”/prosecution disclaimer argument was stressed during trial and was featured prominently in closing. *See* Appx12022:5-24; Appx12024-12035 (lines 20-24); Appx12668-12669 (lines 19-6).

3. The jury verdict and the District Court's denial of ContentGuard's post-trial motions

On September 23, 2015, the jury delivered its verdict. Appx7878. In a general verdict form, it found that Defendants did not infringe any of the five claims ContentGuard pursued at trial—claim 1 of the '859 Patent; claim 1 of the '072 Patent; claim 7 of the '956 Patent; claim 6 of the '007 Patent; and claim 1 of the '053 Patent. Appx7879-7880. The jury also found that none of the five claims at issue had been proven to be invalid. Appx7881.

Following the jury's verdict, ContentGuard moved for judgment as a matter of law or, in the alternative, a new trial. Appx439 (D.I. 400). The District Court denied ContentGuard's motions. Appx161.

IV. SUMMARY OF ARGUMENT

The District Court erred in its construction of usage rights, and compounded its error in its *Daubert* order. The claims of Dr. Stefik's patents, the specification, and the prosecution history all make unambiguously clear that the relationship

between usage rights and content that was contemplated, taught, and claimed by Dr. Stefik includes “association.” It was error for the District Court to find otherwise and thereby preclude ContentGuard from proving Defendants’ infringement based on the express language of the asserted claims.

A new trial is warranted. The District Court’s incorrect construction for usage rights permeated the entirety of the trial and—because the District Court’s construction for repository expressly requires that the three integrities act “in the support of *usage rights*”—directly impacted all of Defendants’ non-infringement arguments. And the error was not harmless because there was sufficient evidence in the record for the jury to find infringement under a correct construction, as Defendants conceded at trial and in post-trial briefing.

A new trial is also warranted based on the evidentiary error that permitted Defendants to present to the jury their “practicing the prior art”/prosecution disclaimer arguments. Practicing the prior art is not a defense to patent infringement, and the question of whether the use of a “license server” was disclaimed during prosecution should not have been submitted to the jury.

V. ARGUMENT

A. The Court should overturn the District Court's construction for usage rights and order a new trial

1. Standard of review

Because it was based solely on intrinsic evidence (*see* Appx35), the District Court's construction for usage rights is subject to *de novo* review. *Avid Tech.*, 812 F.3d at 1044-45 (“We review the district court’s claim construction *de novo* because it was based entirely on an intrinsic-evidence determination about the meaning of the prosecution history, and not on any evidence about extra-patent understandings of language or about other facts.”) (citing *Teva Pharms. USA, Inc. v. Sandoz, Inc.*, 135 S. Ct. 831, 841 (2015)).

A claim construction error “requires at least *vacatur* of the verdict and a remand for a new trial unless [this Court] can conclude that the error was not prejudicial, *i.e.*, was harmless.” *Avid Tech.*, 812 F.3d at 1047. Where, as here, “there was no separate jury determination of non-infringement on a distinct ground, the error in the instruction governing [a] central dispute at trial would be harmless only if a reasonable jury would have been required by the evidence to find non-infringement even without the error.” *Id.* (citing *Ecolab, Inc.*, 285 F.3d at 1374 (error is harmless if it “could not have changed the result,” *i.e.*, if “the same verdict would necessarily be reached absent the error”)).

Where an appellant challenges a district court's disposition of a motion for new trial, this Court applies the law of the regional circuit where the district court sits. *Bettcher Indus. v. Bunzl USA, Inc.*, 661 F.3d 629, 638 (Fed. Cir. 2011). The Fifth Circuit "review[s] the admission or exclusion of evidence for abuse of discretion." *Baisden v. I'm Ready Prods., Inc.*, 693 F.3d 491, 508 (5th Cir. 2012). Even if the district court's evidentiary ruling is found to be an abuse of discretion, it is subject to harmless error analysis and does not justify reversal "unless it affected substantial rights of the complaining party." *Id.* (citation and internal quotation marks omitted). "A ruling has affected the substantial rights of the party if, when considering all of the evidence presented at trial, the ruling had a substantial effect on the outcome of the trial." *U.S. Bank Nat'l Ass'n v. Verizon Commcns, Inc.*, 761 F.3d 409, 430 (5th Cir. 2014).

The Fifth Circuit recognizes the doctrine of "cumulative error," which holds that a number of evidentiary errors may cumulatively warrant a new trial even though none may be sufficient standing alone. *See United States v. Riddle*, 103 F.3d 423, 434-35 (5th Cir. 1997).

2. The District Court’s construction for usage rights was error because it is inconsistent with the claims, the specification, the prosecution history, and the commercial embodiment ContentGuard built based on Dr. Stefik’s inventions

It was error for the District Court to conclude that DRM systems that rely on usage rights that are “associated with” content fall outside the scope of Dr. Stefik’s patents.

This Court has repeatedly emphasized that “claim construction must begin with the words of the claim themselves.” *Amgen Inc. v. Hoechst Marion Roussel, Inc.*, 457 F.3d 1293, 1301 (Fed. Cir. 2006). That is because, “as in all aspects of claim construction, ‘the name of the game is the claim.’” *Apple Inc. v. Motorola, Inc.*, 757 F.3d 1286, 1298 (Fed. Cir. 2014) (quoting Giles Sutherland Rich, *Extent of Protection and Interpretation of Claims—American Perspectives*, 21 INT’L REV. INDUS. PROP. & COPYRIGHT L. 497, 499 (1990)); *In re Hiniker Co.*, 150 F.3d 1362, 1369 (Fed. Cir. 1998) (same). In other words, “the words of the claims themselves . . . define the scope of the patented invention.” *Phillips v. AWH Corp.*, 415 F. 3d 1303, 1313 (Fed. Cir. 2005) (quoting *Innova/Pure Water, Inc. v. Safari Water Filtration Sys., Inc.*, 381 F.3d 1111, 1115 (Fed. Cir. 2004)).

Here, the District Court had to go no further than the claims themselves to capture the relationship between usage rights and content that Dr. Stefik had in mind. Claim 1 of the ’859 Patent reads as follows:

A rendering system adapted for use in a distributed system for managing use of content, said rendering system being operative to rendering content *in accordance with usage rights associated with the content*, said rendering system comprising:

a rendering device configured to render the content;
and

a distributed repository coupled to said rendering device and including a requester mode of operation and server mode of operation,

wherein the server mode of operation is operative to *enforce usage rights associated with the content* and permit the rendering device to render the content in accordance with a manner of use specified by the usage rights,

the requester mode of operation is operative to request access to content from another distributed repository, and

said distributed repository is operative to receive a request to render the content and permit the content to be rendered only if a manner of use specified in the request corresponds to a manner of use specified in the usage rights.

'859 Patent Claim 1, Appx217 at col. 51:16-38 (emphasis added).

Likewise, claim 1 of the '072 Patent reads as follows:

A method for securely rendering digital documents, comprising:

retrieving, by a document platform, a digital document and at least *one usage right associated with the digital document* from a document repository, the at least one usage right specifying a manner of use indicating the manner in which the digital document can be rendered;

storing the digital document and the at least one usage right in separate files in the document platform;

determining, by the document platform, whether the digital document may be rendered based on the at least one usage right; and

if the at least one usage right allows the digital document to be rendered on the document platform, rendering the digital document by the document platform.

'072 Patent Claim 1, Appx264 at col. 52:7-22 (emphasis added).

Given that the parties agreed that the term “usage rights” must carry the same meaning in the four Stefik patents and the '053 Patent (*see* Appx101),⁴ it is plain that the District Court’s construction, as further limited in the *Daubert* order, was error. This construction—which, again, expressly precluded ContentGuard from arguing that usage rights that are “associated with” content fall within the scope of Dr. Stefik’s patents—cannot be reconciled with the manner in which the claims themselves define Dr. Stefik’s inventions.

This Court has repeatedly emphasized that “a claim interpretation that excludes a preferred embodiment from the scope of the claim is rarely, if ever, correct.” *Accent Packaging, Inc. v. Leggett & Platt, Inc.*, 707 F.3d 1318, 1326 (Fed. Cir. 2013) (quotations omitted). In its *Markman* order, the District Court went a step further—it actually excluded a *claimed embodiment*. The District Court

⁴ The parties’ agreement in this regard reflects this Court’s precedent. *Aventis Pharm. Inc. v. Amino Chemicals Ltd.*, 715 F.3d 1363, 1380 (Fed. Cir. 2013) (“[W]e presume, unless otherwise compelled, that the same claim term in the same patent or related patents carries the same construed meaning.”) (quoting *Omega Eng’g v. Raytek Corp.*, 334 F.3d 1314, 1334 (Fed. Cir. 2003)).

thereby committed clear error and its construction for usage rights cannot stand. *Oatey Co. v. IPS Corp.*, 514 F. 3d 1271, 1277 (Fed. Cir. 2008) (“We normally do not interpret claim terms in a way that excludes embodiments”); *MBO Labs., Inc. v. Becton, Dickinson & Co.*, 474 F.3d 1323, 1333 (Fed. Cir. 2007) (rejecting claim construction that would exclude embodiments illustrated in the drawings); *Lava Trading, Inc. v. Sonic Trading Mgmt., LLC*, 445 F.3d 1348, 1353-55 (Fed. Cir. 2006) (rejecting claim construction that “excluded embodiments disclosed in the specification” including embodiments in the drawings); *Vanderlande Indus. Nederland BV v. U.S. Int’l Trade Comm’n*, 366 F.3d 1311, 1320, 1322 (Fed. Cir. 2004) (declining to limit the term “glide surface” to a specific embodiment where the descriptive text includes other embodiments); *Verizon Servs. Corp. v. Vonage Holdings Corp.*, 503 F.3d 1295, 1305 (Fed. Cir. 2007) (rejecting proposed claim interpretation that would exclude disclosed examples in the specification); *Invitrogen Corp. v. Biocrest Mfg., L.P.*, 327 F.3d 1364, 1369 (Fed. Cir. 2003) (finding that the district court’s claim construction erroneously excluded an embodiment described in an example in the specification, where the prosecution history showed no such disavowal of claim scope).

The remainder of the Stefik patents’ disclosure compels the same conclusion as the patent claims themselves. The specification repeatedly references “usage rights” that are “associated with” content:

- “Summary of the Invention: . . . The system includes a rendering device configured to render the content and a repository coupled to the rendering device and operative to enforce *usage rights associated with the content.*” Appx193 col. 3:53-60 (emphasis added).
- “Repository 1 checks the *usage rights associated with the digital work.*” Appx194 col. 6:52-53 (emphasis added).
- “The present invention uses . . . *rights associated with digital works* and their parts.” Appx199 at col. 16:64-66 (emphasis added).
- “The grammar provides a catalog of possible *rights that can be associated with parts of digital works.*” Appx200 at col. 18:39-41 (emphasis added).
- “[U]sage rights *associated with* the folder containing the work.” Appx206 at col. 30:8-9 (emphasis added).
- “[P]rior to initiating a usage transaction, the requestor performs any general tests that are required before *the right associated with the transaction can be exercised . . .*” Appx206 at 30:13-16 (emphasis added).

These repeated references to usage rights that are “associated with” content are not a drafting error; rather they reflect the patentee’s choice to broadly characterize the relationship between usage rights and content. Given the specification’s teachings, the District Court’s construction for “usage rights,” which excludes numerous preferred embodiments, cannot be correct.

Defendants will seize on the fact that, in addition to discussing “association,” the specification also discusses “usage rights” that are “attached” or even “permanently attached” to content. These arguments are unavailing. Attachment is a fiction in the context of digital content or digital data. That is why the first reference to “attachment” in Dr. Stefik’s patents is, literally, surrounded by quotes. Appx194 at col. 6:11-12 (“A key feature of the present invention is that usage rights are permanently ‘attached’ to the digital work.”). Digital information is, by its nature, divisible: it is stored in memory as bits, and those bits are typically represented by the condition of a transistor, typically by putting each individual transistor in an “on” or “off” (*i.e.*, charged or discharged) condition. There is no actual physical “attachment” of one bit of information to another, only *logical links* between data bits or blocks of data bits; the very type of relationship that, in common parlance, is captured by the term “association.”⁵

⁵ In their *Markman* briefing, Defendants recognized that it is a misnomer to use the term “attachment” to describe a relationship between one data bit (defining a “usage right”) and other data bits (representing digital content). After

Indeed, in the *Markman* order, the District Court recognized that the “specification also appears to suggest that usage rights and their associated content can be stored in *separate ‘descriptor’ and ‘contents’ files, respectively,*” and that “[t]he storage can even be *on separate devices.*” Appx31-32 (emphasis added, citing ’859 Patent at col. 8:46-54; Fig. 12, 9:10-25; col. 13:41-47). The District Court further recognized that the specification contemplates “mak[ing] a copy of the digital work *in a place outside of the protection of usage rights.*” Appx35 (emphasis added, citing ’859 Patent at col. 35:64-36:3). Necessarily, the District Court recognized that the claims’ scope encompasses scenarios where usage rights and content are in fact separated—the very type of scenarios captured by a relationship of “association”—yet *logically linked*, such that before any request to render content is executed the usage rights are checked and enforced.

ContentGuard explained to the District Court that it was impossible for data bits to be “attached” to each other (*see* Appx1874-1875), Defendants offered no rebuttal. Rather, Defendants argued that “a person of ordinary skill in the art would understand physical attachment to include, at least, (a) use of header or footer, (b) interspersing a work’s usage rights information with its content, or (c) combining content and usage rights in a single folder.” Appx2426 n.4. Notably, since a “folder” can contain numerous files that are not “attached” to one another and can be accessed individually and independently (*e.g.*, a folder that collects unrelated email), at least option (c) identified by Defendants would require a logical link—or *association*—between one data file representing “usage rights” and another data file comprising content.

Further, as the PTAB held in rejecting an IPR petition filed against a Stefik patent that was included in ContentGuard's initial complaint,⁶ the terms "association" and "attachment" both describe the same relationship between usage rights and content. Appx10215-10216 ("associated with' and 'attached to' refer to *the same relationship* between usage rights and a digital work") (emphasis added).

Finally, as ContentGuard explained in the brief filed in the companion *Apple* appeal, in the flagship RightsEdge product that ContentGuard built based on Dr. Stefik's inventions, usage rights and content originated from different servers and traveled separately to the user device. *See ContentGuard Apple Blue Brief* at 17-18. The fact that RightsEdge operated much like Google's DRM system further underscores that the District Court's construction—which, as interpreted by Defendants and their expert would exclude RightsEdge—is erroneous.

In light of the foregoing, the District Court's construction for usage rights should be overturned and the term should be construed as ContentGuard proposed.

⁶ The patent in question, U.S. Patent No. 7,225,160 ("the '160 Patent") claims priority to the same November 1994 application that provides priority for all the Stefik patents at issue in this appeal. During prosecution of this patent, the patent prosecutor attempted to "remove all possible doubt that 'attached' and 'associated' mean the same thing" by only using the latter term. Appx34. The fact that the PTO issued the '160 Patent as a continuation, rather than as a continuation-in-part, and never characterized the changes made as constituting new matter, demonstrates that the Examiner (and others of ordinary skill in the art) agreed with the prosecuting attorney, and would read the quoted word "attached" in the Stefik patents as being synonymous with the term "associated." *Phillips*, 415 F.3d at 1317.

Appx25 (“an indication of the manner in which a [digital work / digital content / content / a digital document] may be used or distributed as well as any conditions on which use or distribution is premised”).

3. Once the District Court’s construction for usage rights is corrected, a new trial should be ordered

Once the District Court’s construction for usage rights is reversed, a new trial should be ordered because the claim construction error (1) enabled all of Defendants’ non-infringement arguments; and (2) was not harmless because ContentGuard introduced ample evidence that, but for the error, would have supported a verdict of infringement.

a. The District Court’s claim construction error impacted all of Defendants’ non-infringement arguments

Defendants’ non-infringement defense rested on two assertions: (1) that, in the Google DRM scheme, usage rights are not “attached” or “treated as attached” to content; and (2) that the accused customer devices are not “trusted repositories” because they allegedly lack “physical integrity” and “behavioral integrity.” *See* Appx12655:14-18; Appx12669:12-16; Appx9746.

Given that the District Court’s construction for the “repository” limitation expressly requires that the three integrities be maintained “*in the support of usage rights*” (Appx17), the question of whether Google’s DRM scheme utilizes “repositories” hinges on the question of whether it supports “usage rights.” As

such, the District Court's incorrect construction for usage rights directly impacted all of Defendants' non-infringement arguments concerning the Stefik patents⁷ and *vacatur* of the jury verdict and a new trial are appropriate.

b. The District Court's claim construction error was not harmless

Even if the District Court's construction had not impacted all of Defendants' non-infringement arguments, a new trial is warranted because, at the very least, the error was not harmless. At trial, ContentGuard presented ample infringement evidence from two prominent academics, Dr. Michael Goodrich and Dr. David Martin.⁸ ContentGuard's experts provided a detailed, element-by-element analysis to demonstrate that the accused devices literally met each element of the asserted claims. Defendants never attempted to show on cross-examination that ContentGuard's experts incorrectly described the operation of the accused devices.

⁷ The claims of the '956 and '007 Patents do not recite "repository" limitations; rather they recite the limitation "trusted." The parties agree, however, that "repository" and "trusted" mean essentially the same thing, and the District Court adopted essentially similar constructions for these terms, which include the three "integrities." *See* Appx17.

⁸ Dr. Goodrich, who presented opinions on infringement with respect to the Stefik patents, is a Chancellor's Professor at the University of California, Irvine. Appx10847-10848 (lines 13-4). He has written over 300 publications in computer science, including several widely cited textbooks that concern, among other things, computer security and DRM. Appx10848-10849 (lines 15-17). Dr. Martin, who presented opinions on infringement with respect to the '053 Patent, is affiliated with the University of Iowa. Appx11007-11008 (lines 19-3). He has published widely in the area of computer security and has extensive working experience in the software industry. Appx11008-11009 (lines 11-17).

Nor did Defendants' non-infringement expert, Dr. Paul Clark, dispute Drs. Goodrich or Martin's description of the operation of the accused products and services. Instead, through testimony elicited from Dr. Clark and other fact witnesses, Defendants argued that countervailing evidence supported a verdict of non-infringement.

But Defendants' alleged non-infringement evidence does not alter the fact that ContentGuard adduced sufficient evidence of infringement with respect to the three limitations of the five patents-in-suit that Defendants alleged were missing from the Google DRM scheme—physical integrity for the accused customer devices; behavioral integrity for the Google Play apps; and usage rights.

Physical integrity for accused customer devices. The District Court construed “physical integrity” as “preventing access to information in a repository by a non-trusted system.” Appx20-21. Dr. Goodrich's testimony established that the accused Google and Samsung end-user devices meet this limitation through the use of various encryption techniques and keys. Appx10906:7-18; Appx10888-10889 (lines 24-8). In reaching his opinions, Dr. Goodrich relied on documents produced by Google and Samsung. Appx10873-10874 (lines 23-19).

Behavioral integrity for accused Google Play apps. The District Court construed “behavioral integrity” as “requiring software to include a digital certificate in order to be installed in the repository.” Appx23. Dr. Goodrich's

testimony established that the Google Play software on the accused Google and Samsung customer devices requires a digital certificate to be installed or updated. Appx10889-10890 (lines 9-10). In reaching his opinions, Dr. Goodrich relied on documents produced by Google. Appx10890:1-13.

“Usage rights” that are “associated with” content. During trial, Defendants readily acknowledged that in the Google DRM system usage rights are “associated with the content files.” Appx12663:19-20.

c. The jury was not required to accept Defendants’ alleged non-infringement evidence

ContentGuard acknowledges that, through the testimony of Dr. Clark—a professional litigation consultant—Defendants introduced evidence that allegedly established non-infringement. But the jury was not *required* to accept Defendants’ evidence and, based on the trial record, it had ample reasons to find infringement.

Defendants can make no contrary claim since, in post-trial briefing, they repeatedly asserted the opposite. That is, Defendants argued that “the judgment should stand” because “[a]fter seven days of evidence, argument, and instructions, the jury delivered a verdict of non-infringement” that “*appropriately resolved the factual disputes in Defendants’ favor.*” Appx9746 (emphasis added). Defendants repeatedly emphasized that all of their non-infringement arguments raised factual issues:

- With respect to the “attached or treated as attached” limitation, “[t]he jury heard both sides’ evidence and competing expert testimony, and rendered a verdict of non-infringement. *It was entitled to believe Defendants’ expert and evidence* over ContentGuard’s expert, and the assessment of their respective credibility and weight was a core jury function.” Appx9752 (emphasis added).
- “[R]esolution of the experts’ competing application of the Court’s construction to the accused product was *a factual question for the jury.*” Appx9753 (emphasis added).
- “[T]he jury had a right to rely upon Defendants’ evidence . . . and to reject any contrary evidence presented by ContentGuard. . . . Where competing expert testimony is presented on infringement, *the jury is free to credit one side’s expert and not the opposing expert.*” Appx9754 (emphasis added).
- The jury “*was entitled to believe*” that a design document that described usage rights and content as “tied” and “bound” to each other “did not apply to the accused Google Play system. The jury was also *entitled to believe* that even if the statements did not apply to the Google Play system, the usage rights were still not ‘attached or treated

as attached’ to the content. Either way, the jury *decided this factual issue against ContentGuard.*” Appx9757 (emphasis added).

- The question of whether the accused customer devices have “physical integrity” was a “*fact question*” properly submitted to the jury. Appx9768 (emphasis added).
- The question of whether the accused Google Play apps have “behavioral integrity” was a “*fact question*” properly submitted to the jury. Appx9776 (emphasis added).

In summary, Defendants defended the jury’s non-infringement verdict by taking the position that *all of their defenses raised factual questions* that were properly resolved by the jury, and that JMOL should not be entertained.

Defendants’ repeated and unambiguous concessions that a reasonable jury was *not required* to find non-infringement under a correct claim construction compel the conclusion that a new trial is appropriate. *Avid Tech.*, 812 F.3d at 1047 (error is “harmless only if a reasonable jury would have been required by the evidence to find non-infringement even without the error.”); *Ecolab, Inc.*, 285 F.3d at 1374 (error is harmless if it “could not have changed the result,” *i.e.*, if “the same verdict would necessarily be reached absent the error”).

4. The District Court’s evidentiary error that permitted Defendants to advance a “practicing the prior art”/prosecution disclaimer argument warrants a new trial

A new trial is also warranted because the District Court allowed Defendants, over ContentGuard’s objection,⁹ to mount an improper “practicing the prior art”/prosecution disclaimer defense.

At bottom, Defendants took the position that the Google DRM scheme cannot infringe because it allegedly embodies a prior-art approach to DRM that Dr. Stefik had “criticize[d]” and “walked away from” in his patents. Appx10579:14-25. This was stressed throughout trial:

- In opening statements, Defendants argued that *“this case will come down to whether you believe Google is using the license server approach or whether you believe they’re using Dr. Stefik’s trusted system approach, plain and simple. And ContentGuard won’t be able to dispute the evidence that Google uses the license server approach.”* Appx10586:5-9 (emphasis added).
- During the direct examination of Defendants’ non-infringement expert, Dr. Clark, Defendants highlighted alleged similarities between the Google DRM scheme and a prior art reference: “This is the Griswold system on the right. That’s a figure from the patent. And

⁹ See Appx10704-10707 (lines 4-15).

then this is the internal doc from Google that we've been looking at. And *we're going to see how they compare.*" Appx12022:5-24 (emphasis added).

- During Dr. Clark's direct examination, Defendants also stressed that Dr. Stefik "identified some shortcomings from the inventor's perspective of using th[e Griswold] type of system" and "recommended" a different approach "in light of the[] shortcomings" of the Griswold system. Appx12024-12035 (lines 20-24).
- In closing statement, Defendants again stressed the notion that there are "two fundamentally different approaches to DRM": (1) "the license server approach . . . [that] associates the licenses to the digital content using an identifier"; and (2) the "trusted system approach which uses devices that enforce the usage rights that are attached to the content and move with the content throughout the system." Appx12653:1-24. Further, Defendants stressed that the Google DRM scheme allegedly embodies "*a license server approach . . . the one thing that Dr. Stefik in his patent application said he didn't want.*" Appx12661:19-23 (emphasis added). Finally, Defendants argued that "the Griswold system *is like the Google system. . . [s]o if you somehow find that Google's licenses are attached or treated as*

attached to content because of an asset ID, then the Griswold system teaches attached or treated as attached licenses too. And that would make Griswold covered by the patents and that would mean the patents are invalid. They never would have issued in the first place. ContentGuard cannot have it both ways, ladies and gentlemen.”

Appx12668-12669, lines 19-6 (emphasis added).

At bottom, Defendants’ trial presentation suggested to the jury that there can be no infringement if the Google DRM system implements the Griswold “license server” approach. This was impermissible. As counsel for ContentGuard explained when he objected to Google’s plan to argue non-infringement based on the Griswold reference,

If [counsel] wants to prove up he doesn’t infringe, *he needs to start with the claims and the claims as interpreted, not what’s written in the patent specification or what was argued in the prosecution history.* And that is a disclaimer issue.

That is what [counsel] is trying to get this jury to believe that Dr. Stefik said what his patent wasn’t in the specification, *and that is not comparing the accused device to the claims as interpreted.* That is trying to argue a disclaimer by Dr. Stefik in the prosecution.

When he says: You said this Griswold patent isn’t your invention; this Griswold patent is a license server; therefore, a license server is not your invention, that’s a straight up disclaimer. That is not comparing the claims to the accused device and finding a missing piece.

Appx10704-10705 (lines 16-5) (emphasis added).

This Court has repeatedly held that “*there is no ‘practicing the prior art’ defense to literal infringement.*” *Tate Access Floors v. Interface Architectural Res.*, 279 F.3d 1357, 1365 (Fed. Cir. 2002) (emphasis added) (citing *Baxter Healthcare Corp. v. Spectramed, Inc.*, 49 F.3d 1575, 1583 (Fed. Cir. 1995)). See also *In re Omeprazole Patent Litig. v. Apotex Corp.*, 536 F.3d 1361, 1377 (Fed. Cir. 2008) (“It is well-established . . . that ‘*practicing the prior art*’ is not a *defense to infringement.*”) (emphasis added); *Ecolab, Inc.*, 285 F.3d at 1377 (“Paraclype argues that the district court improperly barred it from showing that [it] cannot infringe because its [accused product] more closely resembles the . . . prior art patent than the [plaintiff’s] patent. . . . But ‘practicing the prior art’ is not a defense to literal infringement.”).

It was thus error for the District Court to allow Defendants to argue that the jury should find non-infringement if it concludes that Griswold teaches a system “like the Google system.” And, in overruling ContentGuard’s objection to Defendants’ “practicing the prior art”/prosecution disclaimer assertions, the District Court enabled the very type of argument this Court found impermissible in *Baxter*:

Implicit in [the defendant’s] argument is that [the patentee], in order to establish literal infringement, must prove by a preponderance of the evidence that [the defendant’s] accused devices embody all the limitations in the asserted claims, and in addition, [the defendant’s] accused devices must not be an adoption of the combined

teachings of the prior art. ***This is not a correct statement of the law governing patent infringement.***

Baxter, 49 F.3d at 1583 (emphasis added).

The District Court's evidentiary error is particularly glaring given that, contrary to Defendants' contention, Dr. Stefik did not disclaim the use of a license server. Here is how Dr. Stefik described Griswold and its limitations in the patents' specification:

Griswold requires that the licensed product contain software to involve a license check monitor at predetermined time intervals. The license check monitor generates request datagrams which identify the licensee. ***The request datagrams are sent to a license control system over an appropriate communication facility.*** The license control system then checks the datagram to determine if the datagram is from a valid licensee. The license control system then sends a reply datagram to the license check monitor indicating denial or approval of usage. The license control system will deny usage in the event the request datagrams go unanswered after a predetermined period of time (which may indicate an unauthorized attempt to use the licensed product). In this system, ***usage is managed at a central location by the response datagrams. . . .***

It is argued by Griswold that the described system is advantageous because it can be implemented entirely in software. ***However, the system described by Griswold has limitations. An important limitation is that during the use of the licensed product, the user must always be coupled to an appropriate communication facility in order to send and receive datagrams.*** This creates a dependency on the communication facility. ***So if the communication facility is not available, the licensed product cannot be used.***

Appx192 at col. 2:12-38 (emphasis added). Contrary to Defendants' contention, Dr. Stefik's criticism had nothing to do with Griswold's use of a "license server" to deliver usage rights to consumer devices. Rather, the shortcoming Dr. Stefik identified in Griswold was the requirement that "the user *must always be coupled to an appropriate communication facility* in order to send and receive datagrams," in other words, the requirement that devices that use the Griswold solution remain in constant contact with the "central location." But there is nothing in Griswold that remotely suggests the use of the three "integrities" that comprise the trusted "repository" taught by Dr. Stefik's patents, and, contrary to Defendants' suggestion, a DRM system can implement *both* Griswold's teachings and Dr. Stefik's.

Particularly when coupled with the District Court's incorrect construction for usage rights, the evidentiary ruling that facilitated Defendants' "practicing the prior art"/prosecution disclaimer argument entitles ContentGuard to a new trial. Given how often Defendants repeated their argument that the Google DRM scheme does not infringe because it allegedly embodies Griswold, it is clear that the District Court's evidentiary ruling affected ContentGuard's substantial rights.

VI. CONCLUSION AND RELIEF REQUESTED

For all of these reasons, this Court should (i) construe usage rights as “indications that are associated with [a digital work / digital content / content / a digital document] and that indicate the manner in which the [digital work / digital content / content / digital document] may be used or distributed as well as any conditions on which use or distribution is premised”; and (ii) reverse the jury’s non-infringement verdict and remand this matter for a new trial.

Dated: October 13, 2016

Respectfully submitted,

/s/ Dirk D. Thomas

Dirk D. Thomas
McKool Smith P.C.
1999 K Street, Suite 600
Washington, DC 20006
(202) 370-8302

Robert A. Cote
McKool Smith P.C.
One Bryant Park, 47th Floor
New York, NY 10036
(212) 402-9400

*Attorneys for Plaintiff-Appellant
ContentGuard Holdings, Inc.*

ADDENDUM

2. Pursuant to the stipulations between the parties, (Dkt. No. 712 at 2), Motorola Mobility LLC, Huawei Device USA, Inc., Huawei Technologies Co., Ltd., HTC America, Inc., and HTC Corporation also do not infringe any of the asserted claims of the asserted patents.

3. Google and Samsung are the prevailing parties, and as the prevailing parties, Google and Samsung shall recover their costs from ContentGuard.

4. Any and all pending motions as to Google and Samsung are **DENIED**.

So ORDERED and SIGNED this 12th day of October, 2015.



RODNEY GILSTRAP
UNITED STATES DISTRICT JUDGE

Table of Contents

I. BACKGROUND..... 4

II. LEGAL PRINCIPLES 4

III. CONSTRUCTION OF AGREED TERMS 9

IV. CONSTRUCTION OF DISPUTED TERMS IN THE STEFIK PATENTS 10

 A. “repository” and “trusted” 10

 B. “physical integrity” 16

 C. “communications integrity” 18

 D. “behavioral integrity” 19

 E. “content” and “digital content” 21

 F. “rights,” “usage rights,” and “usage rights information” 23

 G. “usage rights” (‘160 Patent)..... 33

 H. “digital work” 35

 I. “digital document” and “document” 37

 J. “requester mode of operation” and “server mode of operation” 40

 K. “manner of use” 43

 L. “render” and “rendering” 45

 M. “authorization object” 48

 N. “identification certificate” and “digital certificate” 51

 O. “nonce” and “random registration identifier” 53

 P. “distributed repository” 56

 Q. “document platform” 61

 R. “validating” 65

 S. “determining, by the document platform” 68

 T. “grammar” 72

 U. “description structure” 75

 V. “means for communicating with a master repository for obtaining an identification certificate for the repository” 76

 W. “means for processing a request from the means for requesting” 80

 X. “means for checking whether the request is for a permitted rendering of the digital content in accordance with rights specified in the apparatus” 85

Y. “means for receiving the authorization ob[j]ect when it is determined that the request should be granted” 88

Z. “means for requesting a transfer of the digital content from an external memory to the storage” 91

AA. “means for processing the request to make the digital content available to the rendering engine for rendering when the request is for a permitted rendering of the digital [content],” “means for authorizing the repository for making the digital content available for rendering, wherein the digital content can be made available for rendering only by an authorized repository,” and “means for making a request for an authorization object required to be included within the repository for the apparatus to render the digital content” 95

V. CONSTRUCTION OF DISPUTED TERMS IN THE NGUYEN PATENTS 98

A. “repository” 99

B. “license” 99

C. “meta-right” 102

D. “usage rights” 106

E. “manner of use” 108

F. “state variable” 109

G. “the at least one state variable identifies a location where a state of rights is tracked” 114

H. “specifying, in a first license, . . . at least one usage right and at least one meta-right for the item, wherein the usage right and the meta-right include at least one right that is shared among one or more users or devices” 116

I. “means for obtaining a set of rights associated with an item” 119

J. “means for determining whether the rights consumer is entitled to the right specified by the meta-right” 122

K. “means for exercising the meta-right to create the right specified by the meta-right” 125

L. “means for generating a license including the created right, if the rights consumer is entitled to the right specified by the meta-right” 128

VI. CONSTRUCTION OF DISPUTED TERMS IN THE DUNKELD PATENT 130

A. “detect[ing] a transfer” 131

B. “instance” 133

C. “other portion” 136

D. “over said network between user devices” 140

VII. CONCLUSION 143

I. BACKGROUND

Plaintiff brings suit alleging infringement of United States Patents No. 6,963,859 (“the ‘859 Patent”), 7,523,072 (“the ‘072 Patent”), 7,225,160 (“the ‘160 Patent”), 7,269,576 (“the ‘576 Patent”), 8,370,956 (“the ‘956 Patent”), 8,393,007 (“the ‘007 Patent”) (collectively, the “Trusted Repository Patents” or “Stefik Patents”), 7,774,280 (“the ‘280 Patent”), 8,001,053 (“the ‘053 Patent”) (collectively, the “Meta Rights Patents,” “Nguyen/Chen Patents,” or “Nguyen Patents”), and 8,583,556 (“the ‘556 Patent,” also referred to as the “Transaction Tracking Patent” or the “Dunkeld Patent”) (all, collectively, “the patents-in-suit”). (Dkt. No. 304, Exs. A-I.)

The parties have presented the patents-in-suit as three distinct groups, as set forth above, and the Court addresses those three groups in turn, below.

The Court heard oral arguments on February 6, 2015. The parties did not present oral argument as to all disputed terms. Instead, “[g]iven the large number of disputed claim terms,” the parties chose to present oral arguments on terms identified in the parties’ January 23, 2015 Joint Notice Regarding *Markman* Hearing. (Dkt. No. 365.) The parties also presented oral argument regarding one additional group of terms identified by the Court, namely “nonce” and “random registration identifier” in the Stefik Patents. The parties did not present oral arguments regarding any other disputed terms and instead submitted those disputes on the briefing.

II. LEGAL PRINCIPLES

It is understood that “[a] claim in a patent provides the metes and bounds of the right which the patent confers on the patentee to exclude others from making, using or selling the protected invention.” *Burke, Inc. v. Bruno Indep. Living Aids, Inc.*, 183 F.3d 1334, 1340 (Fed. Cir. 1999). Claim construction is clearly an issue of law for the court to decide. *Markman v.*

Westview Instruments, Inc., 52 F.3d 967, 970-71 (Fed. Cir. 1995) (en banc), *aff'd*, 517 U.S. 370 (1996).

To ascertain the meaning of claims, courts look to three primary sources: the claims, the specification, and the prosecution history. *Markman*, 52 F.3d at 979. The specification must contain a written description of the invention that enables one of ordinary skill in the art to make and use the invention. *Id.* A patent's claims must be read in view of the specification, of which they are a part. *Id.* For claim construction purposes, the description may act as a sort of dictionary, which explains the invention and may define terms used in the claims. *Id.* "One purpose for examining the specification is to determine if the patentee has limited the scope of the claims." *Watts v. XL Sys., Inc.*, 232 F.3d 877, 882 (Fed. Cir. 2000).

Nonetheless, it is the function of the claims, not the specification, to set forth the limits of the patentee's invention. Otherwise, there would be no need for claims. *SRI Int'l v. Matsushita Elec. Corp.*, 775 F.2d 1107, 1121 (Fed. Cir. 1985) (en banc). The patentee is free to be his own lexicographer, but any special definition given to a word must be clearly set forth in the specification. *Intellicall, Inc. v. Phonometrics, Inc.*, 952 F.2d 1384, 1388 (Fed. Cir. 1992). Although the specification may indicate that certain embodiments are preferred, particular embodiments appearing in the specification will not be read into the claims when the claim language is broader than the embodiments. *Electro Med. Sys., S.A. v. Cooper Life Sciences, Inc.*, 34 F.3d 1048, 1054 (Fed. Cir. 1994).

This Court's claim construction analysis is substantially guided by the Federal Circuit's decision in *Phillips v. AWH Corporation*, 415 F.3d 1303 (Fed. Cir. 2005) (en banc). In *Phillips*, the court set forth several guideposts that courts should follow when construing claims. In particular, the court reiterated that "the claims of a patent define the invention to which the

patentee is entitled the right to exclude.” 415 F.3d at 1312 (quoting *Innova/Pure Water, Inc. v. Safari Water Filtration Sys., Inc.*, 381 F.3d 1111, 1115 (Fed. Cir. 2004)). To that end, the words used in a claim are generally given their ordinary and customary meaning. *Id.* The ordinary and customary meaning of a claim term “is the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention, i.e., as of the effective filing date of the patent application.” *Id.* at 1313. This principle of patent law flows naturally from the recognition that inventors are usually persons who are skilled in the field of the invention and that patents are addressed to, and intended to be read by, others skilled in the particular art. *Id.*

Despite the importance of claim terms, *Phillips* made clear that “the person of ordinary skill in the art is deemed to read the claim term not only in the context of the particular claim in which the disputed term appears, but in the context of the entire patent, including the specification.” *Id.* Although the claims themselves may provide guidance as to the meaning of particular terms, those terms are part of “a fully integrated written instrument.” *Id.* at 1315 (quoting *Markman*, 52 F.3d at 978). Thus, the *Phillips* court emphasized the specification as being the primary basis for construing the claims. *Id.* at 1314-17. As the Supreme Court stated long ago, “in case of doubt or ambiguity it is proper in all cases to refer back to the descriptive portions of the specification to aid in solving the doubt or in ascertaining the true intent and meaning of the language employed in the claims.” *Bates v. Coe*, 98 U.S. 31, 38 (1878). In addressing the role of the specification, the *Phillips* court quoted with approval its earlier observations from *Renishaw PLC v. Marposs Societa’ per Azioni*, 158 F.3d 1243, 1250 (Fed. Cir. 1998):

Ultimately, the interpretation to be given a term can only be determined and confirmed with a full understanding of what the inventors actually invented and intended to envelop with the claim. The construction that stays true to the claim

language and most naturally aligns with the patent's description of the invention will be, in the end, the correct construction.

Phillips, 415 F.3d at 1316. Consequently, *Phillips* emphasized the important role the specification plays in the claim construction process.

The prosecution history also continues to play an important role in claim interpretation. Like the specification, the prosecution history helps to demonstrate how the inventor and the United States Patent and Trademark Office (“PTO”) understood the patent. *Id.* at 1317. Because the file history, however, “represents an ongoing negotiation between the PTO and the applicant,” it may lack the clarity of the specification and thus be less useful in claim construction proceedings. *Id.* Nevertheless, the prosecution history is intrinsic evidence that is relevant to the determination of how the inventor understood the invention and whether the inventor limited the invention during prosecution by narrowing the scope of the claims. *Id.*; see *Microsoft Corp. v. Multi-Tech Sys., Inc.*, 357 F.3d 1340, 1350 (Fed. Cir. 2004) (noting that “a patentee’s statements during prosecution, whether relied on by the examiner or not, are relevant to claim interpretation”).

Phillips rejected any claim construction approach that sacrificed the intrinsic record in favor of extrinsic evidence, such as dictionary definitions or expert testimony. The *en banc* court condemned the suggestion made by *Texas Digital Systems, Inc. v. Telegenix, Inc.*, 308 F.3d 1193 (Fed. Cir. 2002), that a court should discern the ordinary meaning of the claim terms (through dictionaries or otherwise) before resorting to the specification for certain limited purposes. *Phillips*, 415 F.3d at 1319-24. According to *Phillips*, reliance on dictionary definitions at the expense of the specification had the effect of “focus[ing] the inquiry on the abstract meaning of words rather than on the meaning of claim terms within the context of the patent.” *Id.* at 1321.

Phillips emphasized that the patent system is based on the proposition that the claims cover only the invented subject matter. *Id.*

Phillips does not preclude all uses of dictionaries in claim construction proceedings. Instead, the court assigned dictionaries a role subordinate to the intrinsic record. In doing so, the court emphasized that claim construction issues are not resolved by any magic formula. The court did not impose any particular sequence of steps for a court to follow when it considers disputed claim language. *Id.* at 1323-25. Rather, *Phillips* held that a court must attach the appropriate weight to the intrinsic sources offered in support of a proposed claim construction, bearing in mind the general rule that the claims measure the scope of the patent grant.

In general, prior claim construction proceedings involving the same patents-in-suit are “entitled to reasoned deference under the broad principals of *stare decisis* and the goals articulated by the Supreme Court in *Markman*, even though *stare decisis* may not be applicable *per se.*” *Maurice Mitchell Innovations, LP v. Intel Corp.*, No. 2:04-CV-450, 2006 WL 1751779, at *4 (E.D. Tex. June 21, 2006) (Davis, J.); *see TQP Development, LLC v. Inuit Inc.*, No. 2:12-CV-180, 2014 WL 2810016, at *6 (E.D. Tex. June 20, 2014) (Bryson, J.) (“[P]revious claim constructions in cases involving the same patent are entitled to substantial weight, and the Court has determined that it will not depart from those constructions absent a strong reason for doing so.”); *see also Teva Pharm. USA, Inc. v. Sandoz, Inc.*, 135 S. Ct. 831, 839-40 (2015) (“prior cases will sometimes be binding because of issue preclusion and sometimes will serve as persuasive authority”) (citation omitted).

The Court nonetheless conducts an independent evaluation during claim construction proceedings. *See, e.g., Texas Instruments, Inc. v. Linear Techs. Corp.*, 182 F. Supp. 2d 580, 589-90 (E.D. Tex. 2002); *Burns, Morris & Stewart Ltd. P’ship v. Masonite Int’l Corp.*, 401 F.

Supp. 2d 692, 697 (E.D. Tex. 2005); *Negotiated Data Solutions, Inc. v. Apple, Inc.*, No. 2:11-CV-390, 2012 WL 6494240, at *5 (E.D. Tex. Dec. 13, 2012).

III. CONSTRUCTION OF AGREED TERMS

The Court hereby notes the Parties’ agreed constructions:

Stefik Patents	
<u>Term</u>	<u>Agreed Construction</u>
“rendering engine”	“a processor and associated software that renders”
“master device”	“A special type of device which issues identification certificates and distributes lists of repositories whose integrity has been compromised and which should be denied access to digital works (referred to as repository ‘hotlists’).”
“master repository”	“A special type of repository which issues identification certificates and distributes lists of repositories whose integrity have been compromised and which should be denied access to digital works (referred to as repository ‘hotlists’).”
“session key”	“a cryptographic key for encryption of messages during a single session”
“means for requesting use of the digital content stored in the storage”	“a user interface which is the mechanism by which a user interacts with a repository in order to invoke transactions to gain access to digital content, or exercise usage rights”
Nguyen/Chen Patents	
<u>Term</u>	<u>Agreed Construction</u>
“rights”	“The term ‘right’ in the claims of the ‘280 and ‘053 patents means a ‘meta-right’ or a ‘usage right,’ depending on context”

(Dkt. No. 292, 11/17/2014 Joint Claim Construction and Prehearing Statement, at 2; *see* Dkt. No. 366, Ex. B, 1/23/2015 Joint Claim Construction Chart.)

IV. CONSTRUCTION OF DISPUTED TERMS IN THE STEFIK PATENTS

The earliest issued of the Stefik Patents is the ‘859 Patent. The ‘859 Patent is titled “Content Rendering Repository” and issued on November 8, 2005. The Abstract states:

A rendering system adapted for use in a system for managing use of content and operative to rendering [*sic*] content in accordance with usage rights associated with the content. The system includes a rendering device configured to render the content and a repository coupled to the rendering device and operative to enforce usage rights associated with the content and permit the rendering device to render the content in accordance with a manner of use specified by the usage rights.

Four of the six Stefik Patents have been the subject of *Inter Partes* Review (“IPR”) proceedings at the PTO’s Patent Trial and Appeal Board (“PTAB”). (*See* Dkt. No. 331, Exs. 1-4).

The Stefik Patents all claim priority to an application filed on November 23, 1994. Defendants submit that the specifications of the Stefik Patents are “largely identical” except that, Defendants argue, “the ‘160 patent specification is critically different from the other Stefik specifications,” as discussed further below. (Dkt. No. 331, at 1 n.1.)

The present Memorandum Opinion and Order cites only the specification of the ‘859 Patent unless otherwise indicated.

A. “repository” and “trusted”

“repository”	
Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“a trusted system in that it maintains physical, communications, and behavioral integrity in the support of usage rights”	“a trusted system, which maintains physical, communications and behavioral integrity, and supports usage rights”

“trusted”	
Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“maintains physical, communications, and behavioral integrity in the support of usage rights”	“maintains physical, communications and behavioral integrity”

(Dkt. No. 304, at 1; Dkt. No. 331, at 2.) The parties submit that “repository” appears in Claims 1, 15, 21, 24, 58, 71, and 81 of the ‘859 Patent, Claims 1 and 18 of the ‘576 Patent, and Claims 1 and 10 of the ‘072 Patent. (Dkt. No. 292-1, at 7; Dkt. No. 331, at 2.) The parties submit that “trusted” appears in Claims 1, 7, and 13 of the ‘956 Patent and Claims 1, 6, and 11 of the ‘007 Patent. (Dkt. No. 292-1, at 8; Dkt. No. 331, at 2.)

(1) The Parties’ Positions

Plaintiff argues that its proposed construction for “repository” “adopts th[e] language from the [specification’s] glossary verbatim while Defendants’ proposed construction introduces ambiguities by replacing ‘in that it’ with ‘which’” and “by replacing ‘in the support of usage rights’ with ‘and supports usage rights.’” (Dkt. No. 304, at 2.) Plaintiff submits that although Defendants rely on the construction by the PTAB during an IPR, “[t]he PTAB based its construction on the same glossary definition [Plaintiff] relies on, but provided no reason to depart from the language from the glossary.” (*Id.*)

Defendants respond that “Defendants’ proposed construction of ‘repository’ follows the PTAB’s construction verbatim; and Defendants’ construction of the related term ‘trusted,’ which the PTAB did not construe, mirrors this construction.” (Dkt. No. 331, at 2 (citing, *id.*, Ex. 2, at 8).) Defendants submit that Plaintiff “actually rearranged pieces of the [specification glossary’s] definition to alter the meaning of ‘repository.’” (Dkt. No. 331, at 3.) Defendants

explain that “[u]nder [Plaintiff’s] construction, instead of the three integrities being required at all times, as taught by the Stefik patents and required by the PTAB’s construction, the three integrities only need to be present when supporting usage rights.” (*Id.*)

In an additional, separate responsive brief, Defendant Amazon argues that because the specification defines “repository” and “trusted” in “purely functional language,” those terms are indefinite. (Dkt. No. 336, at 3.) Defendant Amazon cites *Halliburton Energy Services, Inc. v. M-I LLC*, 514 F.3d 1244 (Fed. Cir. 2008), which found indefinite the term “fragile gel.” (*See id.*, at 3-5.)

Plaintiff replies that Defendants’ argument that a repository must “maintain the three integrities at all times” “is not found in the PTAB construction and directly contradicts the Stefik patents’ specification” (Dkt. No. 345, at 1.) Plaintiff concludes that “[t]here is simply no basis for defining ‘repository’ as something that maintains the three integrities *at all times*, even while conducting transactions that do not support usage rights.” (*Id.*)

Plaintiff also replies, as to Amazon’s separate brief, that “Amazon’s arguments should be rejected because they reflect an elementary misunderstanding of the applicable law and are not supported by any evidence.” (Dkt. No. 344, at 1.) Plaintiff notes that Amazon submits no expert opinions on this issue, and Plaintiff submits that “there is ample support in the specification that describes the boundaries of the three integrities that define [Mr.] Stefik’s concept of ‘trust.’” (*Id.*, at 3-5 (citing 11:62-12:50).)

In sur-reply, Defendants argue that, “[l]ogically, [Mr.] Stefik must have intended for repositories and trusted systems to require the three integrities at all times, otherwise his inventions would not solve the digital piracy problem.” (Dkt. No. 353, at 1.) Defendants also note that a “restoration file,” which is used to restore a back-up file, “would be held in [a]

repository,” and “[i]f a repository cannot verify that it is communicating with another trusted repository, then ‘the registration transaction terminates in an error.’” (*Id.*, at 3 (citing ‘859 Patent at 27:3-5 & 36:57-58).)

At the February 6, 2015 hearing, Defendants reiterated that “in support of” is broader than how the PTAB construed the term and injects ambiguity into the claims.

(2) Analysis

The parties disagree as to whether the disputed terms refer to *supporting* usage rights or merely being “*in the support of* usage rights,” as well as whether the three “integrities” must be present at all times.

On one hand, the PTAB construed “repository” to mean “a trusted system which maintains physical, communications and behavioral integrity, and supports usage rights.” (Dkt. No. 304, Ex. J, 6/26/2014 Final Written Decision, at 10-11.). This prior construction is entitled to some deference. *See Maurice Mitchell*, 2006 WL 1751779, at *4; *see also TQP*, 2014 WL 2810016, at *6; *Teva*, 135 S. Ct. at 839-40.

On the other hand, the “Glossary” section of the specification explicitly states:

Repository:

Conceptually a set of functional specifications defining core functionality in the support of usage rights. A repository is a trusted system in that it maintains physical, communications and behavioral integrity.

‘859 Patent at 50:47-51.

On balance, the Court finds that by setting forth an explicit definition in a “Glossary,” the patentee acted as lexicographer and expressly defined the term “repository.” *See Intellicall*, 952 F.2d at 1388; *see also C.R. Bard, Inc. v. U.S. Surgical Corp.*, 388 F.3d 858, 862 (Fed. Cir. 2004) (“[T]he inventor’s written description of the invention . . . is relevant and controlling insofar as it

provides clear lexicography”); *Abbott Labs. v. Syntron Bioresearch, Inc.*, 334 F.3d 1343, 1354 (Fed. Cir. 2003) (“patentee’s lexicography must, of course, appear with reasonable clarity, deliberateness, and precision”) (citation and internal quotation marks omitted).

This lexicography finding is supported by other disclosures in the specification, such as the discussion of “Repositories”:

Repositories

Many of the powerful functions of repositories—such as their ability to “loan” digital works or automatically handle the commercial reuse of digital works—are possible because they are trusted systems. The systems are trusted because they are able to take responsibility for fairly and reliably carrying out the commercial transactions. That the systems can be responsible (“able to respond”) is fundamentally an issue of integrity. The integrity of repositories has three parts: physical integrity, communications integrity, and behavioral integrity.

‘859 Patent at 11:51-61; *see also* 6:29-31 (“the digital work genie only moves from one trusted bottle (repository) to another”). The specification also discloses that a repository may communicate with a non-repository and that not all communications between repositories are secure. *See id.* at 25:37-52, 26:30-67 (“registration transaction”) & 37:12-21 (“non-repository archive storage”).

To whatever extent Defendant Amazon maintains that Plaintiff’s construction is improperly functional rather than structural, that argument is rejected. *See, e.g., Hill-Rom Servs., Inc. v. Stryker Corp.*, 755 F.3d 1367, 1374-75 (Fed. Cir. 2014) (“defining a particular claim term by its function is not improper”); *Funai Elec. Co. v. Daewoo Elecs. Corp.*, 616 F.3d 1357, 1366 (Fed. Cir. 2010) (“The use of comparative and functional language to construe and explain a claim term is not improper. A description of what a component does may add clarity and understanding to the meaning and scope of the claim.”); *Microprocessor Enhancement Corp. v.*

Texas Instruments Inc., 520 F.3d 1367, 1375 (Fed. Cir. 2008) (“claims are not necessarily indefinite for using functional language”).

As to extrinsic evidence, Defendants have also cited an article in which named inventor Mark Stefik stated that a “trusted system” “could always be counted on to follow the rules of the trust,” and “[i]n the case of digital works on repositories, the requirement for trust is that the repositories follow—at all times and in every instance—the rules about how digital works are used.” (Dkt. No. 331, Ex. 5, Mark Stefik, *Letting Loose the Light: Igniting Commerce in Electronic Publication* 12, 24 (1996).) This extrinsic evidence is of insufficient weight, however, to override the explicit lexicography in the specification, as set forth above. *See Phillips*, 415 F.3d at 1317 (“[W]hile extrinsic evidence can shed useful light on the relevant art, we have explained that it is less significant than the intrinsic record in determining the legally operative meaning of claim language.”) (citations and internal quotation marks omitted).

That lexicography, on its face, requires only that the integrities be maintained “in support of” usage rights. Defendants have failed to adequately support their proposed “at all times” interpretation. (*See* Dkt. No. 353, at 1.)

The Court accordingly hereby construes these disputed terms as set forth in the following chart:

<u>Term</u>	<u>Construction</u>
“repository”	“a trusted system in that it maintains physical, communications, and behavioral integrity in the support of usage rights”
“trusted”	“maintains physical, communications, and behavioral integrity in the support of usage rights”

B. “physical integrity”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“prevents access to content by a non-trusted system”	“preventing access to information by a non-trusted system”

(Dkt. No. 304, at 1; Dkt. No. 331, at 4.) Defendants submit that this disputed term appears in Claims 1, 15, 21, 24, 58, 71, and 81 of the ‘859 Patent, Claims 1 and 18 of the ‘576 Patent, Claims 1 and 10 of the ‘072 Patent, Claims 1, 7, and 13 of the ‘956 Patent, and Claims 1, 6, and 11 of the ‘007 Patent. (Dkt. No. 331, at 4.)

(1) The Parties’ Positions

Plaintiff argues that whereas its proposal is supported by the specification and “clarifies for the jury that the relevant ‘information’ is the ‘content,’ as described by the specification,” “Defendants’ proposed construction does not identify what ‘information’ the physical integrity applies to.” (Dkt. No. 304, at 3.)

Defendants argue that “[Plaintiff] attempts to broaden the PTAB’s construction, so that repositories and trusted systems need only prevent non-trusted systems from accessing ‘content’” rather than “information.” (Dkt. No. 331, at 4.) Defendants explain that the PTAB used the word “information” to encompass “data,” “content,” and “digital works.” (*Id.* (citing *id.*, Ex. 2, at 10; citing ‘859 Patent at 11:62-12:20).)

Plaintiff replies that Defendants’ proposal is contrary to the PTAB decision, and “a repository cannot determine whether it is communicating with another trusted repository without first exchanging at least some preliminary, unprotected information” (Dkt. No. 345, at 1.)

At the February 6, 2015 hearing, Plaintiff emphasized that the PTAB found that the terms “content,” “data,” “digital work,” and “information” were used interchangeably. (*See* Dkt.

No. 304, Ex. J, 6/26/2014 Final Written Decision ('576 Patent), at 12; Dkt. No. 331, Ex. 2, 7/1/2014 Final Written Decision ('859 Patent), at 10.)

(2) Analysis

On one hand, the specification refers to protecting “works.” ‘859 Patent at 12:1-5 (“[T]he repository design protects access to the content of digital works. . . . [R]epositories never allow non-trusted systems to access the works directly.”) & 11:63-64 (“Physical integrity applies both to the repositories and to the protected digital works.”). Further, the specification discloses that unsecured communications may be used as part of a “registration process.” *See id.* at 26:30-67 (“registration transaction”).

Also, the PTAB noted that the specification “appears to use” the terms “information” and “content” “interchangeably.” (Dkt. No. 304, Ex. J, 6/26/2014 Final Written Decision ('576 Patent),² at 12 (emphasis added); *see* Dkt. No. 331, Ex. 2, 7/1/2014 Final Written Decision ('859 Patent), at 10 (same).)

On the other hand, the PTAB construed “physical integrity” to mean “preventing access to *information* by a non-trusted system.” (Dkt. No. 304, Ex. J, 6/26/2014 Final Written Decision ('576 Patent), at 12 (emphasis added); *see* Dkt. No. 331, Ex. 2, 7/1/2014 Final Written Decision ('859 Patent), at 10 (same).)

The specification supports such a reading of “physical integrity” by disclosing the importance of protecting the repository itself:

Physical integrity refers to the integrity of the physical devices themselves. Physical integrity applies both to the repositories and to the protected digital works. Thus, the higher security classes of repositories themselves may have sensors that detect when tampering is attempted on their secure cases. In addition

² For convenience, the present Memorandum Opinion and Order in some instances uses parentheticals to identify the patent that is related to a particular exhibit.

to protection of the repository itself, the repository design protects access to the content of digital works. In contrast with the design of conventional magnetic and optical devices—such as floppy disks, CD-ROMs, and videotapes—repositories never allow non-trusted systems to access the works directly. A maker of generic computer systems cannot guarantee that their platform will not be used to make unauthorized copies. The manufacturer provides generic capabilities for reading and writing information, and the general nature of the functionality of the general computing device depends on it. Thus, a copy program can copy arbitrary data. This copying issue is not limited to general purpose computers. It also arises for the unauthorized duplication of entertainment “software” such as video and audio recordings by magnetic recorders. Again, the functionality of the recorders depends on their ability to copy and they have no means to check whether a copy is authorized. In contrast, repositories prevent access to the raw data by general devices and can test explicit rights and conditions before copying or otherwise granting access. *Information is only accessed by protocol between trusted repositories.*

‘859 Patent at 11:62-12:20 (emphasis added).

The Court accordingly hereby construes **“physical integrity”** to mean **“preventing access to information in a repository by a non-trusted system.”**

C. “communications integrity”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“only communicates with other devices that are able to present proof that they are trusted systems, for example, by using security measures such as encryption, exchange of digital certificates, and nonces”	“only communicates with other devices that are able to present proof that they are trusted systems, e.g., by using security measures such as encryption, exchange of digital certificates, and nonces”

(Dkt. No. 304, at 1; Dkt. No. 292-2, at 1.)

Defendants submit that “[t]he parties do not differ substantively about the definition of communications integrity, so the Defendants have not addressed that term in [their response] brief.” (Dkt. No. 331, at 2 n.2.) At the February 6, 2015 hearing, the parties did not address this term.

The Court accordingly hereby construes **“communications integrity”** to mean **“only communicates with other devices that are able to present proof that they are trusted**

systems, for example, by using security measures such as encryption, exchange of digital certificates, and nonces.”

D. “behavioral integrity”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“requires software that is to be installed in the repository to include a digital certificate, in other words, an assurance that the software comes from a source known to the repository”	“requiring software to include a digital certificate in order to be installed in the repository”

(Dkt. No. 304, at 1; Dkt. No. 331, at 4-5.) Defendants submit that this disputed term appears in Claims 1, 15, 21, 24, 58, 71, and 81 of the ‘859 Patent, Claims 1 and 18 of the ‘576 Patent, Claims 1 and 10 of the ‘072 Patent, Claims 1, 7, and 13 of the ‘956 Patent, and Claims 1, 6, and 11 of the ‘007 Patent. (Dkt. No. 331, at 4-5.)

(1) The Parties’ Positions

Plaintiff argues that its proposed construction is supported by the specification as well as by the PTAB’s construction and understanding. (Dkt. No. 304, at 3.)

Defendants respond that “[t]he PTAB rejected essentially the same argument” that Plaintiff has presented here, namely “to redefine the term ‘digital certificate’ to encompass any ‘assurance that the software comes from a source known to the repository.’” (Dkt. No. 331, at 5.) Defendants argue that “[h]aving survived the IPRs based on [the PTAB’s] construction [(requiring the use of a digital certificate)], [Plaintiff] now makes a second attempt at eliminating the digital certificate requirement.” (*Id.*)

Plaintiff replies that Defendants’ attempt to “limit ‘digital certificate’ . . . to an exemplary form” “is contrary to the PTAB’s decisions.” (Dkt. No. 345, at 2.)

(2) Analysis

On one hand, the specification discloses:

The integrity of the software is generally assured only by knowledge of its source. Restated, a user will trust software purchased at a reputable computer store but not trust software obtained off a random (insecure) server on a network. Behavioral integrity is maintained by requiring that repository software be certified and be distributed with proof of such certification, i.e. a *digital certificate*. The purpose of the certificate is to authenticate that the software has been tested by an authorized organization, which attests that the software does what it is supposed to do and that it does not compromise the behavioral integrity of a repository. If the digital certificate cannot be found in the digital work or the master repository which generated the certificate is not known to the repository receiving the software, then the software cannot be installed.

‘859 Patent at 12:36-50 (emphasis added). The PTAB also “credit[ed] the testimony of [Plaintiff’s] expert, Dr. Goodrich, that ‘a person of ordinary skill in the art [in 1994] would [have understood] a digital certificate to be an assurance that downloaded software comes from a reputable source, including a measure of tamper resistance.’” (Dkt. No. 331, Ex. 2, 7/1/2014 Final Written Decision (‘859 Patent), at 26 (square brackets PTAB’s).)

On the other hand, the PTAB construed “behavioral integrity” as “requiring software to include a digital certificate in order to be installed in the repository.” (Dkt. No. 304, Ex. J, 6/26/2014 Final Written Decision (‘576 Patent), at 13; *see* Dkt. No. 331, Ex. 2, 7/1/2014 Final Written Decision (‘859 Patent), at 11 (same).) The PTAB also found:

We do not credit the testimony of the expert witness of [Plaintiff], Dr. Goodrich, that “a person of ordinary skill in the art of 1994 would [have understood] that the ‘859 patent specification refers to the use of digital certificates as only an exemplary method of preserving the behavioral integrity of a repository.” The testimony is unexplained and conclusory; it does not account for the various factors we have considered and discussed above.

(Dkt. No. 331, Ex. 2, 7/1/2014 Final Written Decision (‘859 Patent), at 20 (citation omitted).)

Defendants also object to Plaintiff’s citation of Dr. Goodrich’s statement made in the IPR as hearsay. (Dkt. No. 331, at 6 n.6 (citing Fed. R. Evid. 801(c)).)

On balance, Plaintiff has failed to justify departing from the PTAB’s construction, which is entitled to “reasoned deference.” *Maurice Mitchell*, 2006 WL 1751779, at *4; *see TQP*, 2014

WL 2810016, at *6; *see also Teva*, 135 S. Ct. at 839-40. Plaintiff’s proposed construction, which includes additional language that would tend to broaden the scope of the disputed term, is therefore rejected.

The Court accordingly hereby construes **“behavioral integrity”** to mean **“requiring software to include a digital certificate in order to be installed in the repository.”**

E. “content” and “digital content”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“any work that has been reduced to a digital representation”	“the digital information (i.e., raw bits) representing a digital work”

(Dkt. No. 304, at 4; Dkt. No. 331, at 13-14.) The parties submit that these disputed terms appear in Claims 1 and 58 of the ‘859 Patent, Claims 1, 4, 5, 7, 10, 11, 13, 16, and 17 of the ‘956 Patent, Claims 1, 3, 4, 6, 8, 9, 11, and 13 of the ‘007 Patent, Claims 1, 4, 7, 18, 21, 24, and 34 of the ‘576 Patent, and Claims 1, 9, and 10 of the ‘160 Patent. (Dkt. No. 292-1, at 2; Dkt. No. 331, at 13.)

(1) The Parties’ Positions

Plaintiff argues that Defendants are attempting to “bootstrap ‘digital work’ and their corresponding interpretation into claims that do not contain the term.” (Dkt. No. 304, at 4.) Plaintiff urges that “[t]he final sentence of the ‘digital work’ glossary definition injects an optional feature of the preferred embodiment, and represents a departure from the plain and ordinary meaning of ‘content.’” (Dkt. No. 304, at 5.) “[T]he patentee chose to use ‘content’ rather than ‘digital work’ in the claims,” Plaintiff emphasizes. (*Id.*, at 6.) Finally, Plaintiff submits that “in distinguishing over prior art in a related application, the patentee cited only the first two sentences from the glossary entry as the definition for ‘digital work.’” (*Id.*)

Defendants respond that their proposals for “content” and “digital work” are consistent with one another because, as defined in the specification, “‘digital work’ and ‘content’ are closely related, interdependent terms, with content referring to the ‘digital information (i.e. raw bits)’ that make up a digital work, and digital work referring to encapsulating that ‘digital information.’” (Dkt. No. 331, at 14.)

Plaintiff replies that “[t]here is no basis to rewrite the claims as Defendants propose.” (Dkt. No. 345, at 6.)

(2) Analysis

The parties have not submitted any previous construction, by the PTAB or otherwise, for these disputed terms.

The specification sets forth two definitions for “content.” First, the specification states: “Herein the terms ‘digital work,’ ‘work’ and ‘content’ refer to any work that has been reduced to a digital representation.” ‘859 Patent at 5:64-66. Second, the “Glossary” section of the specification states:

Content:

The digital information (i.e. raw bits) representing a digital work.

Id. at 49:52-54.

The term “digital work,” in turn, is defined in the “Glossary” as follows:

Digital Work (Work):

Any encapsulated digital information. Such digital information may represent music, a magazine or book, or a multimedia composition. Usage rights and fees are attached to the digital work.

Id. at 50:8-12. Fees, however, are optional. *Id.* at 18:19-29 (“In the currently preferred embodiment . . . no fee is required.”).

The definition of “digital work” is probative, particularly given that the first of the above-quoted definitions for “content” purports to define not only “content” but also “digital work.” In light of this, the “Glossary” definition proposed by Defendants is more appropriate.

The Court accordingly hereby construes **“content”** and **“digital content”** to mean **“the digital information (i.e., raw bits) representing a digital work.”**

F. “rights,” “usage rights,” and “usage rights information”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“an indication of the manner in which a [digital work / digital content / content / a digital document] may be used or distributed as well as any conditions on which use or distribution is premised”	“statements in a language for defining the manner in which a digital work may be used or distributed, as well as any conditions on which use or distribution is premised. Usage rights must be permanently attached to the digital work” ³

(Dkt. No. 304, at 4 (square brackets Plaintiff’s); Dkt. No. 331, at 7.) The parties submit that these disputed terms appear in Claims 1 and 58 of the ‘859 Patent, Claims 1, 15, 18, and 32 of the ‘576 Patent, Claims 1, 7, and 13 of the ‘956 Patent, Claims 1, 6, and 11 of the ‘007 Patent, and Claims 1, 8, 10, and 16 of the ‘072 Patent. (Dkt. No. 292-1, at 8; Dkt. No. 331, at 7.)

(1) “language”

(a) The Parties’ Positions

Plaintiff submits:

In the preferred embodiment, the usage rights are *expressed* in a usage rights language, but are not themselves a language. Incongruously, the glossary entry for “usage rights” recites “a language for defining the manner in which a digital work may be used or distributed, as well as any conditions on which use or distribution is premised.” The glossary incorrectly purports to define “usage

³ Defendants previously proposed: “A language for defining the manner in which a digital work may be used or distributed, as well as any conditions on which use or distribution is premised. Rights must be attached to the digital work.” (Dkt. No. 292-2, at 9.)

rights” while actually defining “usage rights language.” . . . This error in the glossary definition was corrected in the ’160 patent.

(Dkt. No. 304, at 8-9 (citing ’859 Patent at 16:63-66 & 51:7-10; citing ’160 Patent at 48:24-26).)

Defendants respond that “[i]n the twenty years of multiple of [*sic*] patent applications prosecuted since it filed the parent application, [Plaintiff] has *never* suggested that the ‘usage rights’ definition appearing across [the] Stefik patent family contains an error or ambiguity.”

(Dkt. No. 331, at 8.)

Plaintiff replies: “[i]f interpretable code expresses a usage right and thus qualifies as a ‘statement in a language,’ then Defendants’ proposal adds nothing. The claims already require computer implementation, and all data recorded, read, or communicated by a computer is encoded. On the other hand, if a code or number interpreted to express a usage right does not meet Defendants’ ‘statement in a language’ limitation, then Defendants’ construction excludes a preferred embodiment.” (Dkt. No. 345, at 6.)

In sur-reply, Defendants reply that the disclosure of a “right code” (quoted below) “merely explains that each right will have a designated right code ‘assigned to’ correspond to it in a descriptor block, not that this code itself constitutes the right.” (Dkt. No. 353, at 3.)

(b) Analysis

The “Glossary” section of the specification of the Stefik Patents (except for the ’160 Patent) states:

Usage Rights:

A language for defining the manner in which a digital work may be used or distributed, as well as any conditions on which use or distribution is premised.

’859 Patent at 51:7-10. The specification of one of the Stefik Patents, the ’160 Patent, includes a “Glossary” section that defines “usage rights” as an “indication” rather than as a language:

Usage Rights—An indication of the manner of use by which a digital work may be used or distributed, as well as any conditions on which manner of use is premised.

‘160 Patent at 48:24-26. As to this disclosure of “indication,” the specification also discloses that “[a] right 1450 [in Fig. 14] has a label (e.g. COPY or PRINT) which *indicate[s]* the use or distribution privileges that are embodied by the right.” ‘859 Patent at 17:25-28 (emphasis added).

The specification includes disclosures of usage rights as being in “a language.” *See id.* at 16:63-25:35 & 51:4-6; *see also id.* at Fig. 15. In particular, the specification discloses:

The present invention uses statements in a high level “*usage rights language*” to define rights associated with digital works and their parts. Usage rights statements are interpreted by repositories and are used to determine what transactions can be successfully carried out for a digital work and also to determine parameters for those transactions. For example, *sentences in the language* determine whether a given digital work can be copied, when and how it can be used, and what fees (if any) are to be charged for that use. Once the usage rights statements are generated, they are encoded in a suitable form for accessing during the processing of transactions.

Defining usage rights in terms of a language in combination with the hierarchical representation of a digital work enables the support of a wide variety of distribution and fee schemes.

Id. at 16:63-17:12 (emphasis added).

The specification also discloses that a usage right can be specified by merely a “right code.” ‘859 Patent at 9:47-57 (“The right code field 1001 will contain a unique code assigned to a right. . . . The rights as stored in the rights portion 304 may typically be in numerical order based on the right code.”).

These disclosures are consistent with reading the definition in the “Glossary” section of the specification as indicating that a “usage right” can be either expressed in a language or with a code. Further, the specification of one of the Stefik Patents, the ‘160 Patent, includes a

“Glossary” section that defines “usage rights” as an “indication” rather than in terms of a language, as set forth above. ‘160 Patent at 48:24-26; *see Aventis Pharm. Inc. v. Amino Chemicals Ltd.*, 715 F.3d 1363, 1380 (Fed. Cir. 2013) (“[W]e presume, unless otherwise compelled, that the same claim term in the same patent or related patents carries the same construed meaning.”) (quoting *Omega Eng’g v. Raytek Corp.*, 334 F.3d 1314, 1334 (Fed. Cir. 2003)).

Defendants’ proposal of requiring a “language” is therefore hereby expressly rejected.

(2) “permanently attached”

(a) The Parties’ Positions

Plaintiff argues that Defendants import a “permanently attached” limitation from a preferred embodiment. (Dkt. No. 304, at 4.) Plaintiff submits that “the agreed portion of the parties’ constructions for the rights terms already encompasses the ‘attachment’ contemplated by the patent.” (*Id.*, at 7.) Plaintiff also argues, for example, that “the content associated with the right may exist both before and after the term or life of the right.” (*Id.*, at 8.)

Defendants respond by citing “the unequivocal statement in the specification that, ‘A key feature of the present invention is that usage rights are permanently ‘attached’ to the digital work.’ [‘859 Patent at] 6:11-12.” (Dkt. No. 331, at 8 (emphasis omitted).) Defendants also submit: “the ‘description tree’ is different and distinct from the actual ‘usage rights’ for a digital work (which are merely ‘described’ by the description tree), and the fact that the ‘description tree’ file can be stored separately from the ‘contents’ file of a work does not, and cannot, negate the patents’ explicit requirement that ‘usage rights are permanently ‘attached’ to the digital work.’” (*Id.*, at 10.) Finally, Defendants argue that “[Plaintiff’s] efforts to eliminate the ‘attachment’ of usage rights teachings, and, instead, to inject ‘associated with’ teachings into the

'160 patent specification, demonstrate that [Plaintiff], itself, recognizes the difference between 'attachment' and mere 'association' of usage rights and content, and that [Plaintiff] did not view the '859, '576, '072, '956 and '007 patents as teaching mere 'association.'" (*Id.*, at 11.)

Plaintiff replies that Defendants' proposal "modifies the glossary definition of 'usage rights' and misinterprets the specification's teachings concerning how usage rights are permanently 'attached.'" (Dkt. No. 345, at 2.) Plaintiff submits that "'attachment' is a metaphor to explain that rights should be associated with content, while 'permanent' attachment means that that association persists for the term or life of the right." (*Id.*, at 2-3.) Plaintiff concludes that "Defendants' attempt to import the undefined requirement of permanent 'attachment' into the claims could only mislead a jury into believing that usage rights must physically be a part of the content file(s) that comprise a digital work." (*Id.*, at 3.) Finally, Plaintiff argues that the "description tree" disclosure cited by Defendants does *not* establish that "the description tree structure describes a single location of content and usage rights (akin to a table of contents)." (*Id.*, at 5.)

In sur-reply, Defendants argue that "[g]iven that 'attachment' is a specific type of association, it comes as no surprise that the Stefik patents occasionally use the phrase 'associated with.'" (Dkt. No. 353, at 4.) Defendants also argue that "[d]escription trees are not the same as usage rights, which [Plaintiff] itself recognized by claiming the two as separate and distinct elements in the '160 patent." (*Id.*, at 5 (citing '160 Patent at 48:31-51 [Claim 1]).)

At the February 6, 2015 hearing, Plaintiff urged that Defendants' proposal of "permanent" attachment must be rejected because a usage right is associated with a work only for the life of the usage right. In other words, Plaintiff argued, the life of the usage right may be

shorter than the life of the work. Defendants responded that the specification consistently discloses that works are inseparable from usage rights.

(b) Analysis

Claim 1 of the '859 Patent, for example, recites "usage rights associated with the content."

The specification discloses, however, that "[a] key feature of the present invention is that usage rights are permanently 'attached' to the digital work." '859 Patent at 6:11-12; *see id.* at 3:50-52 ("It would be desirable to have a distribution system where the means for billing is always transported with the work") & 6:25-32 ("the present invention never separates the fee descriptions from the work"). Similarly, the specification also discloses:

Attaching Usage Rights to a Digital Work

It is fundamental to the present invention that the usage rights are treated as part of the digital work. As the digital work is distributed, the scope of the granted usage rights will remain the same or may be narrowed. For example, when a digital work is transferred from a document server to a repository, the usage rights may include the right to loan a copy for a predetermined period of time (called the original rights). When the repository loans out a copy of the digital work, the usage rights in the loaner copy (called the next set of rights) could be set to prohibit any further rights to loan out the copy. The basic idea is that one cannot grant more rights than they have.

The attachment of usage rights into a digital work may occur in a variety of ways. If the usage rights will be the same for an entire digital work, they could be attached when the digital work is processed for deposit in the digital work server. In the case of a digital work having different usage rights for the various components, this can be done as the digital work is being created. An authoring tool or digital work assembling tool could be utilized which provides for an automated process of attaching the usage rights.

Id. at 10:44-65 (emphasis added); *see id.* at 6:14-16 ("[T]he usage rights and any associated fees assigned by a creator and subsequent distributor will always remain with a digital work."); *see also id.* at 6:22-23 ("[t]he combination of attached usage rights and repositories enable distinct

advantages over prior systems”) & 18:13-17 (“The set of rights attached to a digital work define how that digital work may be transferred, used, performed or played. A set of rights will attach to the entire digital work and in the case of compound digital works, each of the components of the digital work.”).

Further, the “Glossary” section of the specification defines “digital work” and “composite digital work” as requiring that usage rights are “attached”:

Digital Work (Work):

Any encapsulated digital information. Such digital information may represent music, a magazine or book, or a multimedia composition. *Usage rights and fees are attached to the digital work.*

Id. at 50:8-12 (emphasis added).

Composite Digital Work:

A digital work comprised of distinguishable parts. Each of the distinguishable parts is itself a digital work *which have usage rights attached.*

Id. at 49:48-51 (emphasis added).

The specification also appears to suggest that usage rights and their associated content can be stored in separate “descriptor” and “contents” files, respectively. ‘859 Patent at 8:46-54 (“The description tree file makes it possible to examine the rights and fees for a work without reference to the content of the digital work.”); *see id.* at Fig. 12 (illustrating “Descriptor Storage 1203” distinct from “Content Storage 1204”). The storage can even be on separate devices:

The description tree storage 1203 and content storage 1204 need not be of the same type of storage medium, nor are they necessarily on the same physical device. So for example, the descriptor storage 1203 may be stored on a solid state storage (for rapid retrieval of the description tree information), while the content storage 1204 may be on a high capacity storage such as an optical disk.

Id. at 13:41-47; *see also id.* at 9:10-25 (regarding “rights portion 704” of description tree, illustrated in Figure 7). Further, the specification discloses that “usage rights” can be “associated with” content. *See* ’859 Patent at 3:53-60, 6:52-53, 16:64-66, 18:39-41, 30:8-9 & 30:13-16.

On the whole, however, the specification refers to a “description tree” as containing descriptions of usage rights rather than usage rights themselves:

Description Tree:

A structure which *describes* the location of content and the usage rights and usage fees for a digital work. A description tree is comprised of description blocks. Each description block corresponds to a digital work or to an interest (typically a revenue bearing interest) in a digital work.

See ’859 Patent at 50:1-7 (appearing in “Glossary” section of the specification; emphasis added).

Defendants’ proposal of “attached” is thus adequately supported by the explicit glossary definitions as well as the above-quoted discussions of “the present invention.” *See Intellicall*, 952 F.2d at 1388; *see also Regents of the Univ. of Minn. v. AGA Med. Corp.*, 717 F.3d 929, 936 (Fed. Cir. 2013) (“When a patent . . . describes the features of the ‘present invention’ as a whole, this description limits the scope of the invention.”) (quoting *Verizon Servs. Corp. v. Vonage Holdings Corp.*, 503 F.3d 1295, 1308 (Fed. Cir. 2007)); *Honeywell Int’l, Inc. v. ITT Indus., Inc.*, 452 F.3d 1312, 1318 (Fed. Cir. 2006) (construing term in light of description of “the present invention”).

As for the prosecution history, during prosecution of the ’072 Patent the patentee distinguished a prior art reference in part on the basis that the prior art reference at issue required the alleged usage rights to be stored in the same file, and necessarily the same device, as the alleged content:

Independent claim 26, recites, in relevant part, . . . the at least one of the at least one usage rights is stored *separately* from the digital document.

In contrast, as is shown in the diagram below, Perritt is directed to a digital library system, wherein permissions header (PH) is *attached* to a work (W)

(Dkt. No. 304, Ex. M, 5/23/2008 Amendment After Final ('072 Patent) at 8 (emphasis modified).)

[C]ontrary to present independent claims 1, 14 and 26, since the permissions header and the work of Perritt are attached to one another, the permissions header must be stored in the same file as the work.

(*Id.*, Ex. N, 10/15/2008 Response to Office Action ('072 Patent) at 11-12.)

As to extrinsic evidence, Plaintiff submits the opinion of its expert, Dr. Goodrich, that the Stefik Patents teach that “usage rights” and their associated “digital works” exist independently. (See Dkt. No. 304, 11/25/2014 Goodrich Decl. at ¶¶ 46-51).

Defendants submit an article authored by one of the named inventors, Mark Stefik, stating as follows under the heading “Attached Usage Rights”:

We start with an analogy. When we go to a store to buy a shirt, there are various tags attached to it. One kind of tag is a price tag. If we want to buy the shirt, we must pay the amount on the tag. Another tag gives cleaning instructions: for example, wash by hand in cold water or dry clean only. Still another tag might say something about the style of the shirt or the history of the shirt company.

This is roughly the idea of usage rights on digital works. Digital works come with tags on them. * * * [*T*he tags are not removable.

(Dkt. No. 331, Ex. 5, Mark Stefik, *Letting Loose the Light: Igniting Commerce in Electronic Publication* 14 (1996) (emphasis added).)

Defendants also cite prosecution history of a European Patent Office application that names Mark Stefik as inventor:

A skilled person readily knows several ways of how to associate two digital objects (such as a rights object and a content object), for instance by physically attaching both, inserting a unique identifier for one object in the header of the other object, specifying in the header of the first object address or access information for the second object, providing a look-up table linking both objects

together, or storing both objects in different files that have the same unique file name.

(Dkt. No. 353, Ex. 2, 10/25/2004 Response, at 3.)

Further, Plaintiff submits that in *Smartflash LLC v. Apple Inc., et al.*, No. 6:13-cv-447 (E.D. Tex.), Defendant Apple Inc. has described another patent (which also names Mark Stefik as an inventor) as teaching separately stored content and rights information. (Dkt. No. 345, Ex. AB, at 3.)

Finally, Plaintiff has noted that the prosecuting attorney for the '160 Patent has testified that the changes made to the specification of the '160 Patent were intended to remove all possible doubt that "attached" and "associated" mean the same thing. (Dkt. No. 345, Ex. AC, Kaufman Dep. at 171:25-172:14 ("The Stefik application used the terms 'attached' and 'associated' interchangeably. And as I've stated earlier, we think the correct construction was that neither of those terms requires a direct physical coupling. But to a layman that caused some confusion, and we decided to make that consistent. . . . That was our intent.")).

Such prosecutor testimony, however, is of minimal if any weight during claim construction proceedings. *See Markman*, 52 F.3d at 983 ("[T]he testimony of Markman and his patent attorney on the proper construction of the claims is entitled to no deference."), *aff'd*, 517 U.S. 370 (1996); *see also Howmedica Osteonics Corp. v. Wright Med. Tech., Inc.*, 540 F.3d 1337, 1346 (Fed. Cir. 2008) (finding that a letter from prosecuting attorney to inventor, reporting the results of an examiner interview, was "of no value" because the letter was not part of the prosecution history and did not "help educate the court regarding the field of the invention . . . [or] help the court determine what a person of ordinary skill in the art would understand claim terms to mean") (quoting *Phillips*, 415 F.3d at 1319).

On balance, nothing in the prosecution history or the extrinsic evidence cited by the parties warrants departing from the lexicography and descriptions of the invention in the specification, set forth above, which support Defendants’ proposal of requiring that usage rights are “attached” to digital works.

Defendants’ proposal of *permanent* attachment, however, lacks sufficient support in the intrinsic evidence. Instead, for example, the specification contemplates “mak[ing] a copy of the digital work in a place outside of the protection of usage rights.” ‘859 Patent at 35:64-36:3.

(3) Construction

Based on the foregoing analysis of Defendants’ proposals of “language” and of “permanently attached,” the Court hereby construes **“rights,” “usage rights,” and “usage rights information”** to mean **“indications that are attached, or treated as attached, to [a digital work / digital content / content / a digital document] and that indicate the manner in which the [digital work / digital content / content / digital document] may be used or distributed as well as any conditions on which use or distribution is premised.”**

G. “usage rights” (‘160 Patent)

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“an indication of the manner in which a [digital work / digital content / content / a digital document] may be used or distributed as well as any conditions on which use or distribution is premised” ⁴	“an indication of the manner of use by which a digital work may be used or distributed, as well as any conditions on which manner of use is premised”

(Dkt. No. 304, at 9; Dkt. No. 331, at 11; Dkt. No. 366, Ex. B, at 15.) Defendants submit that this disputed term appears in Claims 1, 2, 3, 9, and 10 of the ‘160 Patent. (Dkt. No. 331, at 11.)

⁴ Plaintiff previously proposed: “Same as in Stefik Patents, i.e., ‘an indication of the manner in which a [digital work / digital content / content / a digital document] may be used or distributed as well as any conditions on which use or distribution is premised.’” (Dkt. No. 292-1, at 12.)

(1) The Parties' Positions

Plaintiff argues that “[i]n their proposal, Defendants repeat features already in the claims as well as optional features (e.g., conditions) in a way that differs in form but not in substance from their proposal for the other patents. Defendants’ form over substance approach would only confuse the jury.” (Dkt. No. 304, at 10.)

Defendants respond that the ‘160 Patent is different than the other Stefik Patents because “the ‘160 patent undeniably changed the glossary definition and corresponding teachings of the specification.” (Dkt. No. 331, at 12.) In particular, Defendants urge that “[b]ecause the ‘160 specification eliminated all references to, and requirement of, ‘attachment,’ Defendants propose that the Court adopt a construction of ‘usage rights’ for the ‘160 patent (and only that patent) that does not contain the ‘permanently attached’ requirement.”

Plaintiff replies that having a separate construction for the ‘160 patent “would result in unnecessary jury confusion.” (Dkt. No. 345, at 17.)

Generally, “we presume, unless otherwise compelled, that the same claim term in the same patent or related patents carries the same construed meaning.” *See Aventis*, 715 F.3d at 1380 (quoting *Omega Eng’g*, 334 F.3d at 1334).

(2) Analysis

Here, however, the ‘160 Patent does not contain the same definitive statements cited above as to the other Stefik Patents. The Court therefore does not include an “attached” requirement in the construction of “usage rights” in the ‘160 Patent.

The Court accordingly hereby construes **“usage rights”** in the ‘160 Patent to mean **“an indication of the manner of use by which a digital work may be used or distributed, as well as any conditions on which manner of use is premised.”**

H. “digital work”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
‘859 Patent, ‘576 Patent, ‘072 Patent, ‘956 Patent, and ‘007 Patent: No construction ‘160 Patent: No construction necessary. Alternatively: “any work that has been reduced to a digital representation”	‘859 Patent, ‘576 Patent, ‘072 Patent, ‘956 Patent, and ‘007 Patent: “Any encapsulated digital information. Such digital information may represent music, a magazine or book, or a multimedia composition. Usage rights and fees are attached to the digital work.” ‘160 Patent: “Digital content with any associated usage rights. Such digital content may represent music, a magazine or book, or a multimedia composition.”

(Dkt. No. 304, at 9; Dkt. No. 331, at 12-13; Dkt. No. 366, Ex. B, at 15.) The parties submit that this disputed term appears in Claims 1, 2, 3, 6, 9, and 10 of the ‘160 Patent. (Dkt. No. 292-1, at 2; Dkt. No. 331, at 13.)

(1) The Parties’ Positions

Plaintiff argues “[t]he claims already recite the contents and structure of the claimed digital work, so no separate construction of that term is necessary.” (Dkt. No. 304, at 9.)

Defendants argue: “Although the claims of the ‘859, ‘576, ‘956, ‘007, and ‘072 patents do not explicitly recite the term ‘digital work,’ the glossary defines other terms recited in the claims, such as ‘content,’ ‘description structure,’ ‘requester mode,’ ‘server mode,’ and ‘usage rights,’ in terms of their relationship to a ‘digital work.’ So the jury will need to understand this term.” (Dkt. No. 331, at 12-13.)

As to the ‘160 Patent, Defendants argue: “[Plaintiff] asks the Court to ignore the glossary definition and offers an alternative construction cherry-picked from the specification. Again, the Court should adopt the express definitions provided in the glossary.” (Dkt. No. 331, at 13.)

Plaintiff replies that the term “digital work” appears in the claims of only the ‘160 Patent. (Dkt. No. 345, at 6.)

(2) Analysis

The specification sets forth two definitions for “digital work.” First, the specification states: “Herein the terms ‘digital work,’ ‘work’ and ‘content’ refer to any work that has been reduced to a digital representation.” ‘859 Patent at 5:64-66.

Second, the “Glossary” section of the specification defines “digital work” and “composite digital work” as requiring that usage rights are “attached”:

Digital Work (Work):

Any encapsulated digital information. Such digital information may represent music, a magazine or book, or a multimedia composition. *Usage rights and fees are attached to the digital work.*

Id. at 50:11-12.

Composite Digital Work:

A digital work comprised of distinguishable parts. Each of the distinguishable parts is itself a digital work *which have usage rights attached.*

Id. at 49:49-51 (emphasis added). Also of note, the specification discloses that a “ticket” can be used for “one-time usage rights,” and “[t]ickets are digital works.” ‘859 Patent at 22:20; ‘160 Patent at 21:42.

But although the specifications present these definitions, the term “digital work” appears in the claims of only the ‘160 Patent. The meaning of “digital work” is addressed by the claim language itself (emphasis added):

1. A computer readable medium having embedded thereon a digital work adapted to be distributed within a system for controlling use of digital works, *said digital work comprising:*
 - a digital content portion that is renderable by a rendering device;

a usage rights portion associated with said digital content portion and comprising one or more computer readable instructions configured to permit or prohibit said rendering device to render said digital content portion, said usage rights portion being expressed as statements from a usage rights language having a grammar defining a valid sequence of symbols, and specifying a manner of use relating to one or more purposes for which the digital work can be used by an authorized party; and

a description structure comprising a plurality of description blocks, each of said description blocks comprising address information for at least one part of said digital work, and a usage rights part for associating one or more usage rights portions.

The Court therefore hereby expressly rejects the parties’ proposed constructions. No further construction is necessary. *See U.S. Surgical Corp. v. Ethicon, Inc.*, 103 F.3d 1554, 1568 (Fed. Cir. 1997) (“Claim construction is a matter of resolution of disputed meanings and technical scope, to clarify and when necessary to explain what the patentee covered by the claims, for use in the determination of infringement. It is not an obligatory exercise in redundancy.”); *see also O2 Micro Int’l Ltd. v. Beyond Innovation Tech. Co.*, 521 F.3d 1351, 1362 (Fed. Cir. 2008) (“[D]istrict courts are not (and should not be) required to construe every limitation present in a patent’s asserted claims.”); *Finjan, Inc. v. Secure Computing Corp.*, 626 F.3d 1197, 1207 (Fed. Cir. 2010) (“Unlike *O2 Micro*, where the court failed to resolve the parties’ quarrel, the district court rejected Defendants’ construction.”).

The Court accordingly hereby construes “**digital work**” to have its **plain meaning**.

I. “digital document” and “document”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“any work that has been reduced to a digital representation”	“a type of digital work that is written in or viewable as text, for example a book, magazine article, or a message”

(Dkt. No. 304, at 10; Dkt. No. 331, at 15.) The parties submit that this disputed term appears in Claims 1, 8, 10, and 16 of the ‘072 Patent. (Dkt. No. 292-1, at 1; Dkt. No. 331, at 15.)

(1) The Parties' Positions

Plaintiff argues claim differentiation as to Claims 3, 11, and 19 of the '072 Patent. (Dkt. No. 304, at 10.) Plaintiff also notes that the specification refers to "documents" as including "movies" as well as content that can be "execut[ed]." (*Id.*)

Defendants respond: "the specification defines the terms 'digital work' and 'digital content' broadly, to encompass not only documents (such as magazines and books), but also video, audio and other material. In using the term 'digital document,' the patentees intended to refer not to any form of digital content or digital work, but only to documents." (Dkt. No. 331, at 16.)

Plaintiff replies that "Defendants fail to rebut [Plaintiff's] showing that the Stefik patents contemplated using a document repository for movies, software, and other forms of content, not just written works." (Dkt. No. 345, at 7.)

(2) Analysis

Claims 1 and 3 of the '072 Patent are representative and recite (emphasis added):

1. A method for securely rendering *digital documents*, comprising:
 - retrieving, by a document platform, a *digital document* and at least one usage right associated with the *digital document* from a document repository, the at least one usage right specifying a manner of use indicating the manner in which the *digital document* can be rendered;
 - storing the *digital document* and the at least one usage right in separate files in the document platform;
 - determining, by the document platform, whether the *digital document* may be rendered based on the at least one usage right; and
 - if the at least one usage right allows the *digital document* to be rendered on the document platform, rendering the *digital document* by the document platform.

* * *

3. The method as recited in claim 1, wherein at least a portion of the *digital document* is a software program.

Because dependent Claim 3 limits the “digital document” to being “a software program,” which presumably is much broader than “text,” Claim 3 weighs against Defendants’ proposal that the term “digital document” is limited to text. *See Phillips*, 415 F.3d at 1315 (“[T]he presence of a dependent claim that adds a particular limitation gives rise to a presumption that the limitation in question is not present in the independent claim.”); *see also Liebel-Flarsheim Co. v. Medrad, Inc.*, 358 F.3d 898, 910 (Fed. Cir. 2004) (“[W]here the limitation that is sought to be ‘read into’ an independent claim already appears in a dependent claim, the doctrine of claim differentiation is at its strongest.”); *Wenger Mfg., Inc. v. Coating Mach. Sys., Inc.*, 239 F.3d 1225, 1233 (Fed. Cir. 2001) (“Claim differentiation, while often argued to be controlling when it does not apply, is clearly applicable when there is a dispute over whether a limitation found in a dependent claim should be read into an independent claim, and that limitation is the only meaningful difference between the two claims.”).

Turning to the specification, on one hand the specification discloses that a “Document-Descr[iption]” can be a “string containing various identifying information about a document,” “such as a publisher name, author name, ISBN number, and so on.” ‘859 Patent at 10:17-20.

On the other hand, the specification uses the term “document” in contexts other than text. *See* ‘072 Patent at 14:43-54 (referring to security levels of “document repositories” shortly after noting the importance of security for “some digital works such as a digital copy of a first run movie”) & 26:25-27 (“document playback platform (e.g., for executing or viewing”).

On balance, nothing in the intrinsic evidence warrants reading the seemingly generic term “document” as being limited to text. *See Thorner v. Sony Computer Entm’t Am. LLC*, 669 F.3d 1362, 1367 (Fed. Cir. 2012) (“The patentee is free to choose a broad term and expect to obtain

the full scope of its plain and ordinary meaning unless the patentee explicitly redefines the term or disavows its full scope.”).

Defendants’ proposed construction is therefore hereby expressly rejected. In particular, Defendants have failed to demonstrate that the disputed terms cannot encompass content beyond merely text, such as audio, video, or software.

No further construction is necessary. *See U.S. Surgical*, 103 F.3d at 1568; *see also O2 Micro*, 521 F.3d at 1362; *Finjan*, 626 F.3d at 1207.

The Court accordingly hereby construes **“digital document”** and **“document”** to have their **plain meaning**.

J. “requester mode of operation” and “server mode of operation”

“requester mode of operation”	
Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
No construction necessary in view of language already in the claims	“a mode of a repository where it is requesting access to a digital work”
“server mode of operation”	
Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
No construction necessary in view of language already in the claims	“a mode of a repository where it is processing an incoming request to access the digital work”

(Dkt. No. 304, at 11; Dkt. No. 331, at 14.) The parties submit that these disputed terms appear in Claims 1 and 58 of the ‘859 Patent. (Dkt. No. 292-1, at 9; Dkt. No. 331, at 14.)

(1) The Parties’ Positions

Plaintiff argues that Defendants “wish to insert their construction for ‘digital work’ into claims not already reciting that term.” (Dkt. No. 304, at 11.) Further, Plaintiff submits, “[t]he

asserted claims of the ‘859 patent already recite the features of both the ‘server mode of operation’ and the ‘requester mode of operation’ with greater detail than is supplied by Defendants’ proposed constructions.” (*Id.*)

Defendants respond that Plaintiff “offers no legitimate reason to reject or modify the decisions of the patentees in creating these glossary definitions.” (Dkt. No. 331, at 14.)

Plaintiff replies: “The claims already positively recite that the requester and server modes of operation apply to ‘digital content.’ Thus, Defendants’ proposed construction would add confusion by having the same claim use two different terms, ‘digital content’ and ‘digital work,’ to reference the same thing.” (Dkt. No. 345, at 7.)

At the February 6, 2015 hearing, the parties did not address these disputed terms.

(2) Analysis

The “Glossary” section of the specification states:

Requester Mode:

A mode of repository where it is requesting access to a digital work.

* * *

Server Mode:

A mode of a repository where it is processing an incoming request to access a digital work.

‘859 Patent at 50:53-62.

Claims 1 and 58 of the ‘859 Patent recite:

1. A rendering system adapted for use in a distributed system for managing use of content, said rendering system being operative to rendering [*sic*] content in accordance with usage rights associated with the content, said rendering system comprising:

 a rendering device configured to render the content; and
 a distributed repository coupled to said rendering device and including a *requester mode of operation* and *server mode of operation*,

wherein the *server mode of operation* is operative to enforce usage rights associated with the content and permit the rendering device to render the content in accordance with a manner of use specified by the usage rights,

the *requester mode of operation* is operative to request access to content from another distributed repository, and

said distributed repository is operative to receive a request to render the content and permit the content to be rendered only if a manner of use specified in the request corresponds to a manner of use specified in the usage rights.

* * *

58. A computer readable medium including one or more computer readable instructions embedded therein for use in a distributed system for managing use of content, and operative to render content in accordance with usage rights associated with the content, said computer readable instructions configured to cause one or more computer processors to perform the steps of:

configuring a rendering device to render the content;

configuring a distributed repository coupled to said rendering device to include a *requester mode of operation* and *server mode of operation*;

enforcing usage rights associated with the content and permitting the rendering device to render the content in accordance with a manner of use specified by the usage rights, when in the *server mode of operation*;

requesting access to content from another distributed repository, when in the *requester mode of operation*; and

receiving by said distributed repository a request to render the content and permitting the content to be rendered only if a manner of use specified in the request corresponds to a manner of use specified in the usage rights.

On balance, the surrounding claim language addresses the meaning of the disputed terms such that no construction is necessary. *See U.S. Surgical*, 103 F.3d at 1568; *see also O2 Micro*, 521 F.3d at 1362. Although the specification contains a “Glossary” section that defines “requester mode” and “server mode,” the explicit claim language overrides the specification in this regard. *See Tempo Lighting, Inc. v. Tivoli, LLC*, 742 F.3d 973, 977 (Fed. Cir. 2014) (“In claim construction, this court gives primacy to the language of the claims, followed by the specification.”).

The Court accordingly hereby construes “**requester mode of operation**” and “**server mode of operation**” to have their **plain meaning**.

K. “manner of use”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“a way in which [a digital work / digital content / content / a digital document] may be used”	“a defined way of using or distributing a digital work (for example, PLAY, COPY, or PRINT), as distinct from conditions which must be satisfied before that way of using or distributing the digital work is allowed”

(Dkt. No. 304, at 12 (square brackets Plaintiff’s); Dkt. No. 331, at 16.) The parties submit that this disputed term appears in Claims 1, 19, 20, 58, 75, and 76 of the ‘859 Patent, Claims 1 and 3 of the ‘160 Patent, and Claims 1 and 10 of the ‘072 Patent. (Dkt. No. 292-1, at 3; Dkt. No. 331, at 16.)

(1) The Parties’ Positions

Plaintiff argues that “[t]he specification teaches that conditions can be included in a ‘manner of use’—not that they are distinct.” (Dkt. No. 304, at 12.) “Further,” Plaintiff argues, “Defendants’ construction is improper because it seeks to import Defendants’ construction of the term ‘digital work’ into claims that do not include this term.” (*Id.*)

Defendants argue that “Defendants’ proposed construction preserves th[e] distinction between the two categories of information that can be conveyed by usage rights,” namely “(1) the manner in which a digital work may be used or distributed and (2) the conditions, if any, on which use or distribution is premised.” (Dkt. No. 331, at 16.)

Plaintiff replies that Defendants’ proposal “is made out of whole cloth” and is inconsistent with disclosures showing that “if and when they are defined, conditions are a part of the ‘manner of use.’” (Dkt. No. 345, at 8.)

In sur-reply, Defendants reiterate that “[t]he patents describe ‘manner of use’ (e.g., play, copy, or print) as separate and distinct from conditions that must be satisfied prior to use (e.g.,

copy count, fees, or time), not as one common category as [Plaintiff] argues.” (Dkt. No. 353, at 5 (discussing Figures 14 and 15 of the Stefik Patents).) Finally, Defendants cite Claim 23 of the ‘859 Patent. (Dkt. No. 353, at 5.)

At the February 6, 2015 hearing, Plaintiff argued that the line between manners of use and conditions of use is blurry, so to speak, because the patents-in-suit do not clearly distinguish them from one another. Defendants responded that the distinction is clear and should be set forth in the Court’s construction because Plaintiff may later attempt to argue that a manner of use can be expressed as merely a condition.

(2) Analysis

Claim 23 of the ‘859 Patent recites:

23. A rendering system as recited in claim 1, wherein the usage rights include at least one *condition* that must be satisfied to exercise the *manner of use*, and wherein the system further comprises means for communicating with an authorization repository for authorizing a condition.

This distinct, separate recital of a “condition” and a “manner of use” weighs against Plaintiff’s argument that a “manner of use” can include a condition. *See Phillips*, 415 F.3d at 1314 (“Other claims of the patent in question, both asserted and unasserted, can also be valuable sources of enlightenment as to the meaning of a claim term.”).

The specification also discloses that a “grammar element” can provide a condition. *Id.* at 20:15-16 (“Grammar element 1510 . . . provides a condition . . .”), 20:34-36 (“Grammar element 1511 . . . provides a condition . . .”) & 20:47-50 (“Grammar element 1512 . . . provides for specification of time conditions . . .”). Such disclosures do not demonstrate, however, that a condition is part of a manner of use.

Finally, Defendants have also cited the prosecution of related United States Patent No. 6,708,157, which claims priority to the same application as the Stefik Patents here in suit.

See Ormco Corp. v. Align Tech., Inc., 498 F.3d 1307, 1314-15 (Fed. Cir. 2007) (prosecution history of parent application having specification with “the same content” found to be “relevant in construing the claims” of related patents); *see also Chimie v. PPG Indus., Inc.*, 402 F.3d 1371, 1384 (Fed. Cir. 2005) (“The purpose of consulting the prosecution history in construing a claim is to exclude any interpretation that was disclaimed during prosecution.”) (citation and internal quotation marks omitted). The applicability of this prosecution history is disputed. (*See* Dkt. No. 345, at 8). Even if considered, this prosecution has no impact on the Court’s analysis.

On balance, the intrinsic evidence, especially above-quoted Claim 23 of the ‘859 Patent, is consistent with Defendants’ proposal that a “manner of use” is distinct from a “condition.”

The Court accordingly hereby construes “**manner of use**” to mean “**a way in which [a digital work / digital content / content / a digital document] may be used, as contrasted with a condition that must be satisfied before such use is allowed.**”

L. “render” and “rendering”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“converting into an ephemeral, transitory, or non[-]digital form, such as for playing digital movies, playing digital music, playing a video game, running a computer program, or displaying a document on a display”	“play, print, display, or execute [a digital work] [digital content / content] [a digital document]”

(Dkt. No. 304, at 12; Dkt. No. 331, at 17 (square brackets Defendants’).) The parties submit that these disputed terms appear in Claims 1, 13, 19, 20, 21, 24, 58, 69, and 71 of the ‘859 Patent, Claim 1 of the ‘160 Patent, Claims 1 and 18 of the ‘576 Patent, Claims 1 and 10 of the ‘072 Patent, Claims 1, 7, and 13 of the ‘956 Patent, and Claims 1, 3, 6, 8, 11, and 13 of the ‘007 Patent. (Dkt. No. 292-1, at 7; Dkt. No. 331, at 17.)

(1) The Parties' Positions

Plaintiff submits that its proposal “is taken from the description of rendering in the patents” and “provides examples to guide the jury in understanding that rendering includes, for example, conversion for playing digital movies and displaying a document on a display.” (Dkt. No. 304, at 12-13.) Plaintiff argues that “Defendants’ construction is unduly narrow because it only accounts for specific forms of conversion (i.e. ‘play, print, display, or execute’), while the specification allows for conversion into any ephemeral, transitory, or non-digital form.” (*Id.*, at 13.)

Defendants respond that “[a]lthough the specifications do not explicitly define ‘render,’ their description of types of rendering systems supports Defendants’ construction.” (Dkt. No. 331, at 18.) Further, Defendants argue, “[t]he specification never compares rendering to ‘converting.’ In fact, the words ‘convert’ and ‘converting’ never appear in the patents.” (*Id.*) Defendants conclude: “[Plaintiff’s] construction contains terms that may be difficult for a jury to apply and it is so open-ended that it risks being read to encompass almost any transformation of data, including acts like decryption that persons of skill in the art would not consider to be ‘rendering.’” (*Id.*)

Plaintiff replies that “while the specification defines ‘render’ as any conversion into any ‘ephemeral, transitory, or non-digital’ form, Defendants propose a narrow construction that accounts only for specific types of conversion, i.e., ‘play, print, display, or execute.’” (Dkt. No. 345, at 9.)

(2) Analysis

The “Glossary” section of the specification states:

Rendering System:

The combination of a rendering repository and a rendering device. Examples of rendering systems include printing systems, displaying systems, general purpose computer systems, video systems or audio systems.

Id. at 50:42-46. Other portions of the specification, however, address the meaning of

“rendering” in greater detail:

Rendering Systems

A *rendering* system is generally defined as a system comprising a repository and a *rendering* device which can *render* a digital work into its desired form.

Examples of a *rendering* system may be a computer system, a digital audio system, or a printer.

* * *

Grammar element 1504 “*Render-Code:=*[Play:{Player: Player-ID} | Print:{Printer:Printer-ID}]” lists a category of rights all involving *the making of ephemeral, transitory, or non-digital copies of the digital work*. After use the copies are erased.

Play[:] A process of *rendering* or performing a digital work on some processor. This includes such things as playing digital movies, playing digital music, playing a video game, running a computer program, or displaying a document on a display.

Print[:] To *render* the work in a medium that is not further protected by usage rights, such as printing on paper.

‘859 Patent at 7:50-54 & 18:44-53 (emphasis added). “Printing,” which is disclosed as an example of a type of rendering, is further disclosed as “mak[ing] a copy of the digital work”:

The Print Transaction

A Print transaction is a request to obtain the contents of a work for the purpose of *rendering* them on a “printer.” We use the term “printer” to include the common case of writing with ink on paper. However, the key aspect of “printing” in our use of the term is that it *makes a copy of the digital work* in a place outside of the protection of usage rights.

Id. at 35:64-36:3.

Plaintiff’s proposal of “ephemeral, transitory, or non[-]digital form” is rejected as vague and as tending to confuse rather than clarify the scope of the claims.

Defendants’ proposal, however, is too narrow because it limits the disputed term to four specific types of operation.

Based on the above-quoted disclosures, particularly the disclosure of “rendering or performing,” construction of this disputed term as “presenting,” together with the above-quoted examples set forth in the specification, will be more accurate and of greater help to the finder of fact than either Plaintiff’s or Defendants’ proposal.

The Court accordingly hereby construes **“render”** and **“rendering”** to mean **“present[ing] a digital work, such as by playing a digital movie, playing digital music, playing a video game, running a computer program, displaying a document on a display, or printing on paper.”**

M. “authorization object”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“digital information that must be possessed to gain access to digital content”	“a digital work that can be moved between repositories and, when specified by a usage right attached to another digital work, must be obtained to exercise the usage right”

(Dkt. No. 304, at 13; Dkt. No. 331, at 18.) The parties submit that this disputed term appears in Claims 1 and 18 of the ‘576 Patent, Claims 4, 10, and 16 of the ‘956 Patent, and Claims 3, 8, and 13 of the ‘007 Patent. (Dkt. No. 292-1, at 9; Dkt. No. 331, at 18.)

(1) The Parties’ Positions

Plaintiff argues that its proposed construction “clarifies the role of the ‘authorization object’ set forth in the claims in that it must be possessed to gain access to digital content.” (Dkt. No. 304, at 13.) Plaintiff further argues that Defendants’ proposal “could apply to any ‘digital

work’ that can be moved between repositories” and “imports the term ‘digital work’ into claims that do not use this term.” (*Id.*, at 14.)

Defendants respond that “[t]he specification explains with clarity that an authorization object is a type of ‘digital work,’” and “[b]y stating that authorization objects are only required ‘when specified by a usage right attached to another digital work,’ [Defendants’] construction explains when authorization objects are necessary and that they are separate from the digital content.” (Dkt. No. 331, at 18-19.)

Plaintiff replies, in full: “Defendants’ construction is improper because it attempts to import Defendants’ construction for the term ‘digital work’ into claims that do not use this term. Defendants’ construction is also confusing because it defines the ‘authorization object’ as a ‘digital work,’ even though Defendants concede that ‘authorization objects’ are ‘separate from the digital content.’” (Dkt. No. 345, at 7 (citing Dkt. No. 331, at 19).)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

Claim 4 of the ‘956 Patent, for example, discloses:

4. The method of claim 1, wherein the receiving the digital content comprises:
 - requesting an *authorization object* for the at least one recipient computing device to make the digital content available for use, the *authorization object* being required to receive the digital content and to use the digital content; and
 - receiving the *authorization object* if it is determined that the request for the *authorization object* should be granted.

The specification discloses:

Communication with an authorization repository 202 may occur when a digital work being accessed has a condition requiring an authorization. Conceptually, *an authorization is a digital certificate such that possession of the certificate is required to gain access to the digital work. An authorization is itself a digital work that can be moved between repositories and subjected to fees and usage rights conditions. An authorization may be required by both repositories involved in an access to a digital work.*

'859 Patent at 7:18-26 (emphasis added).

In a transaction involving a repository and a document server, *some usage rights may require that the repository have a particular authorization*, that the server have some authorization, or that both repositories have (possibly different) authorizations. *Authorizations themselves are digital works* (hereinafter referred to as an authorization object) that can be moved between repositories in the same manner as other digital works. Their copying and transferring is subject to the same rights and fees as other digital works. A repository is said to have an authorization if that authorization object is contained within the repository.

Id. at 21:57-67 (emphasis added); *id.* at 40:50-51 (“an authorization object (a digital work in a file of a standard format”).

These consistent, explicit disclosures of authorization objects as being digital works should be given effect in the Court’s construction. *See Nystrom v. TREX Co., Inc.*, 424 F.3d 1136, 1144-45 (Fed. Cir. 2005) (construing term “board” to mean “wood cut from a log” in light of the patentee’s consistent usage of the term; noting that patentee “is not entitled to a claim construction divorced from the context of the written description and prosecution history”); *see also Am. Piledriving Equip., Inc. v. Geoquip, Inc.*, 637 F.3d 1324, 1333 (Fed. Cir. 2011) (“[T]he consistent reference throughout the specification to the ‘eccentric weight portion’ as structure extending from the face of the gear makes it apparent that it relates to the invention as a whole, not just the preferred embodiment.”). Further, the above-quoted reference to authorization for usage rights demonstrates that an authorization object may be required for usage rights other than merely accessing.

Nonetheless, Defendants have failed to adequately justify their proposal that an authorization object is necessarily separate from the digital work for which authorization is required. Indeed, the “Glossary” section of the specification states that a digital work can itself include other digital works:

Composite Digital Work:

A digital work comprised of distinguishable parts. Each of the distinguishable parts is itself a digital work *which have usage rights attached*.

Id. at 49:49-51 (emphasis added).

The Court accordingly hereby construes **“authorization object”** to mean **“a digital work that can be moved between repositories and that must be possessed in order to exercise a usage right.”**

N. “identification certificate” and “digital certificate”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“a signed digital message that attests to the identity of the possessor”	“a signed digital message that attests to the identity of the possessor. Digital certificates are encrypted in the private key of a well-known master repository.”

(Dkt. No. 304, at 14 (Plaintiff addressed only “identification certificate”); Dkt. No. 331, at 14-15.) Defendants submit that “identification certificate” appears in Claims 24 and 81 of the ‘859 Patent, Claims 5, 11, and 17 of the ‘956 Patent, and Claims 4, 5, 9, 10, 14, and 15 of the ‘007 Patent. (Dkt. No. 331, at 14.) Defendants submit that “digital certificate” appears in Claims 1, 15, 21, 24, 58, 71, and 81 of the ‘859 Patent, Claims 1 and 18 of the ‘576 Patent, Claims 1 and 10 of the ‘072 Patent, Claims 1, 7, and 13 of the ‘956 Patent, and Claims 1, 6, and 11 of the ‘007 Patent. (Dkt. No. 331, at 14-15.)

(1) The Parties’ Positions

The parties agree that “identification certificate” and “digital certificate” are used synonymously in the Stefik Patents. (*See* Dkt. No. 331, at 5 n.4.)

Plaintiff submits that this term is defined in the specification's glossary and that Defendants "seek to import an additional requirement from the second sentence of the glossary by removing the word 'typically.'" (Dkt. No. 304, at 14.)

Defendants respond that the specification, as well as the testimony of Plaintiff's expert, Dr. Goodrich, in IPR proceedings, emphasize "the 'extremely important' requirement of tamper resistance, which is provided by encrypting the digital certificate in the private key of a master repository." (Dkt. No. 331, at 15.)

Plaintiff replies that by ignoring the word "typically," Defendants improperly limit the construction to a preferred embodiment. (Dkt. No. 345, at 9.) Plaintiff also submits that in an IPR proceeding, Defendant Apple Inc. has proposed a construction of "identification certificate" that *omits* the second sentence from the "Glossary" definition. (*Id.*, Ex. AE, 12/22/2014 Petition for *Inter Partes* Review of U.S. Patent No. 8,370,956 (IPR2015-00446), at 14.)

(2) Analysis

The "Glossary" section of the specification states:

Identification (Digital) Certificate:

A signed digital message that attests to the identity of the possessor. *Typically*, digital certificates are encrypted in the private key of a well-known master repository.

'859 Patent at 50:16-19 (emphasis added).

Defendants submit that the specification "make[s] clear that identification certificates (also referred to as 'digital certificates') are generated by a 'master repository' and that '[c]ommunication with a master repository . . . occurs in connection with obtaining an identification certificate.'" (Dkt. No. 331, at 15 (citing '859 Patent at 7:32-33 & 12:34-50).)

Defendants have also cited disclosure regarding an “install transaction” that ends with an error if a master repository is not recognized. *See id.* at 41:28-42:6.

On balance, the portions of the specification cited by Defendants do not override the patentee’s use of the word “typically” in the above-quoted “Glossary” definition. *See Abbott Labs.*, 334 F.3d at 1354 (“patentee’s lexicography must, of course, appear with reasonable clarity, deliberateness, and precision”) (citation and internal quotation marks omitted); *cf. MasterObjects, Inc. v. Yahoo!, Inc.*, No. C 11-02539-JSW, 2013 WL 6185475, at *7 (N.D. Cal. Nov. 26, 2013) (“The patentee uses the glossary to describe one particular embodiment Yahoo! seeks to import a specific limitation from a glossary which is expressly limited to a preferred embodiment. This is not permitted.”).

The Court accordingly hereby construes **“identification certificate”** and **“digital certificate”** to mean **“a signed digital message that attests to the identity of the possessor.”**

O. “nonce” and “random registration identifier”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“random or variable information generated to establish a cryptographic connection” ⁵	“nonce”: “random and variable information used only once to establish a cryptographic connection” “random registration identifier”: Same as “nonce”; Alternatively, indefinite.

(Dkt. No. 304, at 15; Dkt. No. 331, at 20.) The parties submit that “nonce” appears in Claims 6, 12, and 18 of the ‘956 Patent and Claims 5, 10, and 15 of the ‘007 Patent. (Dkt. No. 292-1,

⁵ Plaintiff previously proposed: “nonce” means “Random or variable information for determining whether a system can correctly perform a cryptographic operation”; and “random registration identifier” means “Random or variable information for identifying a communication session.” (Dkt. No. 292-1, at 3-4.)

at 3-4; Dkt. No. 331, at 20.) The parties submit that “random registration identifier” appears in Claims 5, 11, and 17 of the ‘956 Patent and Claims 4, 9, and 14 of the ‘007 Patent. (Dkt. No. 292-1, at 4; Dkt. No. 331, at 20.)

(1) The Parties’ Positions

Plaintiff argues that the specification describes a “nonce” as being random *or* variable. (Dkt. No. 304, at 15.) Also, Plaintiff argues, “[b]oth the ‘random registration identifier’ and the ‘nonce’ are used several times in different ways during the registration procedure for repositories” (*Id.*; *see id.* at 15-16 (citing ‘956 Patent at Figure 16).) Plaintiff further argues:

Any random number of finite length used as part of such a procedure would with some probability repeat a previously generated random number. That is why the specification does not rely on the random registration identifier alone to secure the message: the repository using the registration identifier also uses the time and the names of the repositories to verify the session.

(*Id.*, at 16 (citing ‘956 Patent at 27:37-45 & 27:52-57).)

Defendants respond that “[i]f they [(nonces and random registration identifiers)] were used more than once, a counterparty could fool a repository into creating a connection by using an unencrypted nonce or identifier from an earlier session.” (Dkt. No. 331, at 20.) Defendants also submit that “[b]oth parties to a transaction do use them, but that is a single use of the nonce.” (*Id.*) Finally, Defendants urge that “although it is theoretically possible that a number could be randomly generated more than once, that remote possibility is unlikely to confuse a jury.” (*Id.*)

Plaintiff replies that “Defendants thus admit that their ‘only once’ limitation is scientifically incorrect” (Dkt. No. 345, at 9 (citing Dkt. No. 331, at 20)), and Defendants’ proposal of “random *and* variable” “would exclude the two examples provided in the

specification (time and temperature), both of which are ‘variable only’ nonces.” (Dkt. No. 345, at 9-10 (citing Dkt. No. 304 at 15).)

(2) Analysis

The specification discloses a “random registration identifier” as being unique to a “session”:

A registration message is comprised of an identifier of a master repository, the identification certificate for the repository-1 and an encrypted *random registration identifier*. * * * The registration identifier is a number generated by the repository for this registration. The registration identifier is *unique to the session* and is encrypted in repository-1’s private key. The registration identifier is used to improve security of authentication by detecting certain kinds of communications[-]based attacks.

‘859 Patent at 26:51-66 (emphasis added). The specification also discloses a “nonce” as follows:

When a sending repository transmits a message to a receiving repository, the sending repository encrypts all of its data using the public writing key of the receiving repository. The sending repository includes its name, the name of the receiving repository, a *session identifier* such as a *nonce* (described below), and a message counter in each message. In this way, the communication can only be read (to a high probability) by the receiving repository, which holds the private checking key for decryption. The auxiliary data is used to guard against various replay attacks to security. If messages ever arrive with the wrong counter or an old *nonce*, the repositories can assume that someone is interfering with communication and the transaction [is] terminated.

* * *

A nonce is a generated message based on some random and variable information (e.g., the time or the temperature.) The nonce is used to check whether repository-1 can actually exhibit correct encrypting of a message using the private keys it claims to have, on a message that it has never seen before.

‘859 Patent at 26:14-26 & 27:45-50 (emphasis added).

This explicit definition of a “nonce” as being based on random *and* variable information should be given effect in the Court’s construction. *See Intellicall*, 952 F.2d at 1388; *see also C.R. Bard*, 388 F.3d at 862; *Abbott Labs.*, 334 F.3d at 1354. Also, the above-quoted disclosures

explain that a “nonce” or “random registration identifier” is “unique” to a particular “session.”

‘859 Patent at 26:62-63.

Such a construction is also consistent with the extrinsic dictionary submitted by Defendants, which defines “nonce” as meaning “occurring, used, or made only once or for a special occasion.” (Dkt. No. 331, Ex. 12, *The Merriam-Webster Dictionary* 355 (1998).)

The Court accordingly hereby construes **“nonce”** and **“random registration identifier”** to mean **“random and variable information unique to a cryptographic session.”**

P. “distributed repository”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
No construction necessary apart from construction of the term “repository” Alternatively: “a repository adapted for use in a distributed system”	Indefinite

(Dkt. No. 304, at 16; Dkt. No. 331, at 23.) The parties submit that this disputed term appears in Claims 1 and 58 of the ‘859 Patent. (Dkt. No. 292-1, at 3; Dkt. No. 331, at 23.)

(1) The Parties’ Positions

“In view of the lack of uncertainty regarding ‘distributed’ before the PTAB, [Plaintiff] submits that ‘distributed repository’ cannot be indefinite.” (Dkt. No. 304, at 16.) Plaintiff also argues that “[t]he repositories described in the specification are connected as a part of a distributed network or system such as the internet.” (*Id.* at 16-17 (citing ‘859 Patent at Fig. 2).)

Defendants argue that because the intrinsic evidence contains no basis for determining what it means for a repository to be “distributed,” “[Plaintiff] . . . urges the Court to rewrite the claim so that ‘distributed’ modifies the system as a whole, rather than the repository.” (Dkt. No. 331, at 23.) Defendants urge that such rewriting would be improper because “(1) courts

cannot rewrite claims and (2) it is inconsistent with the claim language, which recites both a ‘distributed repository’ and a ‘distributed system.’” (*Id.*)

Plaintiff replies that its proposal is supported by the claims and that the prosecution history contains no disclaimer. (Dkt. No. 345, at 12.)

At the February 6, 2015 hearing, Defendants argued that because the preambles of the claims at issue use the phrase “adapted for use in a distributed system,” the patentee knew how to set forth such a description but did not do so as to the term “distributed repository.” Defendants concluded that the term “distributed repository” must refer to the repository itself being “distributed” rather than merely being adapted for use in a distributed system.

(2) Analysis

The PTAB did not construe “distributed repository” apart from its construction of “repository.” (*See* Dkt. No. 304, Ex. Q, 7/1/2014 Final Written Decision (‘859 Patent), at 22-23.)

Claims 1 and 58 of the ‘859 Patent recite (emphasis added):

1. A rendering system adapted for use in a distributed system for managing use of content, said rendering system being operative to rendering [*sic*] content in accordance with usage rights associated with the content, said rendering system comprising:

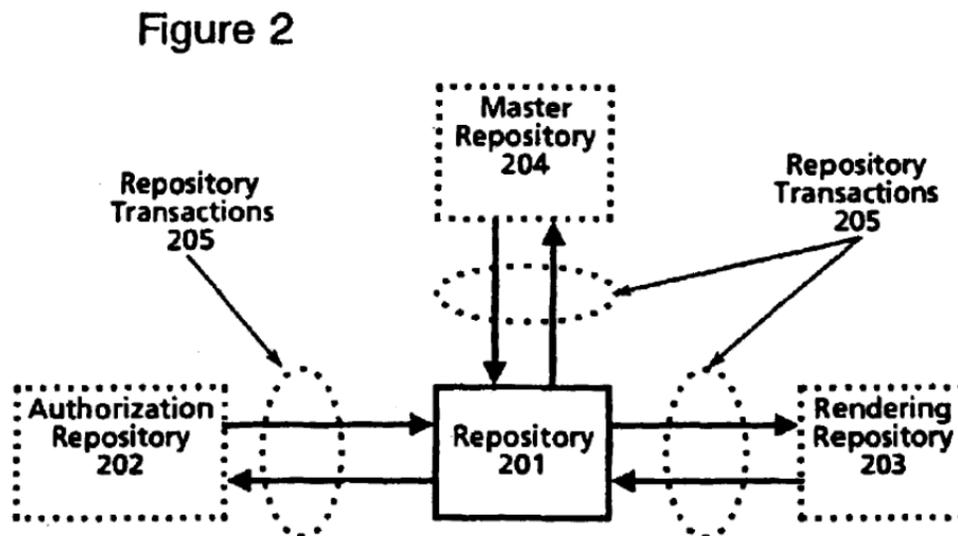
- a rendering device configured to render the content; and
- a *distributed repository* coupled to said rendering device and including a requester mode of operation and server mode of operation,
 - wherein the server mode of operation is operative to enforce usage rights associated with the content and permit the rendering device to render the content in accordance with a manner of use specified by the usage rights,
 - the requester mode of operation is operative to request access to content from another *distributed repository*, and
 - said *distributed repository* is operative to receive a request to render the content and permit the content to be rendered only if a manner of use specified in the request corresponds to a manner of use specified in the usage rights.

* * *

58. A computer readable medium including one or more computer readable instructions embedded therein for use in a distributed system for managing use of content, and operative to render content in accordance with usage rights associated with the content, said computer readable instructions configured to cause one or more computer processors to perform the steps of:

- configuring a rendering device to render the content;
- configuring a *distributed repository* coupled to said rendering device to include a requester mode of operation and server mode of operation;
- enforcing usage rights associated with the content and permitting the rendering device to render the content in accordance with a manner of use specified by the usage rights, when in the server mode of operation;
- requesting access to content from another *distributed repository*, when in the requester mode of operation; and
- receiving by said *distributed repository* a request to render the content and permitting the content to be rendered only if a manner of use specified in the request corresponds to a manner of use specified in the usage rights.

Figure 2 of the '859 Patent, cited by Plaintiff, is reproduced here:



During prosecution of the '859 Patent, Defendants argue, the patentee distinguished United States Patent No. 5,260,999 (“Wyman”) as disclosing a “centralized” system rather than a “distributed repository” as claimed by the patentee:

Wyman discloses a *centralized* license management system used to account for software product usage, wherein each licensed product upon start-up makes a call to a license server to check on whether usage is permitted (Abstract). However, “the purpose of the license management facility” of Wyman “is not that of

enforcement, nor that of ‘copy protection’, but instead is merely that of license management” (col. 14, lines 9-11). Moreover, the license management system of Wyman is *centralized* (see, e.g., Wyman Fig. 1). Accordingly, Wyman fails to disclose, teach or suggest rights that are enforced by a *distributed* repository, as recited in the independent claims.

(Dkt. No. 331, Ex. 16, 3/8/2005 Response, at 13-14 (emphasis modified).)

Defendants argue that Plaintiff’s proposed construction must fail because the patentee “distinguish[ed] [Wyman] by arguing that a license server connected with other devices in a distributed network is *not* a distributed repository.” (Dkt. No. 331, at 23.)

Plaintiff replies that “whether Wyman discloses a repository, let alone one that is distributed, was never discussed during prosecution.” (Dkt. No. 345, at 12.) Plaintiff submits that in the Office Action to which the above-quoted Response was directed, the Examiner cited a different reference, Risberg, that allegedly taught a repository. (*Id.*, Ex. AH, 9/8/2004 Office Action, at 4.) Also, at the February 6, 2015 hearing, Plaintiff emphasized that the Wyman reference was not concerned with enforcement of rights.

On balance, the prosecution history submitted by Defendants is not sufficiently clear to warrant finding any disclaimer. *See Golight, Inc. v. Wal-Mart Stores, Inc.*, 355 F.3d 1327, 1332 (Fed. Cir. 2004) (“Because the statements in the prosecution history are subject to multiple reasonable interpretations, they do not constitute a clear and unmistakable departure from the ordinary meaning of the term ‘rotating.’”); *see also Omega Eng’g*, 334 F.3d at 1324 (“As a basic principle of claim interpretation, prosecution disclaimer promotes the public notice function of the intrinsic evidence and protects the public’s reliance on *definitive* statements made during prosecution.”) (emphasis added); *id.* at 1325-26 (“[F]or prosecution disclaimer to attach, our precedent requires that the alleged disavowing actions or statements made during prosecution be both *clear and unmistakable*”) (emphasis added); *id.* at 1330 (“[T]here is more than one

reasonable basis for the amendment, rendering the intent underlying the amendment ambiguous and thus negating the possibility of the disclaimer being unmistakable.”).

As to extrinsic evidence, Plaintiff’s expert, Dr. Goodrich, opines that “[a] person of ordinary skill in the art at the time of the patent would understand a distributed repository to be capable of communicating with other repositories over a network, such as the Internet.” (Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶ 55.)

Defendants’ expert, Dr. Grimes, responds that although “[i]n computer science, the term ‘distributed’ is used to refer to systems that include a number of separate devices working in a cooperative manner, as opposed to a single device at a single location,” “[t]here is no description in the specification of any repository that is distributed over different nodes of the network.” (Dkt. No. 331, 12/22/2004 Grimes Decl., at ¶¶ 59-60.)

Finally, Plaintiff submits that in a recent IPR petition, Defendant Apple Inc. described a “distributed repository” as “a type of repository that must be able to interact with other repositories over a network, *i.e.*, in a distributed system.” (Dkt. No. 345, Ex. AG, 12/22/2014 Petition for *Inter Partes* Review (‘859 Patent), at 19-20.)

On balance, the Court finds no prosecution disclaimer (as set forth above), and the Court finds the opinions of Plaintiff’s expert more credible as to this disputed term. In particular, the Court finds more credible Plaintiff’s expert’s opinion that a person of ordinary skill in the art would be able to understand the constituent term “distributed” in the context of a repository. The Court therefore hereby expressly rejects Defendants’ indefiniteness argument.

As for the proper construction, the claim language and the above-quoted prosecution history are consistent with reading “distributed” as referring to the relative location of a repository on a network rather than the nature of the repository itself.

The Court accordingly hereby construes “**distributed repository**” to mean “**a repository adapted for use in a distributed system.**”

Q. “document platform”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“any computing system that holds a digital document, such as software” ⁶	Indefinite. Alternatively: “a repository for rendering a digital document”

(Dkt. No. 304, at 17; Dkt. No. 331, at 21.) Defendants submit that this disputed term appears in Claims 1, 8, 10, and 16 of the ‘072 Patent. (Dkt. No. 331, at 21.)

(1) The Parties’ Positions

Plaintiff argues that “[a]s the PTAB found, the specification provides sufficient context for this term.” (Dkt. No. 304, at 17.)

“First and foremost,” Defendants respond, “[Plaintiff’s] proposed construction would inject an inconsistency into the claims by failing to require the document platform to be a repository even though it communicates with a ‘document repository.’” (Dkt. No. 331, at 22.) “Second,” Defendants argue, “[Plaintiff’s] proposed construction conflates ‘document repository’ and ‘document platform.’” (*Id.*) “Third,” Defendants argue that “[Plaintiff’s] proposed construction incorrectly uses ‘software’ as an example of a ‘digital document,’ as explained more fully above [as to the term “digital document].” (*Id.*, at 23.) Defendants conclude: “While a ‘document platform’ must at least be a repository, [Plaintiff] chose to claim its alleged invention using a term that is otherwise without meaning. Given the lack of any

⁶ Plaintiff previously proposed: “a repository for rendering a digital document.” (Dkt. No. 292-1, Pl.’s P.R. 4-3 Disclosures, at 1.)

definition in the specification, [Plaintiff's] own difficulty discerning its meaning, and the problems with [Plaintiff's] latest construction, the term 'document platform' is indefinite." (*Id.*)

Plaintiff replies, "[f]irst, the fact that [Plaintiff's] construction of 'document platform' does not include the limitation 'repository' is irrelevant given that the surrounding claim language already requires the claimed document platform to include a repository." (Dkt. No. 345, at 11.) "Second, Defendants' argument that a document platform cannot also be used to render the digital documents it holds is nonsensical given that Defendants accept that the claimed document platform must include a repository that renders content." (*Id.*) Third, Plaintiff argues claim differentiation as to Claim 3 of the '072 Patent. (*Id.*) Finally, as to the change in Plaintiff's proposed construction, Plaintiff replies that "[Plaintiff] should not be penalized for attempting to work with Defendants to arrive at an agreed construction for this term." (*Id.*, at 11-12 n.13.)

At the February 6, 2015 hearing, Defendants urged that a "document platform" must be a repository, as Plaintiff itself previously proposed (as footnoted above), because use of a repository is fundamental to the Stefik Patents. Plaintiff responded that it does not dispute that a "document platform" must include a repository. Plaintiff submitted that the change in its proposed construction (so as to propose the PTAB construction) was an effort to reach a compromise, but Plaintiff expressed that the construction Plaintiff previously proposed would be acceptable.

(2) Analysis

Claim 1 of the '072 Patent is representative and recites:

1. A method for securely rendering digital documents, comprising:
retrieving, by a *document platform*, a digital document and at least one usage right associated with the digital document from a document repository, the

at least one usage right specifying a manner of use indicating the manner in which the digital document can be rendered;
storing the digital document and the at least one usage right in separate files in the *document platform*;
determining, by the *document platform*, whether the digital document may be rendered based on the at least one usage right; and
if the at least one usage right allows the digital document to be rendered on the *document platform*, rendering the digital document by the *document platform*.

As discussed regarding the terms “digital document” and “document,” above, Claim 3 of the ‘072 Patent weighs against limiting the term “document” to text.

As to whether a “document platform” should be construed as being a repository, the specification discloses:

Transactions occur between two repositories (one acting as a server), *between a repository and a document playback platform* (e.g. for executing or viewing), between a repository and a credit server or between a repository and an authorization server.

‘859 Patent at 25:48-52 (emphasis added).

The PTAB, in its Decision instituting an IPR of the ‘072 Patent, cited this above-quoted passage and found:

In our view, the discussion of “a document playback platform” set forth above provides sufficient context for construing the claim term “document platform.”

Accordingly, applying the broadest reasonable interpretation consistent with the specification, we construe the claim term “document platform” as “any computing system that holds a digital document, such as software.” For example, a “document platform” may be a computing system that executes or views software.

(Dkt. No. 304, Ex. R, 7/1/2013 Decision (‘072 Patent), at 17.)

The specification also states that “repositories will only communicate with other devices that are able to present proof that they are certified repositories.” ‘859 Patent at 12:25-27.

Because a “document platform” is disclosed as communicating with repositories, this passage

suggests that a document platform must itself be a repository. Plaintiff, however, does not appear to contend that a document platform need not at least include a repository.

Further, the specification discloses that a repository can be used for rendering content:

Rendering Repository:

A special type of repository which is typically coupled to a rendering system. The rendering repository will be typically be [*sic*] embodied within the secure boundaries of a rendering system.

‘859 Patent at 50:37-41.

As for the prosecution history, the patentee stated:

Applicants respectfully submit that newly amended independent claims 1, 14 and 26 recite various steps being performed by “a document platform”. The document platform is the client[-]side engagement that controls the document.

(Dkt. No. 331, Ex. 14, 10/15/2008 Response to Office Action, at 12.)

Finally, Plaintiff’s expert, Dr. Goodrich, has opined:

The claims of the ‘072 patent require the recited document platform to also be a rendering repository, as it receives access to content from a repository, receives usage rights from a repository, interprets and enforces usage rights, and renders content when permitted by the usage rights.

(Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶ 56.)

On balance, the intrinsic evidence, as well as the opinions of Plaintiff’s expert, are persuasive that “document platform” would be readily understandable to a person of ordinary skill in the art. The Court hereby expressly rejects Defendants’ indefiniteness argument.

In light of Plaintiff’s statement at the February 6, 2015 hearing that Plaintiff’s prior proposed construction (which is the same as Defendants’ alternative proposed construction) would be acceptable to Plaintiff, the Court adopts that agreed-upon construction.

The Court therefore hereby expressly rejects Defendants’ indefiniteness argument and hereby construes **“document platform”** to mean **“a repository for rendering a digital document.”**

R. “validating”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
The phrase “The method of claim 1, wherein the validating comprises:” should be corrected to read: “The method of claim 4, wherein the validating comprises:”	Indefinite

(Dkt. No. 304, at 17; Dkt. No. 331, at 25.) The parties submit that this disputed term appears in Claim 5 of the ‘007 Patent. (Dkt. No. 292-1, at 9-10; Dkt. No. 331, at 25.)

(1) The Parties’ Positions

Plaintiff argues that “[t]his obvious error should be corrected so that claims 4-5 parallel their mirror image claims 9-10 and 14-15” in the ‘007 Patent. (Dkt. No. 304, at 18.) Plaintiff also submits that “Defendants have not offered an alternative plausible construction, and the prosecution history does not suggest an alternate interpretation of the claims.” (*Id.*)

Defendants respond that because the requested correction is “subject to reasonable debate,” judicial correction would be inappropriate. (Dkt. No. 331, at 25.)

Plaintiff replies “Defendants’ proposed alternative correction (replacing the term ‘validating’ with ‘determining’ to correct the misstated claim dependency, instead of changing claim 5 to depend from claim 4, as [Plaintiff] proposes) is not reasonable.” (Dkt. No. 345, at 13.) Further, Plaintiff argues, “Defendants also misinterpret the prosecution history—the dependency of claim 5 was originally to application claim 10, an apparatus claim similar to claim 5. Claim 5 is a method that adds steps to claim 4, which in turn depends on claim 1. This is consistent with the patentee’s remarks.” (*Id.*, at 13 n.14.)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

Judicial correction of an error in a patent may be available “if (1) the correction is not subject to reasonable debate based on consideration of the claim language and the specification and (2) the prosecution history does not suggest a different interpretation of the claims.” *Novo Indus. v. Micro Molds Corp.*, 350 F.3d 1348, 1354 (Fed. Cir. 2003); *see LG Elecs., Inc. v. Quanta Computer Inc.*, 566 F. Supp. 2d 910, 913 (W.D. Wis. 2008) (noting the “nearly impossible standard for judicial correction of a patent” and citing *Novo*, 350 F.3d 1348, which “refus[ed] to correct ‘a’ to ‘and’ because other possibilities for correction existed”).

Claims 1, 4, and 5 of the ‘007 Patent recite (emphasis added):

1. A computer-implemented method of distributing digital content to at least one recipient computing device to be rendered by the at least one recipient computing device in accordance with usage rights information, the method comprising:

determining, by at least one sending computing device, if the at least one recipient computing device is trusted to receive the digital content from the at least one sending computing device;

sending the digital content, by the at least one sending computing device, to the at least one recipient computing device only if the at least one recipient computing device has been determined to be trusted to receive the digital content from the at least one sending computing device; and

sending usage rights information indicating how the digital content may be rendered by the at least one recipient computing device, the usage rights information being enforceable by the at least on [*sic*, one] recipient computing device.

4. The method of claim 1, wherein the determination of trust comprises:

receiving a registration message from the at least one recipient device, the registration message including an identification certificate of the recipient computing device and a random registration identifier, the identification certificate being certified by a master device;

validating the authenticity of the at least one recipient device;

exchanging messages including at least one session key with the at least one recipient device, the session key to be used in communications; and

conducting a secure transaction using the session key, wherein the secure transaction includes sending the digital content to the at least one recipient device.

5. *The method of claim 1*, wherein the *validating* comprises:
verifying the identification certificate of the at least one recipient device;
generating a message to test the authenticity of the at least one recipient device, the generated message including a nonce;
sending the generated message to the at least one recipient device; and
verifying if the at least one recipient device correctly processed the generated message.

Defendants suggest that the error could be corrected by replacing “validating” with “determining,” and Defendants cite Claims 6 and 8-10 of the ‘007 Patent.

Plaintiff replies that Defendants’ proposed alternative correction “would break the parallel structure of the three claim families of the ’007 patent and would generate additional antecedent basis problems. For example, the term ‘the identification certificate’ in claim 5 lacks antecedent basis in claim 1, but has antecedent basis in claim 4.” (Dkt. No. 345, at 13.)

Nonetheless, during prosecution, the patentee amended claim 5 of the ’007 Patent and expressly explained:

Claim 5 stands objected to as incorrectly stating a dependency on claim 10 rather than claim 1. Claim 5 is amended herein to correctly depend on claim 1. Thus, this objection should be withdrawn.

(Dkt. No. 331, Ex. 17, 11/5/2012 Response to Office Action, at 7.)

On balance, the correction sought by Plaintiff is “subject to reasonable debate.” *Novo*, 350 F.3d at 1354. Plaintiff’s request for judicial correction is therefore denied. *Novo*, 350 F.3d at 1354; *see LG*, 566 F. Supp. 2d at 913; *see also Allen Eng’g Corp. v. Bartell Indus., Inc.*, 299 F.3d 1336, 1349 (Fed. Cir. 2002) (“It is not our function to rewrite claims to preserve their validity . . .”).

Thus, based on the lack of antecedent basis, the Court hereby finds that “validating” in Claim 5 of the ‘007 Patent is **indefinite**.

S. “determining, by the document platform”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“determining by the document platform” should be corrected to: “determining, by the document repository”	Indefinite

(Dkt. No. 304, at 18; Dkt. No. 331, at 24.) The parties submit that this disputed term appears in Claim 10 of the ‘072 Patent. (Dkt. No. 292-1, at 1; Dkt. No. 331, at 24.)

(1) The Parties’ Positions

Plaintiff argues that “[t]he claim itself makes both the error and its correction apparent, and the prosecution history suggests no alternate interpretation.” (Dkt. No. 304, at 18.) Plaintiff explains that “[t]he error was introduced when other determining steps involving client-side activity were clarified to indicate that those determinations were being performed on the client.” (Dkt. No. 304, at 19 (citing *id.*, Ex. N, 10/15/2008 Response to Office Action at 2 & 9-10).)

Defendants respond that “[Plaintiff] cannot establish that the requested correction is ‘not subject to reasonable debate.’” (Dkt. No. 331, at 24.)

Plaintiff replies that “Defendants argue that the patentee intentionally wrote and prosecuted an indefinite claim. That is nonsense” (Dkt. No. 345, at 13.) Plaintiff further explains:

Defendants admit that the asserted claims of the ‘072 patent recite a document repository (the server-side engagement) and a document platform (the client-side engagement). Dkt. 331 at 23. With this understanding, the error of having the client device (the “document platform”) decide whether to grant its own request is clear, and thus susceptible to correction. Dkt. 304 at 18-19; Goodrich Reply Decl. ¶¶ 39-42. It is equally clear from the claim language that the server-side device (the document repository) must perform the step of determining if the client is authorized to receive the content.

(Dkt. No. 345, at 13.)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

Judicial correction of an error in a patent may be available “if (1) the correction is not subject to reasonable debate based on consideration of the claim language and the specification and (2) the prosecution history does not suggest a different interpretation of the claims.” *Novo*, 350 F.3d at 1354; *see LG*, 566 F. Supp. 2d at 913 (noting the “nearly impossible standard for judicial correction of a patent” and citing *Novo*, 350 F.3d 1348, which “refus[ed] to correct ‘a’ to ‘and’ because other possibilities for correction existed”).

Claim 10 of the ‘072 Patent recites (emphasis added):

10. A method for securely rendering digital documents, comprising:
storing a digital document and at least one usage right in separate files in a document repository, wherein the at least one usage right is associated with the digital document;
receiving a request from a document platform for access to the digital document;
determining, by the document platform, whether the request may be granted based on the at least one usage right, the determining step including authenticating the document platform and determining whether the at least one usage right includes a manner of use that allows transfer of the digital document to the document platform;
if the at least one usage right allows the transfer of the digital document to the document platform, transferring the digital document and the at least one usage right associated with the digital document to the document platform;
storing the digital document and the at least one usage right in the document platform, wherein the at least one usage right is stored in a separate file from the digital document; and
rendering the digital document by the document platform.

Plaintiff’s expert, Dr. Goodrich, opines:

A person of ordinary skill in the art would recognize an error in claim 10 of the ‘072 patent where the document platform of the claim is required to determine whether a request the document platform made in the previous step should be granted. When the determining step is performed, the request subject to the determining step has just been received from the document platform. The determining step requires the determining entity (document platform or document repository) to determine whether a usage right includes a manner of use that allows a digital document to be transferred to the document platform. The document platform does not possess the subject usage right until, later in the

claim, only after the determining step has been resolved in favor of granting the request; the usage right is transferred to the document platform with the digital document. Prior to the “receiving” step, claim 10 introduces the digital document and the usage right as being stored by the document repository. Accordingly, one having ordinary skill in the art would understand that the determining step should be performed by the document repository (which is storing the right used to perform the determining step) and not the document platform (which cannot store the usage right until after the determining step).

(Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶ 57.)

On balance, the correction sought by Plaintiff is “subject to reasonable debate.” *Novo*, 350 F.3d at 1354. For example, as argued by Defendants, Claim 10 of the ‘072 Patent could be corrected by deleting the entire phrase “by the document platform” or by moving the phrase “by the document platform” to after the word “request.”

Moreover, during prosecution the patentee relied upon the phrase “determining, by the document platform” when distinguishing a prior art reference:

... *Perritt* fails to disclose, teach or suggest the steps of determining, *by the document platform*, whether the request may be granted based on the at least one usage right, and storing the digital document and the at least one usage right in the document platform, wherein the at least one usage right is stored in a separate file from the digital document, as recited in independent claim 14.

(Dkt. No. 331, Ex. 14, 10/15/2008 Response to Office Action, at 10 (emphasis in original).)

Plaintiff replies that “[n]owhere in their response to the Office Action do applicants attempt to explain how the ‘document platform’ makes a determination about whether it is authorized to receive the requested content.” (Dkt. No. 345, at 13.) Nonetheless, this prosecution history weighs against finding that the claim contains an error for which the proper correction is not “subject to reasonable debate.” *Novo*, 350 F.3d at 1354.

Plaintiff’s request for judicial correction is therefore denied. *Novo*, 350 F.3d at 1354; *see LG*, 566 F. Supp. 2d at 913. This finding also comports with the principle that “[c]ourts do not rewrite claims; instead, we give effect to the terms chosen by the patentee.” *K-2 Corp. v.*

Salomon S.A., 191 F.3d 1356, 1364 (Fed. Cir. 1999); *see Chef Am., Inc. v. Lamb-Weston, Inc.*, 358 F.3d 1371, 1374 (Fed. Cir. 2004) (“courts may not redraft claims, whether to make them operable or to sustain their validity”); *see also Allen Eng’g*, 299 F.3d at 1349 (“It is not our function to rewrite claims to preserve their validity . . .”).

The remaining issue is whether the disputed term renders the uncorrected claim invalid as indefinite.

Defendants have failed to demonstrate any inconsistency that would preclude a person of ordinary skill in the art from understanding the meaning of the disputed term. Instead, the parties simply seem to submit that the claim, as written, does not reflect what the patentee intended to claim. Defendants have not established that this is an appropriate basis for finding indefiniteness.

Defendants’ indefiniteness argument is therefore hereby expressly rejected. No further construction is required. *See PPG Indus. v. Guardian Indus. Corp.*, 156 F.3d 1351, 1355 (Fed. Cir. 1998) (“[A]fter the court has defined the claim with whatever specificity and precision is warranted by the language of the claim and the evidence bearing on the proper construction, the task of determining whether the construed claim reads on the accused product is for the finder of fact.”); *see also U.S. Surgical*, 103 F.3d at 1568; *O2 Micro*, 521 F.3d at 1362.

The Court accordingly hereby construes “**determining, by the document platform**” to have its **plain meaning**.

T. “grammar”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“a manner of defining a valid sequence of symbols for a language”	“a manner of defining a valid sequence of symbols consisting of brackets, bars and braces used to describe the language of usage rights sentences, parentheses used to group items together in lists, keywords followed by colons used to indicate a single value, typically an identifier or list of identifiers, and the suffix ‘ID’” Alternatively, indefinite.

(Dkt. No. 304, at 19; Dkt. No. 331, at 19.) The parties submit that this disputed term appears in Claim 1 of the ‘160 Patent. (Dkt. No. 292-1, at 4; Dkt. No. 331, at 19.)

(1) The Parties’ Positions

Plaintiff argues that “Defendants seek to shoehorn additional limitations into the term ‘grammar’ from a subsequent portion of the specification that discusses a particular ‘notation’ that can be used ‘[i]n describing the grammar’ of the preferred embodiment.” (Dkt. No. 304, at 19.)

Defendants respond that “[t]he patentees did not use ‘grammar’ in the ‘160 patent as a generic term but rather a specifically disclosed sequence of symbols used to create the ‘usage rights language.’” (Dkt. No. 331, at 19.) Defendants further explain: “There are many ways to use and arrange the grammar elements, but the grammar used to describe ‘usage rights’ must be the particular grammar described by the specification. If not, then grammar does not give any meaning to the claims in which it is used, and it would be indefinite.” (*Id.*, at 19-20.)

Plaintiff replies that “[t]he claim . . . is directed to the statements generated in accordance with a grammar to express usage rights (‘statements from a usage rights language’), not a notation used to express a grammar.” (Dkt. No. 345, at 10.)

(2) Analysis

Claim 1 of the '160 Patent recites (emphasis added):

1. A computer readable medium having embedded thereon a digital work adapted to be distributed within a system for controlling use of digital works, said digital work comprising:

a digital content portion that is renderable by a rendering device;

a usage rights portion associated with said digital content portion and comprising one or more computer readable instructions configured to permit or prohibit said rendering device to render said digital content portion, said usage rights portion being expressed as statements from a usage rights language having a *grammar* defining a valid sequence of symbols, and specifying a manner of use relating to one or more purposes for which the digital work can be used by an authorized party; and

a description structure comprising a plurality of description blocks, each of said description blocks comprising address information for at least one part of said digital work, and a usage rights part for associating one or more usage rights portions.

The specification discloses:

The usage rights language is based on the *grammar* described below. A *grammar* is a convenient means for defining valid sequence [*sic*] of symbols for a language. In describing the *grammar* the notation “[a|b|c]” is used to indicate distinct choices among alternatives. In this example, a sentence can have either an “a”, “b” or “c”. It must include exactly one of them. The braces { } are used to indicate optional items. Note that brackets, bars and braces are used to describe the language of usage rights sentences but do not appear in actual sentences in the language.

'859 Patent at 17:41-50 (emphasis added; square brackets in “[a|b|c]” as in original); *see id.*

at 17:57-60 (regarding “keywords”) & 28:4-8 (regarding how to specify “things” that “need to be identified”).

As to extrinsic evidence, Plaintiff has cited the following definitions: “grammar” in the context of “computing” means “a set of rules governing what strings are valid or allowable in a language or text” (Dkt. No. 304, Ex. S, “Google.com Dictionary”); “grammar” means “[a] normative or prescriptive set of rules setting forth a standard of usage” (*id.*, Ex. T, *American Heritage College Dictionary* 602 (2002)); and “language” in the context of computer science

means “[a] system of symbols and rules used for communication with or between computers”

(*id.*, Ex. U, “thefreedictionary.com”).

Plaintiff also submits the opinion of is expert, Dr. Goodrich, that:

The term “grammar” in computer science has been well understood for decades to comprise a set of rules for a language that govern what constitutes valid or permissible strings or sequences of symbols in that language. * * * [T]o a person of ordinary skill in the art, the claim element requiring that statements be expressed from a “language having a grammar” would be well understood to require that statements in that language be generated and interpreted according to a set of rules making up the grammar, which as noted above could be represented using a variety of notational techniques, and would not include the limitation of a particular notation “consisting of brackets, bars and braces,” or the like.

(Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶¶ 65 & 67.)

Defendants’ expert, Dr. Grimes, responds that “[t]he term [‘grammar’] is deprived of any real meaning until it is applied to a particular computer language,” and the specification “defines the term ‘grammar’ by distinguishing it from any computer language” as set forth above. (Dkt. No. 331, Ex. 11, 12/22/2014 Grimes Decl. at ¶¶ 56-57.)

On balance, the surrounding claim language, the specification, the extrinsic dictionary definitions, and Plaintiff’s expert’s opinion, together, are persuasive evidence that grammar is a generic term that should not be limited to the specific features proposed by Defendants.

Defendants’ proposal imports limitations from preferred embodiments and is therefore rejected.

See Electro Med., 34 F.3d at 1054.

The Court accordingly hereby construes “**grammar**” to mean “**a manner of defining a valid sequence of symbols for a language.**”

U. “description structure”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“a structure which describes the location of content and the usage rights for a digital work that is comprised of description blocks, each of which corresponds to a digital work or to an interest in a digital work”	“any acyclic structure that represents the relationship between the components of a digital work”

(Dkt. No. 304, at 20; Dkt. No. 292-2, at 15-16.) Plaintiff submits that this term appears in Claim 1 of the ‘160 Patent. (Dkt. No. 292-1, at 4; Dkt. No. 292-2, at 15-16.)

Plaintiff argues that “Defendants ignore the glossary and rely on a portion of the detailed description of a preferred embodiment” (Dkt. No. 304, at 20.)

Defendants’ response brief does not address this disputed term. (*See* Dkt. No. 331.)

Plaintiff replies that “Defendants have dropped their proposal (which improperly includes an ‘acyclic’ limitation) and agreed to a modified version of [Plaintiff’s] construction,” namely: “A structure which describes the location of content and the usage rights and usage fees, if any such usage fees are required, for a digital work that is comprised of description blocks, each of which corresponds to a digital work or to an interest in a digital work.” (Dkt. No. 345, at 10 & n.10). The parties’ January 23, 2015 Joint Claim Construction Chart confirms that the parties have reached agreement as to this term (Dkt. No. 366, Ex. B, at 16), and the parties did not address this disputed term otherwise at the February 6, 2015 hearing.

The Court accordingly hereby construes **“description structure”** to mean **“a structure which describes the location of content and the usage rights and usage fees, if any such usage fees are required, for a digital work that is comprised of description blocks, each of which corresponds to a digital work or to an interest in a digital work.”**

V. “means for communicating with a master repository for obtaining an identification certificate for the repository”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“Corresponding Structure: an external interface that provides for a signal connection with another device described at 13:52-59” ⁷	Indefinite

(Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at p. 16; Dkt. No. 331, at 25; Dkt. No. 366, Ex. B, at 6.) The parties submit that this disputed term appears in Claim 24 of the ‘859 Patent. (Dkt. No. 292-1, at 5; Dkt. No. 331, at 25.)

(1) The Parties’ Positions

Plaintiff’s expert submits that the cited structure “is identical to those [*sic*] given by the PTAB for these means in the ‘859 patent.” (Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶ 35 (citation omitted).)

Defendants argue that “[t]he recited function is indefinite because ‘the repository’ could refer to the ‘distributed repository,’ the ‘another distributed repository,’ or the ‘master repository.’” (Dkt. No. 331, at 25.)

Plaintiff replies that because “[m]aster repositories issue identification certificates to other ‘distributed repositories,’” “the reference to communicating with a master repository to

⁷ Plaintiff previously proposed: “Corresponding Structure: algorithm/structure necessary for performing the recited function set forth in the ‘859 Spec., including from the following portions: 7:16-17; 13:52-59, Fig. 2, Fig. 12, and equivalents thereof.” (Dkt. No. 292-1, at 5.)

obtain a certificate can only be attributed to the previously recited ‘distributed repository.’”

(Dkt. No. 345, at 14.)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

In a Decision instituting an IPR of the ‘859 Patent, the PTAB found:

Claim 23 recites “means for communicating with an authorization repository for authorizing a condition.” Claim 24 recites “means for communication with a master repository for obtaining an identification certificate.”

For corresponding structure, [Petitioner] ZTE contends the following:

Claim 23 requires means for communicating with an authorization repository for authorizing a condition, and claim 24 requires means for communication with a master repository for obtaining an identification certificate. The corresponding structure for performing the claimed function of means for communicating is external interface 1206 (Fig. 12; 13:52-59: “The external interface means 1206 provides for the signal connection to other repositories.”).

(Pet. 7) We agree. Each of the “means for communicating . . .” covers an external interface that provides for a signal connection with another device and equivalents thereof.

(Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl., Ex. 2, 7/1/2013 Decision (IPR2013-00137)

at 16 (p. 102 of 204 of Ex. K) (citation omitted).)

The parties’ present dispute boils down to whether a reasonably clear antecedent basis exists for “the repository” in Claim 24 of the ‘859 Patent. Claims 1 and 24 of the ‘859 Patent recite (emphasis added):

1. A rendering system adapted for use in a distributed system for managing use of content, said rendering system being operative to rendering [*sic*] content in accordance with usage rights associated with the content, said rendering system comprising:

a rendering device configured to render the content; and
a *distributed repository* coupled to said rendering device and including a requester mode of operation and server mode of operation,

wherein the server mode of operation is operative to enforce usage rights associated with the content and permit the rendering device to render the content in accordance with a manner of use specified by the usage rights,

the requester mode of operation is operative to request access to content from *another distributed repository*, and

said distributed repository is operative to receive a request to render the content and permit the content to be rendered only if a manner of use specified in the request corresponds to a manner of use specified in the usage rights.

* * *

24. A rendering system as recited in claim 1, further comprising means for communicating with a master repository for obtaining an identification certificate for *the repository*.

The specification discloses:

As a prerequisite to operation, a repository will require possession of an identification certificate. Identification certificates are encrypted to prevent forgery and are issued by a Master repository. A master repository plays the role of an authorization agent to enable repositories to receive digital works. Identification certificates must be updated on a periodic basis. Identification certificates are described in greater detail below with respect to the registration transaction.

‘859 Patent at 13:1-9.

Communications integrity refers to the integrity of the communications channels between repositories. Roughly speaking, communications integrity means that repositories cannot be easily fooled by “telling them lies.” Integrity in this case refers to the property that repositories will only communicate with other devices that are able to present proof that they are certified repositories, and furthermore, that the repositories monitor the communications to detect “impostors” and malicious or accidental interference.

Id. at 12:21-29; *see id.* at 26:43-57 (disclosing a “registration transaction between two repositories”).

Plaintiff also submits that in a recent IPR petition, Defendant Apple Inc. stated:

Claim 24 specifies the rendering system of claim 1 comprises “means for communicating with a master repository for obtaining an identification certificate for the repository.” In IPR2013-00137, the Board construed this means “to be the external interface 1206, which provides a signal connection with another device.”

. . . This conclusion is supported by the '859 disclosure. *See* Ex. 1001 [('859 Patent)] at 13:20-22, 13:52-59.

(Dkt. No. 345, Ex. AG, 12/22/2014 Petition for *Inter Partes* Review ('859 Patent), at 22.)

Finally, Plaintiff's expert, Dr. Goodrich, opines that a person of ordinary skill in the art would understand from the intrinsic evidence that "a master repository is the issuer of an identification certificate, not the recipient That is, the line of communication for obtaining an identification certificate always goes from a master repository to a recipient repository."

(Dkt. No. 345, Ex. AA, 1/9/2015 Goodrich Decl., at ¶ 23.) Plaintiff's expert concludes that "it is clear from [the specification and the claims] that to someone with ordinary skill in the art 'the repository' referenced in claim 24 must be the 'distributed repository' referenced in claim 1."

(*Id.*)

On balance, the Court finds credible Plaintiff's expert's assessment of how a person of ordinary skill in the art would understand Claim 24 in light of the intrinsic evidence.

The Court therefore hereby finds that the antecedent basis for "the repository" in Claim 24 of the '859 Patent is "a distributed repository" in Claim 1 of the '859 Patent. Defendants' indefiniteness argument is accordingly hereby expressly rejected.

The parties do not appear to otherwise dispute the claimed function, and Defendants have not proposed any alternative corresponding structure. Further, the Court affords "reasoned deference" to the above-quoted finding of the PTAB. *Maurice Mitchell*, 2006 WL 1751779, at *4; *see TQP*, 2014 WL 2810016, at *6; *see also Teva*, 135 S. Ct. at 839-40.

The Court accordingly hereby finds that for the "**means for communicating with a master repository for obtaining an identification certificate for the repository,**" the function is "**communicating with a master repository for obtaining an identification certificate for**

the repository,” and the corresponding structure is “the external interface means 1206 described in the ‘859 Patent at 13:52-59; and equivalents thereof.”

W. “means for processing a request from the means for requesting”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
<p>Corresponding Structure: “One or more processors that implement the following algorithm: Processing begins when a requester repository has sent a message to initiate a request. (36:35-40, 37:9-13.)”⁸</p>	<p>“Processing begins when a requester repository has sent a message to initiate a request. First, the server repository checks the compatibility of the requester repository and the validity of the requester’s identification. Second, the requester and server repositories perform the common opening transaction steps of determining whether authorization is needed, and whether the requested transaction is permitted given the usage rights. Third, requester and server repositories perform a transmission protocol to read and write blocks of data, and then the requester repository renders the digital work. Fourth, the contents are removed from the rendering device and the requester repository. Finally, the requester and server repositories perform the common closing transaction steps of updating the usage rights and billing information.”⁹</p>

⁸ Plaintiff previously proposed: “Corresponding Structure: algorithm/structure necessary for performing the recited function set forth in the ‘576 Spec., including from the following portions: 30:25-30, 30:59-31:49, 32:20-31, 36:35-54, 37:9-26, and equivalents thereof; see also, e.g., ‘576 PTAB Decision at 21-22.” (Dkt. No. 292-1, at 6.) Plaintiff also previously proposed: “processing begins when a requester repository has sent a message to initiate a request (36:35-40, 37:9-13.)” (Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at p. 7.)

⁹ Defendants previously proposed: “A computer processor and processor memory containing software that, sequentially, instructs the server repository to check the compatibility of the requester repository and the validity of the requester’s identification, instructs the requester and server repositories to perform the common opening transaction steps of determining whether authorization is needed and whether the requested transaction is permitted given the usage rights, instructs the requester and sever [*sic*, server] repositories to perform a transmission protocol to read and write blocks of data and then instructs the requester repository to render the digital work, instructs the rendering device and the requester repositories to remove the contents, and instructs the requester and server repositories to perform closing transaction steps of updating the usage rights and billing information. Alternatively, indefinite.” (Dkt. No. 292-2, at 27-28.)

(Dkt. No. 366, Ex. B, at 21; Dkt. No. 331, at 26; Dkt. No. 366, Ex. B, at 21.) The parties submit that this disputed term appears in Claim 1 of the '576 Patent. (Dkt. No. 292-1, at 6; Dkt. No. 331, at 26.)

(1) The Parties' Positions

Plaintiff's expert opines: "A person of ordinary skill in the art would identify the algorithm described by the PTAB as 'processing begins when a requester repository has sent a message to initiate a request' as being clearly linked or associated to the recited function. This algorithm is described in the specification at 36:35-40 and 37:9-13, which includes the contents of the requesting message, as cited by the PTAB on page 21. . . . The remaining portions of the algorithm included in the Defendants' proposed construction are not necessary to perform the recited function." (Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶¶ 18-19.)

Defendants argue that Plaintiff's proposal "only describes prerequisites for the processing to 'begin[]' and not how it is carried out," which is set forth by the five-step algorithm that appears in the PTAB's construction. (Dkt. No. 331, at 26.) Defendants also argue that the passages cited by Plaintiff "at most restate the function without explaining the steps conducted to implement it." (*Id.*, at 27 (citing *Aristocrat Techs. Austral. Pty Ltd. v. Int'l Game Tech.*, 521 F.3d 1328, 1335 (Fed. Cir. 2008) (rejecting an alleged "algorithm" that was "at best, a description of the claimed function of the means-plus-function claim").) Finally, Defendants argue that rather than merely citing passages from the specification, "that text should actually be reproduced in the construction, to avoid burdening jurors with the need to look up each citation in the patent or confusing them as to whether the cited text is required structure." (Dkt. No. 331, at 27.)

Plaintiff replies that Defendants have proposed "numerous steps that are not necessary to perform the recited function," and "[b]ecause the 'means for processing' requires no more than

merely processing, a general-purpose computer processor is sufficient structure.” (Dkt. No. 345, at 14.)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

Claim 1 of the ‘576 Patent recites (emphasis added):

1. An apparatus for rendering digital content in accordance with rights that are enforced by the apparatus, said apparatus comprising:
 - a rendering engine configured to render digital content;
 - a storage for storing the digital content;
 - means for requesting use of the digital content stored in the storage; and
 - a repository coupled to the rendering engine, wherein the repository includes:
 - means for processing a request from the means for requesting,*
 - means for checking whether the request is for a permitted rendering of the digital content in accordance with rights specified in the apparatus,
 - means for processing the request to make the digital content available to the rendering engine for rendering when the request is for a permitted rendering of the digital [sic]; and
 - means for authorizing the repository for making the digital content available for rendering, wherein the digital content can be made available for rendering only by an authorized repository, the repository comprising:
 - means for making a re[quest] for an authorization ob[j]ect required to be included within the repository for the apparatus to render the digital content; and
 - means for receiving the authorization ob[j]ect when it is determined that the request should be granted.

In instituting an IPR of the ‘576 Patent, the PTAB found as to this disputed term:

Processing begins when a requester repository has sent a message to initiate a request. (Ex. 1001 [(‘576 Patent)], 36:35-40, 37:9-13.) First, the server repository checks the compatibility of the requester repository and the validity of the requester’s identification. (Ex. 1001, 36:41-44, 37:14-17.) Second, the requester and server repositories perform the common opening transaction steps of determining whether authorization is needed, and whether the requested transaction is permitted given the usage rights. (Ex. 1001, 36:45-46, 37:18-19, 30:59-31:49.) Third, requester and server repositories perform a transmission protocol to read and write blocks of data, and then the requester repository renders the digital work. (Ex. 1001, 36:47-50, 37:20-22.) Fourth, the contents are removed from the rendering device and the requester repository. (Ex. 1001, 36:51-52, 37:23-24.) Finally, the requester and server repositories perform the

common closing transaction steps of updating the usage rights and billing information. (Ex. 1001, 36:53-54, 37:25-26, 32:20-31.)

The above-noted description constitutes a sufficiently disclosed algorithm, expressed in prose, which, in combination with processor 1200, constitutes a corresponding structure for the claimed “means for processing a request from the means for requesting.”

(Dkt. No. 331, Ex. 18, 7/9/2013 Decision (‘576 Patent), at 21-22.)

On one hand, the PTAB Decision is entitled to some deference. *See Maurice Mitchell*, 2006 WL 1751779, at *4; *see also TQP*, 2014 WL 2810016, at *6; *Teva*, 135 S. Ct. at 839-40.

On the other hand, this Court conducts an independent review of claim construction disputes. *See, e.g., Texas Instruments*, 182 F. Supp. 2d at 589-90; *Burns, Morris*, 401 F. Supp. at 697; *Negotiated Data Solutions*, 2012 WL 6494240, at *5.

“[A] means-plus-function claim element for which the only disclosed structure is a general purpose computer is invalid if the specification fails to disclose an algorithm for performing the claimed function.” *Net MoneyIN Inc. v. VeriSign, Inc.*, 545 F.3d 1359, 1367 (Fed. Cir. 2008); *see WMS Gaming, Inc. v. Int’l Game Tech.*, 184 F.3d 1339, 1349 (Fed. Cir. 1999) (“In a means-plus-function claim in which the disclosed structure is a computer, or microprocessor, programmed to carry out an algorithm, the disclosed structure is not the general purpose computer, but rather the special purpose computer programmed to perform the disclosed algorithm.”); *see also Noah Sys. v. Intuit Inc.*, 675 F.3d 1302, 1319 (Fed. Cir. 2012) (“Computer-implemented means-plus-function claims are indefinite unless the specification discloses an algorithm to perform the function associated with the limitation.”).

There is, however, an exception to the general rule requiring an algorithm. Specifically, when the corresponding structure is a general purpose computer, an algorithm is required *unless* the recited function can be achieved by any general purpose computer without special

programming. *In re Katz Interactive Call Processing Patent Litig.*, 639 F.3d 1303, 1316 (Fed. Cir. 2011) (“Absent a possible narrower construction of the terms ‘processing,’ ‘receiving,’ and ‘storing,’ . . . those functions can be achieved by any general purpose computer without special programming. As such, it was not necessary to disclose more structure than the general purpose processor that performs those functions.”); *accord Ergo Licensing, LLC v. CareFusion 303, Inc.*, 673 F.3d 1361, 1365 (Fed. Cir. 2012) (“In *In re Katz*, we held that ‘[a]bsent a possible narrower construction’ of the terms ‘processing,’ ‘receiving,’ and ‘storing,’ the disclosure of a general-purpose computer was sufficient. . . . In other words, a general-purpose computer is sufficient structure if the function of a term such as ‘means for processing’ requires no more than merely ‘processing,’ which any general-purpose computer may do without any special programming.”) (citations omitted); *but see id.* (“It is only in the rare circumstances where any general-purpose computer without any special programming can perform the function that an algorithm need not be disclosed.”).

The claimed function of “processing a request from the means for requesting” does not include anything beyond merely generic “processing.” As found in *Katz*, this function “can be achieved by any general purpose computer without special programming.” *In re Katz*, 639 F.3d at 1316.

The Court accordingly hereby finds that for the **“means for processing a request from the means for requesting,”** the function is **“processing a request from the means for requesting,”** and the corresponding structure is **“a general-purpose computer; and equivalents thereof.”**

X. “means for checking whether the request is for a permitted rendering of the digital content in accordance with rights specified in the apparatus”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
<p>“Corresponding Structure: One or more processors that implement the following algorithm: ‘the server repository determines whether the right is granted and whether any specified time, security, and access based conditions are satisfied (31:13-33.)’”¹⁰</p>	<p>“First, the requester repository determines whether an authorization certificate or a digital ticket is needed. Second, the server repository generates a transaction identifier. Third, the server repository determines whether the right is granted and whether time, security, and access based conditions are satisfied. Finally, the server repository determines whether there are sufficient copies of the work to distribute”¹¹</p>

(Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at p. 10; Dkt. No. 331, at 27; Dkt. No. 366, Ex. B, at 22.) The parties submit that this disputed term appears in Claim 1 of the ‘576 Patent. (Dkt. No. 292-1, at 5; Dkt. No. 331, at 27.)

(1) The Parties’ Positions

Plaintiff’s expert opines: “A person of ordinary skill in the art would identify the algorithm described by the PTAB as ‘the server repository determines whether the right is granted and whether time, security, and access based conditions are satisfied’ as being clearly linked or associated to the recited function. This algorithm is described in the ‘576 patent at

¹⁰ Plaintiff previously proposed: “Corresponding Structure: algorithm/structure necessary for performing the recited function set forth in the ‘576 Spec., including from the following portions: 30:59-31:49, 31:3-49, 36:45-46, 37:18-19, and equivalents thereof; see also, e.g., ‘576 PTAB Decision at 22-23.” (Dkt. No. 292-1, at 5.)

¹¹ Defendants previously proposed: “A computer processor and processor memory containing software that, sequentially, instructs the requester repository to determine whether an authorization certificate or a digital ticket is needed, instructs the server repository to generate a transaction identifier, instructs the server repository to determine whether the right is granted and whether time, security, and access based conditions are satisfied, and instructs the server repository to determine whether there are sufficient copies of the work to distribute. Alternatively, indefinite.” (Dkt. No. 292-2, at 28.)

31:13-33, which describes in prose the functions shown in the flowchart of Figure 18 boxes 1804 through 1807, inclusive.” (Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶ 26 (citation omitted).)

Defendants argue that Plaintiff’s proposal “does little more than restate the function” and “wrongly dismisses the PTAB’s other corresponding steps.” (Dkt. No. 331, at 28.)

Plaintiff replies that “[o]nly the second step [proposed by Defendants] is necessary to check whether the request is for a permitted rendering of the digital content.” (Dkt. No. 345, at 14.)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

Claim 1 of the ‘576 Patent recites (emphasis added):

1. An apparatus for rendering digital content in accordance with rights that are enforced by the apparatus, said apparatus comprising:
 - a rendering engine configured to render digital content;
 - a storage for storing the digital content;
 - means for requesting use of the digital content stored in the storage; and
 - a repository coupled to the rendering engine, wherein the repository includes:
 - means for processing a request from the means for requesting,
 - means for checking whether the request is for a permitted rendering of the digital content in accordance with rights specified in the apparatus,*
 - means for processing the request to make the digital content available to the rendering engine for rendering when the request is for a permitted rendering of the digital [*sic*]; and
 - means for authorizing the repository for making the digital content available for rendering, wherein the digital content can be made available for rendering only by an authorized repository, the repository comprising:
 - means for making a re[qu]est for an authorization ob[j]ect required to be included within the repository for the apparatus to render the digital content; and
 - means for receiving the authorization ob[j]ect when it is determined that the request should be granted.

In instituting an IPR of the ‘576 Patent, the PTAB found as to this disputed term:

First, the requester repository determines whether an authorization certificate or a digital ticket is needed. (Ex. 1001 [('576 Patent)], 31:3-9.) Second, the server repository generates a transaction identifier. (Ex. 1001, 31:10-13.) Third, the server repository determines whether the right is granted and whether time, security, and access based conditions are satisfied. (Ex. 1001, 31:13-33.) Finally, the server repository determines whether there are sufficient copies of the work to distribute. (Ex. 1001, 31:34-49.) The above-noted description constitutes a sufficiently disclosed algorithm, expressed in prose, which, in combination with processor 1200, constitutes a corresponding structure for the claimed “means for checking whether the request is for a permitted rendering of the digital content in accordance with rights specified in the apparatus.”

(Dkt. No. 331, Ex. 18, 7/9/2013 Decision ('576 Patent), at 22-23.)

The parties' experts dispute whether the steps of the PTAB's construction are necessary to perform the claimed function. (Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶¶ 27-29; Dkt. No. 331, Ex. 11, Grimes Decl. at ¶¶ 16-24.) On balance, the Court finds Defendants' expert more persuasive. In particular, Plaintiff's expert has failed to persuasively demonstrate that the additional steps identified by the PTAB are not part of the algorithm disclosed for performing the claimed function. *See* '859 Patent at 30:16-62. Finally, the Court affords “reasoned deference” to the PTAB Decision. *Maurice Mitchell*, 2006 WL 1751779, at *4; *see TQP*, 2014 WL 2810016, at *6; *see also Teva*, 135 S. Ct. at 839-40.

The Court accordingly hereby finds that for the **“means for checking whether the request is for a permitted rendering of the digital content in accordance with rights specified in the apparatus,”** the function is **“checking whether the request is for a permitted rendering of the digital content in accordance with rights specified in the apparatus,”** and the corresponding structure is **“a processor configured to perform the following algorithm and equivalents thereof: first, the requester repository determines whether an authorization certificate or a digital ticket is needed; second, the server repository generates a transaction identifier; third, the server repository determines whether the right**

is granted and whether time, security, and access based conditions are satisfied; and finally, the server repository determines whether there are sufficient copies of the work to distribute; and equivalents thereof.”

Y. “means for receiving the authorization ob[j]ect when it is determined that the request should be granted”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
<p>“Corresponding Structure: One or more processors that implement the following algorithm: The remote repository transmits a block of data to the server repository and waits for an acknowledgement, which the server provides when the block of data has been completely received. (33:9-15.) Unless a communications failure terminates the transaction, that process repeats until there are no more blocks to transmit. (33:16-38, 33:46-49.) Finally, the server repository sends a completion acknowledgement to the remote repository. (33:39-45.)”¹²</p>	<p>“[T]he server repository transmits a block of data to the requester repository and waits for an acknowledgement, which the requester provides when the block of data has been completely received. Unless a communications failure terminates the transaction, that process repeats until there are no more blocks to transmit. Finally, the requester repository sends a completion acknowledgement to the server repository.</p> <p>In that regard, the specification states: “The key property is that both the server and the requester cancel a transaction if it is interrupted before all of the data blocks are delivered, and commits to it if all of the data blocks have been delivered.”¹³</p>

¹² Plaintiff previously proposed: “Corresponding Structure: algorithm/structure necessary for performing the recited function set forth in the ‘576 Spec., including from the following portions: 32:33, 33:9-49, 36:47-50, 41:50-65, and equivalents thereof; see also, e.g., ‘576 PTAB Decision at 26.” (Dkt. No. 292-1, at 6.)

¹³ Defendants previously proposed: “A computer processor and processor memory containing software that, sequentially, instructs the server repository to transmit a block of data to the requester repository and to wait for an acknowledgment from the requester that the block of data has been completely received, instructs the server repository to repeat that transmission and waiting for acknowledgment until there are no more blocks to transmit, instructs the requester repository to send a completion acknowledgment to the server repository, and instructs both the server and the requester to cancel a transaction if it is interrupted before all the data blocks are delivered, and instructs both the server and the requester to commit to the transaction if all the data blocks have been delivered. Alternatively, indefinite.” (Dkt. No. 292-2, at 30-31.)

(Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at p. 14; Dkt. No. 331, at 28; Dkt. No. 366, Ex. B, at 26.) The parties submit that this disputed term appears in Claim 1 of the ‘576 Patent. (Dkt. No. 292-1, at 6; Dkt. No. 331, at 28.)

(1) The Parties’ Positions

Plaintiff’s expert opines:

The PTAB construction incorrectly referred to the “server repository” and “requester repository.” However, the PTAB construction[] . . . for the “means for authorizing the repository for making the digital content available for rendering” makes clear that the “authorization object” is transmitted from the “remote repository” to the “server repository.” In fact, the PTAB stated that “It is described in the specification that the authorization process invokes a ‘play’ transaction to acquire an authorization object from a remote repository.” Accordingly, I have corrected this apparent clerical error in the PTAB construction.

(Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶ 33 (citation omitted).)

Defendants argue that the PTAB’s construction should be adopted because “[t]he PTAB’s terminology . . . comes straight from the cited passages of the patent.” (Dkt. No. 331, at 28 (citing ‘576 Patent at 33:9-10).)

Plaintiff replies that “Defendants’ construction fails to account for the ‘remote repository.’” (Dkt. No. 345, at 15; *see id.*, Ex. AA, 1/9/2015 Goodrich Decl., at ¶ 17.)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

Claim 1 of the ‘576 Patent recites (emphasis added):

1. An apparatus for rendering digital content in accordance with rights that are enforced by the apparatus, said apparatus comprising:
 - a rendering engine configured to render digital content;
 - a storage for storing the digital content;
 - means for requesting use of the digital content stored in the storage; and
 - a repository coupled to the rendering engine,wherein the repository includes:

means for processing a request from the means for requesting,
means for checking whether the request is for a permitted rendering of the digital content in accordance with rights specified in the apparatus,
means for processing the request to make the digital content available to the rendering engine for rendering when the request is for a permitted rendering of the digital [sic]; and
means for authorizing the repository for making the digital content available for rendering, wherein the digital content can be made available for rendering only by an authorized repository, the repository comprising:
means for making a re[qu]est for an authorization ob[j]ect required to be included within the repository for the apparatus to render the digital content; and
means for receiving the authorization ob[j]ect when it is determined that the request should be granted.

In instituting an IPR of the ‘576 Patent, the PTAB found as to this disputed term:

[T]he server repository transmits a block of data to the requester repository and waits for an acknowledgement, which the requester provides when the block of data has been completely received. (Ex. 1001, 33:9-15.) Unless a communications failure terminates the transaction, that process repeats until there are no more blocks to transmit. (Ex. 1001, 33:16-38, 33:46-49.) Finally, the requester repository sends a completion acknowledgement to the server repository. (Ex. 1001, 33:39-45.) In that regard, the specification states (Ex. 1001, 33:45-48): “The key property is that both the server and the requester cancel a transaction if it is interrupted before all of the data blocks are delivered, and commits to it if all of the data blocks have been delivered.” The above-noted description constitutes a sufficiently disclosed algorithm, expressed in prose, which, in combination with processor 1200, constitutes a corresponding structure for the claimed “means for receiving the authorization object when it is determined that the request should be granted.”

(Dkt. No. 331, Ex. 18, 7/9/2013 Decision (‘576 Patent), at 26.)

The PTAB’s construction is consistent with the context of the claim because the “means for receiving” is part of the rendering apparatus that is recited as requesting access to the content. The Court therefore affords “reasoned deference” to the PTAB Decision. *Maurice Mitchell*, 2006 WL 1751779, at *4; *see TQP*, 2014 WL 2810016, at *6; *see also Teva*, 135 S. Ct. at 839-40.

The Court accordingly hereby finds that for the “**means for receiving the authorization ob[j]ect when it is determined that the request should be granted,**” the function is “**receiving**

the authorization object when it is determined that the request should be granted,” and the corresponding structure is “a processor configured to perform the following algorithm and equivalents thereof: the server repository transmits a block of data to the requester repository and waits for an acknowledgement, which the requester provides when the block of data has been completely received; unless a communications failure terminates the transaction, that process repeats until there are no more blocks to transmit; finally, the requester repository sends a completion acknowledgement to the server repository; both the server and the requester cancel a transaction if it is interrupted before all of the data blocks are delivered, and commits to it if all of the data blocks have been delivered; and equivalents thereof.”

Z. “means for requesting a transfer of the digital content from an external memory to the storage”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“Corresponding Structure: user interface 1305 described at 16:35-44” ¹⁴	Indefinite

(Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at p. 15; Dkt. No. 331, at 29.) The parties submit that this disputed term appears in Claim 4 of the ‘576 Patent. (Dkt. No. 292-1, at 6-7; Dkt. No. 331, at 29.) “The PTAB did not construe this claim term.” (Dkt. No. 331, at 29.)

(1) The Parties’ Positions

Plaintiff’s expert opines: “A person of ordinary skill in the art would understand that a user interface is necessary for a user to request a transfer of the digital content from an external

¹⁴ Plaintiff previously proposed: “Corresponding Structure: algorithm/structure necessary for performing the recited function set forth in the ‘576 Spec., including from the following portions: 30:25-30, 30:59-31:49, 32:20-31, 36:35-54, 37:9-26, and equivalents thereof; see also, e.g., ‘576 PTAB Decision at 20-22.” (Dkt. No. 292-1, at 6-7.)

memory to the storage and clearly linked to that function.” (Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶ 34.)

Defendants argue that “[Plaintiff] fails to include an algorithm because there is none, thus the term is indefinite.” (Dkt. No. 331, at 29.)

Plaintiff replies that “[n]o algorithm is required because the specification discloses the precise structure (i.e., a user interface) used for performing the function.” (Dkt. No. 345, at 15.)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

“[A] means-plus-function claim element for which the only disclosed structure is a general purpose computer is invalid if the specification fails to disclose an algorithm for performing the claimed function.” *Net MoneyIN*, 545 F.3d at 1367; *see WMS Gaming*, 184 F.3d at 1349; *see also Noah*, 675 F.3d at 1319.

If an algorithm is required, that algorithm may be disclosed in any understandable form. *See Typhoon Touch Techs., Inc. v. Dell, Inc.*, 659 F.3d 1376, 1386 (Fed. Cir. 2011) (“Indeed, the mathematical algorithm of the programmer is not included in the specification. However, as precedent establishes, it suffices if the specification recites in prose the algorithm to be implemented by the programmer.”); *see also Finisar Corp. v. DirecTV Group, Inc.*, 523 F.3d 1323, 1340 (Fed. Cir. 2008) (noting that “a patentee [may] express th[e] algorithm in any understandable terms including as a mathematical formula, in prose, or as a flow chart, or in any other manner that provides sufficient structure”) (citation omitted); *TecSec, Inc. v. Int’l Bus. Machs.*, 731 F.3d 1336, 1348 (Fed. Cir. 2013) (quoting *Finisar*).

Nonetheless, the purported algorithm cannot “merely provide[] functional language” and must provide a “step-by-step procedure” for accomplishing the claimed function. *Ergo*

Licensing, 673 F.3d at 1365. Further, “[i]t is well settled that simply disclosing software, however, without providing some detail about the means to accomplish the function, is not enough.” Finally, when citing sections of the specification, a patentee should demonstrate “how these sections explain to one of ordinary skill in the art the manner in which the claimed functions are implemented.” *Personalized Media Commc’n, LLC v. Motorola, Inc.*, No. 2:08-CV-70-CE, 2011 WL 4591898, at *38 (E.D. Tex. Sept. 30, 2011); see *Function Media, L.L.C. v. Google, Inc.*, 708 F.3d 1310, 1318 (Fed. Cir. 2013).

Claim 4 of the ‘576 Patent recites: “4. The apparatus as recited in claim 1, further comprising means for requesting a transfer of the digital content from an external memory to the storage.”

The specification discloses:

Repository User Interface

A user interface is broadly defined as the mechanism by which a user interacts with a repository in order to invoke transactions to gain access to a digital work, or exercise usage rights. As described above, a repository may be embodied in various forms. The user interface for a repository will differ depending on the particular embodiment. The user interface may be a graphical user interface having icons representing the digital works and the various transactions that may be performed. The user interface may be a generated dialog in which a user is prompted for information.

The user interface itself need not be part of the repository. As a repository may be embedded in some other device, the user interface may merely be a part of the device in which the repository is embedded. For example, the repository could be embedded in a “card” that is inserted into an available slot in a computer system. The user interface may be combination of a display, keyboard, cursor control device and software executing on the computer system.

At a minimum, the user interface must permit a user to input information such as access requests and alpha numeric data and provide feedback as to transaction status. The user interface will then cause the repository to initiate the suitable transactions to service the request. Other facets of a particular user interface will depend on the functionality that a repository will provide.

'859 Patent at 15:44-16:2.

Defendants' expert, Dr. Grimes, opines that "[t]he 'user interface 1305, shown in Figure 13' consists of a rectangle containing the text 'User Interface 1305.' This is not an algorithm or even a block diagram, only a block." (Dkt. No. 331, Ex. 11, 12/22/2014 Grimes Decl. at ¶ 31.) Further, Dr. Grimes opines that the specification disclosure cited by Plaintiff, which is quoted above, "is not a description of an algorithm for the claimed function." (*Id.*, at ¶ 32.)

On balance, the disclosure set forth in the above-quoted "Repository User Interface" section of the specification is sufficient for performing the claimed function. The Court finds Plaintiff's expert's opinion credible in that regard. (Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶ 34.)

The Court accordingly hereby finds that for the **"means for requesting a transfer of the digital content from an external memory to the storage,"** the function is **"requesting a transfer of the digital content from an external memory to the storage,"** and the corresponding structure is **"user interface 1305, as described in the '576 Patent at 16:20-46; and equivalents thereof."**

AA. “means for processing the request to make the digital content available to the rendering engine for rendering when the request is for a permitted rendering of the digital [content],” “means for authorizing the repository for making the digital content available for rendering, wherein the digital content can be made available for rendering only by an authorized repository,” and “means for making a request for an authorization object required to be included within the repository for the apparatus to render the digital content”

“means for processing the request to make the digital content available to the rendering engine for rendering when the request is for a permitted rendering of the digital [content]”	
Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
<p>“Corresponding Structure: One or more processors that implement the following algorithm: First, transaction information is provided to the server repository by the requester repository. (33:3-8.) Second, the server repository transmits a block of data to the requester repository and waits for an acknowledgement provided by the requester when the block of data has been received. (33:9-15.) Unless a communications failure terminates the transaction, that process repeats until there are no more blocks to transmit. (33:16-38, 33:46-49.) Finally, the requester repository commits the transaction and sends an acknowledgement to the server repository. (33:39-45.)”</p>	<p>“One or more processors that implement the following algorithm: First, transaction information is provided to the server repository by the requester repository. Second, the server repository transmits a block of data to the requester repository and waits for an acknowledgement provided by the requester when the block of data has been received. Unless a communications failure terminates the transaction, that process repeats until there are no more blocks to transmit. Finally, the requester repository commits the transaction and sends an acknowledgement to the server repository.”</p>

“means for authorizing the repository for making the digital content available for rendering, wherein the digital content can be made available for rendering only by an authorized repository”	
Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
<p>“Corresponding Structure: One or more processors that implement the following algorithm: First, a communication channel is set up between the server and the remote repository. (41:52-54.) Second, the server repository performs a registration process with the remote repository. (41:55-57.) Third, the server repository invokes a “play” transaction to acquire the authorization object. (41:58-64.) Finally, the server repository performs tests on the authorization object or executes a script before signaling that authorization has been granted for rendering content. (41:65-42:16.)”</p>	<p>“One or more processors that implement the following algorithm: First, a communication channel is set up between the server and the remote repository. Second, the server repository performs a registration process with the remote repository. Third, the server repository invokes a “play” transaction to acquire the authorization object. Finally, the server repository performs tests on the authorization object or executes a script before signaling that authorization has been granted for rendering content.”</p>
“means for making a re[q]uest for an authorization ob[j]ect required to be included within the repository for the apparatus to render the digital content”	
Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
<p>“Corresponding Structure: One or more processors that implement the following algorithm: First, a communications channel is set up between the server repository and the remote repository. (41:52-54.) Second, the server repository performs a registration process with the remote repository. (41:55-57.) Third, the server repository invokes a “play” transaction to request the authorization object. (41:58-64.)”</p>	<p>“One or more processors that implement the following algorithm: First, a communications channel is set up between the server repository and the remote repository. Second, the server repository performs a registration process with the remote repository. Third, the server repository invokes a “play” transaction to request the authorization object.”</p>

(Dkt. No. 366, Ex. B, at 23-25.) These disputed terms appear in Claim 1 of the ‘576 Patent.

Defendants submit:

[T]o avoid jury confusion, the Court should decline [Plaintiff’s] invitation to include parenthetical citations in its constructions of “means for processing the

request to make the digital content available to the rendering engine for rendering when the request is for a permitted rendering of the digital [content]”; “means for authorizing the repository for making the digital content available for rendering”; and “means for making a request for an authorization object required to be included within the repository for the apparatus to render the digital content.” Outside of the parenthetical citations, the parties now agree on the substance of these three terms.

(Dkt. No. 331, at 27 n.15 (square brackets Defendants’).) At the February 6, 2015 hearing, the parties did not address these terms.

On balance, the Court agrees with Defendants that including the parentheticals proposed by Plaintiff is unnecessary in light of the parties having otherwise reached agreement as to the proper corresponding structure for these means-plus-function terms.

The Court accordingly hereby construes these disputed terms as set forth in the following chart:

<u>Term</u>	<u>Corresponding Structure</u>
“means for processing the request to make the digital content available to the rendering engine for rendering when the request is for a permitted rendering of the digital [content]”	“One or more processors that implement the following algorithm and equivalents thereof: First, transaction information is provided to the server repository by the requester repository. Second, the server repository transmits a block of data to the requester repository and waits for an acknowledgement provided by the requester when the block of data has been received. Unless a communications failure terminates the transaction, that process repeats until there are no more blocks to transmit. Finally, the requester repository commits the transaction and sends an acknowledgement to the server repository.”
“means for authorizing the repository for making the digital content available for rendering, wherein the digital content can be made available for rendering only by an authorized repository”	“One or more processors that implement the following algorithm and equivalents thereof: First, a communication channel is set up between the server and the remote repository. Second, the server repository performs a registration process with the remote repository. Third, the server repository invokes a “play”

	transaction to acquire the authorization object. Finally, the server repository performs tests on the authorization object or executes a script before signaling that authorization has been granted for rendering content.”
“means for making a re[q]uest for an authorization ob[j]ect required to be included within the repository for the apparatus to render the digital content”	“One or more processors that implement the following algorithm and equivalents thereof: First, a communications channel is set up between the server repository and the remote repository. Second, the server repository performs a registration process with the remote repository. Third, the server repository invokes a “play” transaction to request the authorization object.”

V. CONSTRUCTION OF DISPUTED TERMS IN THE NGUYEN PATENTS

The ‘280 Patent is titled “System and Method for Managing Transfer of Rights Using Shared State Variables.” The ‘280 Patent issued on August 10, 2010, and bears a priority date of June 7, 2001. The Abstract states:

A method, system and device for transferring rights adapted to be associated with items from a rights supplier to a rights consumer, including obtaining a set of rights associated with an item, the set of rights including meta-rights specifying derivable rights that can be derived from the meta- [sic, meta-right]; determining whether the rights consumer is entitled to the derivable rights specified by the meta-rights; and deriving at least one right from the derivable rights, if the rights consumer is entitled to the derivable rights specified by the meta-rights, wherein the derived right includes at least one state variable based on the set of rights and used for determining a state of the derived right.

The ‘053 Patent claims priority to a provisional application to which the ‘280 Patent also claims priority.

A. “repository”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“Same as in Stefik Patents, i.e.: a trusted system in that it maintains physical, communications, and behavioral integrity in the support of usage rights”	“a trusted system, which maintains physical, communications and behavioral integrity, and supports usage rights”

(Dkt. No. 292-1, at 12; Dkt. No. 331, at 29; Dkt. No. 366, Ex. B, at 46.) The parties submit that this disputed term appears in Claims 1 and 12 of the ‘280 Patent and Claims 1 and 15 of the ‘053 Patent. (Dkt. No. 292-1, at 12; Dkt. No. 331, at 29.)

“[Plaintiff] and Defendants agree that ‘repository’ should be construed the same way in the Stefik and Nguyen patents.” (Dkt. No. 331, at 30.)

For the same reasons set forth above as to the Stefik Patents, the Court hereby construes **“repository”** to mean **“a trusted system in that it maintains physical, communications, and behavioral integrity in the support of usage rights.”**

B. “license”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“data embodying a grant of usage rights and/or meta-rights”	“a data structure containing both a usage right and meta-right”

(Dkt. No. 304, at 21; Dkt. No. 334, at 34.) The parties submit that this disputed term appears in Claims 11 and 22 of the ‘280 Patent and Claims 1, 3, 4, 5, 15, and 23 of the ‘053 Patent. (Dkt. No. 292-1, at 11; Dkt. No. 331, at 34.)¹⁵

¹⁵ Defendants submit in their response brief, evidently erroneously, that this disputed term appears also in Claim 12 of the ‘280 Patent. (Dkt. No. 331, at 34.)

(1) The Parties' Positions

Plaintiff submits that “the specification repeatedly uses ‘and/or’ language to make clear that a ‘license’ can embody usage rights, meta-rights, or both.” (Dkt. No. 304, at 21.) Plaintiff also argues that Defendants’ proposal of a “data structure” is based on a portion of the specification but, “[a]mong other issues, . . . is incomplete.” (*Id.*, at 21-22.)

Defendants respond that the Nguyen Patents “describe licenses not as mere permissions or authorizations, but as particular data constructs” (Dkt. No. 331, at 34-35 (citing ‘280 Patent at 4:7-14; citing ‘053 Patent at 4:40-47).)

Plaintiff replies that “the patent specification repeatedly uses ‘and/or’ language to teach that a license can embody usage rights, meta-rights, or both” (Dkt. No. 345, at 15.) Further, Plaintiff submits, “Defendants also fail to explain why the Court should limit the term license to a ‘data structure’ when the patent teaches that rights can also be expressed in ‘symbols, elements, or sets of rules.’” (*Id.*, at 15 n.17.)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

The specification discloses:

License 52 includes the appropriate rights, such as usage rights *and/or* meta-rights, and can be downloaded from license server 50 or an associated device.

‘280 Patent at 5:13-16 (emphasis added).

As shown in FIG. 10, license 52 includes license 52a, grant 52b, and digital signature 52c. Grant 52b includes granted usage rights *and/or* meta-rights selected from label. The structure of the grant also includes one or more principals, to whom the specified usage rights and/or meta-rights are granted, a list of conditions, and state variables required to enforce the license. Like usage rights, access and exercise of the granted meta-rights are controlled by the condition list and state variables as described below.

'053 Patent at 4:59-64 (emphasis omitted); *see id.* at 10:15-16 (“[G]rant 52b of license 52 can include usage rights *and/or* meta-rights.”) (emphasis added); *see also id.* at 6:66-7:2 (“An ‘offer of rights’ or ‘rights offer’ expresses how a consumer (e.g. a content distributor or user) can acquire a particular instance of content together with its associated usage rights *and/or* meta-rights.”) (emphasis added).

The specification also discloses “licenses” both with and without reference to a meta-right:

These licenses embody the actual granting of usage rights to an end user. For example, rights label 40 may include usage rights permitting a recipient to view content for a fee of five dollars and view and print content for a fee of ten dollars. License 52 can be issued for the view right when the five dollar fee has been paid, for example. Client component 60 interprets and enforces the rights that have been specified in license 52.

'280 Patent at 4:7-14.

FIG. 4 is an example of license 52 encoded in XrML™. The provider grants the distributor a meta right to issue a usage right (i.e., play) to the content (i.e., a book) to any end user. With this meta right, the distributor may issue the right to play the book within the U.S. region and subject to some additional conditions that the distributor may impose upon the user, as long as the distributor pays \$1 to the provider each time the distributor issues a license for an end user.

Id. at 8:17-24.

Licenses 52 embody the actual granting of rights, including usage rights and meta-rights, to an end user. For example, rights offer 40 may permit a user to view content for a fee of five dollars and print content for a fee of ten dollars, or it may permit a user to offer rights to another user, for example, by utilizing the concept of meta-rights described below. License 52 can be issued for the view right when the five dollar fee has been paid. Client component 60 interprets and enforces the rights, including usage rights and meta-rights, that have been specified in the license.

'053 Patent at 4:40-50 (emphasis added).

Plaintiff argues that “[t]he plain import of this sentence [(“Licenses 52 embody the actual granting of rights, including usage rights and meta-rights, to an end user.”)] is that the scope of

the word ‘rights’ in licenses generally can encompass both usage rights and meta-rights. It does not follow, however, that every license specifically must contain both types of rights.” (Dkt. No. 345, at 15.)

On balance, the above-quoted disclosures demonstrate that Defendants’ proposal of “usage right *and* meta-right” should be rejected.

As to whether a “license” must be a data structure, as Defendants have proposed, the specification discloses:

Meta-rights can be expressed by use of a grammar or rights language including data structures, symbols, elements, or sets of rules.

‘280 Patent at 7:40-42; *see id.* at 4:7-14; *see also id.* at 4:40-48. Plaintiff urges that because Defendants’ proposed construction “ignores the reference to ‘symbols,’ ‘elements’ and ‘sets of rules, . . . [Plaintiff’s] construction of ‘data embodying a grant’ is more appropriate. (Dkt. No. 304, at 22.)

On balance, Defendants’ proposal of “data structure” lacks sufficient support in the intrinsic evidence and would tend to confuse rather than clarify the scope of the claims.

The Court accordingly hereby construes “**license**” to mean “**data embodying a grant of usage rights and/or meta-rights.**”

C. “meta-right”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“a right that when exercised creates or disposes of usage rights or other meta[-]rights but is not itself a usage right, i.e., actions to content do not result from exercising meta-rights” ¹⁶	“a data structure that is used by a repository to create or dispose of ‘usage rights’ or other meta-rights relating to an item of content and is distinct from a usage right associated with the item of content”

¹⁶ Plaintiff previously proposed: “a right that when exercised creates or disposes of usage rights or other meta-rights but is not itself a usage right.” (Dkt. No. 304, at 22.)

(Dkt. No. 304, at 22; Dkt. No. 331, at 31; Dkt. No. 366, Ex. B, at 45.) The parties submit that these disputed terms appear in Claims 1, 11, 12, and 22 of the ‘280 Patent and Claims 1, 3, 4, and 15 of the ‘053 Patent. (Dkt. No. 292-1, at 11; Dkt. No. 331, at 31.)

(1) The Parties’ Positions

Plaintiff argues that whereas its proposed construction is “based on an express definition in the specification,” Defendants “seek to graft additional language onto this definition that is nowhere found in the specification and which obscures rather than clarifies the distinction between meta-rights and usage rights.” (Dkt. No. 304, at 22.) Plaintiff explains that “in contrast to usage rights which control access to content, meta-rights control and manage usage rights (or other meta-rights).” (*Id.*, at 23 (citing ‘280 Patent at 7:23-31).) Finally, Plaintiff argues that “it is unclear whether Defendants’ construction means: (a) that a meta-right is not itself a usage right (in which case [Plaintiff’s] construction is a clearer way of saying that); (b) that a meta-right *can* be a usage right so long as it is distinct from at least one other usage right associated with the same content (which would be at odds with the distinctions recited in the specification); or (c) something else entirely.” (Dkt. No. 304, at 24.)

Defendants respond that “[m]eta-rights’ in the Nguyen patents are the rights to grant ‘usage rights’ (or additional meta-rights) to others.” (Dkt. No. 331, at 31.) Defendants submit that their proposed construction “avoids reciting pure legal concepts and captures the three distinguishing characteristics of ‘meta-rights’ in the Nguyen patents: ‘meta-rights’ are (1) formed as a data construct associated with a particular digital work; (2) used by a repository to create or dispose of usage rights or other meta-rights; and (3) distinct from a usage right.” (*Id.*, at 31-32.) Defendants emphasize that “[e]very . . . example in the specifications . . . presents meta-rights as data.” (*Id.*, at 32.)

Plaintiff replies that including Defendants' "data structure" and "repository" limitations would "create[] nonsensical redundancy." (Dkt. No. 345, at 16.) Plaintiff also argues that its proposed construction makes clear that "a meta-right can be exercised independent of an action to content." (*Id.*)

At the February 6, 2015 hearing, Defendants argued that Plaintiff's proposed "i.e." phrase is confusing and should be omitted.

(2) Analysis

Claim 1 of the '280 Patent, for example, recites (emphasis added):

1. A computer-implemented method for transferring rights adapted to be associated with items from a rights supplier to a rights consumer, the method comprising:

obtaining a set of rights associated with an item, the set of rights including *a meta-right specifying a right that can be created when the meta-right is exercised, wherein the meta-right is provided in digital form and is enforceable by a repository;*

determining, by a repository, whether the rights consumer is entitled to the right specified by the *meta-right*; and

exercising the *meta-right* to create the right specified by the *meta-right* if the rights consumer is entitled to the right specified by the *meta-right*, wherein the created right includes at least one state variable based on the set of rights and used for determining a state of the created right.

The specification discloses:

Meta-rights are the rights that one has to generate, manipulate, modify, dispose of or otherwise derive other rights. Meta-rights can be thought of as usage rights to usage rights (or other meta-rights). * * * Meta-rights can be hierarchical and can be structured as objects within objects.

'280 Patent at 5:47-50 & 5:60-62; *see* '053 Patent at 5:22-23 (similar).

At a high level the process of enforcing and exercising meta-rights are [*sic*, is] the same as for usage rights. However, the difference between usage rights and meta-rights are [*sic*, is] the result from exercising the rights. *When exercising usage rights, actions to content result.* For example usage rights can be for viewing, printing, or copying digital content. *When meta-rights are exercised, new rights are created from the meta-rights or existing rights are disposed as the result of exercising the meta-rights.* The recipient of the new rights may be the same

principal (same person, entity, or machine, etc), who exercises the meta-rights. Alternatively, the recipient of meta-rights can be a new principal. The principals who receive the derived rights may be authenticated and authorized before receiving/storing the derived rights. Thus, the mechanism for exercising and enforcing a meta-right can be the same as that for a usage right. . . .

Meta-rights can be expressed by use of a grammar or rights language including data structures, symbols, elements, or sets of rules. For example, the XrML™ rights language can be used. As illustrated in FIG. 3, the structure of license 52 can consist of one or more grants 300 and one or more digital signatures 310.

‘280 Patent at 7:23-45 (emphasis added).

In FIG. 9, rights 902 and 903 derived from an offer 901 are exclusive to each respective consumer. The offer 901 is a type of *meta-right* of which the recipients have the rights to obtain specific derivative rights when the conditions for obtaining such rights are satisfied.

‘280 Patent at 11:54-58 (emphasis added); *see* ‘280 Patent at Figs. 9-16; *see also* ‘053 Patent at Figs. 13-20 (same).

Plaintiff also submits that in an appeal during prosecution of United States Patent No. 7,774,279, which claims priority to one of the same applications to which the Nguyen Patents claim priority, the Board of Patent Appeals and Interferences found that “the Specification provides an express definition of ‘meta-rights.’ . . . The definition for ‘meta-rights’ given in the Specification governs the construction to be given that term in the claims.” (Dkt. No. 304, Ex. V, 12/16/2009 Decision on Appeal, at 6.)

Plaintiff also submits that in a recent Petition for Covered Business Method Patent Review at the PTO, Defendant Google Inc. argued that “the broadest reasonable construction of a ‘meta-right’ is ‘a right about a right.’” (Dkt. No. 345, Ex. AJ, at 27.)

On balance, Defendants’ proposal of “data structure” lacks sufficient support in the intrinsic evidence and would tend to confuse rather than clarify the scope of the claims. The parties are otherwise essentially in substantive agreement, and the above-discussed evidence

demonstrates that a “meta-right” is a right governing another right rather than governing any action to content.

The Court accordingly hereby construes **“meta-right”** to mean **“a right that, when exercised, creates or disposes of usage rights (or other meta-rights) but that is not itself a usage right because exercising a meta-right does not result in action to content.”**

D. “usage rights”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
Same as in Stefik patents, i.e., “an indication of the manner in which a [digital work / digital content / content / a digital document] may be used or distributed as well as any conditions on which use or distribution is premised”	“statements in a language for defining the manner in which a digital work may be used or distributed, as well as any conditions on which use or distribution is premised. Usage rights must be permanently attached to the digital work” ¹⁷

(Dkt. No. 304, at 23; Dkt. No. 331 at 33; Dkt. No. 366, Ex. B, at 50.) The parties submit that this disputed term appears in Claims 1, 3, 4, and 15 of the ‘053 Patent. (Dkt. No. 292-1, at 12; Dkt. No. 331, at 33.)¹⁸

(1) The Parties’ Positions

Plaintiff argues: “Defendants’ scattershot approach will result in jury confusion and does not reflect how a person of skill in the art would understand these patents. From beginning to end, the specification confirms that ‘usage rights’ should be construed the same in both the meta-rights patents and the Stefik patents.” (Dkt. No. 304, at 24.)

¹⁷ Defendants previously proposed: “A data structure that persists with digital content and that defines the manner in which the content may be used or distributed, as well as any conditions on which use or distribution is premised.” (Dkt. No. 292-2, at 21.)

¹⁸ Plaintiff submitted in its P.R. 4-3 statement, evidently erroneously, that this disputed term appears also in Claims 5 and 23 of the ‘053 Patent. (Dkt. No. 292-1, at 12.)

Defendants respond “[t]he parties agree that the Court should construe ‘usage rights’ in the Nguyen ‘053 patent in the same way the Court construes that term for the majority of the Stefik patents because the ‘053 patent incorporates four Stefik patents by reference. . . . The parties disagree about which Stefik patents to base this construction on.” (Dkt. No. 331, at 33.) Defendants emphasize that whereas the ‘053 Patent incorporates-by-reference several Stefik Patents, the Stefik ‘160 Patent relied upon by Plaintiff is not incorporated by the ‘053 Patent. (*Id.*)

Plaintiff replies that having a different construction for the Nguyen Patents “would result in unnecessary jury confusion.” (Dkt. No. 345, at 17.)

(2) Analysis

The specification discloses:

A predetermined set of usage transaction steps define a protocol used by the repositories for enforcing usage rights associated with a document. *Usage rights persist with the document content.* The usage rights can permit various manners of use such as, viewing only, use once, distribution, and the like. Usage rights can be contingent on payment or other conditions.

* * *

The interpretation and enforcement of usage rights are well known generally. ‘053 Patent at 2:39-45 & 6:23-25 (emphasis added); *see id.* at 15:25-27 (“Access to the various documents, and elements thereof, can be controlled using known techniques.”)

For substantially the same reasons discussed above as to the Stefik Patents, the Court reaches the same construction for the Nguyen Patents as for the Stefik Patents.

The Court accordingly hereby construes **“usage rights”** to mean **“indications that are attached, or treated as attached, to [a digital work / digital content / content / a digital document] and that indicate the manner in which the [digital work / digital content /**

content / digital document] may be used or distributed as well as any conditions on which use or distribution is premised.”

E. “manner of use”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
Same as in Stefik patents, i.e., “a way in which [a digital work / digital content / content / a digital document] may be used”	“a defined way of using or distributing a digital work (for example, PLAY, COPY, or PRINT), as distinct from conditions which must be satisfied before that way of using or distributing the digital work is allowed” [Same as in Stefik Patents]

(Dkt. No. 304, at 23; Dkt. No. 331 at 31; Dkt. No. 366, Ex. B, at 50.) The parties submit that this disputed term appears in Claims 1 and 15 of the ‘053 Patent. (Dkt. No. 292-1, at 12; Dkt. No. 331, at 31.)

“Both sides propose their respective constructions of ‘manner(s) of use’ for both the Stefik and Nguyen patents.” (Dkt. No. 331, at 31.)

Defendants argue that “[j]ust like the Stefik patents, the ‘053 patent differentiates manners of use from conditions on use or distribution.” (Dkt. No. 331, at 31 (citing ‘053 Patent at 1:58-64 & 2:42-44).) “Essentially,” Defendants submit, “authorization determines whether a user is allowed to access digital content, while usage rights define particular manners in which the authorized content can be used.” (*Id.*)

Plaintiff’s reply brief does not specifically address this disputed term. (*See* Dkt. No. 345, at 16-17.)

The specification discloses:

The usage rights can permit various manners of use such as, viewing only, use once, distribution, and the like. Usage rights can be contingent on payment or other conditions.

'280 Patent at 2:16-19.

Because the parties appear to agree that this disputed term should be given the same construction in the Nguyen Patents as in the Stefik Patents, the Court hereby construes **“manner of use”** to mean **“a way in which [a digital work / digital content / content / a digital document] may be used, as contrasted with a condition associated with such use.”**

F. “state variable”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“a variable having a value, or identifying a location at which a value is stored, that represents status of an item, rights, license, or other potentially dynamic conditions”	“a variable having a value that represents the status of usage rights, license, or other dynamic conditions”

(Dkt. No. 304, at 25; Dkt. No. 331, at 34.) The parties submit that this disputed term appears in Claims 1, 5, and 12 of the '280 Patent and Claims 1, 4, 5, 15, and 23 of the '053 Patent. (Dkt. No. 292-1, at 13; Dkt. No. 331, at 34.)

(1) The Parties’ Positions

Plaintiff argues that the specification and the prosecution history explain that a “state variable” can identify location information. (Dkt. No. 304, at 25-26.)

Defendants respond that the Nguyen Patents expressly define “state variables” in the manner Defendants have proposed. (Dkt. No. 331, at 34.) Defendants also argue that “[Plaintiff] does not cite any part of the patents to support its construction” and, “[a]s for [Plaintiff]’s prosecution history argument, new subject matter added to the '053 patent by amendment cannot be used to expand the scope of this term.” (*Id.*) Finally, Defendants urge that “[c]onditions are either dynamic or not. [Plaintiff]’s proposal of [p]otentially’ turns the claim inside out by

allowing it to encompass conditions that are not dynamic and might never become dynamic.”

(*Id.*)

Plaintiff replies that based on Figure 15 and accompanying description, “there can be no dispute that the specification amply discloses the use of a state variable to identify a location on a remote server.” (Dkt. No. 345, at 17.) Plaintiff also emphasizes disclosure of “track[ing] *potentially* dynamic states [*sic*] conditions.” (*Id.*, at 18 (quoting ‘053 Patent at 5:42-44) (emphasis Plaintiff’s).)

At the February 6, 2015 hearing, Defendants argued that Plaintiff’s proposal of referring to a location cannot be correct because a state variable is something that holds a value, not a location where a value might be found. Defendants also argued that the illustrations of a “state variable id” (in the Figures of the Nguyen Patents) are references to the *names* of state variables, not the *values* of those variables. Defendants concluded that the illustration of “state variable id = www.foo.edu” does *not* demonstrate that a state variable can identify a location.

(2) Analysis

Claim 1 of the ‘053 Patent recites (emphasis added):

1. A method for sharing rights adapted to be associated with an item, the method comprising:
 - specifying, in a first license, using a processor, at least one usage right and at least one meta-right for the item, wherein the usage right and the meta-right include at least one right that is shared among one or more users or devices;
 - defining, via the at least one usage right, using a processor, a manner of use selected from a plurality of permitted manners of use for the item;
 - defining, via the at least one meta-right, using a processor, a manner of rights creation for the item, wherein said at least one meta-right is enforceable by a repository and allows said one or more users or devices to create new rights;
 - associating, using a processor, at least one state variable with the at least one right in the first license, *wherein the at least one state variable identifies a location where a state of rights is tracked*;
 - generating, in a second license, using a processor, one or more rights based on the meta-right in the first license, wherein the one or more rights in the

second license includes at least one right that is shared among one or more users or devices; and

associating at least one state variable with the at least one right that is shared in the second license, wherein the at least one state variable that is associated with the second license is based on the at least one state variable that is associated with the first license.

This claim itself thus contemplates that a “state variable” can identify a location at which a value is stored.

The specification discloses:

State variables track *potentially dynamic* states [*sic*] conditions. *State variables are variables having values that represent status of an item, usage rights, license or other dynamic conditions.* State variables can be tracked, by clearinghouse 90 license or server 30 another device [*sic*], based on identification mechanisms in license 52. Further, the value of state variables can be used in a condition. For example, a usage right can be the right to print content 42 three times. Each time the usage right is exercised, the value of the state variable “number of prints” is incremented. In this example, when the value of the state variable is three, the condition is no[] longer satisfied and content 42 cannot be printed. Another example of a state variable is time. A condition of license 52 may require that content 42 is printed within thirty days. A state variable can be used to track the expiration of thirty days. Further, the state of a right can be tracked as a collection of state variables. The collection of the change is [*sic, in*] the state of a usage right represents the usage history of that right.

‘053 Patent at 5:42-59 (emphasis added); *see* ‘280 Patent at 7:66-8:12 (similar).

In light of this disclosure of “potentially” dynamic conditions, Plaintiff’s proposal of “potentially” is appropriate.

As to Plaintiff’s proposal of “identifying a location at which a value is stored,” the specification discloses:

There are multiple ways to specify the scope of state variables, each of which can affect whether the derivative state variables can be shared, how the derivative state variables can be shared, and the like. For example, a state variable can be local, and solely confined to a recipient or can be global, and shared by a predetermined group of recipients. *A global state variable can be shared by a group of recipients* not determined when derived rights are issued, but to be specified later, perhaps based on certain rules defined in the license or based on other means. A global state variable can be shared between one or more rights

suppliers, predetermined recipients, un-specified recipients, and the like. Advantageously, depending on the sharing employed with a given a [sic] business model and the rights granted in the meta-rights, state variables can be created at different stages of the value chain.

‘053 Patent at 17:4-18 (emphasis added); *see* ‘280 Patent at 11:29-43 (same); *see also id.* at 11:25-28 & 12:19-21. Also of note, Figure 17 of the ‘280 Patent discloses “state variable id = www.foo.edu,” which a person of ordinary skill in the art would understand as an Internet location. (Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶ 68; Dkt. No. 345, Ex. AA, 1/9/2015 Goodrich Decl., at ¶ 43.)

Plaintiff also submits that during prosecution of the ‘053 Patent, the patentee stated that “*a state variable referring to a location on a server can be used to infer that the right is shared among multiple devices.*” (Dkt. No. 397, Ex. W, Amendment After Non-Final Rejection at 13 (emphasis added); *see id.* at 13-14 (“a state variable is not ‘the number of copies’ or ‘rental terms,’ as asserted by the present Office Action, but rather references, for example, a counter or variable where ‘the number of copies’ or ‘rental terms’ is maintained, and wherein such a counter or variable can be located on a local device or a remote server. The ability to choose the location of a state keeper instead of a specific number, advantageously, provides a mechanism for the rights owner to control rights sharing.”).)¹⁹

¹⁹ At the February 6, 2015 hearing, the Court requested that Plaintiff file a more complete version of the excerpted Exhibit W attached to Plaintiff’s opening brief. Plaintiff did so. (Dkt. No. 397.) Defendants have responded by submitted a subsequent office action. (*See* Dkt. No. 399, Ex. A, 5/21/2007 Office Action.) Defendants explain that submission of this subsequent office action is necessary for “completeness.” (Dkt. No. 399 at 1.) Defendants have not argued that they were not aware of the content or significance of Plaintiff’s Exhibit W. Indeed, at the February 6, 2015 hearing, Defendants’ counsel offered immediately to hand up paper copies of a complete version of the document that Plaintiff had submitted in excerpted form as Exhibit W. Defendants had an opportunity to respond to Exhibit W and submit additional exhibits at the time Defendants filed their responsive claim construction brief. Thus, Defendants’ submission (Dkt. No. 399) is untimely, and the Court does not consider it.

Finally, Plaintiff's expert, Dr. Goodrich, has opined:

To someone of ordinary skill in the art the concept that a variable can store a location where something is tracked is well understood. For example, the computer language C includes the concept of a variable storing the address for another variable and this address can be used to read or write the value of the other variable. Likewise, at the time of the '053 patent it was well understood that objects on the Internet could be referenced by a Uniform Resource Identifier (URI), such as a Uniform Resource Locator (URL) which is standard format for specifying addresses of objects on the Internet, or a Uniform Resource Name (URN), which identifies a resource on a network using a unique name. * * * It would be well known to someone with ordinary skill in the art . . . that variables can be of many different types and the same is true of state variables as taught in the patent.

(Dkt. No. 345, Ex. AA, 1/9/2015 Goodrich Decl., at ¶ 43.)

Defendants' expert, Dr. Grimes, responds that "[i]n the context of claims and the intrinsic evidence, a person of ordinary skill in the art would not understand whether the state variable stores 1) an address or file in which a value is stored, 2) the identity of a group/organization for which a state variable is tracked, or 3) the right itself." (Dkt. No. 331, Ex. 11, 12/22/2014 Grimes Decl., at ¶ 73.) Further, as to the above-mentioned disclosure of a web address, Dr. Grimes responds that "the specification never discloses what, if anything, is maintained at that location." (*Id.*)

On balance, in light of the above-discussed intrinsic evidence, the opinions of Plaintiff's expert are more credible as to how a person of ordinary skill in the art would understand a "state variable" with regard to a location. Defendants' expert's opinions regarding purported ambiguity are unpersuasive, particularly given the above-quoted prosecution history and the above-quoted disclosure that state variables can be shared. *See* '280 Patent at 11:29-43; *see also id.* at 11:25-28 ("[A] shared state variable can include a data variable that is updated in response to actions by a plurality of users and which is globally applied to each of the users.") & 12:19-21 ("[A] shared state can be managed by an entity that is accessible by all sharing principals.").

The Court accordingly hereby construes “**state variable**” to mean “**a variable having a value, or identifying a location at which a value is stored, that represents status of an item, rights, license, or other potentially dynamic conditions.**”

G. “the at least one state variable identifies a location where a state of rights is tracked”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
Ordinary and customary meaning	Indefinite

(Dkt. No. 304, at 25; Dkt. No. 331, at 36). The parties submit that this disputed term appears in Claims 1 and 15 of the ‘053 Patent. (Dkt. No. 292-1, at 14; Dkt. No. 331, at 36.)

(1) The Parties’ Positions

Plaintiff’s opening brief does not address this term separately from the term “state variable,” which is addressed above. (*See* Dkt. No. 304, at 25-26.)

Defendants respond that “One of ordinary skill in the art cannot determine with reasonable certainty whether the ‘state variable’ is and/or stores (1) an address or file in which another value is stored; (2) the identity of a group/organization for which a state variable is tracked; or (3) the ‘state of rights’ itself.” (Dkt. No. 331, at 36.)

Plaintiff’s reply brief does not address this term separately from the term “state variable,” which is addressed above. (*See* Dkt. No. 345, at 17-18.)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

Defendants submit that the specification refers to “state variables,” a “state variable identification,” and a “state variable id” interchangeably. *See, e.g.*, ‘053 Patent at 4:64, 20:2 & Fig. 17. The specification illustrates the state variable “id” as being a number (Fig. 17 (“40”)), a right (Fig. 13 (“AlicePlayEbook”)), a priority (Fig. 18 (“priority_1”)), or an organization (Fig. 16

(“um.foo.club”). Defendants submit that “[t]he specification never explains if these values or elements are themselves ‘state variables’ or merely ‘identification[s]’ of the state variables.”

(Dkt. No. 331, at 36 (square brackets Defendants’).)

Further, Defendants argue, “though the specification alludes to a state variable storing a location (such as a Web site address), the specification never discloses what is maintained at that location.” (*Id.* (citing ‘053 Patent at 18:14-21 & Fig. 15).) Finally, Defendants submit that the prosecution history of the ‘053 Patent fails to clarify the meaning by stating that “a state variable referring to a location on a server can be used to infer that the right is shared among multiple devices” (Dkt. No. 331, Ex. 23, 12/31/2008 Reply Brief, at 6.)

For substantially the same reasons set forth above as to the term “state variable,” in particular as to disclosures regarding sharing of state variables, Defendants’ arguments are unpersuasive.

The Court therefore hereby expressly rejects Defendants’ indefiniteness argument. No further construction is necessary. *See U.S. Surgical*, 103 F.3d at 1568; *see also O2 Micro*, 521 F.3d at 1362; *Finjan*, 626 F.3d at 1207.

The Court accordingly hereby construes “**the at least one state variable identifies a location where a state of rights is tracked**” to have its **plain meaning** apart from the construction of the term “state variable,” above.

H. “specifying, in a first license, . . . at least one usage right and at least one meta-right for the item, wherein the usage right and the meta-right include at least one right that is shared among one or more users or devices”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“specifying in a first license, at least one usage right and at least one meta-right for the item, wherein at least one of the meta-right or the usage right is shared among one or more users or devices”	Indefinite

(Dkt. No. 304, at 27; Dkt. No. 331, at 35.) Defendants submit that this disputed term appears in Claims 1 and 15 of the ‘053 Patent. (Dkt. No. 331, at 35.)

(1) The Parties’ Positions

Plaintiff argues that “the surrounding claim language and the specification make clear exactly what the claim covers. In particular, the requirement that a license ‘include’ at last one shared right merely means that at least one right specified in the license be shared with other users or devices.” (Dkt. No. 304, at 27.)

Defendants respond that “given that [Plaintiff] admits that a meta-right ‘is not itself a usage right,’ it is nonsensical that a meta-right and a usage right could possibly both include a ‘shared’ right.” (Dkt. No. 331, at 35 (citing Dkt. No. 304, at 23).) Defendants emphasize that the disputed term itself recites “the usage right *and* the meta-right,” *not* “the usage right *or* the meta-right.” (Dkt. No. 331, at 35.)

Plaintiff replies that “[t]he claim language simply requires that the first license specify a set of rights (i.e., at least one usage right and at least one meta-right), where this set includes at least one right that is shared. This limitation is satisfied so long as at least one member of the set is shared.” (Dkt. No. 345, at 18.) Plaintiff also highlights Figure 15 and the accompanying written description. (*Id.*)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

The specification discloses:

When a usage right is to be shared among a predetermined set of recipients, a state variable for tracking a corresponding usage right can be specified in a meta-right using a same state variable identification for all recipients. During a process of exercising the meta-right, the same state variable identification is included in every derived right.

FIG. 15 illustrates the use of state variable in deriving rights that are shared among a known set of rights recipients, according to the present invention.

‘053 Patent at 18:8-16; *see id.* at 18:17-24.

Defendants’ expert, Dr. Grimes, opines that “[t]he specification fails to explain how either the usage right or the meta-right ‘includes’ a shared right[,] and one of ordinary skill in the art would not understand what this means without further explanation.” (Dkt. No. 331, Ex. 11, 12/22/2014 Grimes Decl., at ¶ 76.)

Plaintiff’s expert, Dr. Goodrich, replies that “in light of the specification, it is clear that the claim language does not require that the usage right and the meta-right must each contain another right that is shared, but rather means that the usage right and meta-right comprise a set whereby that set includes at least one right (either the meta-right or the usage right) that is shared.” (Dkt. No. 345, Ex. AA, 1/9/2015 Goodrich Decl., at ¶ 45.)

Claim 1 of the ‘530 Patent, for example, recites (emphasis added):

1. A method for sharing rights adapted to be associated with an item, the method comprising:

specifying, in a first license, using a processor, at least one usage right and at least one meta-right for the item, wherein the usage right and the meta-right include at least one right that is shared among one or more users or devices;

defining, via the at least one usage right, using a processor, a manner of use selected from a plurality of permitted manners of use for the item;

defining, via the at least one meta-right, using a processor, a manner of rights creation for the item, wherein said at least one meta-right is enforceable by

a repository and allows said one or more users or devices to create new rights;
 associating, using a processor, at least one state variable with the at least one right in the first license, wherein the at least one state variable identifies a location where a state of rights is tracked;
 generating, in a second license, using a processor, one or more rights based on the meta-right in the first license, wherein the one or more rights in the second license includes at least one right that is shared among one or more users or devices; and
 associating at least one state variable with the at least one right that is shared in the second license, wherein the at least one state variable that is associated with the second license is based on the at least one state variable that is associated with the first license.

Consistent with Plaintiff’s proposed construction, the language of the disputed term itself refers to “at least one right” from among “the usage right and the meta-right.” Plaintiff’s expert’s opinion supports such a reading of the plain language of the claims and is persuasive. The Court therefore adopts Plaintiff’s proposed construction so as to aid clarity, and the Court hereby expressly rejects Defendants’ indefiniteness argument.

The Court accordingly hereby construes **“specifying, in a first license, . . . at least one usage right and at least one meta-right for the item, wherein the usage right and the meta-right include at least one right that is shared among one or more users or devices”** to mean **“specifying in a first license, at least one usage right and at least one meta-right for the item, wherein at least one of the meta-right or the usage right is shared among one or more users or devices.”**

I. “means for obtaining a set of rights associated with an item”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“Corresponding Structure: a client environment 30 capable of connecting to a web server 80 and storing a license 52 including meta-rights and/or usage rights in a license repository 818, which can be interpreted by a license interpreter 802 (4:67-5:13; 10:44-45) ^{20,,21}	Indefinite

(Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at p. 35; Dkt. No. 331, at 36; Dkt. No. 366, Ex. B, at 47.) The parties submit that this disputed term appears in Claim 12 of the ‘280 Patent. (Dkt. No. 292-1, at 14; Dkt. No. 331, at 36.)

(1) The Parties’ Positions

Plaintiff’s expert opines: “A person of ordinary skill in the art would understand that the ability to connect to a web server and to store and interpret licenses embodying a grant of rights is necessary to obtain a set of rights associated with an item, and clearly linked to that function.” (Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶ 69.)

Defendants argue that the specification “does not identify a specific structure [such as an algorithm] that ‘obtains a set of rights’” but instead “provides only a conceptual description in which rights, content, or both are sent from a content provider to a recipient.” (Dkt. No. 331, at 37 (citing ’280 Patent at 6:27-31).)

Plaintiff’s reply brief does not address this term. (*See* Dkt. No. 345, at 15-18.)

²⁰ This parenthetical does not appear in the parties’ January 23, 2015 Joint Claim Construction Chart. (*See* Dkt. No. 366, Ex. B, at 47.)

²¹ Plaintiff previously proposed: “Corresponding Structure: algorithm/structure necessary for performing the recited function set forth in the ’280 Spec., including from the following portions: Figs. 1, 5, 8 and 4:66-5:16, 5:18-21, 6:10-17, 8:31-35, 9:54-58, 10:37-45, and equivalents thereof.” (Dkt. No. 292-1, at 14.)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

“[A] means-plus-function claim element for which the only disclosed structure is a general purpose computer is invalid if the specification fails to disclose an algorithm for performing the claimed function.” *Net MoneyIN*, 545 F.3d at 1367; *see WMS Gaming*, 184 F.3d at 1349; *see also Noah*, 675 F.3d at 1319.

If an algorithm is required, that algorithm may be disclosed in any understandable form. *See Typhoon Touch*, 659 F.3d at 1386 (“Indeed, the mathematical algorithm of the programmer is not included in the specification. However, as precedent establishes, it suffices if the specification recites in prose the algorithm to be implemented by the programmer.”); *see also Finisar*, 523 F.3d at 1340; *TecSec*, 731 F.3d at 1348.

Nonetheless, the purported algorithm cannot “merely provide[] functional language” and must provide a “step-by-step procedure” for accomplishing the claimed function. *Ergo Licensing*, 673 F.3d at 1365. Further, “[i]t is well settled that simply disclosing software, however, without providing some detail about the means to accomplish the function, is not enough.” *Function Media*, 708 F.3d at 1318 (citation and internal quotations and alterations omitted).

Claim 12 of the ‘280 Patent recites (emphasis added):

12. A system for transferring rights adapted to be associated with items from a rights supplier to a rights consumer, the system comprising:
- means for obtaining a set of rights associated with an item*, the set of rights including a meta-right specifying a right that can be created when the meta-right is exercised, wherein the meta-right is provided in digital form and is enforceable by a repository;
 - means for determining whether the rights consumer is entitled to the right specified by the meta-right; and
 - means for exercising the meta-right to create the right specified by the meta-right if the rights consumer is entitled to the right specified by the meta-

right, wherein the created right includes at least one state variable based on the set of rights and used for determining a state of the created right.

The specification discloses:

When a recipient wishes to obtain specific content 42, the recipient makes a request for content 42. For example, a user, as a recipient, might browse a Web site running on Web server 80, using a browser installed in client environment 30, and request content 42. During this process, the user may go through a series of steps possibly including a fee transaction (as in the sale of content) or other transactions (such as collection of information). When the appropriate conditions and other prerequisites, such as the collection of a fee and verification that the user has been activated, are satisfied, Web server 80 contacts license server 50 through a secure communications channel, such as a channel using a Secure Sockets Layer (SSL). License server 50 then generates license 52 for content 42 and Web server 80 causes both the content and license 52 to be downloaded. License 52 includes the appropriate rights, such as usage rights and/or meta-rights, and can be downloaded from license server 50 or an associated device. Content 42 can be downloaded from computer 70 associated with a vendor, distributor, or other party.

Client component 60 in client environment 30 will then proceed to interpret license 52 and allow use of content 42 based on the usage rights and conditions specified in license 52. The interpretation and enforcement of usage rights are well known generally and described in the patents referenced above, for example. The steps described above may take place sequentially or approximately simultaneously or in various orders.

'280 Patent at 4:66-5:25.

On balance, this disclosure amounts to a sufficient algorithm "in prose" for implementing the claimed function. *Typhoon Touch*, 659 F.3d at 1386.

The Court accordingly hereby finds that for the **"means for obtaining a set of rights associated with an item,"** the function is **"obtaining a set of rights associated with an item,"** and the corresponding structure is **"a client environment 30 with a browser capable of connecting to a web server 80 and storing a license 52 that can be interpreted by client component 60, as set forth in the '280 Patent at 4:66-5:21; and equivalents thereof."**

J. “means for determining whether the rights consumer is entitled to the right specified by the meta-right”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“Corresponding Structure: authorization manager 508 that authenticates the rights consumer 304 and verifies that the conditions 306 of the license 52 have been satisfied (8:66-9:8; 9:15-18; 9:63-10:2) ^{22,23}	Indefinite

(Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at p. 36; Dkt. No. 331, at 37.) The parties submit that this disputed term appears in Claim 12 of the ‘280 Patent. (Dkt. No. 292-1, at 14; Dkt. No. 331, at 37.)

(1) The Parties’ Positions

Plaintiff’s expert opines: “A person of ordinary skill in the art would understand that authenticating the rights consumer and verifying that the conditions of the license have been satisfied is necessary to determine whether the rights consumer is entitled to the right specified by the meta-right and clearly linked to that function.” (Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶ 70.)

Defendants argue that “[t]here are no algorithms or other descriptions of software that show how the consumer’s entitlement is determined. Instead, the ‘280 patent provides only conceptual and abstract restatements of the claimed function.” (Dkt. No. 331, at 37 (citations omitted).)

²² This parenthetical does not appear in the parties’ January 23, 2015 Joint Claim Construction Chart. (See Dkt. No. 366, Ex. B, at 48.)

²³ Plaintiff previously proposed: “Corresponding Structure: algorithm/structure necessary for performing the recited function set forth in the ’280 Spec., including from the following portions: Figs. 5, 7, 8 and 8:66-9:8, 9:66-10:2, 10:35-45, and equivalents thereof.” (Dkt. No. 292-1, at 14.)

Plaintiff's reply brief does not address this term. (*See* Dkt. No. 345, at 15-18.)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

Claim 12 of the '280 Patent recites (emphasis added):

12. A system for transferring rights adapted to be associated with items from a rights supplier to a rights consumer, the system comprising:
- means for obtaining a set of rights associated with an item, the set of rights including a meta-right specifying a right that can be created when the meta-right is exercised, wherein the meta-right is provided in digital form and is enforceable by a repository;
 - means for determining whether the rights consumer is entitled to the right specified by the meta-right;* and
 - means for exercising the meta-right to create the right specified by the meta-right if the rights consumer is entitled to the right specified by the meta-right, wherein the created right includes at least one state variable based on the set of rights and used for determining a state of the created right.

The specification discloses:

Authorization module 508 instructs license manager 503 to fetch state variable 308 and conditions 306 of license 52. Authorization manager 508 then determines which state variables are required to enforce to enforce [*sic*] license 52. State of rights manager 504 then supplies the current value of each required state variable to authorization module 508. Authorization module 508 then passes conditions 306 and the required state variables to condition validator 506. If all conditions 306 are satisfied, authorization module 508 returns "authorized" to meta-rights manager module 510.

* * *

Rights manager module 512 uses authorization module 508 to verify that recipient of the newly created rights or derived rights is intended principal 304.

* * *

In step 702 [of Fig. 7], principal 304 of license 52 is *authenticated in a known manner*. In other words, it is determined if the party exercising meta-right 302 has the appropriate license to do so. If the principal is not authorized, the procedure terminates in step 704. If the principal is authorized, the procedures [*sic*] advances to step 706 in which meta right 302 is exercised and transmitted to the consumer in the form of license 52 having derived rights in the manner set forth above. In step 708 the principal of this new license is *authenticated*. In

other words, it is determined if the party exercising the derived rights has the appropriate license to do so. If the principal is not authorized, the procedure terminates in step 710. If the principal is authorized, the procedures [sic] advances to step 712 in which the derived right is stored. The procedure then returns to step 708 for each additional right in the license and terminates in step 714 when all rights have been processed.

‘280 Patent at 8:66-9:8, 9:15-18 & 9:58-10:7 (emphasis added).

Defendants’ expert, Dr. Grimes, opines:

For instance, “authenticating the rights consumer” is not an algorithm. Rather, it is a function that similarly requires structural support, which Dr. Goodrich [(Plaintiff’s expert)] does not identify. “Verifying that conditions of the license have been satisfied” is no better. This language is insufficient to define an algorithm.

(Dkt. No. 331, Ex. 11, 12/22/2014 Grimes Decl., at ¶ 41.)

Plaintiff’s expert, Dr. Goodrich, opines that a person of ordinary skill in the art would understand the above-quoted passages as sufficient disclosure of an algorithm in prose form.

(Dkt. No. 345, Ex. AA, 1/9/2015 Goodrich Decl., at ¶ 23.)

The Court finds Plaintiff’s expert’s opinion more credible in this regard. For example, Defendants’ have not rebutted the above-quoted disclosure in the specification that authentication was “known.” On one hand, “[t]he inquiry is whether one of skill in the art would understand the specification itself to disclose a structure, not simply whether that person would be capable of implementing a structure.” *Biomedino, LLC v. Waters Techs. Corp.*, 490 F.3d 946, 953 (Fed. Cir. 2007). On the other hand, “the amount of detail that must be included in the specification depends on the subject matter that is described and its role in the invention as a whole, in view of the existing knowledge in the field of the invention.” *Typhoon Touch*, 659 F.3d at 1385. On balance, the latter is the applicable principle here, and the above-quoted disclosure amounts to a sufficient algorithm in prose form. *Id.* at 1386.

The Court accordingly hereby finds that for the **“means for determining whether the rights consumer is entitled to the right specified by the meta-right,”** the function is **“determining whether the rights consumer is entitled to the right specified by the meta-right,”** and the corresponding structure is **“authorization manager 508 that authenticates the rights consumer 304 and verifies that the conditions 306 of the license 52 have been satisfied, as described in the ‘280 Patent at 8:66-9:8, 9:15-18 & 9:63-10:2; and equivalents thereof.”**

K. “means for exercising the meta-right to create the right specified by the meta-right”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“Corresponding Structure: meta-rights manager module 510 that derives new rights from meta-rights 302 in accordance with a set of rules or other logic and updates the state of rights and the current value of the conditions in a state of rights repository (9:9-13; 9:33-50) ^{24,25}	Indefinite

(Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at p. 37; Dkt. No. 331, at 37.) The parties submit that this disputed term appears in Claim 12 of the ‘280 Patent. (Dkt. No. 292-1, at 15; Dkt. No. 331, at 37.)

(1) The Parties’ Positions

Plaintiff’s expert opines: “A person of ordinary skill in the art would understand that deriving new rights from meta-rights and updating the state of rights and the current value of the

²⁴ This parenthetical does not appear in the parties’ January 23, 2015 Joint Claim Construction Chart. (See Dkt. No. 366, Ex. B, at 48.)

²⁵ Plaintiff previously proposed: “Corresponding Structure: algorithm/structure necessary for performing the recited function set forth in the ‘280 Spec., including from the following portions: Figs. 5, 7, 8 and 8:56-57, 9:9-22, 9:33-6, 10:2-7, 10:35-45, 10:62-66, and equivalents thereof.” (Dkt. No. 292-1, at 15.)

conditions in a state of rights repository is necessary in exercising the meta-right to create the right specified by the meta-right, and clearly linked to that function.” (Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶ 71.)

Defendants argue that “[t]he ‘280 patent explains that to ‘exercise’ a meta-right is to create a usage right (or another meta-right) using the meta-right. The ‘280 patent, however, provides no description of any algorithm for performing this ‘exercising’ function.” (Dkt. No. 331, at 37.)

Plaintiff’s reply brief does not address this term. (*See* Dkt. No. 345, at 15-18.)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

Claim 12 of the ‘280 Patent recites (emphasis added):

12. A system for transferring rights adapted to be associated with items from a rights supplier to a rights consumer, the system comprising:

means for obtaining a set of rights associated with an item, the set of rights including a meta-right specifying a right that can be created when the meta-right is exercised, wherein the meta-right is provided in digital form and is enforceable by a repository;

means for determining whether the rights consumer is entitled to the right specified by the meta-right; and

means for exercising the meta-right to create the right specified by the meta-right if the rights consumer is entitled to the right specified by the meta-right, wherein the created right includes at least one state variable based on the set of rights and used for determining a state of the created right.

The specification discloses:

Meta-rights manager module 510 verifies license 52 and meta-rights 302 therein, to authorize the request to exercise meta-rights 302, to derive new rights from meta-rights 302, and to update the state of rights and the current value of the conditions.

* * *

Once a request to exercise a meta-rights [*sic*] has been authorized, the meta-right can be exercised. Meta-rights manager module 510 informs state of rights

module 504 that it has started exercising the requested meta-rights. State of rights module 504 then records the usage history and changes its current value of the state variables. Meta-rights manager module 510 exercises the requested meta-rights in a manner similar to known procedures for usage rights. If new rights are derived, then meta-rights manager module 510 invokes license manager module 504 to create new rights as the result of exercising the target meta-rights. Each new right is then sent to the corresponding rights manager module 512 of the consumer and stored in a repository associated with the consumer. Rights manager module 512 of the consumer will authenticate and authorize the consumer before receiving and storing the newly created right. New rights can be derived from meta-rights in accordance with a set of rules or other logic. For example, one rule can dictate that a consumed right to offer a license for use will result in the consumer having the right to offer a usage right and grant a license to that usage right to another consumer.

‘280 Patent at 9:9-13 & 9:33-53.

Plaintiff’s expert, Dr. Goodrich, opines that a person of ordinary skill in the art would recognize this as disclosure of an algorithm in prose form. (Dkt. No. 345, Ex. AA, 1/9/2015 Goodrich Decl., at ¶ 22; *see* Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶ 71.)

The Court finds Plaintiff’s expert’s opinion credible in this regard. On one hand, “[t]he inquiry is whether one of skill in the art would understand the specification itself to disclose a structure, not simply whether that person would be capable of implementing a structure.” *Biomedino*, 490 F.3d at 953. On the other hand, “the amount of detail that must be included in the specification depends on the subject matter that is described and its role in the invention as a whole, in view of the existing knowledge in the field of the invention.” *Typhoon Touch*, 659 F.3d at 1385. On balance, the latter is the applicable principle here, and the above-quoted disclosure amounts to a sufficient algorithm in prose form. *Id.* at 1386.

The Court accordingly hereby finds that for the **“means for exercising the meta-right to create the right specified by the meta-right,”** the function is **“exercising the meta-right to create the right specified by the meta-right,”** and the corresponding structure is **“meta-rights manager module 510 that derives new rights from meta-rights 302 in accordance with a set**

of rules or other logic and updates the state of rights and the current value of the conditions in a state of rights repository, as described in the ‘280 Patent at 9:9-13 & 9:33-50; and equivalents thereof.’”

L. “means for generating a license including the created right, if the rights consumer is entitled to the right specified by the meta-right”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“Corresponding Structure: License Server 50/License Manager 803 (4:5-14; 5:6-13; 10:35-45; 10:62-11:16) ^{26,27}	Indefinite

(Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at p. 37; Dkt. No. 331, at 38; Dkt. No. 366, Ex. B, at 49.) Plaintiff submits that this disputed term appears in Claim 22 of the ‘280 Patent. (Dkt. No. 292-1, at 15.)

(1) The Parties’ Positions

Plaintiff’s expert opines: “A person of ordinary skill in the art would understand that a license server/license manager is necessary to generate a license including the created right and [is] clearly linked to that function.” (Dkt. No. 304, Ex. K, 11/25/2014 Goodrich Decl. at ¶ 72.)

Defendants argue that the only relevant passage in the specification “simply restates the function of generating a license.” (Dkt. No. 331, at 38 (citing ‘280 Patent at 5:11-13).)

Plaintiff’s reply brief does not address this term. (*See* Dkt. No. 345, at 15-18.)

At the February 6, 2015 hearing, the parties did not address this disputed term.

²⁶ This parenthetical does not appear in the parties’ January 23, 2015 Joint Claim Construction Chart. (*See* Dkt. No. 366, Ex. B, at 49.)

²⁷ Plaintiff previously proposed: “Corresponding Structure: algorithm/structure necessary for performing the recited function set forth in the ‘280 Spec., including from the following portions: Figs. 1, 3, 4, 5, 7, 8 and 4:2-49, 5:6-13, 8:31-35, 8:56-57, 9:9-22, 9:33-66, 10:2-7, 10:35-45, 10:62-66, and equivalents thereof.” (Dkt. No. 292-1, at 15.)

(2) Analysis

Claim 22 of the '280 Patent recites: "The system of claim 12, further comprising means for generating a license including the created right, if the rights consumer is entitled to the right specified by the meta-right."

The portions of the specification cited by Plaintiff disclose as follows:

License Server 50 manages the encryption keys and issues licenses for protected content. These licenses embody the actual granting of usage rights to an end user. For example, rights label 40 may include usage rights permitting a recipient to view content for a fee of five dollars and view and print content for a fee of ten dollars. License 52 can be issued for the view right when the five dollar fee has been paid, for example. Client component 60 interprets and enforces the rights that have been specified in license 52.

* * *

When the appropriate conditions and other prerequisites, such as the collection of a fee and verification that the user has been activated, are satisfied, Web server 80 contacts license server 50 through a secure communications channel, such as a channel using a Secure Sockets Layer (SSL). *License server 50 then generates license 52 for content 42 and Web server 80 causes both the content and license 52 to be downloaded.*

* * *

FIG. 8 illustrates an exemplary system including a common state-of-rights server, according to the present invention. In FIG. 8, the exemplary system can include a common state-of-rights server of the system 801, including a state-of-rights manager 809, and one or more state-of-rights repositories 814, and one or more license servers 800, including a meta-rights manager 810, a usage rights manager 812, an authorization component 808, a condition validator 806, a state-of-rights manager 804, one or more state-of-rights repositories 816, a license manager 803, a license interpreter 802, and one or more license repositories 818.

* * *

The *license manager 803 derives new rights* based on an offer, which can include any suitable machine-readable expression, and optionally including meta-rights. While deriving rights, the license manager 803 can create new state variables to be associated with derived rights. The creation of state variables and their scopes can be prescribed in the offer or by some other function in the system. The state variables can be created in one or more instances, for example, prior to rights

derivation, during rights derivation, upon fulfillment of conditions, during a first exercise of rights associated with the state variables, and the like. The state variables can be designated exclusively for a specific rights consumer, can be shared among rights consumers, and can be shared among rights consumers and other entities, such as rights suppliers, and the like. The license manager 803 can interact with the state-of-rights manager 804 to associate new state variables with physical addresses in one or more of the state-of-rights repositories 816. The state-of-rights manager 804 can access the one or more state-of-rights repositories 816 and can interact with the state-of-rights server 801 to access shared state variables from one or more of the state-of-rights repositories 814.

‘280 Patent at 4:5-14, 5:6-13, 10:35-45 & 10:62-11:16 (emphasis added).

On balance, these disclosures set forth a “license server 50” and a “license manager 803” that are “clearly linked or associated with the claimed function,” *Med. Instrumentation & Diagnostics Corp. v. Elekta AB*, 344 F.3d 1205, 1219 (Fed. Cir. 2003), and that connote structure as opposed to merely restating the claimed function, especially in light of the surrounding disclosure set forth above.

The Court accordingly hereby finds that for the **“means for generating a license including the created right, if the rights consumer is entitled to the right specified by the meta-right,”** the function is **“generating a license including the created right, if the rights consumer is entitled to the right specified by the meta-right,”** and the corresponding structure is **“license server 50 or license manager 803 as described in the ‘280 Patent at 4:5-14, 5:6-13, 10:35-45, and 10:62-11:16; and equivalents thereof.”**

VI. CONSTRUCTION OF DISPUTED TERMS IN THE DUNKELD PATENT

The ‘556 Patent is titled “Method of Providing a Digital Asset for Distribution.” The ‘556 Patent issued on November 12, 2013, and bears a priority date of December 10, 2001. The Abstract states:

Digital assets are provided for distribution within an electronic network. The digital asset includes digital content that is associated with a digital rights holder. A serial number is provided for (embedded within) the asset; this number

uniquely identifies a first introduction of digital asset for distribution within the electronic network. The asset is then posted in a number of locations so that it can be distributed to users. A transaction database is updated to reflect occurrences of different instantiations of the asset.

A. “detect[ing] a transfer”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“to discover or determine the existence, presence, or fact of a transfer”	“to discover the occurrence of a transfer”

(Dkt. No. 304, at 28; Dkt. No. 331, at 38.) Defendants submit that these disputed terms appear in Claims 1 and 12 of the ‘556 Patent. (Dkt. No. 331, at 38.)

(1) The Parties’ Positions

Plaintiff argues, in full: “Both parties are proposing dictionary definitions. [Plaintiff’s] proposal, which relies on the Merriam-Webster dictionary (<http://www.merriam-webster.com/dictionary/detect>), is slightly broader than Defendants’, and as such should be adopted.” (Dkt. No. 304, at 28 (citing *Cephalon Inc. v. Mylan Pharms. Inc.*, 962 F. Supp. 2d 688, 699 (D. Del. 2013)).)

Defendants respond that “[t]he specification does not include the broad[] scope urged by [Plaintiff],” and “Defendants’ more clear and concise proposal will be easier for a jury to understand.” (Dkt. No. 331, at 38.)

Plaintiff replies that “[t]he parties’ . . . dispute boils down to whether the Court should pick Defendants’ narrow definition . . . or [Plaintiff’s] slightly broader definition” (Dkt. No. 345, at 19.) “Nothing in the specification indicates that the patentee had in mind a narrow definition for ‘detecting,’” Plaintiff urges. (*Id.*)

(2) Analysis

The specification discloses:

One key purpose of the present inventions is to allow individual customers to trade digital assets with each other while compensating rights holders for their work. The described system allows each asset to be *identified and tracked* (preferably) at the time the asset is *transferred*.

‘556 Patent at 10:13-17 (emphasis added).

At the completion of the transfer, Host Server Network Device initiates step 324 by contacting Serial Number Reconciliation Module 120 to report completion of the transfer or its abandonment and the reasons for such.

* * *

Customer client server module 124 also contacts Serial Number Reconciliation Module 120 in step 328. It reports the transaction as being complete and also indicates whether and where the second instantiation of the digital asset can be found for transfer to other customers in system 100.

Id. at 15:20-23 & 15:38-42.

In another variation of the present invention, *detection* of “rogue” assets is performed prior to *transfers*. By this it is meant that a first customer may attempt to download a digital asset from a second customer, and in the process of doing so, System Network Device 106 may *detect* that there is no appropriate tracking record reflecting a prior authorized *transfer* to such second customer.

Id. at 16:10-16 (emphasis added).

Nothing in the intrinsic evidence suggests that the patentee gave a special meaning to the term “detect.” *Thorner*, 669 F.3d at 1367 (“The patentee is free to choose a broad term and expect to obtain the full scope of its plain and ordinary meaning unless the patentee explicitly redefines the term or disavows its full scope.”). In other words, the proposals by both sides lack support or justification in the intrinsic evidence and would tend to confuse rather than clarify the scope of the claims.

Further, upon discussion at the February 6, 2015 hearing, both sides agreed that this disputed term could be construed to have its plain meaning.

The Court accordingly hereby construes “**detect[ing] a transfer**” to have its **plain meaning**.

B. “instance”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“instantiation”	“a file containing the digital asset that is distinct from other files containing the same digital asset”

(Dkt. No. 304, at 28; Dkt. No. 331, at 40.) The parties submit that this disputed term appears in Claims 1 and 12 of the ‘556 Patent. (Dkt. No. 292-1, at 16; Dkt. No. 331, at 40.)

(1) The Parties’ Positions

Plaintiff argues that whereas the specification and the prosecution history support Plaintiff’s proposal, Defendants’ proposal “is both unsupported by the evidence and needlessly verbose.” (Dkt. No. 304, at 28; *see id.*, at 28 n.15 (citing *Encap LLC v. Oldcastle Retail, Inc.*, 2012 WL 2339095, at *9 (E.D. Wis. June 19, 2012) (“Claim construction is not intended to allow for needless substitution of more complicated language for terms easily understood by a lay jury.”); *Am. Patent Dev. Corp. v. Movielink, LLC*, 604 F. Supp. 2d 704, 716 (D. Del. 2009) (rejecting construction that, “in the Court’s view, is merely a verbose paraphrasing of the claim language that otherwise offers little to assist one of skill in the art in understanding the claims”)).)

Defendants respond, in full: “Defendants’ proposal clearly explains the meaning of ‘instance,’ while [Plaintiff’s] proposal would be more confusing to the jury than ‘instance’ itself. Moreover, ‘instance’ was changed from ‘instantiation’ during prosecution. The Court should prevent [Plaintiff] from reclaiming scope it surrendered to obtain the claims.” (Dkt. No. 331, at 40 (citation omitted).)

Plaintiff replies that whereas Defendants' proposal lacks support in any intrinsic evidence or any dictionary definition, Plaintiff's proposal is consistent with the intrinsic record. (Dkt. No. 345, at 20.) Plaintiff also argues that "Defendants are incorrect that the patentee effectuated a 'surrender[r]' of claim scope by replacing 'instantiation' in the claims with its synonym 'instance.'" (*Id.*)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

The specification repeatedly uses the term "instantiation":

A related object [of the invention] is to provide a tracking mechanism and method that relies primarily on creating *separate instantiations of a digital asset to facilitate tracking* of the latter[.]

* * *

This architecture ensures security, compliance, and accountability for each *instantiation* of the asset.

* * *

[T]o assist the tracking of the digital asset, *a separate and new instantiation of the digital asset is created for each transfer* occurring over the network between peer devices.

* * *

The present invention treats each instantiation of an asset as unique and as such the terms of acquisition can be flexible with respect to time, parties involved in the transaction, prior purchasing, intended usage, etc.

* * *

As noted earlier, a *first instantiation* of the digital asset is created based on an original offset and serial number embedded within the digital content. To allow for *tracking of the particular transfer*, a *new instantiation* of the digital asset is made. In step 316 Client Server module 124 gets a new serial number and new offset for this transaction from the Serial Number Assignment Module 118. The new serial number and new offset are used to create a *unique instantiation* of the digital asset for the particular transaction. Thus, instead of merely copying the

digital content as part of the transfer, *the present invention creates a separate instantiation to facilitate tracking of each transfer* (or transaction) within system 100.

In some applications where security and accounting is [*sic*, are] not as critical (or can be remedied by other mechanisms consistent with the present teachings) it is possible that actual *separate instantiations* of the digital asset might not be required. Instead, it might be more practical to simply track the point-to-point movement of a digital asset across network between one or more Customer Network devices 112, and/or Host Server Network Device 110.

'556 Patent at 2:32-34, 3:27-28, 3:43-46, 8:23-27 & 14:27-46 (emphasis added); *see id.* at 4:24-38 (“first instantiation” and “second instantiation”); *see also id.* at 5:37-44 (similar) & 6:53-62 (similar).

Plaintiff also submits that during prosecution of the '556 Patent, the patentee submitted: “Instantiation=To create an instance of an object, *Microsoft Computer Dictionary*, 3rd Edition, Microsoft Press, Redmond, WA, 1997[.]” (Dkt. No. 304, Ex. X, at 4.) Defendants likewise note that the patentee amended the claims so as to replace “instantiation” with “instance.” (Dkt. No. 331, Ex. 20, 10/9/2012 Amendment and Response, at 2-3 & 5-6.)

The intrinsic evidence thus demonstrates that the term “instance” in the claims refers to what is described as an “instantiation” in the specification. Defendants’ proposed construction properly conveys the essential feature of an “instance” as set forth in the disclosures above. Specifically, as quoted above, the essence of one “instance” of an item of digital content, as compared to another such instance, is that each instance contains the same digital content but is nonetheless uniquely identifiable. *See* '556 Patent at 8:23-27 & 14:27-38 (describing “the present invention”); *see, e.g., Verizon*, 503 F.3d at 1308.

The Court accordingly hereby construes **“instance”** to mean **“a file that contains a digital asset, and the file is distinguishable from other files containing the same digital asset.”**

C. “other portion”

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
“an unused part of the digital asset or information prepended or postpended to the digital asset”	“a part of the digital asset wherein embedding information does not affect the user-perceptible portion of content”

(Dkt. No. 304, at 29; Dkt. No. 331, at 39.) The parties submit that this disputed term appears in Claims 1 and 12 of the ‘556 Patent. (Dkt. No. 292-1, at 16; Dkt. No. 331, at 39.)

(1) The Parties’ Positions

Plaintiff argues that whereas its proposed construction “faithfully tracks the disclosure in the specification,” “Defendants’ construction reflects language that is expressly tied to a preferred embodiment (*see* ‘556 Patent at col. 3:35-39) that is nowhere present in the claims.”

(Dkt. No. 304, at 29.) Plaintiff submits that although the prosecution history of the ‘556 Patent reveals that the application claims at one time included a limitation of “without altering user perceptible content of the first digital media asset” (Dkt. No. 304, Ex. Y, 10/9/2012 Amendment and Response, at 3), the issued Claims 1 and 12 do not include such a limitation.

Defendants respond that “[Plaintiff’s] proposed construction ignores the specification, which shows that a customer identification can be embedded into many portions of the digital asset.” (Dkt. No. 331, at 39 (citing ‘556 Patent at 12:43-47 & 19:28-31).) Defendants also urge that “Defendants’ proposed construction takes into account a key part of the invention: that embedding the customer identification does not affect user-perceptible content.” (Dkt. No. 331, at 39 (citing ‘556 Patent at 1:24-26, 2:8-10, 3:37-39 & 8:8-9).)

Plaintiff replies that its proposal “mirrors verbatim the specification’s disclosure concerning the ‘portion[s]’ of the digital assets that are to contain identifying information.” (Dkt. No. 345, at 19 (citing ‘556 Patent at 19:36-38).) Plaintiff also argues that “Defendants’

proposed construction (1) describes a preferred embodiment; (2) rests on a portion of the specification that says nothing about *which part* of the digital asset is to contain identifying information; and (3) imports into the claims a limitation (‘embedding information does not affect the user-perceptible portion of content’) that was removed during prosecution.” (*Id.*, at 19-20 (citation omitted).)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

Claims 1 and 23 of the ‘556 Patent recite, in relevant part:

1. A method implemented by one or more computing devices for providing a digital asset for distribution, the method comprising:
 storing, by at least one of the one or more computing devices, the digital asset, the digital asset including *digital content*; . . .
 in response to the request for the digital asset, creating, by at least one of the one or more computing devices, a second instance of the digital asset for transfer to the user device, the second instance of the digital asset including *content* and at least one *other portion*, and embedding in the at least one *other portion* of the second instance of the digital asset at least a customer identification associated with the user and the asset identifier, wherein other instances of the digital asset have customer identifications embedded therein and the customer identifications are used to track instances of the digital asset;

. . . .

* * *

23. The method of claim 1, wherein the at least one *other portion* is an unused portion.

Claim 23, a dependent claim, thus suggests that the “other portion” can be either an “unused” portion or, presumably, a “used” portion. *See Phillips*, 415 F.3d at 1315 (“[T]he presence of a dependent claim that adds a particular limitation gives rise to a presumption that the limitation in question is not present in the independent claim.”); *see also Liebel-Flarsheim*, 358 F.3d at 910 (“[W]here the limitation that is sought to be ‘read into’ an independent claim already appears in a dependent claim, the doctrine of claim differentiation is at its strongest.”);

Wenger, 239 F.3d at 1233 (“Claim differentiation, while often argued to be controlling when it does not apply, is clearly applicable when there is a dispute over whether a limitation found in a dependent claim should be read into an independent claim, and that limitation is the only meaningful difference between the two claims.”).

Nonetheless, “the doctrine of claim differentiation can not broaden claims beyond their correct scope, determined in light of the specification and the prosecution history and any relevant extrinsic evidence.” *Multiform Desiccants, Inc. v. Medzam, Ltd.*, 133 F.3d 1473, 1480 (Fed. Cir. 1998); *see N. Am. Vaccine, Inc. v. Am. Cyanamid Co.*, 7 F.3d 1571, 1577 (Fed. Cir. 1993) (“While it is true that dependent claims can aid in interpreting the scope of claims from which they depend, they are only an aid to interpretation and are not conclusive. The dependent claim tail cannot wag the independent claim dog.”).

At first blush, reading “other portion” to encompass the “content” set forth in Claim 1 would seemingly be at odds with the plain language of Claim 1, which appears to contrast the “other portion” with “content,” as quoted above. Nonetheless, Claim 1 recites “content,” not a “content portion,” so Claim 1 does not on its face recite whether or not the “other portion” can be embedded with the content.

Turning to the other intrinsic evidence, the specification discloses substantially the language that Defendants have proposed:

In a preferred embodiment the digital asset is modified for each transfer, and this modification is used by the third management server for generating the tracking records. Again, the modification *does not alter user-perceptible content* of the digital asset.

* * *

A preferred approach is to use the Offset to specify a valid frame and word count within the MP3 file to begin the marking. The Serial Number is then encoded one

bit at a time in the least significant bit of successive data words until the entire Serial Number is encoded.

'556 Patent at 3:35-39 & 12:43-47 (emphasis added); *see id.* at 1:24-26 (“intellectual property assets that can be digitized can now be reproduced and distributed without quality degradation”); *see also id.* at 8:8-9 (“digital media formats are receptive to steganographic techniques without noticeable quality degradation”).

Defendants’ proposed reference to what is “user-perceptible” thus has some support in the specification, as set forth above, but the specification also explains that one of the purposes of using steganographic techniques is to allow use of existing media formats and rights management. *See* '556 Patent at 8:8-18; *see also id.* at 8:50-52 & 9:61-64. Thus, user-perceptible portions of content may be altered, albeit perhaps in a way that a user would not notice. Defendants’ proposal of “user-perceptible” would also introduce a potentially subjective element that would tend to confuse rather than clarify the scope of the claims. The Court therefore hereby expressly rejects Defendants’ proposed construction.

The specification also discloses that information can be “prepended or postpended,” as Plaintiff has proposed:

Steganographic Variations

While the preferred embodiment discussed above uses a steganographic technique for embedding a serial number in an MP3 file, there are many other approaches that could accomplish this same function. Furthermore, it is expected that the particular mechanism used to provide and associated serial numbers will be different from application to application, because various digital asset formats are receptive to different approaches.

In addition, as alluded to earlier, digital asset serial numbers could be *prependded or postpendded*; alternatively, *unused portions* of the digital asset could be used to store the serial numbers. Finally, a modified format for a digital asset could be created to accommodate the serial number, such as new variation of an MP3 file, MPEG file, etc. For example, one or more standards groups or industry groups

may utilize a form of digital asset that includes fields intended to accommodate a serial number.

Id. at 19:27-44 (emphasis added)

Plaintiff's proposal of "prepending or postpending" is thus supported by the specification. Plaintiff's proposal of referring to an "unused part of the digital asset," however, would apparently exclude a "used" portion. Such a reading is disfavored by the doctrine of claim differentiation, as set forth above.

The Court accordingly hereby construes "**other portion**" to mean "**any part of the digital asset, or information prepended or postpending to the digital asset.**"

D. "over said network between user devices"

Plaintiff's Proposed Construction	Defendants' Proposed Construction
Error – "over said network" should be deleted from both claims 8 and 19	Indefinite

(Dkt. No. 304, at 29; Dkt. No. 331, at 40; Dkt. No. 366, Ex. B, at 57.) The parties submit that this disputed term appears in Claims 8 and 19 of the '556 Patent. (Dkt. No. 292-1, at 16; Dkt. No. 331, at 40.)

(1) The Parties' Positions

Plaintiff submits that "[o]ver said network" appears in claims 8 and 19 because it was overlooked in a complicated amendment when similar language was deleted from other claims." (Dkt. No. 304, at 30.)

Defendants respond that "[t]he parties agree that claims 8 and 19 of the '556 patent are indefinite as written," and "[t]he Court should not grant [Plaintiff's] request [for judicial correction] because the claims are subject to multiple potential 'corrections,' e.g., changing

‘said’ to ‘a’ or reciting a ‘network’ in claim 1.” (Dkt. No. 331, at 40 (citing *Novo Indus.*, 350 F.3d at 1354).)

Plaintiff replies:

[F]rom the face of the patent, it is plain that, as they presently stand and unless “over said network” is removed, claims 8 and 19 lack antecedent bases in the claims from which they depend, independent claims 1 and 12. Put differently, [Plaintiff] is not asking the Court to impermissibly remove a limitation from a defect-free, perfectly coherent claim. Rather, [Plaintiff] is asking for a correction of an obvious clerical error to make these claims coherent.

(Dkt. No. 345, at 20.)

At the February 6, 2015 hearing, the parties did not address this disputed term.

(2) Analysis

Claims 1 and 8 of the ‘556 Patent are representative and recite (emphasis added):

1. A method implemented by one or more computing devices for providing a digital asset for distribution, the method comprising:
 - storing, by at least one of the one or more computing devices, the digital asset, the digital asset including digital content;
 - associating, by at least one of the one or more computing devices, an asset identifier with the digital asset to thereby generate a first instance of the digital asset, the asset identifier identifying the digital asset;
 - receiving from a user, by at least one of the one or more computing devices, an acceptance of terms of use of digital assets;
 - providing, by at least one of the one or more computing devices, a list of one or more digital assets to the user, the list including the digital asset;
 - receiving from the user, by at least one of the one or more computing devices, a request for the digital asset;
 - in response to the request for the digital asset, creating, by at least one of the one or more computing devices, a second instance of the digital asset for transfer to the user device, the second instance of the digital asset including content and at least one other portion, and embedding in the at least one other portion of the second instance of the digital asset at least a customer identification associated with the user and the asset identifier, wherein other instances of the digital asset have customer identifications embedded therein and the customer identifications are used to track instances of the digital asset;
 - detecting, by at least one of the one or more computing devices, a transfer of the second instance of the digital asset to the user based at least in part on the customer identification;

debiting an account of the user related to the transfer of the second instance of the digital media asset to the user; and updating, by at least one of the one or more computing devices, a transaction database to reflect a transfer of the second instance of the digital media asset to the user.

* * *

8. The method of claim 1, wherein distributions of said digital asset *over said network* between user devices are not preconditioned on securing authorization for individual copies of said digital asset.

During prosecution, the patentee added new application claims 40 and 43 (which issued as Claim 1 and Claim 12, respectively), which recited no “network” limitation. (Dkt. No. 304, Ex. Z, 8/14/2014 Amendment, at 6-7.) At the same time, the patentee modified various dependent claims so as to depend from the new application claims 40 and 43, and the patentee removed the “network” from various dependent claims. (*See id.*, at 2-6.) Plaintiff submits that the patentee’s failure to remove “over said network” from the claims that issued as Claim 8 and Claim 19 was an “oversight.” (Dkt. No. 304, at 30.)

“A district court can correct a patent only if, among other things, the error is evident from the face of the patent.” *H-W Tech., LC v. Overstock.com, Inc.*, 758 F.3d 1329, 1333 (Fed. Cir. 2014) (citation and internal quotation marks omitted). Further, “evidence of error in the prosecution history [is] alone insufficient to allow the district court to correct the error.” *Id.* at 1334.

Here, Plaintiff argues that because the erroneousness of the claims is self-evident on their face, the Court can look to the prosecution history to determine the nature of the error and the appropriate correction. The above-cited authority, however, appears to require that “*the error*,” not merely the presence of some error, must be evident from the face of the patent. *Id.* Because

Plaintiff must resort to the prosecution history to demonstrate the error, the Court hereby expressly rejects Plaintiff's argument.

This finding is also consistent with the principle that “[c]ourts do not rewrite claims; instead, we give effect to the terms chosen by the patentee.” *K-2*, 191 F.3d at 1364; *see Chef Am.*, 358 F.3d at 1374 (“courts may not redraft claims, whether to make them operable or to sustain their validity”).

The Court accordingly hereby finds that the term “**over said network**” in Claims 8 and 19 of the ‘556 Patent is **indefinite**.

VII. CONCLUSION

The Court adopts the constructions set forth in this opinion for the disputed terms of the patents-in-suit. The parties are ordered that they may not refer, directly or indirectly, to each other's claim construction positions in the presence of the jury. Likewise, the parties are ordered to refrain from mentioning any portion of this opinion, other than the actual definitions adopted by the Court, in the presence of the jury. Any reference to claim construction proceedings is limited to informing the jury of the definitions adopted by the Court.

Having found that the term “validating” in Claim 5 of the ‘007 Patent is indefinite, as discussed above, the Court hereby finds that Claim 5 of the ‘007 Patent is invalid.

Also, having found that the term “over said network” in Claims 8 and 19 of the ‘556 Patent is indefinite, as discussed above, the Court hereby finds that Claims 8 and 19 of the ‘556 Patent are invalid.

Within thirty (30) days of the issuance of this Memorandum Opinion and Order, the parties are hereby ORDERED, in good faith, to mediate this case with the mediator agreed upon by the parties. As a part of such mediation, each party shall appear by counsel and by at least

one corporate officer possessing sufficient authority and control to unilaterally make binding decisions for the corporation adequate to address any good faith offer or counteroffer of settlement that might arise during such mediation. Failure to do so shall be deemed by the Court as a failure to mediate in good faith and may subject that party to such sanctions as the Court deems appropriate.

So ORDERED and SIGNED this 20th day of March, 2015.



RODNEY GILSTRAP
UNITED STATES DISTRICT JUDGE

and Testimony of Michael T. Goodrich and David Martin (Dkt. No. 721). At the Court's request, the Defendants consolidated the separate motions filed by various defendants on June 26, 2015, at docket number 232 in Case No. 2:14-cv-61, and docket numbers 665, 668, 673, 675, 677, 687, and 691 in Case No. 2:13-cv-1112 into the Combined Motion to Strike Portions of the Expert Reports and Testimony of Michael T. Goodrich and David Martin (Dkt. No. 721). The Court requested that the parties submit copies of each expert report in dispute, (Dkt. No. 782), which have subsequently been reviewed by the Court. The Court held a hearing on these motions on August 5, 2015. (Dkt. No. 827.) For the reasons set forth below, the motions to strike are **GRANTED** to the extent specified below, and are otherwise **DENIED**.

I. Background

On December 18, 2013, ContentGuard filed suit against Amazon, Apple, BlackBerry, Huawei, and Motorola Mobility asserting claims of patent infringement of the patents in this suit. (Dkt. No. 1). On January 17, 2014, ContentGuard filed an amended complaint asserting the same patents against HTC and Samsung. (Dkt. No. 22).

ContentGuard has asserted the following twenty claims from six related patents issued to Mark Stefik: Claims 1, 3, 6, 8, 11, and 13 from U.S. Patent No. 8,393,007 ("the '007 patent"); Claims 1, 7, and 13 from U.S. Patent No. 8,370,956 ("the '956 patent"); Claims 1 and 8 from U.S. Patent No. 7,523,072 ("the '072 patent"); Claims 18, 21, and 34 from U.S. Patent No. 7,269,576 ("the '576 patent"); and Claims 1, 21, and 58 from U.S. Patent No. 6,963,859 ("the '859 patent") (collectively, the "Stefik patents"). ContentGuard has also asserted the following five claims from two related patents issued to Mai Nguyen: Claims 1 and 5 of U.S. Patent No. 7,774,280 ("the '280 patent"); and Claims 1, 3, and 5 from U.S. Patent No. 8,001,053 ("the '053 patent") (collectively, the "Nguyen patents").

II. LEGAL STANDARD

An expert witness may provide opinion testimony if “(a) the expert’s scientific, technical, or other specialized knowledge will help the trier of fact to understand the evidence or to determine a fact in issue; (b) the testimony is based on sufficient facts or data; (c) the testimony is the product of reliable principles and methods; and (d) the expert has reliably applied the principles and methods to the facts of the case.” FED. R. EVID. 702.

Rule 702 requires a district court to make a preliminary determination, when requested, as to whether the requirements of the rule are satisfied with regard to a particular expert’s proposed testimony. See *Kumho Tire Co. v. Carmichael*, 526 U.S. 137, 149 (1999); *Daubert v. Merrell Dow Pharm., Inc.*, 509 U.S. 579, 592–93 (1993). District courts are accorded broad discretion in making Rule 702 determinations. *Kumho Tire*, 526 U.S. at 152 (“[T]he trial judge must have considerable leeway in deciding in a particular case how to go about determining whether particular expert testimony is reliable.”). Although the Fifth Circuit and other courts have identified various factors that the district court may consider in determining whether an expert’s testimony should be admitted, the common nature of these factors direct the trial court to consider as its ultimate inquiry whether the expert’s testimony is sufficiently reliable and relevant to be helpful to the finder of fact and thus to warrant admission at trial. *United States v. Valencia*, 600 F.3d 389, 424 (5th Cir. 2010).

Importantly, in a jury trial setting, the Court’s role under *Daubert* is not to weigh the expert testimony to the point of supplanting the jury’s fact-finding role. See *Micro Chem., Inc. v. Lextron, Inc.*, 317 F.3d 1387, 1391–92 (Fed. Cir. 2003) (applying Fifth Circuit law) (“When, as here, the parties’ experts rely on conflicting sets of facts, it is not the role of the trial court to evaluate the correctness of facts underlying one expert’s testimony.”); *Pipitone v. Biomatrix, Inc.*, 288 F.3d 239, 249–50 (5th Cir. 2002) (“[t]he trial court’s role as gatekeeper [under

Daubert] is not intended to serve as a replacement for the adversary system.’ . . . Thus, while exercising its role as a [gatekeeper], a trial court must take care not to transform a *Daubert* hearing into a trial on the merits”) (quoting FED. R. EVID. 702 advisory committee note). Instead, the Court’s role is limited to that of a gatekeeper, ensuring that the evidence in dispute is at least sufficiently reliable and relevant to the issue before the jury so as to be appropriate for the jury’s consideration. *See Pipitone*, 288 F.3d at 249–50. As the Supreme Court explained in *Daubert*, 509 U.S. at 596, “[v]igorous cross-examination, presentation of contrary evidence, and careful instruction on the burden of proof are the traditional and appropriate means of attacking shaky but admissible evidence.” *See Mathis v. Exxon Corp.*, 302 F.3d 448, 461 (5th Cir. 2002).

III. DISCUSSION

A. Prior Art Related Grounds for the Motions to Strike

1. Pfleeger References

The Court previously excluded the Pfleeger references from Defendants’ invalidity contentions because the Court found that Defendants had not shown good cause sufficient to add a reference to their invalidity contentions late in the litigation when they had previously known about the reference but not asserted it for over a year. (June 23, 2015, H’rg Tr., Dkt. No. 660, at 35:8–13.) Plaintiff argues that Defendants Apple, Google, HTC, Huawei, Motorola, and Samsung are now attempting to maneuver around the Court’s previous order by including Pfleeger as a state-of-the-art reference, rather than an invalidity reference. (Dkt. No. 679, at 11–12.) Defendants argue that even though the Court excluded Pfleeger as an invalidity reference, Pfleeger can still be used to show the state of the art, particularly as it was known to the inventor, at the time the patent was filed. (Dkt. No. 749, at 15.) Further, Defendants explicitly affirmed that they would not attempt to show Pfleeger as an anticipation or obviousness reference. (*Id.*); *see also* (Aug. 5, 2015, H’rg Tr., Dkt. No. 827, at 9:23–10:2.)

The Court is sympathetic to Defendants' position regarding the need to show the state of the prior art and accepts counsel's representation that Defendants would not overtly use Pfleeger as a *per se* invalidity reference. However, the Court, after considering the briefing and oral argument, finds that the portions of Mr. Ward's report containing discussion of the substance of Pfleeger must be stricken in view of the Court's previous order, including at least from Paragraphs 113, 121, 132, 134, 139, 140, 141, 178, 195, 391-94, 402-05, 511-524, 580, 615-16, 627-28, 641-42, 655-56, 669-70, 683-84, 702-03, 716-17, 739-40, and 746 of Mr. Ward's report. After examining the paragraphs in question, including the paragraphs that Defendants expressly identified as only touching on the state of the prior art, the Court finds it difficult to completely separate the use of Pfleeger as a state-of-the-art reference from the use of Pfleeger as an invalidity reference: the use for one purpose unavoidably bleeds into the other. Accordingly, the Court **GRANTS** Plaintiff's motion (Dkt. No. 679) as it pertains to the Pfleeger references in the above cited paragraphs.

2. VDE References

The Court previously denied Apple's request to amend its invalidity contentions with any VDE references other than U.S. Patent. No. 5,892,900 ("Ginter"), because the Court found that Apple had not shown an adequate basis to overcome the prejudice to ContentGuard of adding the VDE references this late in litigation. (April 28, 2015, H'rg Tr., Dkt. No. 577, at 44:5–16.) The Court also granted a subsequent motion to exclude based on its earlier order. (Dkt. No. 820.) Plaintiff argues that Defendant Apple has never asserted the "VDE system" against the Nguyen patents and is now attempting to circumvent the Court's previous order by "recit[ing] contentions (including two figures) about 'the VDE system' that are based not on anything in the public Ginter patent reference, but on deposition testimony from Mr. Ginter regarding the operation of the precluded VDE system." (Dkt. No. 685, at 14.) Apple argues that Ginter was

always asserted against the Nguyen patents and that the “two embodiments (‘traveling objects’ and ‘stationary objects’) described and depicted in paragraph 269 of Dr. Prakash’s report, and also described by Mr. Ginter during his deposition, are indeed disclosed in the Ginter patent.” (Dkt. No. 748, at 14.)

After considering the briefing and oral argument, the Court finds that the portions of the sentences from Paragraph 269 of Dr. Prakash’s report, which contain information about the VDE systems which is not directly from the Ginter patent, including the portions of the report based on information obtained from Mr. Ginter’s deposition testimony, must be stricken in view of the Court’s previous order. Accordingly, the Court **GRANTS** Plaintiff’s motion (Dkt. No. 685) as it pertains to information about the VDE system that is not disclosed directly from the Ginter patent.

3. Wyman References

Plaintiff argues that “[d]espite amending their Invalidity Contentions multiple times, Defendants never asserted [U.S. Patent Nos. 5,260,999 (the “’999 Patent”) and 5,204,897 (the “’897 Patent”) (collectively, “the Wyman patents”)] as prior art against the meta-rights patents in their Invalidity Contentions.” (Dkt. No. 679, at 11; Dkt. No. 685, at 13.) Plaintiff further argues that Defendants are now attempting to assert “[the Wyman patents] against the meta-rights patents for the first time in [their] expert report[s].” (Dkt. No. 679, at 11; Dkt. No. 685, at 13.) The ’999 Patent had previously been asserted against at least the Stefik patents. (Dkt. No. 749, at 14.) The non-Apple Defendants argue that they did not “appreciate the full relevance of [the ’999 Patent] to the Nguyen patents,” (*id.*), until “ContentGuard asserted an entirely new theory of infringement regarding ‘meta-rights’ in its April 20, 2015 fifth infringement contentions,” (Dkt. No. 805, at 4). Defendants further argue that they “promptly identified [the ’999 Patent] as being relevant to the invalidity of the Nguyen patents [upon realizing the ’999 Patent’s

relevance],” (Dkt. No. 749, at 14). Apple argues that “Dr. Prakash’s report does not assert [the ‘897 Patent] as a § 102 or 103 reference against the Nguyen patents.” (Dkt. No. 748, at 13.)

After considering the briefing and oral argument, the Court finds that assertion of the Wyman patents against the Nguyen patents is untimely, and as a result, those portions of Mr. Ward’s report (Paragraphs 882–887 and Exhibits X1 and X2) and Dr. Prakash’s report (Paragraphs 182 and 222) should be stricken. Accordingly, the Court **GRANTS** Plaintiff’s motions (Dkt. Nos. 679 and 685) as they pertain to the Wyman patents as invalidity references against the Nguyen patents.

B. Claim Construction Related Grounds for the Motions to Strike

Plaintiff argues that portions of Defendants’ various experts’ reports and testimony are improper because Defendants’ respective experts misapply the Court’s claim construction. *See* (Dkt. No. 679, at 5–11; Dkt. No. 683, at 4–12; Dkt. No. 684, at 4–9; Dkt. No. 685, at 5–13; Dkt. No. 690, at 4–13; Dkt. No. 692, at 4–12.) Similarly, Defendants argue that portions of Plaintiff’s experts’ reports and testimony are improper because Plaintiff’s experts misapply the Court’s claim construction. *See* (Dkt. No. 721, at 1.)

As an initial matter, the Court notes, that during a hearing held on July 27, 2015, the Court asked the parties whether any outstanding claim construction issues needed to be dealt with prior to trial. *See* (July 27, 2015, H’rg Tr., Dkt. No. 818, at 31:23–32:1.) Other than one discrete issue that has already been completely briefed (a motion for reconsideration, Dkt. No. 480), no party identified any remaining claim construction issues. *See (id. at 32:2–13.)* Regarding the particular expert reports and testimony disputes at hand, **IT IS ORDERED** that no experts are to render any conclusions regarding the scope of the patents-in-suit or particular claim limitations that deviate from this Court’s Claim Construction Memorandum and Order (Dkt. No. 459). Accordingly, all experts, whether Plaintiff’s or Defendants’, are hereby

excluded from providing any opinions that violate these constraints, and any portions of their reports in conflict with this Order are stricken.

Further, all experts are hereby excluded from providing any opinions based on an interpretation of the Court's construction that is the equivalent of any construction that the Court previously considered and expressly rejected. In particular, with regard to the following previously construed claim terms, the identified claim construction arguments have previously been considered and expressly rejected in the Court's Claim Construction Order, and therefore, the Court **ORDERS** that:

- **“rights,” “usage rights,” “usage rights information”** – No expert may opine or insinuate that a mere association between the content and the usage rights is enough to meet the requirement that the usage rights be “attached” to the content. *See* (Dkt. No. 459, at 33.) A mere reference, with nothing more to indicate that the usage rights should be attached or treated as attached to the content, is not enough. The bolded portion from the following sentence from paragraph 48 of Mr. Goodrich's report is an exemplary opinion implicated by this Order and must be stricken: “Thus usage rights may be attached to content by placing them inside the same data structure or file or through the use of links or **references** to the content from the data structure containing the usage right.” (Dkt. No. 721-7, at ¶48 (emphasis added).)
- **“rights,” “usage rights,” “usage rights information”** – No expert may opine or insinuate that the attachment between the content and the usage rights must be permanent. *See* (Dkt. No. 459, at 33.) The bolded portion from the following sentences from paragraph 160 of Mr. Ward's report are exemplary opinions

implicated by this Order and must be stricken: “A key feature of the present invention is that usage rights are **permanently** ‘attached’ to the digital work. . . . Thus, the usage rights and any associated fees assigned by a creator and subsequent distributor will **always** remain with a digital work.” (Dkt. No. 679-1, at ¶ 160 (emphasis added).)

- **“repository” and “trusted”** – No expert may opine or insinuate that a “repository” or “trusted device” must maintain the three integrities “at all times.” *See* (Dkt. No. 459, at 15.) The bolded portion from the following sentence from paragraph 138 of Mr. Clark’s report is an exemplary opinion implicated by this Order and must be stricken: “Thus, if an accused repository, recipient computing device, or recipient apparatus allows access to **any** ‘information’ by a nontrusted system (including, but not limited to ‘content’), then it cannot be found to possess the required ‘physical integrity’ and, accordingly, cannot be a ‘repository,’ a ‘trusted’ device, or ‘trusted’ apparatus.” (Dkt. No. 690-1, at ¶ 138 (emphasis added).)

Accordingly, the Court **GRANTS** Plaintiff’s motions (Dkt. No. 679, Dkt. No. 683, Dkt. No. 684, Dkt. No. 685, Dkt. No. 690, Dkt. No. 692) and Defendants’ Motion (Dkt. No. 721) only as they relate to the claim construction positions explicitly identified above. The claim construction positions of Plaintiff’s motion and Defendants’ motions are **DENIED** in all other respects.

C. **Doctrine of Equivalents Related Grounds for the Motions to Strike**

Defendants argue that Plaintiff’s disclosure of Plaintiff’s Doctrine of Equivalents (“DOE”) theory is untimely. *See* (Dkt. No. 721, at 25–26.) Defendants also argue that the DOE theory improperly attempts to revive a claim construction position previously rejected by the

Court, vitiates the Court's claim construction ruling, and is barred by prosecution history estoppel ("PHE"). *See (id. at 16–25.)* Relatedly, Apple argues that Plaintiff did not adequately or timely disclose that the use of the "Messages" program to transfer files between Apple employees met the "behavioral integrity" requirement of the patents-in-suit. *See (id. at 26–28.)* Plaintiff responds that, as an initial matter, Plaintiff properly amended its infringement contentions within 30 days of the Court's Claim Construction Order, as allowed under the Local Rules, and adequately disclosed Plaintiff's infringement theories in its infringement contentions. *See (Dkt. No. 740, at 18–21.)* Further, Plaintiff argues that it is not attempting to revive any rejected claim construction positions and that though the Court rejected Plaintiff's proposed claim construction, the Court did not expressly find that the "behavioral integrity" limitation could not be met through the use of an equivalent to a "digital certificate." *See (id. at 11–14.)* Plaintiff also argues that, in regard to Apple, any late disclosure of infringement theories, to the extent that such disclosure was late, was a result of Apple's own "deficient source code production and belated depositions." *See (id. at 20–21.)*

The Court will address the PHE arguments later in this Order. After considering the briefing and oral argument, the Court finds the DOE disclosures were timely. Accordingly, the Court **DENIES** Defendants' motion (Dkt. No. 721) as it pertains striking the DOE theories disclosed in Mr. Goodrich's and Mr. Martin's reports. The Court notes, however, that the Court expressly rejected the inclusion of the language, "in other words, an assurance that the software comes from a source known to the repository," in the construction of "behavioral integrity," because the "additional language . . . would tend to broaden the scope of the disputed term." (Dkt. No. 459, at 19–21.) In accordance with that decision, **IT IS ORDERED** that no expert

may opine or insinuate that a “digital certificate” is simply “an assurance that the software comes from a source known to the repository.”

D. Prosecution History Estoppel Related Grounds for the Motions to Strike

Plaintiff argues that portions of Dr. Kelly’s report improperly raise legal issues, such as ensnarement and PHE, which should not be presented to the jury. *See* (Dkt. No. 692, at 12–13.) Apple specifically responds that Dr. Kelly’s report properly discloses the underlying facts necessary to present Apple’s PHE arguments. *See* (Dkt. No. 738, at 11–12.) Relatedly, Defendants argue that Mr. Goodrich’s and Mr. Martin’s theories that “behavioral integrity” is met through the DOE are barred by PHE. *See* (Dkt. No. 721, at 21.) ContentGuard responds by arguing that, procedurally, the PHE arguments have been waived by Defendants’ silence prior to the *Daubert* motion and that, substantively, the patentee did not clearly surrender subject matter. *See* (Dkt. No. 740, at 15.) Defendants respond by arguing that the DOE infringement theory was not disclosed until April 20, 2015, a day before the deadline to file the letter briefs for dispositive motions, which is why the PHE arguments were not raised prior to the *Daubert* stage. *See* (Aug. 5, 2015, H’rg Tr., Dkt. No. 827, at 107:11–108:2.)

After considering the briefing and oral argument, the Court finds that PHE arguments are untimely, and as a result, those portions of Dr. Kelly’s report (Paragraphs 296–336) are stricken. Though Defendants argue that the PHE arguments are a response to Plaintiff’s Amended Infringement Contentions served on the day that the letter briefing was due, Defendants did not request an extension or any relief to rectify the problems which they assert were created by this alleged late amendment. For example, Defendants did not request leave to file a motion for summary judgment in which they could have laid out the particular timing issues at play here. Accordingly, the Court **GRANTS** Plaintiff’s motion (Dkt. No. 692) as it pertains to the prosecution history estoppel arguments contained in Dr. Kelly’s report and **DENIES**

Defendants' motion (Dkt. No. 721) as it pertains to a prosecution history estoppel bar to ContentGuard's assertion of the doctrine of equivalents.

E. ContentGuard's *Daubert* Motion to Exclude Portions of the Noninfringement Reports and Testimony Related to Rooting, Jailbreaking, and TunesKit Software

Plaintiff argues that portions of Defendants' noninfringement experts' reports and testimony are improper because Defendants' respective experts relied on facts, including tests, that were not properly disclosed during fact discovery, modified the accused instrumentalities such that any test results are irrelevant, and failed to disclose enough facts to determine whether the tests were reliable. *See* (Dkt. No. 688, at 3–15.) Defendants first respond by arguing that the facts underlying their tests were timely disclosed and that they did not need to disclose their testing methodology. *See, e.g.*, (Dkt. No. 750, at 3–5.) Defendants further argue that their test results are relevant because they prove that the accused instrumentalities do not meet the claim limitations of the Patents-in-Suit, particularly as they relate to the three integrities. *See, e.g.*, (Dkt. No. 721, at 1.) Finally, Defendants argue that the experts provided more than sufficient detail such that the tests could be reproduced and determined reliable. *See, e.g.*, (Dkt. No. 750, at 12–14.).

As an initial matter, the Court does not find the timing of the disclosure of this evidence to be improper. Defendants timely disclosed the facts that are the subject of the tests. However, after considering the briefing and oral argument, the Court does find that the rooting and jailbreaking related modifications made by Dr. Clark, Dr. Noble, and Dr. Kelly render the test results irrelevant and the portions from their respective reports and testimonies relating to such are stricken. After much consideration, the Court has determined that the issue is not whether it is possible to eventually modify the accused instrumentalities so that they no longer meet the claim limitations of the asserted patents. Rather, the question is whether the accused

instrumentalities, as sold, meet all the claim limitations as construed by the Court. Similarly, the Court also finds that the installation of the TunesKit Application renders Dr. Kelly's test results irrelevant and the portions from his report and testimony relating to such are stricken. Accordingly, the Court **GRANTS** Plaintiff's motion (Dkt. No. 688).

IV. CONCLUSION

Having considered all of Plaintiff's objections, Plaintiff's (1) Motion to Exclude Portions of the Reports and Testimony of Jean Renard Ward (Dkt. No. 679); (2) Motion to Exclude Portions of the Reports and Testimony of Dr. Brian Noble (Dkt. No. 683); (3) Motion to Exclude Portions of the Reports and Testimony of Dr. Steve White (Dkt. No. 684), (4) Motion to Exclude Portions of the Reports and Testimony of Atul Prakash (Dkt. No. 685), (5) Motion to Exclude Portions of the Noninfringement Reports and Testimony of Paul Clark, Brian Noble, John Kelly, and Gene Tsudik (Dkt. No. 688), (6) Motion to Exclude Portions of the Reports and Testimony of Dr. Paul Clark (Dkt. No. 690), (7) Motion to Exclude Portions of the Reports and Testimony of Dr. John P.J. Kelly (Dkt. No. 692) are **GRANTED** as specifically set forth above and **DENIED** in all other respects.

Having considered all of Defendants' objections, Defendants' Combined Motion to Strike Portions of the Expert Reports and Testimony of Michael T. Goodrich and David Martin (Dkt. No. 721) is **GRANTED** as specifically set forth above, **CARRIED** as to Mr. Goodrich's indirect infringement opinions regarding the OEM Defendants, and **DENIED** in all other respects.

Additionally, having ordered further briefing from the Parties regarding the Motion to Exclude Portions of the Reports and Testimony of Dr. John P.J. Kelly (Dkt. No. 692), the Court **CARRIES** that motion specifically in regard to ContentGuard's allegation that Apple relies on evidence unrelated to the Representative Products.

So ORDERED and SIGNED this 19th day of August, 2015.



RODNEY GILSTRAP
UNITED STATES DISTRICT JUDGE

**IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
MARSHALL DIVISION**

CONTENTGUARD HOLDINGS, INC.,

Plaintiff,

v.

GOOGLE, INC.,

Defendant.

§
§
§
§
§
§
§
§
§
§

Case No. 2:14-CV-61-JRG

MEMORANDUM OPINION AND ORDER

Before the Court are the following motions filed by Plaintiff ContentGuard Holdings, Inc. (“ContentGuard”) and Defendants Google, Inc., HTC America, Inc., HTC Corporation, Huawei Device USA, Inc., Huawei Technologies Co., Ltd., Motorola Mobility LLC, Samsung Electronics Co., Ltd., and Samsung TeleCommunications America, LLC (collectively, “Defendants”): (1) ContentGuard’s Motion for Judgment as a Matter of Law with Respect to the Google-Samsung Trial or, in the Alternative, for a New Trial (Dkt. No. 400¹; Dkt. No. 1038 in Case No. 2:13-cv-1112); (2) Defendants’ Motion for Judgment of Invalidity as a Matter of Law Pursuant to Federal Rule of Civil Procedure 50(b), and in the Alternative, Request for a New Trial Pursuant to Federal Rule of Civil Procedure 49 (Dkt. No. 397; Dkt. No. 1034 in Case No. 2:13-cv-1112); (3) Defendants’ Conditional Motion for Bench Trial on Defendants’ Inequitable Conduct Defenses (Dkt. No. 396; Dkt. No. 1032 in Case No. 2:13-cv-1112); and (4) Google’s Motion for Judgment of Laches (Dkt. No. 394; Dkt. No. 1038 in Case No. 2:13-cv-1112). For the reasons set forth below, the Court finds that each of these motions should be **DENIED**.

¹ Unless otherwise indicated, references to the docket are for Case No. 2:14-cv-61.

I. BACKGROUND

The Court held a jury trial in this case and the jury returned a unanimous verdict on September 23, 2015, that Defendants had not infringed United States Patents Nos. 6,963,859 (“the ’859 Patent”), 7,523,072 (“the ’072 Patent”), 8,370,956 (“the ’956 Patent”), and 8,393,007 (“the ’007 Patent”) (collectively, the “Trusted Repository Patents” or “Stefik Patents”); and 8,001,053 (“the ’053 Patent”) (the “Meta Rights Patent,” “Nguyen/Chen Patent,” or “Nguyen Patent”) (all, collectively, “the patents-in-suit”). The jury also found that Defendants had not proved that ContentGuard’s patents were invalid. ContentGuard and Defendants now uniformly assert that, in the approximately 36 hours of testimony and evidence presented at trial, the jury did not have sufficient evidence for its findings. The Court disagrees.

II. APPLICABLE LAW

Upon a party’s renewed motion for judgment as a matter of law following a jury verdict, the Court asks whether “the state of proof is such that reasonable and impartial minds could reach the conclusion the jury expressed in its verdict.” Fed. R. Civ. P. 50(b); *Am. Home Assur. Co. v. United Space Alliance*, 378 F.3d 482, 487 (5th Cir. 2004). “The grant or denial of a motion for judgment as a matter of law is a procedural issue not unique to patent law, reviewed under the law of the regional circuit in which the appeal from the district court would usually lie.” *Finisar Corp. v. DirectTV Group, Inc.*, 523 F.3d 1323, 1332 (Fed. Cir. 2008). “A JMOL may only be granted when, ‘viewing the evidence in the light most favorable to the verdict, the evidence points so strongly and overwhelmingly in favor of one party that the court believes that reasonable jurors could not arrive at any contrary conclusion.’” *Versata Software, Inc. v. SAP Am., Inc.*, 717 F.3d 1255, 1261 (Fed. Cir. 2013) (quoting *Dresser-Rand Co. v. Virtual Automation, Inc.*, 361 F.3d 831, 838 (5th Cir. 2004)).

Under Fifth Circuit law, a court is to be “especially deferential” to a jury’s verdict, and must not reverse the jury’s findings unless they are not supported by substantial evidence. *Baisden v. I’m Ready Productions, Inc.*, 693 F.3d 491, 499 (5th Cir. 2012). “Substantial evidence is defined as evidence of such quality and weight that reasonable and fair-minded men in the exercise of impartial judgment might reach different conclusions.” *Threlkeld v. Total Petroleum, Inc.*, 211 F.3d 887, 891 (5th Cir. 2000). A motion for judgment as a matter of law must be denied “unless the facts and inferences point so strongly and overwhelmingly in the movant’s favor that reasonable jurors could not reach a contrary conclusion.” *Baisden* 393 F.3d at 498 (citation omitted). However, “[t]here must be more than a mere scintilla of evidence in the record to prevent judgment as a matter of law in favor of the movant.” *Arismendez v. Nightingale Home Health Care, Inc.*, 493 F.3d 602, 606 (5th Cir. 2007).

In evaluating a motion for judgment as a matter of law, a court must “draw all reasonable inferences in the light most favorable to the verdict and cannot substitute other inferences that [the court] might regard as more reasonable.” *E.E.O.C. v. Boh Bros. Const. Co., L.L.C.*, 731 F.3d 444, 451 (5th Cir. 2013) (citation omitted). However, “[c]redibility determinations, the weighing of the evidence, and the drawing of legitimate inferences from the facts are jury functions, not those of a judge.” *Reeves v. Sanderson Plumbing Prods., Inc.*, 530 U.S. 133, 150 (2000). “[T]he court should give credence to the evidence favoring the nonmovant as well as that ‘evidence supporting the moving party that is uncontradicted and unimpeached, at least to the extent that that evidence comes from disinterested witnesses.’” *Id.* at 151 (citation omitted).

III. ANALYSIS

To prove infringement under 35 U.S.C. § 271, a plaintiff must show the presence of every element, or its equivalent, in the accused product or service. *Lemelson v. United States*,

752 F.2d 1538, 1551 (Fed. Cir. 1985). First, the claim must be construed to determine its scope and meaning; and second, the construed claim must be compared to the accused device or service. *Absolute Software, Inc. v. Stealth Signal, Inc.*, 659 F.3d 1121, 1129 (Fed. Cir. 2011) (citing *Carroll Touch, Inc. v. Electro Mech. Sys., Inc.*, 15 F.3d 1573, 1576 (Fed. Cir. 1993)). “A determination of infringement is a question of fact that is reviewed for substantial evidence when tried to a jury.” *ACCO Brands, Inc. v. ABA Locks Mfr. Co.*, 501 F.3d 1307, 1311 (Fed. Cir. 2007).

An issued patent is presumed valid. 35 U.S.C. § 282; *Fox Grp., Inc. v. Cree, Inc.*, 700 F.3d 1300, 1304 (Fed. Cir. 2012). Samsung has the burden to show by clear and convincing evidence that the asserted claims were anticipated by or obvious over the prior art. *Microsoft Corp. v. i4i Ltd. P’ship*, 131 S. Ct. 2238, 2242 (2011). To prevail on judgment as a matter of law, moreover, Samsung must show that no reasonable jury would have a legally sufficient evidentiary basis to find for the Plaintiff. FED. R. CIV. P. 50. “Generally, a party seeking to invalidate a patent as obvious must demonstrate by clear and convincing evidence that a skilled artisan would have had reason to combine the teaching of the prior art references to achieve the claimed invention, and that the skilled artisan would have had a reasonable expectation of success from doing so.” *In re Cyclobenzaprine Hydrochloride*, 676 F.3d 1063 (Fed. Cir. 2012) (internal quotation marks omitted).

A. Infringement of the Patents-In-Suit

ContentGuard argues that the Court should enter judgment of infringement as a matter of law because Defendants “failed to present a meritorious non-infringement defense” and instead “misled the jury to a verdict of non-infringement by repeatedly urging arguments the Court had already rejected in its *Markman* and other pre-trial orders.” (Inf. JMOL at 1.) More specifically,

ContentGuard argues that because “there is no dispute concerning the structure or operation of the accused device, . . . the issue of whether the claim language reads on the device is purely one of claim construction properly resolved by the Court.” *See (id. at 5.)* Further, ContentGuard argues that because all of Defendants’ noninfringement arguments “were contrary to the Court’s *Markman* and *Daubert* Orders, and thus legally incorrect,” the Court should enter judgment of infringement as a matter of law. *See (id. at 6.)*

For example, ContentGuard asserts that Defendants’ argument that the content and the usage rights must travel together was inconsistent with both the Court’s *Markman* and *Daubert* orders rejecting the idea of “permanent” attachment. (*Id. at 6–10.*) Similarly, ContentGuard claims that “non-infringement argument Defendants advanced based on the file-moving experiments performed by Dr. Clark[, Defendants’ technical expert] should also be rejected” because, “as a matter of law, copying and moving encrypted content is not the same as ‘access[ing]’ the content.” *See (id. at 10–12.)* ContentGuard also argues that Defendants’ “file-moving defense” violated the Court’s *Daubert* order, which prohibited argument that the three integrities must be maintained “at all times.” (*Id. at 11.*) Finally, ContentGuard argues that because books and movies are data, rather than software, Defendants’ argument that movie and book files are downloaded from the Google Play store without digital certificates and thus their products lack “behavioral integrity” is incorrect. *See (id. at 11–13.)*

In the alternative, ContentGuard argues that new trial should be ordered because “Defendants’ conduct clearly violated the Court’s *Markman* and *Daubert* orders,” Defendants unfairly used the Court’s *Daubert* order to create alleged inconsistencies in ContentGuard’s positions, and “that the Court erred when it permitted Defendants to pursue a ‘practicing the prior art’ defense before the jury.” (*Id. at 13–18.*)

Defendants respond by first arguing that “ContentGuard admits that Defendants never expressly violated the Court’s claim construction and *Daubert* orders, but argues that Defendants implicitly argued to the jury that the attachment of usage rights must be permanent.” *See* (Dkt. No. 451, “Inf. JMOL Resp.”, at 3.) Further, Defendants argue that “Defendants never suggested to the jury that attachment must be permanent,” but rather “Defendants’ evidence addressed how to apply the Court’s constructions to the accused Google Play system.” (*Id.* at 3, 5–13) Defendants also argue that “the Court already determined that the application of its ‘attached or treated as attached’ construction for ‘usage rights’ was a fact question for the jury, [that] ContentGuard agreed with the Court on this point,” that “ContentGuard failed to object on this basis on the record at trial when the complained-of evidence was introduced,” and that the Court’s construction required that “attachment” be more than mere “association.” *See* (*id.* at 3–5, 14–23.)

As to the factual issue of whether the Google Play system uses “usage rights,” Defendants argue that they “presented substantial and compelling evidence to the jury.” (*Id.* at 4.) For example, Dr. Clark, Defendants’ technical expert, testified as follows regarding a test he performed on the accused devices:

Question: And, sir, can you please describe the demonstration that you did with those phones in order to come to your opinions in this case?

Answer: Sure. So the -- the test that I performed was that I took Device 1, this device (indicating), and I purchased a movie on it as a user. And then on this device, I rented a movie -- the same movie as a different user. And then I downloaded the movie files, and then I deleted the movie file from this file -- from the rental device and transferred the purchased movie to the rental device. And then when I played it on the rental device, it was still treated as a rental, and that told me that the usage rights from the purchased movie were not attached and were not transferred when I copied it over.

(9/21/2015 P.M. Trial Tr. (Clark), Dkt. No. 440, at 78:1–16.) Dr. Clark then testified as follows

about the implication of the test results to an infringement analysis:

Question: And, sir, what did you conclude from that demonstration?

Answer: That the system could not be understood to be treating the license and the content as attached if I could move them around independently and have different usage conditions associated with each device in the same file.

(*Id.* at 84:18–23.)

In regard to the issue of “physical integrity,” Defendants assert that (1) “Dr. Clark presented evidence of testing he performed that showed that the accused Google Play apps did not exhibit physical integrity,” including manipulating purchased content on the accused devices, and (2) ContentGuard did not even address Dr. Clark’s testimony that Google Play’s support of “removable memory prevents the accused devices from maintaining physical security, [and thus] the accused system lacked physical integrity.” (Inf. JMOL Resp. at 23–25.) Defendants further assert that “Defendants never contended that physical integrity must be present at all times, and ContentGuard does not point to any such evidence or argument.” (*Id.* at 26–28.) Defendants argue that, in fact, ContentGuard’s own witnesses’ testimony confirmed that Defendants’ explanation of the “physical integrity” limitation was correct. (*Id.* at 26–29.)

As to “behavioral integrity,” Defendants argue that “Defendants presented both testimony and documentary evidence showing that Google Play digital content (movies, TV, and books) is “software” and that “much of this evidence was from the Stefik patents and Dr. Stefik himself.” (*Id.* at 31–34.) Further, Defendants argue “[a]lthough ContentGuard suggested to the jury that the explicit inclusion of digital content files in the definition of ‘software’ in language in both the patents and the glossary definitions of Dr. Stefik’s article was merely ‘metaphorical,’ the jury was entitled to disbelieve this argument.” (*Id.* at 33–34.) Finally, Defendants argue (1) that the Court had previously considered and rejected ContentGuard’s argument that Google Play digital

content can be “software” as used in the patents-in-suit and (2) ContentGuard failed to object during trial to this evidence. (*Id.* at 38.)

As to ContentGuard’s arguments regarding the alleged “practicing the prior art” defense, Defendants first argue that “ContentGuard points to no argument by Defendants that the accused Google Play system did not infringe because it practiced the prior art.” (*Id.* at 39.) Defendants argue that, in fact, they “did not claim that they practiced the prior-art Griswold system, but rather that they practiced a license server system similar to the Griswold system, which Dr. Stefik himself had distinguished from his trusted system with attached usage rights.” (*Id.*) For example, Dr. Clark testified as follows regarding the “license server” approach and the Stefik approach:

Question: Sir, what are you showing in this slide?

Answer: This is actually a ContentGuard presentation in which there appear to be -- distinguishing the trusted system, that is the system where the -- the user devices are trusted from what they show as a conventional DRM design using a license server.

Question: And, sir, which of these ContentGuard documents shows, in your view, the license server approach?

Answer: The one on the left that has the license server in it.

Question: And, sir, what does the slide on the right then show?

Answer: The one on the right shows that the retailer next to the publisher is going to attach rights and conditions, basically he’s going to package the digital work, the protected content with those things before it goes to the consumer.

Question: And in which of these is the license attached to the content in your view?

Answer: In the trusted system approach.

Question: And, sir, does one of these slides correspond to the Griswold license server system, in your view?

Answer: Sure. Griswold is kind of a classic license server DRM system.

Question: And does one of these correspond to Dr. Stefik's approach from his patent, in your view?

Answer: Yes. I mean, you need trusted systems and you need those systems to have attached usage rights that accompany the works, and those are enforced by the devices, which is shown on the right.

Question: And does -- does one of these diagrams correspond to what you understand Google's Google Play and Google Play Books and Movies apps, what their system looks like?

Answer: As -- as we've seen, the Google Play Movies and Books use the license server.

(9/21/2015 P.M. Trial Tr. (Clark), Dkt. No. 440, at 99:15–100:22.) Defendants also argue that they “never argued to the jury that the scope of the patent claims was narrower because of the specification,” and “[i]nstead, Defendants argued that Google Play did not meet the Court's construction of the claim terms ‘usage rights’ and ‘trusted’ repositories.” (Inf. JMOL Resp. at 44–45.)

First, the Court agrees with Defendants that their noninfringement arguments were within the scope of the Court's prior orders. ContentGuard has already raised and the Court has previously addressed the claim construction issue in regard to “usage rights.” *See, e.g.*, (9/15/2015 A.M. Trial Tr., Dkt. No. 428, at 6:13–24, 8:19–9:3, 16:16–23.) The Court declines to revisit that ruling. Further, the jury was properly instructed on the law, was free to judge the credibility of witnesses, and weigh all competing evidence. After consideration of the evidence presented at trial, including the tests that Dr. Clark performed on the accused devices, the jury found that Defendants had not infringed the patents-in-suit. *See, e.g.*, (9/21/2015 P.M. Trial Tr. (Clark), Dkt. No. 440, at 78:1–16, 84:18–23.) The jury, acting under a preponderance of the evidence standard as to this disputed factual issue, unanimously reached a reasoned and supportable decision. Accordingly, the Court will not supplant the judgment of the jury. The Court does not find that no reasonable jury could have found that Defendants had not infringed

the patents-in-suit, and thus the Court must deny the motion for judgment as a matter of law on this issue.

Furthermore, the issues raised by ContentGuard do not warrant a new trial. The Court does not find that Defendants presented a “practicing the prior art” defense. Instead, Defendants properly distinguished their system from the systems described in the patents-in-suit. Further, the Court properly and specifically instructed the jury that when answering the question of infringement, they were only to compare the accused products to the patent claims, and were never to compare the accused products to the prior art. *See* (9/23/2015 P.M. Trial Tr., Dkt. No. 445, at 52:10–21.) The Court does not find that a new trial is warranted in this case. Accordingly, ContentGuard’s motion for judgment of infringement as a matter of law or, alternatively, a new trial is **DENIED** in all respects.

B. Invalidity of the Patents-in-Suit

Defendants request that the Court overturn the jury’s verdict and enter judgment that the patents-in-suit are invalid as obvious as a matter of law. (Dkt. No. 397, “Inv. JMOL,” at 3.) Defendants argue that “evidence presented at trial compelled a finding that the Asserted Claims of the Stefik Patents were anticipated by the DOD Orange Book, formally titled ‘Trusted Computer System Evaluation.’” (*Id.*) In particular, Defendants argue that ContentGuard only “disputed that two elements of the asserted Stefik Patent claims were disclosed in the DOD Orange Book: a ‘repository’/‘trusted’ system and ‘usage rights’” and that “[i]n view of the evidence presented, however, no reasonable jury could conclude that the DOD Orange Book did not teach these elements.” (*Id.*) Further, Defendants argue that the claims were obvious in view of the DOD Orange Book in combination with Harn or the CNI Proceedings Book or the Kahn and Cerf References. (*Id.* at 12–16.) Additionally, Defendants argue that they are obvious in

view of the Griswold patent publication and the CNI Proceedings. (*Id.* at 16–18.) Defendants also assert that the Meta-Rights patent is invalid in light of the '980 patent. (*Id.* at 19–23.) Finally, Defendants argue that new trial is warranted because of attorney misconduct during closing argument. (*Id.* at 23–26.)

In addressing the challenges to Mr. Ward's testimony on the disclosure of the "trusted," "repository," and "usage rights" limitations in the DOD Orange Book, Defendants argue that Dr. Goodrich only gave conclusory testimony that the DOD Orange Book did not disclose these limitations as taught in the patents-in-suit. (*Id.* at 4–12.) Similarly, Defendants argue that "Dr. Goodrich's wholly conclusory statements [about the DOD Orange Book combinations] could not create a fact issue that the jury could determine against Defendants." (*Id.* at 12–16.) As for ContentGuard's criticism that Griswold "did not teach usage rights because the usage rights are never on the same device as the digital content," Defendants argue that "[a] reasonable jury could not rely on this testimony to find against Defendants on invalidity because the Griswold references clearly disclosed that an alternative, off-line, embodiment was also possible, and that off-line embodiment did not involve datagrams repeatedly checking for authorization." (*Id.* at 16–19.) Additionally, Defendants argue that no reasonable jury could find that the Stefik '980 patent did not disclose the "meta-rights" and "state variable" elements through the '980 patent's discussion of "Next-Set-of-Rights" and "Digital Work State Information." (*Id.* at 21–23) Finally, Defendants argue that certain comments made by ContentGuard's attorney during closing were improper and prejudicial and thus new trial is warranted. (*Id.* at 23–26.)

ContentGuard responds by arguing that "[a]s a threshold matter, JMOL should be denied because Defendants' invalidity expert, Mr. Ward, failed to establish that the [DOD] Orange Book discloses a number of elements of the asserted claims." (Dkt. No. 450, "Inv. JMOL Resp.,")

at 3–5.) Further, ContentGuard argues that “Mr. Ward failed to establish that the [DOD] Orange Book discloses *any* of the three integrities.” (*Id.* at 5–9 (emphasis in original)) For example, ContentGuard argues that “Mr. Ward admitted during direct-examination that the [DOD] Orange Book has no explicit disclosure of a *digital certificate*”:

Question: Sir, I don’t see the words “digital certificate” anywhere on these pages; is that correct?

Answer: That’s correct.

(*Id.* at 5; 9/22/2015 A.M. Tr. (Ward), Dkt. No. 441, at 62:23–25.) ContentGuard also argues that “Mr. Ward provided no testimony and pointed to no disclosure in the [DOD] Orange Book that required the use of *digital certificates*” and that “Mr. Ward failed to establish that any of the prior art references teach *digital certificates* as required by the Court’s claim construction.” (*Id.* at 6–7, 10–12 (emphasis in original).) Additionally, ContentGuard argues that “Dr. Goodrich explained that the Orange Book discloses that software is installed after a months-long process that involves ‘hands-on’ involvement with and testing of the source code” and that such a process is “completely different from the digital certificate required by the Court’s construction.” (*Id.* at 6.) Moreover, ContentGuard argues that Mr. Ward only provided conclusory testimony as to the disclosure of “communications integrity” and “physical integrity.” (*Id.* at 7–9.)

In regard to the “usage rights,” ContentGuard asserts that Mr. Ward provided merely conclusory testimony that “modes of access” discloses the manner of use necessary for “usage rights,” while Dr. Goodrich provided specific testimony explaining why “usage rights” were not taught by the DOD Orange Book. (*Id.* at 9–10.) For example, Dr. Goodrich testified as follows regarding “usage rights”:

Q. And why did you include this phrase for enforcing usage rights after the three integrities?

A. Because in -- in the Orange Book system, there isn't usage rights the way the Court is construing this term. Instead, they have labels that come out to say, if it's top secret, secret, and so on. And users, then -- not some other system where usage rights would be attached or treated as attached to content, but users themselves just get to say what they can do with it, based on the security clearance they have.

(9/22/2015 A.M. Trial Tr. (Goodrich), Dkt. No. 442, at 94:15–24.) Similarly, in regard to the combination of Griswold and the CNI Proceedings Book, ContentGuard argues that “Mr. Ward failed to establish that the combination of the Griswold Patent Publication and Griswold’s chapter in the CNI Proceedings Book disclosed any of the three integrities in support of usage rights.” (*Id.* at 12–13.) Further, in regard to the Stefik ’980 patent, ContentGuard argues that “Mr. Ward merely asserted in conclusory fashion that the prior art disclosed a ‘meta-right’ and a ‘state variable’ without ever stating whether or how the claim limitations were met.” (*Id.* at 14.) ContentGuard also notes that “the portion of Mr. Ward’s testimony cited by Defendants spans less than three pages, and does not even purport to cover the limitations recited in the asserted Meta-Rights Patent claim.” (*Id.*) Finally, in regard to the issue of attorney misconduct, ContentGuard argues that the comments made during closing were proper characterizations of the witness’s testimony and that the other alleged misconduct, to the extent it even occurred, related to infringement rather than invalidity. (*Id.* at 16–18.)

After consideration of the evidence presented, including testimony from Mr. Ward admitting that the DOD Orange Book did not use the term “digital certificates,” the jury found that Defendants had not proved the patents-in-suit were invalid. *See* (9/22/2015 A.M. Tr. (Ward), Dkt. No. 441, at 62:23–25.) The jury reached a reasoned and supportable decision, and the Court does not find that no reasonable jury could have found that the patents-in-suit are valid. Further, the Court does not find that a new trial is warranted in this case. During oral argument on post-trial motions, Defendants did not deny that any attorney misconduct that may have occurred was

substantially directed toward infringement, an issue on which Defendants prevailed, as opposed to invalidity. Thus, Defendants' motion for judgment as a matter of law or, alternatively, for a new trial as to the validity of the patents-in-suit is **DENIED**.

C. Motion for Judgment as to Laches and Motion for Bench Trial on Defendants' Inequitable Conduct Defenses

Defendants conditionally request a bench trial on their inequitable conduct defenses. (Dkt. No. 396 at 1.) Additionally, Google requests judgment that ContentGuard's damages are limited due to laches. (Dkt. No. 394 at 1.) Both requests are predicated upon a finding of infringement. However, the jury in this case did not find infringement and awarded no damages. As explained above, the Court declines to overturn the jury's verdict of non-infringement. Resultantly, the Court finds that both of these motions (Dkt. Nos. 396 and 394) should be and hereby are **DENIED AS MOOT**.

IV. CONCLUSION

For the reasons set forth above, the Court finds that the jury's verdict of both noninfringement and validity should stand. The jury's verdict as a whole is supported by adequate evidence presented at trial, and it should not be disturbed. Accordingly, ContentGuard's Motion for Judgment as a Matter of Law with Respect to the Google-Samsung Trial or, in the Alternative, for a New Trial (Dkt. No. 400; Dkt. No. 1038 in Case No. 2:13-cv-1112) and Defendants' Motion for Judgment of Invalidity as a Matter of Law Pursuant to Federal Rule of Civil Procedure 50(b), and in the Alternative, Request for a New Trial Pursuant to Federal Rule of Civil Procedure 49 (Dkt. No. 397; Dkt. No. 1034 in Case No. 2:13-cv-1112) are **DENIED**. Further, after reviewing the motions and the arguments contained within, the Court finds that Defendants' Conditional Motion for Bench Trial on Defendants' Inequitable Conduct Defenses (Dkt. No. 396; Dkt. No. 1032 in Case No. 2:13-cv-1112); and Google's Motion for

Judgment of Laches (Dkt. No. 394) are **MOOT** and they should be and are hereby **DENIED** as such.

So ORDERED and SIGNED this 8th day of July, 2016.



RODNEY GILSTRAP
UNITED STATES DISTRICT JUDGE



US006963859B2

(12) **United States Patent**
Stefik et al.

(10) **Patent No.: US 6,963,859 B2**
 (45) **Date of Patent: Nov. 8, 2005**

(54) **CONTENT RENDERING REPOSITORY**
 (75) Inventors: **Mark J. Stefik**, Portola Valley, CA (US); **Peter L. Pirolli**, San Francisco, CA (US)

3,790,700 A 2/1974 Callais et al.
 3,798,605 A 3/1974 Feistel
 4,159,468 A 6/1979 Barnes et al.
 4,220,991 A 9/1980 Hamano et al.

(Continued)

(73) Assignee: **ContentGuard Holdings, Inc.**,
 Wilmington, DE (US)

FOREIGN PATENT DOCUMENTS

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 78 days.

EP 0 084 441 7/1983
 EP 0 180 460 5/1986
 EP 0 332 707 9/1989
 EP 0 651 554 5/1995
 EP 0 668 695 8/1995

(Continued)

(21) Appl. No.: **10/345,390**

(22) Filed: **Jan. 16, 2003**

OTHER PUBLICATIONS

(65) **Prior Publication Data**

US 2003/0225699 A1 Dec. 4, 2003

Weber, Robert. Digital Rights Management Technologies. Oct. 1995. Retrieved from IDS.*
 "National Semiconductor and EPR Partner for Information Metering/Data Security Cards" Mar. 4, 1994, Press Release from Electronic Publishing Resources, Inc.

Related U.S. Application Data

(Continued)

(63) Continuation of application No. 09/778,006, filed on Feb. 7, 2001, now Pat. No. 6,714,921, which is a division of application No. 08/967,084, filed on Nov. 10, 1997, now Pat. No. 6,236,971, which is a continuation of application No. 08/344,760, filed on Nov. 23, 1994, now abandoned.

Primary Examiner—James A Reagan

(74) *Attorney, Agent, or Firm*—Marc S. Kaufman; Nixon Peabody, LLP

(51) **Int. Cl.**⁷ **G06F 17/60**
 (52) **U.S. Cl.** **705/51; 705/52; 705/53; 705/54; 705/55; 705/56; 705/57; 705/58; 705/59; 705/50; 380/201; 707/9; 707/104.1; 713/182; 713/183; 713/184; 713/185; 713/186**
 (58) **Field of Search** **705/50–59; 380/201, 380/30; 707/9, 104.1; 713/182–186, 156; 235/449; 379/93**

(57) **ABSTRACT**

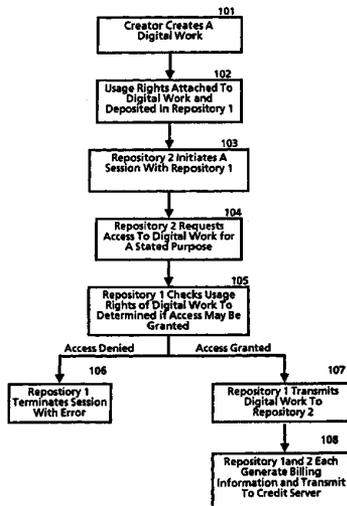
A rendering system adapted for use in a system for managing use of content and operative to rendering content in accordance with usage rights associated with the content. The system includes a rendering device configured to render the content and a repository coupled to the rendering device and operative to enforce usage rights associated with the content and permit the rendering device to render the content in accordance with a manner of use specified by the usage rights.

(56) **References Cited**

U.S. PATENT DOCUMENTS

3,263,158 A 7/1966 Bargaen et al.
 3,609,697 A 9/1971 Blevins et al.

84 Claims, 13 Drawing Sheets



US 6,963,859 B2

Page 2

U.S. PATENT DOCUMENTS

4,278,837 A 7/1981 Best
 4,323,921 A 4/1982 Guillou
 4,442,486 A 4/1984 Mayer
 4,529,870 A 7/1985 Chaum
 4,558,176 A 12/1985 Arnold et al.
 4,593,376 A 6/1986 Volk
 4,614,861 A 9/1986 Pavlov et al.
 4,644,493 A 2/1987 Chandra et al.
 4,658,093 A 4/1987 Hellman
 4,713,753 A 12/1987 Boebert et al.
 4,796,220 A 1/1989 Wolfe
 4,817,140 A 3/1989 Chandra et al.
 4,827,508 A 5/1989 Shear
 4,868,376 A 9/1989 Lessin et al.
 4,891,838 A 1/1990 Faber
 4,924,378 A 5/1990 Hershey et al.
 4,932,054 A 6/1990 Chou et al.
 4,937,863 A 6/1990 Robert et al.
 4,949,187 A 8/1990 Cohen
 4,953,209 A 8/1990 Ryder, Sr. et al.
 4,961,142 A 10/1990 Elliott et al.
 4,975,647 A 12/1990 Downer et al.
 4,977,594 A 12/1990 Shear
 4,999,806 A 3/1991 Chernow et al.
 5,010,571 A 4/1991 Katznelson
 5,014,234 A 5/1991 Edwards, Jr.
 5,023,907 A 6/1991 Johnson et al.
 5,047,928 A 9/1991 Wiedemer
 5,050,213 A 9/1991 Shear
 5,052,040 A 9/1991 Preston et al.
 5,058,164 A 10/1991 Elmer et al.
 5,103,476 A 4/1992 Waite et al.
 5,113,519 A 5/1992 Johnson et al.
 5,136,643 A 8/1992 Fischer
 5,138,712 A * 8/1992 Corbin 713/200
 5,146,499 A 9/1992 Geffrotin
 5,148,481 A 9/1992 Abraham et al.
 5,159,182 A 10/1992 Eisele
 5,183,404 A 2/1993 Aldous et al.
 5,191,193 A 3/1993 Le Roux
 5,204,897 A 4/1993 Wyman
 5,222,134 A 6/1993 Waite et al.
 5,235,642 A 8/1993 Wobber et al.
 5,247,575 A 9/1993 Sprague et al.
 5,255,106 A 10/1993 Castro
 5,260,999 A * 11/1993 Wyman 705/59
 5,263,157 A 11/1993 Janis
 5,263,158 A 11/1993 Janis
 5,276,444 A 1/1994 McNair
 5,276,735 A 1/1994 Boebert et al.
 5,291,596 A 3/1994 Mita
 5,295,266 A * 3/1994 Hinsley et al. 718/101
 5,301,231 A 4/1994 Abraham et al.
 5,311,591 A 5/1994 Fischer
 5,319,705 A 6/1994 Halter et al.
 5,335,346 A * 8/1994 Fabbio 711/163
 5,337,357 A 8/1994 Chou et al.
 5,339,091 A 8/1994 Yamazaki et al.
 5,339,392 A * 8/1994 Risberg et al. 345/762
 5,341,429 A 8/1994 Stringer et al.
 5,347,579 A 9/1994 Blandford
 5,381,526 A 1/1995 Ellson
 5,394,469 A 2/1995 Nagel et al.
 5,410,598 A 4/1995 Shear
 5,412,717 A 5/1995 Fischer
 5,428,606 A 6/1995 Moskowitz
 5,432,849 A 7/1995 Johnson et al.
 5,438,508 A 8/1995 Wyman
 5,444,779 A 8/1995 Daniele
 5,453,601 A 9/1995 Rosen

5,455,953 A 10/1995 Russell
 5,457,746 A 10/1995 Dolphin
 5,473,687 A 12/1995 Lipscomb et al.
 5,473,692 A 12/1995 Davis
 5,499,298 A 3/1996 Narasimhalu et al.
 5,502,766 A 3/1996 Boebert et al.
 5,504,814 A 4/1996 Miyahara
 5,504,818 A 4/1996 Okano
 5,504,837 A 4/1996 Griffeth et al.
 5,509,070 A 4/1996 Schull
 5,530,235 A 6/1996 Stefik et al.
 5,532,920 A 7/1996 Hartrick et al.
 5,534,975 A 7/1996 Stefik et al.
 5,539,735 A 7/1996 Moskowitz
 5,563,946 A 10/1996 Cooper et al.
 5,568,552 A 10/1996 Davis
 5,621,797 A 4/1997 Rosen
 5,629,980 A 5/1997 Stefik et al.
 5,633,932 A 5/1997 Davis et al.
 5,634,012 A 5/1997 Stefik et al.
 5,638,443 A 6/1997 Stefik et al.
 5,649,013 A 7/1997 Stuckey et al.
 5,655,077 A 8/1997 Jones et al.
 5,708,717 A 1/1998 Alasia
 5,734,823 A 3/1998 Saigh et al.
 5,734,891 A 3/1998 Saigh
 5,737,413 A 4/1998 Akiyama et al.
 5,737,416 A 4/1998 Cooper et al.
 5,745,569 A 4/1998 Moskowitz et al.
 5,748,783 A 5/1998 Rhoads
 5,757,907 A 5/1998 Cooper et al.
 5,761,686 A 6/1998 Bloomberg
 5,765,152 A 6/1998 Erickson
 5,768,426 A 6/1998 Rhoads
 5,825,892 A 10/1998 Braudaway et al.
 5,892,900 A 4/1999 Ginter et al.
 5,910,987 A 6/1999 Ginter et al.
 5,915,019 A 6/1999 Ginter et al.
 5,917,912 A 6/1999 Ginter et al.
 5,920,861 A 7/1999 Hall et al.
 5,940,504 A 8/1999 Griswold
 5,943,422 A 8/1999 Van Wie et al.
 5,949,876 A 9/1999 Ginter et al.
 5,982,891 A 11/1999 Ginter et al.
 5,999,949 A 12/1999 Crandall
 6,047,067 A 4/2000 Rosen
 6,112,181 A 8/2000 Shear et al.
 6,115,471 A 9/2000 Oki et al.
 6,138,119 A 10/2000 Hall et al.
 6,157,721 A 12/2000 Shear et al.
 6,185,683 B1 2/2001 Ginter et al.
 6,226,618 B1 5/2001 Downs et al.
 6,233,684 B1 5/2001 Stefik et al.
 6,237,786 B1 5/2001 Ginter et al.
 6,240,185 B1 5/2001 Van Wie et al.
 6,253,193 B1 6/2001 Ginter et al.
 6,266,618 B1 7/2001 Ye et al.
 6,292,569 B1 9/2001 Shear et al.
 6,301,660 B1 10/2001 Benson
 6,327,652 B1 12/2001 England et al.
 6,330,670 B1 12/2001 England et al.
 6,345,256 B1 2/2002 Milsted et al.
 6,363,488 B1 3/2002 Ginter et al.
 6,389,402 B1 5/2002 Ginter et al.

FOREIGN PATENT DOCUMENTS

EP 0 725 376 8/1996
 GB 2 136 175 9/1984
 GB 2 236 604 4/1991
 JP 62-241061 10/1987
 JP 64-068835 3/1989

US 6,963,859 B2

Page 3

JP	H03-282733	* 12/1991 G06F/9/06
JP	04-369068	12/1992	
JP	05-268415	10/1993	
JP	06-175794	6/1994	
JP	06-215010	8/1994	
JP	07-084852	3/1995	
JP	07-200317	8/1995	
JP	07-244639	9/1995	
JP	0 715 241	6/1996	
WO	WO 92/20022	11/1992	
WO	WO 93/01550	1/1993	
WO	WO 94/01821	1/1994	
WO	WO 96/24092	8/1996	
WO	WO 97/48203	12/1997	
WO	WO 98/11690	3/1998	
WO	WO 98/42098	9/1998	
WO	WO 99/49615	9/1999	
WO	WO 01/63528	8/2001	

OTHER PUBLICATIONS

Weber, R., "Digital Rights Management Technology" Oct. 1995.

Flasche, U. et al., "Decentralized Processing of Documents", pp. 119-131, 1986, *Comput. & Graphics*, vol. 10, No. 2.

Mori, R. et al., "Superdistribution: The Concept and the Architecture", pp. 1133-1146, 1990. *The Transactions of the IEICE*, Vol. E 73, No. 7, Tokyo, JP.

Weber, R., "Metering Technologies for Digital Intellectual Property", pp. 1-29, Oct. 1994, A Report to the International Federation of Reproduction Rights Organizations.

Clark, P.C. et al., "Bits: A Smartcard protected Operating System", pp. 66-70 and 94, Nov. 1994, *Communications of the ACM*, vol. 37, No. 11.

Ross, P.E., "Data Guard", pp. 101, Jun. 6, 1994, *Forbes*.

Saigh, W.K., "Knowledge is Sacred", 1992, Video Pocket/Page Reader Systems, Ltd.

Kahn, R.E., "Deposit, Registration and Recordation in an Electronic Copyright Management System", pp. 1-19, Aug. 1992, Corporation for National Research Initiatives, Virginia.

Hilts, P. et al., "Books While U Wait", pp. 48-50, Jan. 3, 1994, *Publishers Weekly*.

Stratner, A., "Cash Register on a Chip may Revolutionize Software Pricing and Distribution; Wave Systems Corp.", pp. 1-3, Apr. 1994, *Computer Shopper*, vol. 14, No. 4, ISSN 0886-0556.

O'Conner, M., "New Distribution Option for Electronic Publishers; iOpener Data Encryption and Metering System for CD-ROM use; Column", pp. 1-6, Mar. 1994, *CD-ROM Professional*, vol. 7, No. 2, ISSN: 1409-0833.

Willett, S., "Metered PCs: Is Your System Watching You? Wave System beta tests new technology", pp. 84, May 2, 1994, *InfoWorld*.

Linn, R., "Copyright and Information Services in the Context of the National Research and Education Network", pp. 9-20, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1, Issue 1.

Perrit, Jr., H., "Permission Headers and Contract Law", pp. 27-48, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1, Issue 1.

Upthegrove, L., "Intellectual Property Header Descriptors: A Dynamic Approach", pp. 63-66, Jan. 1994, *IMA Intellectual Property Proceedings*, vol. 1, Issue 1.

Sirbu, M., "Internet Billing Service Design and prototype Implementation", pp. 67-80, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1, Issue 1.

Simmell, S. et al., "Metering and Licensing of Resources: Kala's General Purpose Approach", pp. 81-110, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1, Issue 1.

Kahn, R., "Deposit Registration and Recordation in an Electronic Copyright Management System", pp. 111-120, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1, Issue 1.

Tygar, J. et al., "Dyad: A System for Using Physically Secure Coprocessors", pp. 121-152, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1, Issue 1.

Griswold, G., "A Method for Protecting Copyright on Networks", pp. 169-178, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1 Issue 1.

Nelson, T., "A Publishing and Royalty Model for Networked Documents", pp. 257-259, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1, Issue 1.

Robinson, E., "Redefining Mobile Computing", pp. 238-240, 247-248 and 252, Jul. 1993, *PC Computing*.

Abadi, M. et al., "Authentication and Delegation with Smart-cards", pp. 1-24, 1990, *Research Report DEC Systems Research Center*.

Mark Stefik, "Letting Loose the Light: Igniting Commerce in Electronic Publication", pp. 219-253, 1996, *Internet Dreams: Archetypes, Myths, and Metaphors*, IDSN 0-262-19373-6.

Mark Stefik, "Letting Loose the Light: Igniting Commerce in Electronic Publication", pp. 2-35, Feb. 8, 1995, *Internet Dreams: Archetypes, Myths and Metaphors*.

Henry H. Perritt, Jr., "Technological Strategies for Protecting Intellectual Property in the Networked Multimedia Environment", Apr. 2-3, 1993, *Knowbots, Permissions Headers & Contract Law*.

* cited by examiner

Figure 1

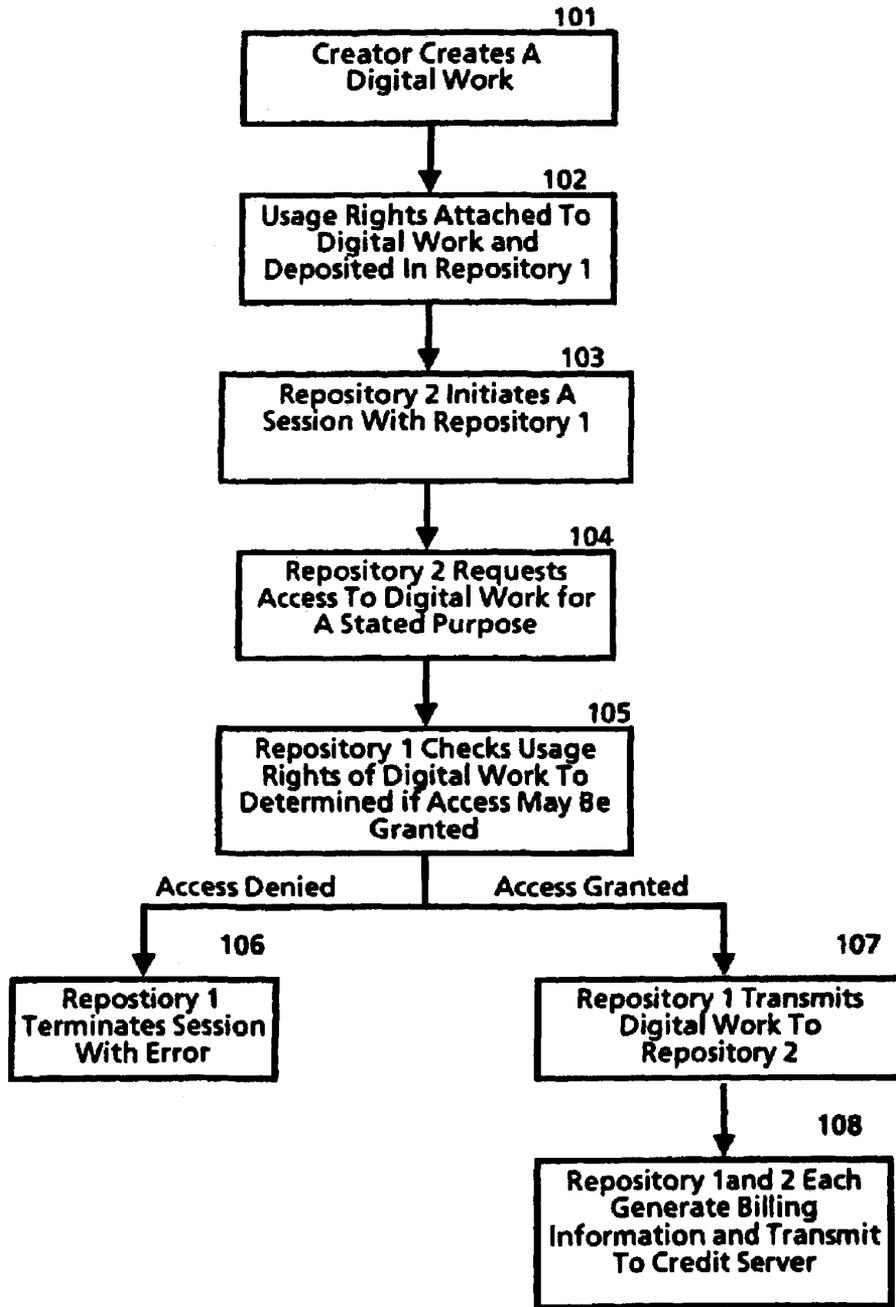


Figure 2

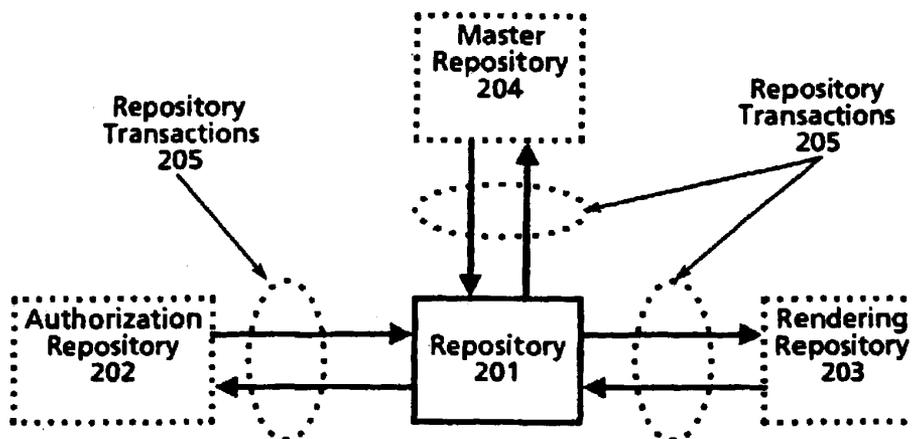


Figure 3

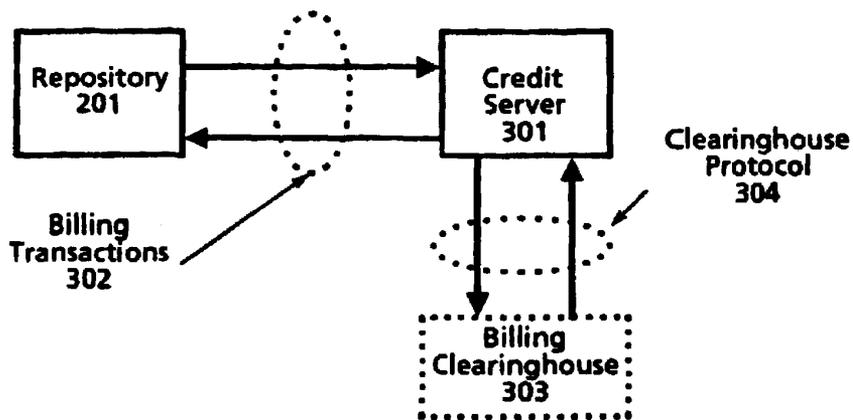


Figure 4a

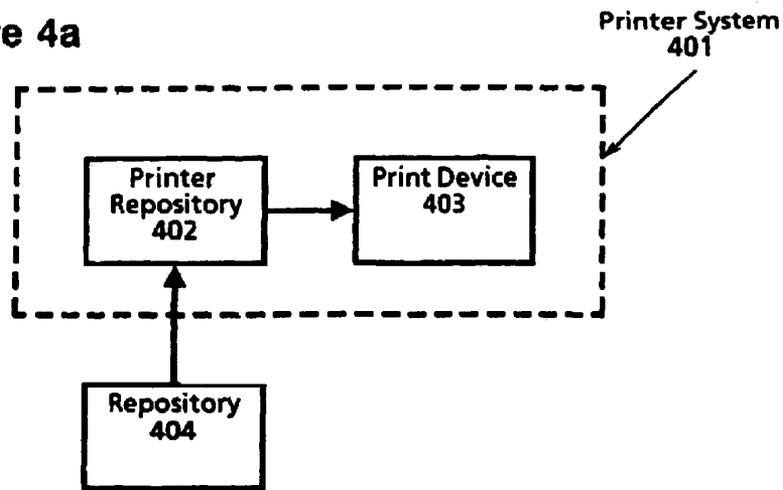
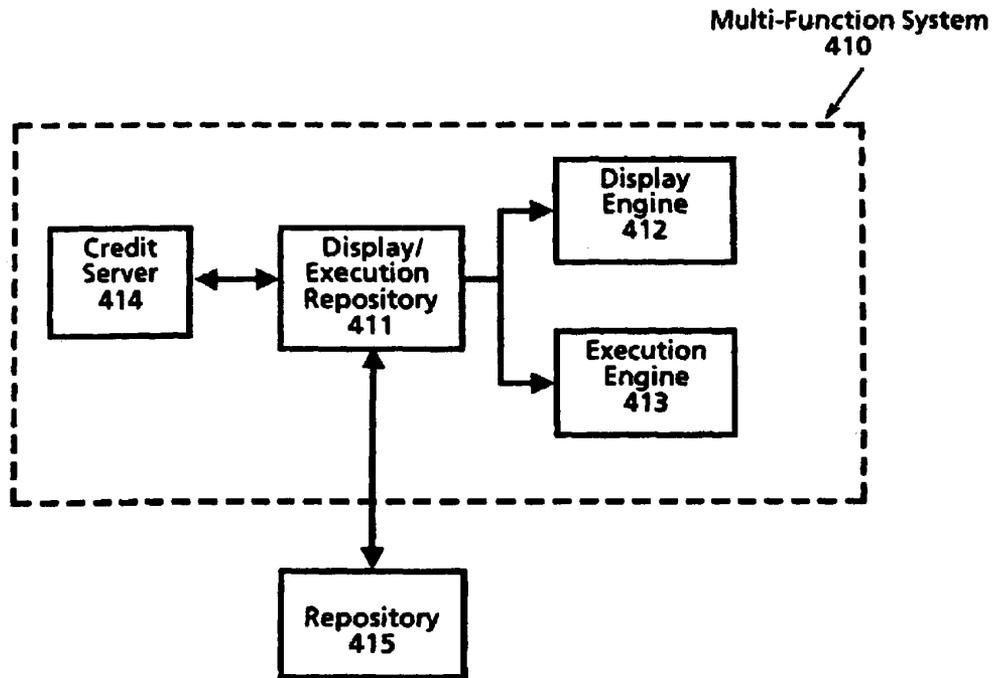


Figure 4b



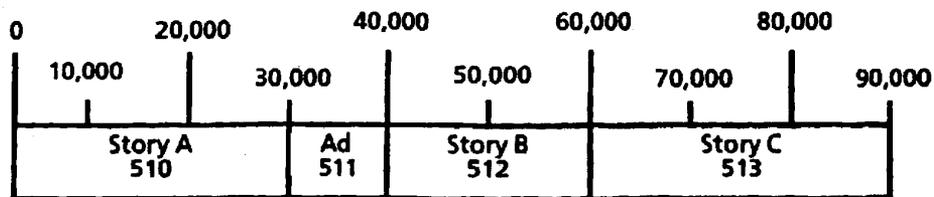


Figure 5

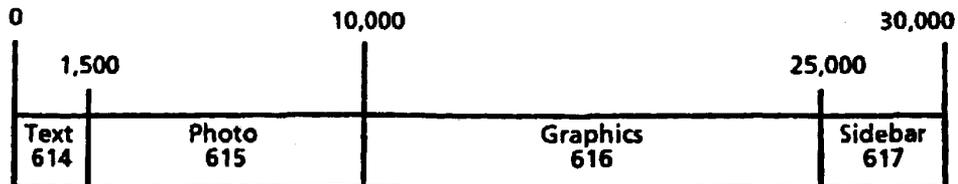


Figure 6

Figure 7

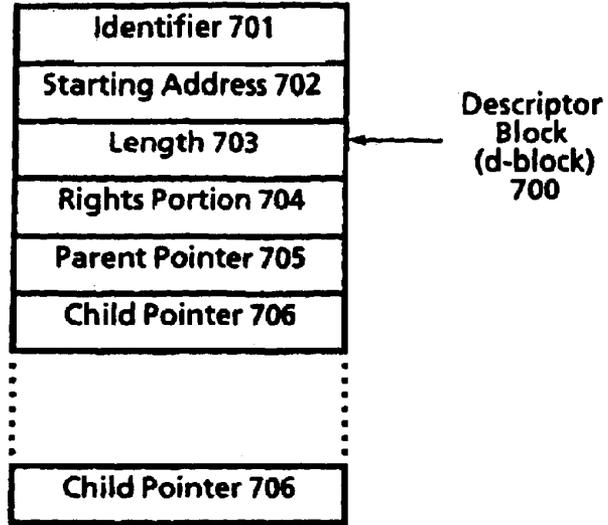


Figure 8

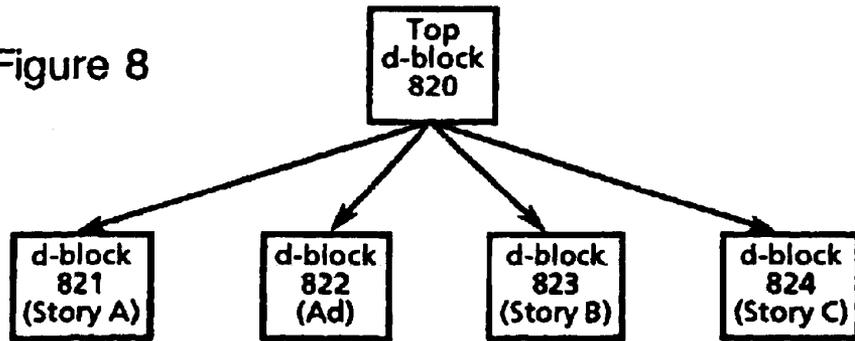


Figure 9

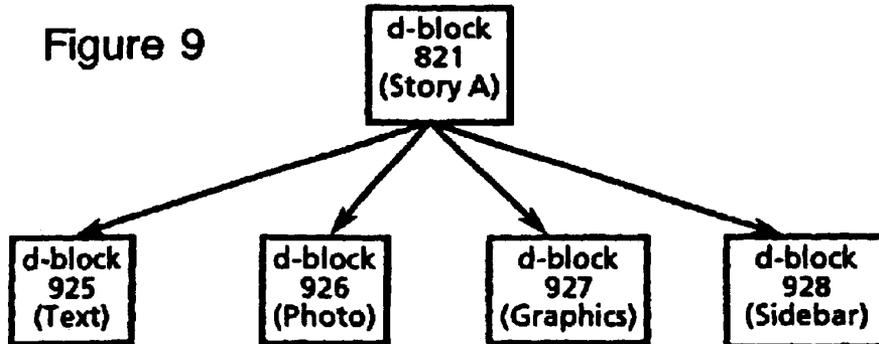


Figure 10

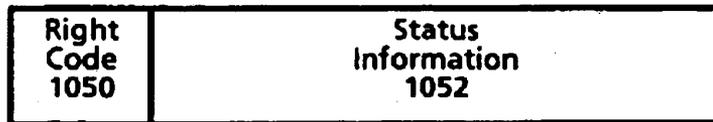


Figure 14

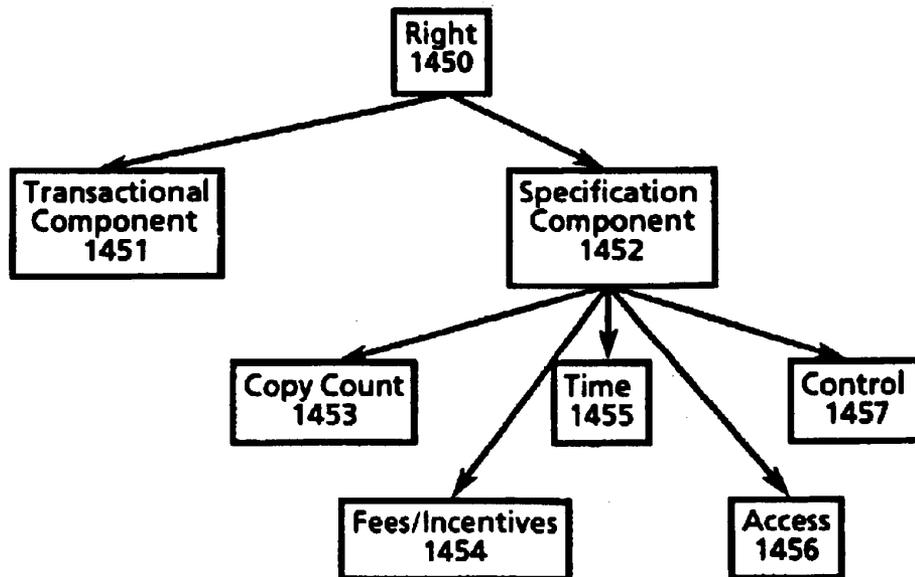


Figure 11

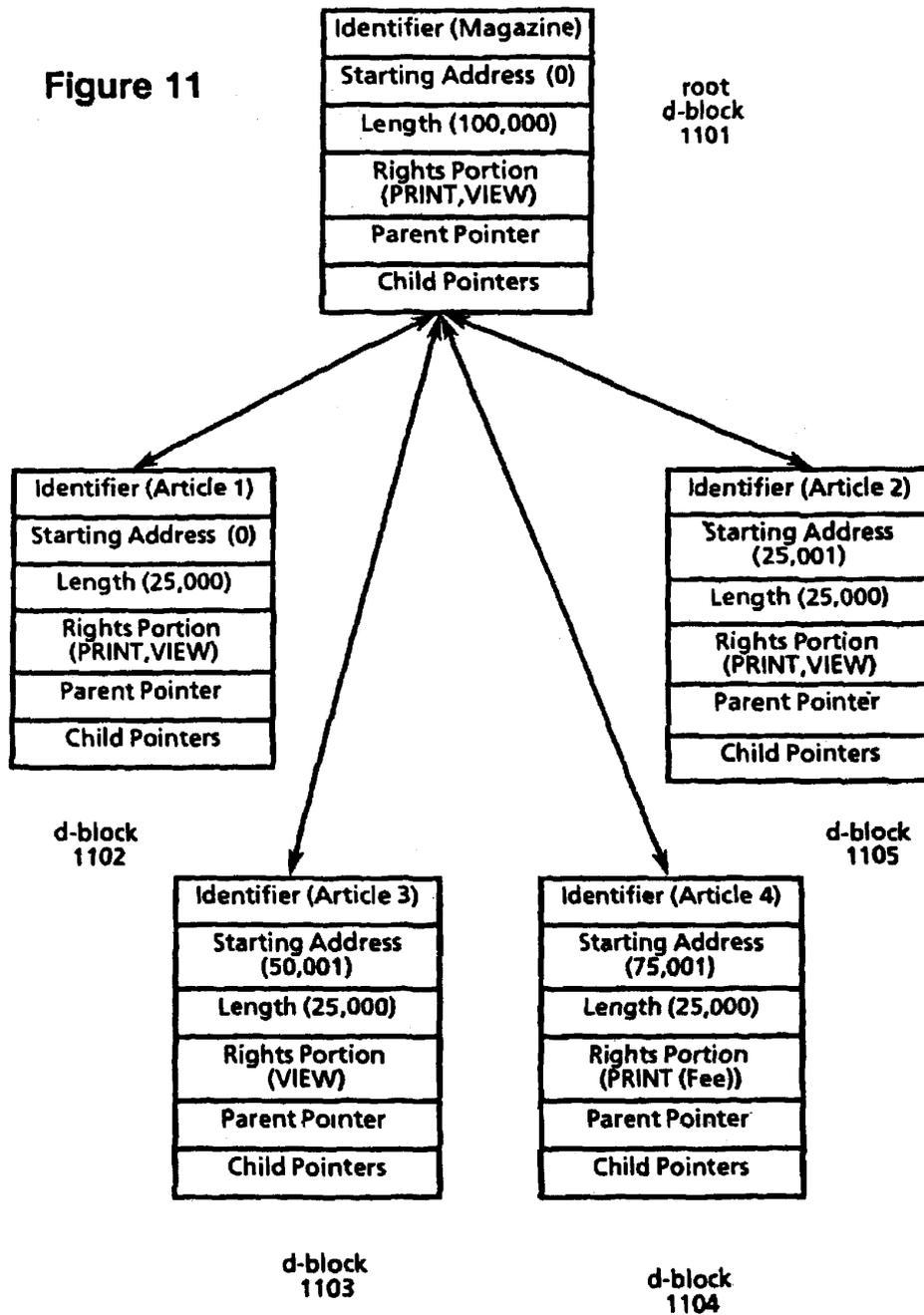


Figure 12

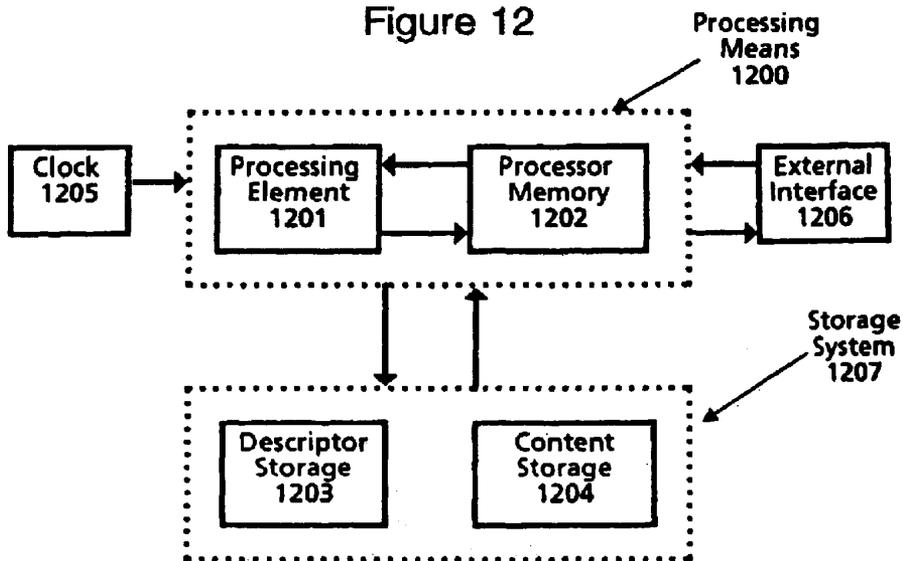
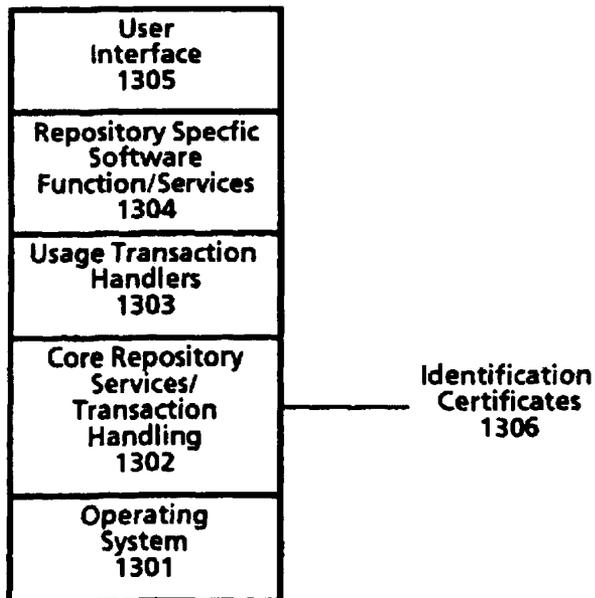


Figure 13



U.S. Patent

Nov. 8, 2005

Sheet 9 of 13

US 6,963,859 B2

1501 ~ Digital Work Rights := (Rights*)
1502 ~ Right := (Right-Code {Copy-Count} {Control-Spec} {Time-Spec} {Access-Spec} {Fee-Spec})
1503 ~ Right-Code := Render-Code | Transport-Code | File-Management-Code | Derivative-Works-Code | Configuration-Code
1504 ~ Render-Code := [Play : {Player: Player-ID} | Print: {Printer: Printer-ID}]
1505 ~ Transport-Code := [Copy | Transfer | Loan {Remaining-Rights: Next-Set-of-Rights}] { (Next-Copy-Rights: Next-Set-of-Rights) }
1506 ~ File-Management-Code := Backup {Back-Up-Copy-Rights: Next-Set-of-Rights} | Restore | Delete | Folder | Directory {Name: Hide-Local | Hide-Remote} {Parts: Hide-Local | Hide-Remote}
1507 ~ Derivative-Works-Code := [Extract | Embed | Edit {Process: Process-ID}] {Next-Copy-Rights: Next-Set-of-Rights}
1508 ~ Configuration-Code := Install | Uninstall
1509 ~ Next-Set-of-Rights := { (Add: Set-Of-Rights) } { (Delete: Set-Of-Rights) } { (Replace: Set-Of-Rights) } { (Keep: Set-Of-Rights) }
1510 ~ Copy-Count := (Copies: positive-integer | 0 | Unlimited)
1511 ~ Control-Spec := (Control: {Restrictable | Unrestrictable} {Unchargeable | Chargeable})
1512 ~ Time-Spec := { (Fixed-Interval | Sliding-Interval | Meter-Time) } Until: Expiration-Date
1513 ~ Fixed-Interval := From: Start-Time
1514 ~ Sliding-Interval := Interval: Use-Duration
1515 ~ Meter-Time := Time-Remaining: Remaining-Use
1516 ~ Access-Spec := { (SC: Security-Class) } { Authorization: Authorization-ID* } { Other-Authorization: Authorization-ID* } { Ticket: Ticket-ID }
1517 ~ Fee-Spec := { Scheduled-Discount } Regular-Fee-Spec | Scheduled-Fee-Spec | Markup-Spec
1518 ~ Scheduled-Discount := Scheduled-Discount: (Scheduled-Discount: (Time-Spec Percentage)*)
1519 ~ Regular-Fee-Spec := { (Fee: | Incentive:) } [Per-Use-Spec | Metered-Rate-Spec | Best-Price-Spec | Call-For-Price-Spec] { (Min: Money-Unit Per: Time-Spec) } { (Max: Money-Unit Per: Time-Spec) } To: Account-ID
1520 ~ Per-Use-Spec := Per-Use: Money-unit
1521 ~ Metered-Rate-Spec := Metered: Money-Unit Per: Time-Spec
1522 ~ Best-Price-Spec := Best-Price: Money-unit Max: Money-unit
1523 ~ Call-For-Price-Spec := Call-For -Price
1524 ~ Scheduled-Fee-Spec := (Schedule: (Time-Spec Regular-Fee-Spec))
1525 ~ Markup-Spec := Markup: percentage To: Account-ID

Fig. 15

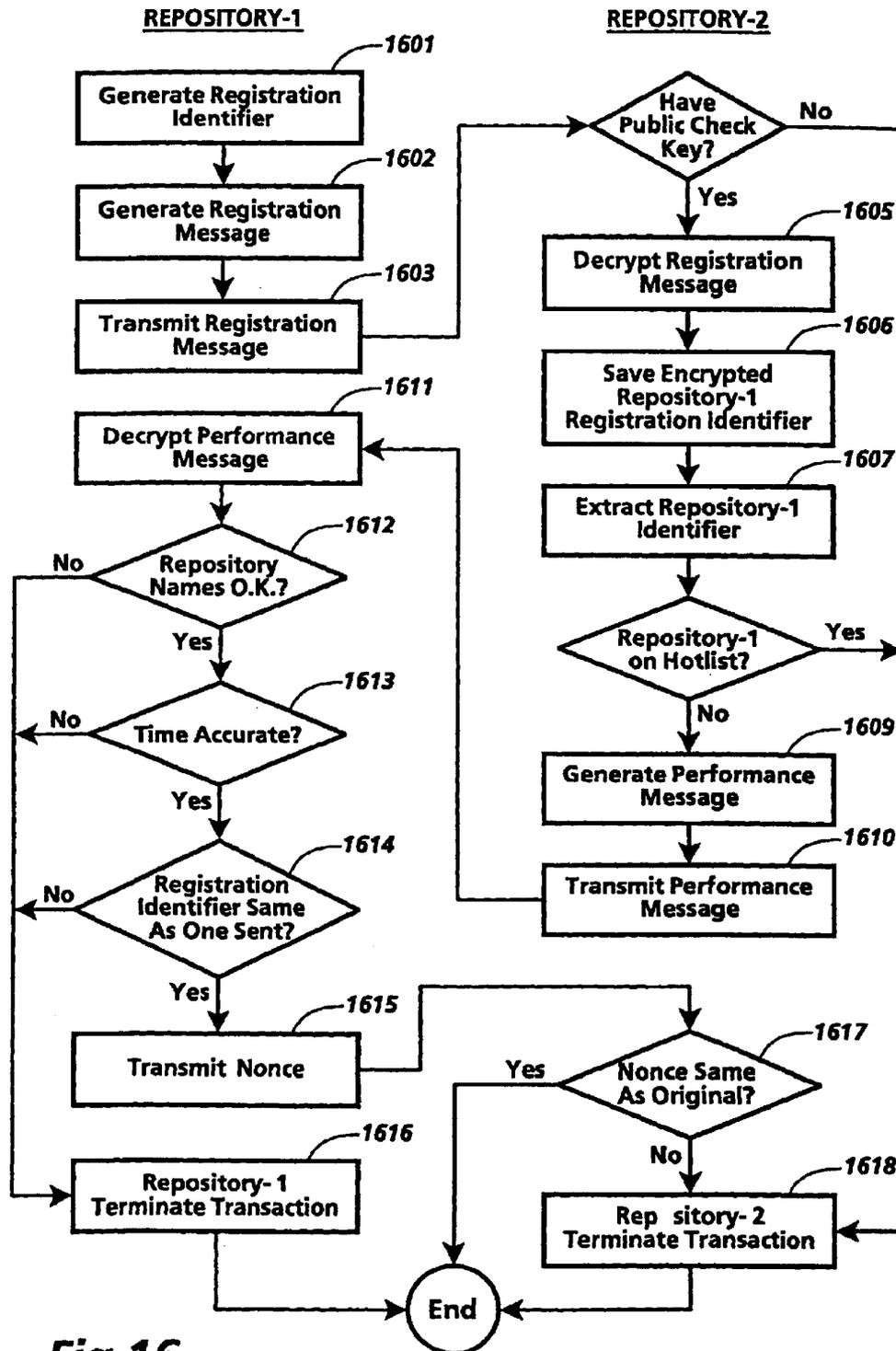


Fig. 16

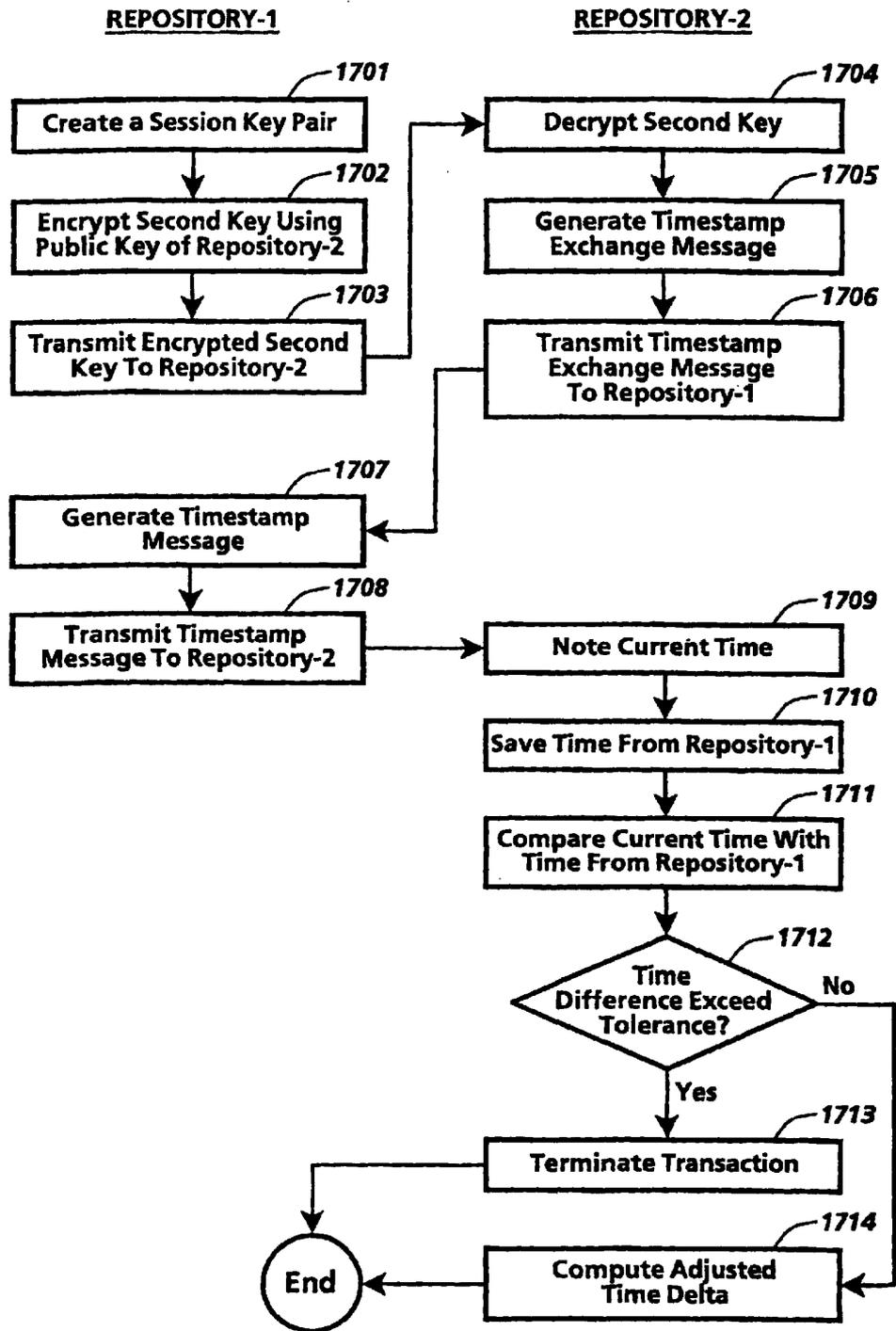


Fig.17

Figure 18

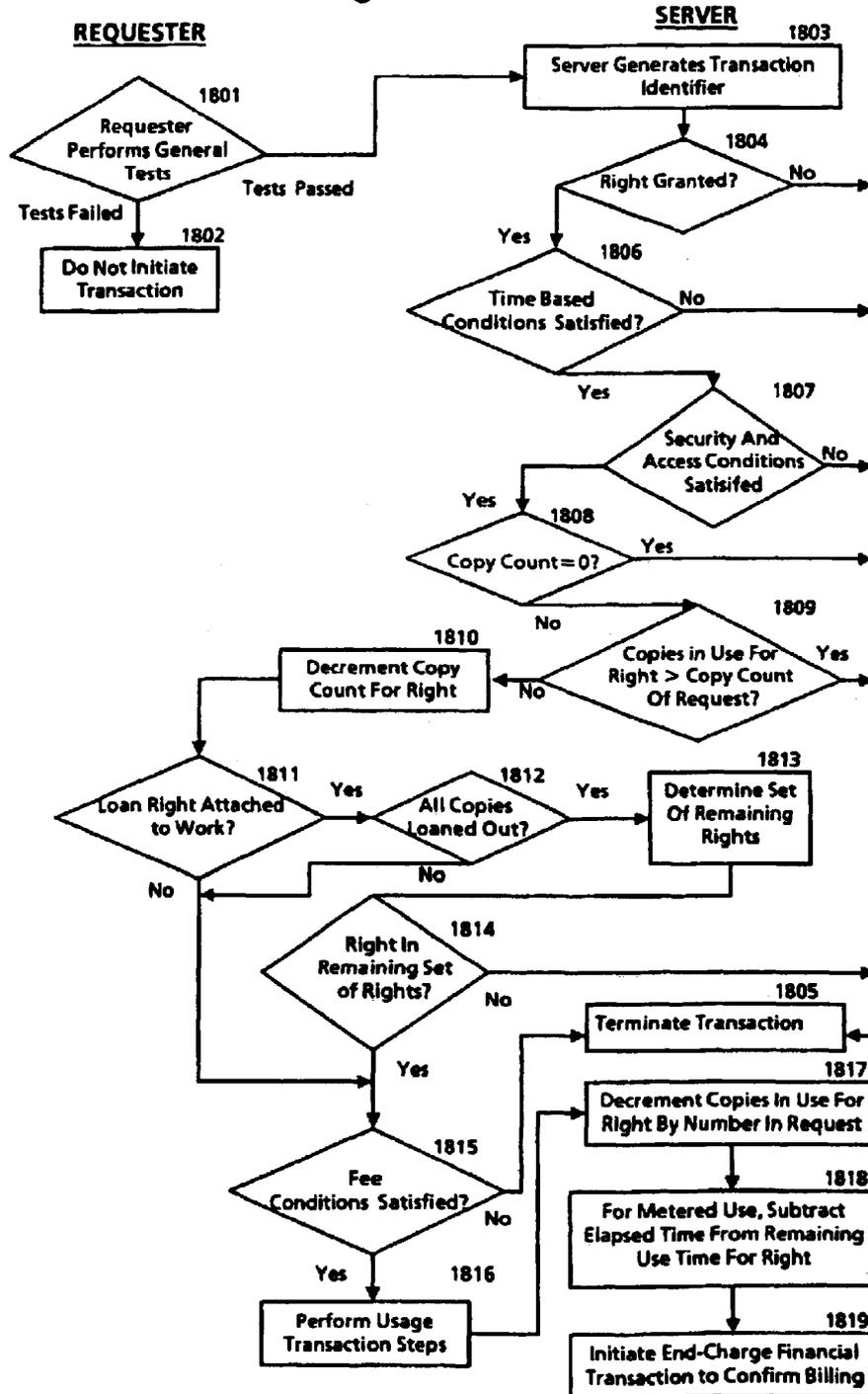
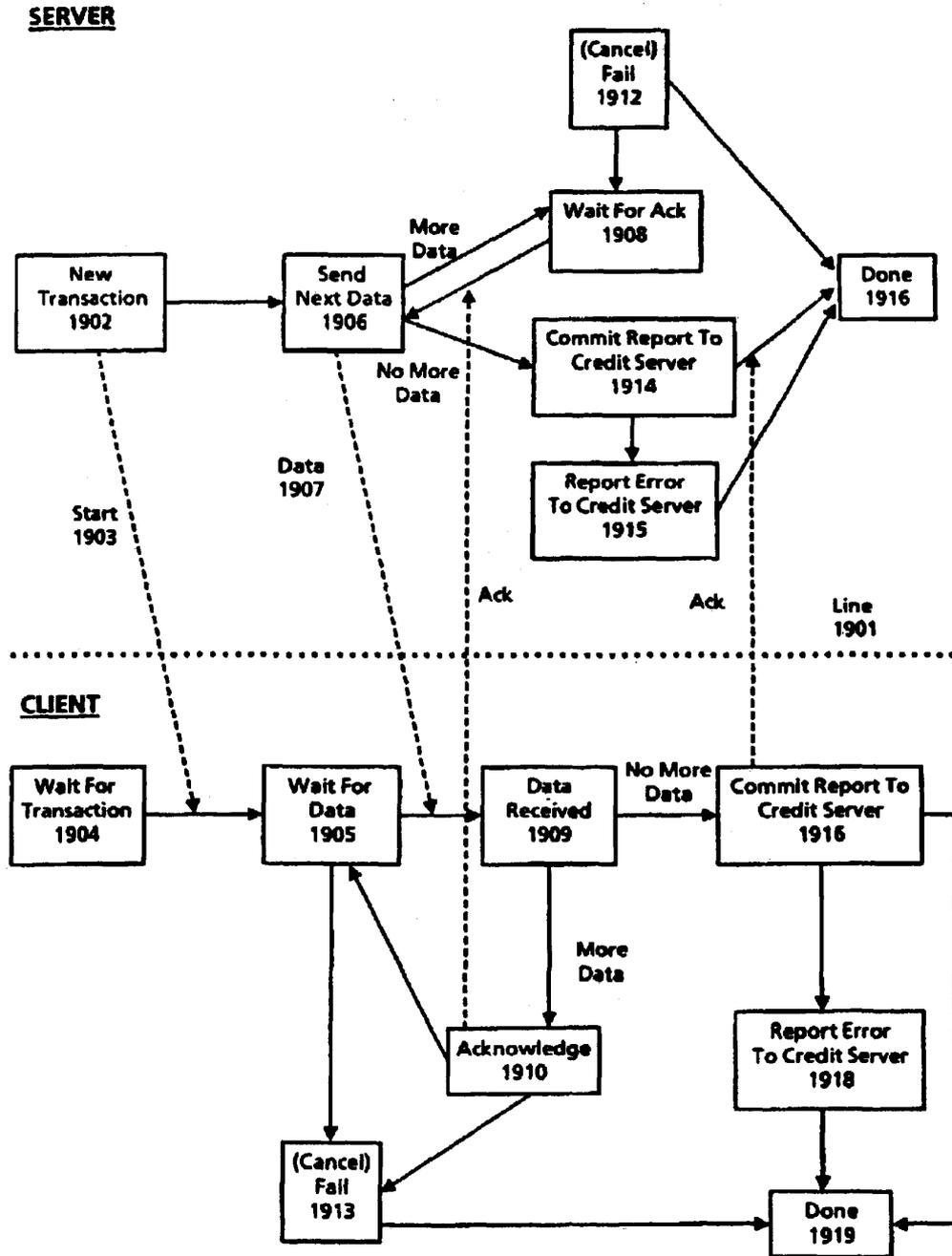


Figure 19



US 6,963,859 B2

1

CONTENT RENDERING REPOSITORY

Continuation of prior application Ser. No.: 09/778,006 filed Feb. 7, 2001, now U.S. Pat. No. 6,714,921; which is a Division of U.S. Ser. No.: 08/967,084 filed Nov. 10, 1997, now U.S. Pat. No. 6,236,971 and which is a Continuation of U.S. Ser. No.: 08/344,760 filed Nov. 23, 1994, now abandoned.

FIELD OF THE INVENTION

The present invention relates to the field of distribution and usage rights enforcement for digitally encoded works.

BACKGROUND OF THE INVENTION

A fundamental issue facing the publishing and information industries as they consider electronic publishing is how to prevent the unauthorized and unaccounted distribution or usage of electronically published materials. Electronically published materials are typically distributed in a digital form and recreated on a computer based system having the capability to recreate the materials. Audio and video recordings, software, books and multimedia works are all being electronically published. Companies in these industries receive royalties for each accounted for delivery of the materials, e.g. the sale of an audio CD at a retail outlet. Any unaccounted distribution of a work results in an unpaid royalty (e.g. copying the audio recording CD to another digital medium.)

The ease in which electronically published works can be "perfectly" reproduced and distributed is a major concern. The transmission of digital works over networks is commonplace. One such widely used network is the Internet. The Internet is a widespread network facility by which computer users in many universities, corporations and government entities communicate and trade ideas and information. Computer bulletin boards found on the Internet and commercial networks such as CompuServ and Prodigy allow for the posting and retrieving of digital information. Information services such as Dialog and LEXIS/NEXIS provide databases of current, information on a wide variety of topics. Another factor which will exacerbate the situation is the development and expansion of the National Information Infrastructure (the NII). It is anticipated that, as the NII grows, the transmission of digital works over networks will increase many times over. It would be desirable to utilize the NII for distribution of digital works without the fear of widespread unauthorized copying.

The most straightforward way to curb unaccounted distribution is to prevent unauthorized copying and transmission. For existing materials that are distributed in digital form, various safeguards are used. In the case of software, copy protection schemes which limit the number of copies that can be made or which corrupt the output when copying is detected have been employed. Another scheme causes software to become disabled after a predetermined period of time has lapsed. A technique used for workstation based software is to require that a special hardware device must be present on the workstation in order for the software to run, e.g., see U.S. Pat. No. 4,932,054 entitled "Method and Apparatus for Protecting Computer Software Utilizing Coded Filter Network in Conjunction with an Active Coded Hardware Device." Such devices are provided with the software and are commonly referred to as dongles.

Yet another scheme is to distribute software, but which requires a "key" to enable it's use. This is employed in distribution schemes where "demos" of the software are

2

provided on a medium along with the entire product. The demos can be freely used, but in order to use the actual product, the key must be purchased. These scheme do not hinder copying of the software once the key is initially purchased.

A system for ensuring that licenses are in place for using licensed products is described in PCT Publication WO 93/01550 to Griswold entitled "License Management System and Method." The licensed product may be any electronically published work but is most effective for use with works that are used for extended periods of time such as software programs. Griswold requires that the licensed product contain software to invoke a license check monitor at predetermined time intervals. The license check monitor generates request datagrams which identify the licensee. The request datagrams are sent to a license control system over an appropriate communication facility. The license control system then checks the datagram to determine if the datagram is from a valid licensee. The license control system then sends a reply datagram to the license check monitor indicating denial or approval of usage. The license control system will deny usage in the event that request datagrams go unanswered after a predetermined period of time (which may indicate an unauthorized attempt to use the licensed product). In this system, usage is managed at a central location by the response datagrams. So for example if license fees have not been paid, access to the licensed product is terminated.

It is argued by Griswold that the described system is advantageous because it can be implemented entirely in software. However, the system described by Griswold has limitations. An important limitation is that during the use of the licensed product, the user must always be coupled to an appropriate communication facility in order to send and receive datagrams. This creates a dependency on the communication facility. So if the communication facility is not available, the licensed product cannot be used. Moreover, some party must absorb the cost of communicating with the license server.

A system for controlling the distribution of digitally encoded books is embodied in a system available from VPR Systems, LTD. of St. Louis, Mo. The VPR system is self-contained and is comprised of: (1) point of sale kiosks for storing and downloading of books, (2) personal storage mediums (cartridges) to which the books are downloaded, and (3) readers for viewing the book. In a purchase transaction, a purchaser will purchase a voucher card representing the desired book. The voucher will contain sufficient information to identify the book purchased and perhaps some demographic information relating to the sales transaction. To download the book, the voucher and the cartridge are inserted into the kiosk.

The VPR system may also be used as a library. In such an embodiment, the kiosk manages the number of "copies" that may be checked out at one time. Further, the copy of the book is erased from the users cartridge after a certain check-out time has expired. However, individuals cannot loan books because the cartridges may only be used with the owners reader.

The foregoing distribution and protection schemes operate in part by preventing subsequent distribution of the work. While this certainly prevents unauthorized distributions, it does so by sacrificing the potential for subsequent revenue bearing uses. For example, it may be desirable to allow the lending of a purchased work to permit exposure of the work to potential buyers. Another example would be to permit the

US 6,963,859 B2

3

creation of a derivative work for a fee. Yet another example would be to permit copying the work for a fee (essentially purchasing it). Thus, it would be desirable to provide flexibility in how the owner of a digital work may allow it to be distributed.

While flexibility in distribution is a concern, the owners of a work want to make sure they are paid for such distributions. In U.S. Pat. No. 4,977,594 to Shear, entitled "Database Usage Metering and Protection System and Method," a system for metering and billing for usage of information distributed on a CD-ROM is described. The system requires the addition of a billing module to the computer system. The billing module may operate in a number of different ways. First, it may periodically communicate billing data to a central billing facility, whereupon the user may be billed. Second, billing may occur by disconnecting the billing module and the user sending it to a central billing facility where the data is read and a user bill generated.

U.S. Pat. No. 5,247,575, Sprague et al., entitled "Information Distribution System", describes an information distribution system which provides and charges only for user selected information. A plurality of encrypted information packages (IPs) are provided at the user site, via high and/or low density storage media and/or by broadcast transmission. Some of the IPs may be of no interest to the user. The IPs of interest are selected by the user and are decrypted and stored locally. The IPs may be printed, displayed or even copied to other storage medias. The charges for the selected IP's are accumulated within a user apparatus and periodically reported by telephone to a central accounting facility. The central accounting facility also issues keys to decrypt the IPs. The keys are changed periodically. If the central accounting facility has not issued a new key for a particular user station, the station is unable to retrieve information from the system when the key is changed.

A system available from Wave Systems Corp. of Princeton, N.Y., provides for metering of software usage on a personal computer. The system is installed onto a computer and collects information on what software is in use, encrypts it and then transmits the information to a transaction center. From the transaction center, a bill is generated and sent to the user. The transaction center also maintains customer accounts so that licensing fees may be forwarded directly to the software providers. Software operating under this system must be modified so that usage can be accounted.

Known techniques for billing do not provide for billing of copies made of the work. For example, if data is copied from the CD-ROM described in Shear, any subsequent use of the copy of the information cannot be metered or billed. In other words, the means for billing runs with the media rather than the underlying work. It would be desirable to have a distribution system where the means for billing is always transported with the work.

SUMMARY OF THE INVENTION

An aspect of the invention is a rendering system adapted for use in a system for managing use of content and operative to rendering content in accordance with usage rights associated with the content. The system includes a rendering device configured to render the content and a repository coupled to the rendering device and operative to enforce usage rights associated with the content and permit the rendering device to render the content in accordance with a manner of use specified by the usage rights.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a flowchart illustrating a simple instantiation of the operation of the currently preferred embodiment of the present invention.

4

FIG. 2 is a block diagram illustrating the various repository types and the repository transaction flow between them in the currently preferred embodiment of the present invention.

FIG. 3 is a block diagram of a repository coupled with a credit server in the currently preferred embodiment of the present invention.

FIGS. 4a and 4b are examples of rendering systems as may be utilized in the currently preferred embodiment of the present invention.

FIG. 5 illustrates a contents file layout for a digital work as may be utilized in the currently preferred embodiment of the present invention.

FIG. 6 illustrates a contents file layout for an individual digital work of the digital work of FIG. 5 as may be utilized in the currently preferred embodiment of the present invention.

FIG. 7 illustrates the components of a description block of the currently preferred embodiment of the present invention.

FIG. 8 illustrates a description tree for the contents file layout of the digital work illustrated in FIG. 5.

FIG. 9 illustrates a portion of a description tree corresponding to the individual digital work illustrated in FIG. 6.

FIG. 10 illustrates a layout for the rights portion of a description block as may be utilized in the currently preferred embodiment of the present invention.

FIG. 11 is a description tree wherein certain d-blocks have PRINT usage rights and is used to illustrate "strict" and "lenient" rules for resolving usage rights conflicts.

FIG. 12 is a block diagram of the hardware components of a repository as are utilized in the currently preferred embodiment of the present invention.

FIG. 13 is a block diagram of the functional (logical) components of a repository as are utilized in the currently preferred embodiment of the present invention.

FIG. 14 is diagram illustrating the basic components of a usage right in the currently preferred embodiment of the present invention.

FIG. 15 lists the usage rights grammar of the currently preferred embodiment of the present invention.

FIG. 16 is a flowchart illustrating the steps of certificate delivery, hotlist checking and performance testing as performed in a registration transaction as may be performed in the currently preferred embodiment of the present invention.

FIG. 17 is a flowchart illustrating the steps of session information exchange and clock synchronization as may be performed in the currently preferred embodiment of the present invention, after each repository in the registration transaction has successfully completed the steps described in FIG. 16.

FIG. 18 is a flowchart illustrating the basic flow for a usage transaction, including the common opening and closing step, as may be performed in the currently preferred embodiment of the present invention.

FIG. 19 is a state diagram of server and client repositories in accordance with a transport protocol followed when moving a digital work from the server to the client repositories, as may be performed in the currently preferred embodiment of the present invention.

US 6,963,859 B2

5

DETAILED DESCRIPTION OF THE
PREFERRED EMBODIMENT

TABLE OF CONTENTS

OVERVIEW
RENDERING SYSTEMS
ATTACHING USAGE RIGHTS TO A DIGITAL WORK
Resolving Conflicting Rights
REPOSITORIES
Repository Security Classes
Repository User Interface
CREDIT SERVICES
USAGE RIGHTS LANGUAGE
Copy Count Specification
Control Specification
Time Specification
Security Class and Authorization Specification
Usage Fees and Incentives Specification
Examples of Sets of Usage Rights
REPOSITORY TRANSACTIONS
Message Transmission
Session Initiation Transactions
Billing Transactions
Transmission Protocol
The Copy Transaction
The Transfer Transaction
The Loan Transaction
The Play Transaction
The Print Transaction
The Backup Transaction
The Restore Transaction
The Delete Transaction
The Directory Transaction
The Folder Transaction
The Extract Transaction
The Embed Transaction
The Edit Transaction
The Authorization Transaction
The Install Transaction
The Uninstall Transaction
DISTRIBUTION AND USE SCENARIOS
APPENDIX A GLOSSARY
Overview

A system for controlling use and distribution of digital works is disclosed. The present invention is directed to supporting commercial transactions involving digital works. The transition to digital works profoundly and fundamentally changes how creativity and commerce can work. It changes the cost of transporting or storing works because digital property is almost "massless." Digital property can be transported at electronic speeds and requires almost no warehousing. Keeping an unlimited supply of virtual copies on hand requires essentially no more space than keeping one copy on hand. The digital medium also lowers the costs of alteration, reuse and billing.

There is a market for digital works because creators are strongly motivated to reuse portions of digital works from others rather than creating their own completely. This is because it is usually so much easier to use an existing stock photo or music clip than to create a new one from scratch.

Herein the terms "digital work", "work" and "content" refer to any work that has been reduced to a digital representation. This would include any audio, video, text, or multimedia work and any accompanying interpreter (e.g.

6

software) that may be required for recreating the work. The term composite work refers to a digital work comprised of a collection of other digital works. The term "usage rights" or "rights" is a term which refers to rights granted to a recipient of a digital work. Generally, these rights define how a digital work can be used and if it can be further distributed. Each usage right may have one or more specified conditions which must be satisfied before the right may be exercised. Appendix 1 provides a Glossary of the terms used herein.

A key feature of the present invention is that usage rights are permanently "attached" to the digital work. Copies made of a digital work will also have usage rights attached. Thus, the usage rights and any associated fees assigned by a creator and subsequent distributor will always remain with a digital work.

The enforcement elements of the present invention are embodied in repositories. Among other things, repositories are used to store digital works, control access to digital works, bill for access to digital works and maintain the security and integrity of the system.

The combination of attached usage rights and repositories enable distinct advantages over prior systems. As noted in the prior art, payment of fees are primarily for the initial access. In such approaches, once a work has been read, computational control over that copy is gone. Metaphorically, "the content genie is out of the bottle and no more fees can be billed." In contrast, the present invention never separates the fee descriptions from the work. Thus, the digital work genie only moves from one trusted bottle (repository) to another, and all uses of copies are potentially controlled and billable.

FIG. 1 is a high level flowchart omitting various details but which demonstrates the basic operation of the present invention. Referring to FIG. 1, a creator creates a digital work, step 101. The creator will then determine appropriate usage rights and fees, attach them to the digital work, and store them in Repository 1, step 102. The determination of appropriate usage rights and fees will depend on various economic factors. The digital work remains securely in Repository 1 until a request for access is received. The request for access begins with a session initiation by another repository. Here a Repository 2 initiates a session with Repository 1, step 103. As will be described in greater detail below, this session initiation includes steps which helps to insure that the respective repositories are trustworthy. Assuming that a session can be established, Repository 2 may then request access to the Digital Work for a stated purpose, step 104. The purpose may be, for example, to print the digital work or to obtain a copy of the digital work. The purpose will correspond to a specific usage right. In any event, Repository 1 checks the usage rights associated with the digital work to determine if the access to the digital work may be granted, step 105. The check of the usage rights essentially involves a determination of whether a right associated with the access request has been attached to the digital work and if all conditions associated with the right are satisfied. If the access is denied, repository 1 terminates the session with an error message, step 106. If access is granted, repository 1 transmits the digital work to repository 2, step 107. Once the digital work has been transmitted to repository 2, repository 1 and 2 each generate billing information for the access which is transmitted to a credit server, step 108. Such double billing reporting is done to insure against attempts to circumvent the billing process.

FIG. 2 illustrates the basic interactions between repository types in the present invention. As will become apparent from

US 6,963,859 B2

7

FIG. 2, the various repository types will serve different functions. It is fundamental that repositories will share a core set of functionality which will enable secure and trusted communications. Referring to FIG. 2, a repository **201** represents the general instance of a repository. The repository **201** has two modes of operation; a server mode and a requester mode. When in the server mode, the repository will be receiving and processing access requests to digital works. When in the requester mode, the repository will be initiating requests to access digital works. Repository **201** is general in the sense that its primary purpose is as an exchange medium for digital works. During the course of operation, the repository **201** may communicate with a plurality of other repositories, namely authorization repository **202**, rendering repository **203** and master repository **204**. Communication between repositories occurs utilizing a repository transaction protocol **205**.

Communication with an authorization repository **202** may occur when a digital work being accessed has a condition requiring an authorization. Conceptually, an authorization is a digital certificate such that possession of the certificate is required to gain access to the digital work. An authorization is itself a digital work that can be moved between repositories and subjected to fees and usage rights conditions. An authorization may be required by both repositories involved in an access to a digital work.

Communication with a rendering repository **203** occurs in connection with the rendering of a digital work. As will be described in greater detail below, a rendering repository is coupled with a rendering device (e.g. a printer device) to comprise a rendering system.

Communication with a master repository **205** occurs in connection with obtaining an identification certificate. Identification certificates are the means by which a repository is identified as "trustworthy". The use of identification certificates is described below with respect to the registration transaction.

FIG. 3 illustrates the repository **201** coupled to a credit server **301**. The credit server **301** is a device which accumulates billing information for the repository **201**. The credit server **301** communicates with repository **201** via billing transactions **302** to record billing transactions. Billing transactions are reported to a billing clearinghouse **303** by the credit server **301** on a periodic basis. The credit server **301** communicates to the billing clearinghouse **303** via clearinghouse transactions **304**. The clearinghouse transactions **304** enable a secure and encrypted transmission of information to the billing clearinghouse **303**.

Rendering Systems

A rendering system is generally defined as a system comprising a repository and a rendering device which can render a digital work into its desired form. Examples of a rendering system may be a computer system, a digital audio system, or a printer. A rendering system has the same security features as a repository. The coupling of a rendering repository with the rendering device may occur in a manner suitable for the type of rendering device.

FIG. 4a illustrates a printer as an example of a rendering system. Referring to FIG. 4, printer system **401** has contained therein a printer repository **402** and a print device **403**. It should be noted that the dashed line defining printer system **401** defines a secure system boundary. Communications within the boundary is assumed to be secure. Depending on the security level, the boundary also represents a barrier intended to provide physical integrity. The printer repository **402** is an instantiation of the rendering repository **205** of FIG. 2. The printer repository **402** will in

8

some instances contain an ephemeral copy of a digital work which remains until it is printed out by the print engine **403**. In other instances, the printer repository **402** may contain digital works such as fonts, which will remain and can be billed based on use. This design assures that all communication lines between printers and printing devices are encrypted, unless they are within a physically secure boundary. This design feature eliminates a potential "fault" point through which the digital work could be improperly obtained. The printer device **403** represents the printer components used to create the printed output.

Also illustrated in FIG. 4a is the repository **404**. The repository **404** is coupled to the printer repository **402**. The repository **404** represents an external repository which contains digital works.

FIG. 4b is an example of a computer system as a rendering system. A computer system may constitute a "multi-function" device since it may execute digital works (e.g. software programs) and display digital works (e.g. a digitized photograph). Logically, each rendering device can be viewed as having its own repository, although only one physical repository is needed. Referring to FIG. 4b, a computer system **410** has contained therein a display/execution repository **411**. The display/execution repository **411** is coupled to display device, **412** and execution device **413**. The dashed box surrounding the computer system **410** represents a security boundary within which communications are assumed to be secure. The display/execution repository **411** is further coupled to a credit server **414** to report any fees to be billed for access to a digital work and a repository **415** for accessing digital works stored therein.

Structure of Digital Works

Usage rights are attached directly to digital works. Thus, it is important to understand the structure of a digital work. The structure of a digital work, in particular composite digital works, may be naturally organized into an acyclic structure such as a hierarchy. For example, a magazine has various articles and photographs which may have been created and are owned by different persons. Each of the articles and photographs may represent a node in a hierarchical structure. Consequently, controls, i.e. usage rights, may be placed on each node by the creator. By enabling control and fee billing to be associated with each node, a creator of a work can be assured that the rights and fees are not circumvented.

In the currently preferred embodiment, the file information for a digital work is divided into two files: a "contents" file and a "description tree" file. From the perspective of a repository, the "contents" file is a stream of addressable bytes whose format depends completely on the interpreter used to play, display or print the digital work. The description tree file makes it possible to examine the rights and fees for a work without reference to the content of the digital work. It should be noted that the term description tree as used herein refers to any type of acyclic structure used to represent the relationship between the various components of a digital work.

FIG. 5 illustrates the layout of a contents file. Referring to FIG. 5, a digital work **509** is comprised of story A **510**, advertisement **511**, story B **512** and story C **513**. It is assumed that the digital work is stored starting at a relative address of 0. Each of the parts of the digital work are stored linearly so that story A **510** is stored at approximately addresses 0-30,000, advertisement **511** at addresses 30,001-40,000, story B **512** at addresses 40,001-60,000 and story C **513** at addresses 60,001-85K. The detail of story A **510** is illustrated in FIG. 6. Referring to FIG. 6, the story A

US 6,963,859 B2

9

510 is further broken down to show text 614 stored at address 0–1500, soldier photo 615 at addresses 1501–10,000, graphics 616 stored at addresses 10,001–25,000 and sidebar 617 stored address 25,001–30,000. Note that the data in the contents file may be compressed (for saving storage) or encrypted (for security).

From FIGS. 5 and 6 it is readily observed that a digital work can be represented by its component parts as a hierarchy. The description tree for a digital work is comprised of a set of related descriptor blocks (d-blocks). The contents of each d-block is described with respect to FIG. 7. Referring to FIG. 7, a d-block 700 includes an identifier 701 which is a unique identifier for the work in the repository, a starting address 702 providing the start address of the first byte of the work, a length 703 giving the number of bytes in the work, a rights portion 704 wherein the granted usage rights and their status data are maintained, a parent pointer 705 for pointing to a parent d-block and child pointers 706 for pointing to the child d-blocks. In the currently preferred embodiment, the identifier 701 has two parts. The first part is a unique number assigned to the repository upon manufacture. The second part is a unique number assigned to the work upon creation. The rights portion 704 will contain a data structure, such as a look-up table, wherein the various information associated with a right is maintained. The information required by the respective usage rights is described in more detail below. D-blocks form a strict hierarchy. The top d-block of a work has no parent; all other d-blocks have one parent. The relationship of usage rights between parent and child d-blocks and how conflicts are resolved is described below.

A special type of d-block is a “shell” d-block. A shell d-block adds no new content beyond the content of its parts. A shell d-block is used to add rights and fee information, typically by distributors of digital works.

FIG. 8 illustrates a description tree for the digital work of FIG. 5. Referring to FIG. 8, a top d-block 820 for the digital work points to the various stories and advertisements contained therein. Here, the top d-block 820 points to d-block 821 (representing story A. 510), d-block 822 (representing the advertisement 511), d-block 823 (representing story B 512) and d-block 824 (representing story C 513).

The portion of the description tree for Story A 510 is illustrated in FIG. 9. D-block 925 represents text 614, d-block 926 represents photo 615, d-block 927 represents graphics 616 by and d-block 928 represents sidebar 617.

The rights portion 704 of a descriptor block is further illustrated in FIG. 10. FIG. 10 illustrates a structure which is repeated in the rights portion 704 for each right. Referring to FIG. 10, each right will have a right code field 1001 and status information field 1002. The right code field 1001 will contain a unique code assigned to a right. The status information field 1002 will contain information relating to the state of a right and the digital work. Such information is indicated below in Table 1. The rights as stored in the rights portion 304 may typically be in numerical order based on the right code.

The approach for representing digital works by separating description data from content assumes that parts of a file are contiguous but takes no position on the actual representation of content. In particular, it is neutral to the question of whether content representation may take an object oriented approach. It would be natural to represent content as objects. In principle, it may be convenient to have content objects that include the billing structure and rights information that is represented in the d-blocks. Such variations in the design of the representation are possible and are

10

TABLE 1

DIGITAL WORK STATE INFORMATION		
Property	Value	Use
Copies -in-Use	Number	A counter of the number of copies of a work that are in use. Incremented when another copy is used; decremented when use is completed.
Loan-Period	Time-Units	Indicator of the maximum number of time-units that a document can be loaned out
Loaner-Copy	Boolean	Indicator that the current work is a loaned out copy of an authorized digital work.
Remaining-Time	Time-Units	Indicator of the remaining time of use on a metered document right.
Document-Descr	String	A string containing various identifying information about a document. The exact format of this is not specified, but it can include information such as a publisher name, author name, ISBN number, and so on.
Revenue-Owner	RO-Descr	A handle identifying a revenue owner for a digital work. This is used for reporting usage fees.
Publication-Date	Date-Descr	The date that the digital work was published.
History-list	History-Rec	A list of events recording the repositories and dates for operations that copy, transfer, backup, or restore a digital work.

viable alternatives but may introduce processing overhead, e.g. the interpretation of the objects.

Digital works are stored in a repository as part of a hierarchical file system. Folders (also termed directories and sub-directories) contain the digital works as well as other folders. Digital works and folders in a folder are ordered in alphabetical order. The digital works are typed to reflect how the files are used. Usage rights can be attached to folders so that the folder itself is treated as a digital work. Access to the folder would then be handled in the same fashion as any other digital work. As will be described in more detail below, the contents of the folder are subject to their own rights. Moreover, file management rights may be attached to the folder which define how folder contents can be managed.

Attaching Usage Rights to a Digital Work

It is fundamental to the present invention that the usage rights are treated as part of the digital work. As the digital work is distributed, the scope of the granted usage rights will remain the same or may be narrowed. For example, when a digital work is transferred from a document server to a repository, the usage rights may include the right to loan a copy for a predetermined period of time (called the original rights). When the repository loans out a copy of the digital work, the usage rights in the loaner copy (called the next set of rights) could be set to prohibit any further rights to loan out the copy. The basic idea is that one cannot grant more rights than they have.

The attachment of usage rights into a digital work may occur in a variety of ways. If the usage rights will be the same for an entire digital work, they could be attached when the digital work is processed for deposit in the digital work server. In the case of a digital work having different usage rights for the various components, this can be done as the digital work is being created. An authoring tool or digital work assembling tool could be utilized which provides for an automated process of attaching the usage rights.

As will be described below, when a digital work is copied, transferred or loaned, a “next set of rights” can be specified.

US 6,963,859 B2

11

The “next set of rights” will be attached to the digital work as it is transported.

Resolving Conflicting Rights

Because each part of a digital work may have its own usage rights, there will be instances where the rights of a “contained part” are different from its parent or container part. As a result, conflict rules must be established to dictate when and how a right may be exercised. The hierarchical structure of a digital work facilitates the enforcement of such rules. A “strict” rule would be as follows: a right for a part in a digital work is sanctioned if and only if it is sanctioned for the part, for ancestor d-blocks containing the part and for all descendent d-blocks. By sanctioned, it is meant that (1) each of the respective parts must have the right, and (2) any conditions for exercising the right are satisfied.

It also possible to implement the present invention using a more lenient rule. In the more lenient rule, access to the part may be enabled to the descendent parts which have the right, but access is denied to the descendents which do not.

Example of applying both the strict rule and lenient is illustrated with reference to FIG. 11. Referring to FIG. 11, a root d-block 1101 has child d-blocks 1102–1105. In this case, root d-block represents a magazine, and each of the child d-blocks 1102–1105 represent articles in the magazine. Suppose that a request is made to PRINT the digital work represented by root d-block 1101 wherein the strict rule is followed. The rights for the root d-block 1101 and child d-blocks 1102 and 1105 have been granted PRINT rights. Child d-block 1103 has not been granted PRINT rights and child d-block 1104 has PRINT rights conditioned on payment of a usage fee.

Under the strict rule the PRINT right cannot be exercised because the child d-block does not have the PRINT right. Under the lenient rule, the result would be different. The digital works represented by child d-blocks 1102 and 1105 could be printed and the digital work represented by d-block 1104 could be printed so long as the usage fee is paid. Only the digital work represented by d-block 1103 could not be printed. This same result would be accomplished under the strict rule if the requests were directed to each of the individual digital works.

The present invention supports various combinations of allowing and disallowing access. Moreover, as will be described below, the usage rights grammar permits the owner of a digital work to specify if constraints may be imposed on the work by a container part. The manner in which digital works may be sanctioned because of usage rights conflicts would be implementation specific and would depend on the nature of the digital works.

Repositories

Many of the powerful functions of repositories—such as their ability to “loan” digital works or automatically handle the commercial reuse of digital works—are possible because they are trusted systems. The systems are trusted because they are able to take responsibility for fairly and reliably carrying out the commercial transactions. That the systems can be responsible (“able to respond”) is fundamentally an issue of integrity. The integrity of repositories has three parts: physical integrity, communications integrity, and behavioral integrity.

Physical integrity refers to the integrity of the physical devices themselves. Physical integrity applies both to the repositories and to the protected digital works. Thus, the higher security classes of repositories themselves may have sensors that detect when tampering is attempted on their secure cases. In addition to protection of the repository

12

itself, the repository design protects access to the content of digital works. In contrast with the design of conventional magnetic and optical devices—such as floppy disks, CD-ROMs, and videotapes—repositories never allow non-trusted systems to access the works directly. A maker of generic computer systems cannot guarantee that their platform will not be used to make unauthorized copies. The manufacturer provides generic capabilities for reading and writing information, and the general nature of the functionality of the general computing device depends on it. Thus, a copy program can copy arbitrary data. This copying issue is not limited to general purpose computers. It also arises for the unauthorized duplication of entertainment “software” such as video and audio recordings by magnetic recorders. Again, the functionality of the recorders depends on their ability to copy and they have no means to check whether a copy is authorized. In contrast, repositories prevent access to the raw data by general devices and can test explicit rights and conditions before copying or otherwise granting access. Information is only accessed by protocol between trusted repositories.

Communications integrity refers to the integrity of the communications channels between repositories. Roughly speaking, communications integrity means that repositories cannot be easily fooled by “telling them lies.” Integrity in this case refers to the property that repositories will only communicate with other devices that are able to present proof that they are certified repositories, and furthermore, that the repositories monitor the communications to detect “impostors” and malicious or accidental interference. Thus the security measures involving encryption, exchange of digital certificates, and nonces described below are all security measures aimed at reliable communication in a world known to contain active adversaries.

Behavioral integrity refers to the integrity in what repositories do. What repositories do is determined by the software that they execute. The integrity of the software is generally assured only by knowledge of its source. Restated, a user will trust software purchased at a reputable computer store but not trust software obtained off a random (insecure) server on a network. Behavioral integrity is maintained by requiring that repository software be certified and be distributed with proof of such certification, i.e. a digital certificate. The purpose of the certificate is to authenticate that the software has been tested by an authorized organization, which attests that the software does what it is supposed to do and that it does not compromise the behavioral integrity of a repository. If the digital certificate cannot be found in the digital work or the master repository which generated the certificate is not known to the repository receiving the software, then the software cannot be installed.

In the description of FIG. 2, it was indicated that repositories come in various forms. All repositories provide a core set of services for the transmission of digital works. The manner in which digital works are exchanged is the basis for all transaction between repositories. The various repository types differ in the ultimate functions that they perform. Repositories may be devices themselves, or they may be incorporated into other systems. An example is the rendering repository 205 of FIG. 2.

A repository will have associated with it a repository identifier. Typically, the repository identifier would be a unique number assigned to the repository at the time of manufacture. Each repository will also be classified as being in a particular security class. Certain communications and transactions may be conditioned on a repository being in a particular security class. The various security classes are described in greater detail below.

US 6,963,859 B2

13

As a prerequisite to operation, a repository will require possession of an identification certificate. Identification certificates are encrypted to prevent forgery and are issued by a Master repository. A master repository plays the role of an authorization agent to enable repositories to receive digital works. Identification certificates must be updated on a periodic basis. Identification certificates are described in greater detail below with respect to the registration transaction.

A repository has both a hardware and functional embodiment. The functional embodiment is typically software executing on the hardware embodiment. Alternatively, the functional embodiment may be embedded in the hardware embodiment such as an Application Specific Integrated Circuit (ASIC) chip.

The hardware embodiment of a repository will be enclosed in a secure housing which if compromised, may cause the repository to be disabled. The basic components of the hardware embodiment of a repository are described with reference to FIG. 12. Referring to FIG. 12, a repository is comprised of a processing means 1200, storage system 1207, clock 1205 and external interface 1206. The processing means 1200 is comprised of a processor element 1201 and processor memory 1202. The processing means 1201, provides controller, repository transaction, and usage rights transaction functions for the repository. Various functions in the operation of the repository such as decryption and/or decompression of digital works and transaction messages are also performed by the processing means 1200. The processor element 1201 may be a microprocessor or other suitable computing component. The processor memory 1202 would typically be further comprised of Read Only Memories (ROM) and Random Access Memories (RAM). Such memories would contain the software instructions utilized by the processor element 1201 in performing the functions of the repository.

The storage system 1207 is further comprised of descriptor storage 1203 and content storage 1204. The description tree storage 1203 will store the description tree for the digital work and the content storage will store the associated content. The description tree storage 1203 and content storage 1204 need not be of the same type of storage medium, nor are they necessarily on the same physical device. So for example, the descriptor storage 1203 may be stored on a solid state storage (for rapid retrieval of the description tree information), while the content storage 1204 may be on a high capacity storage such as an optical disk.

The clock 1205 is used to time-stamp various time based conditions for usage rights or for metering usage fees which may be associated with the digital works. The clock 1205 will have an uninterruptable power supply, e.g. a battery, in order to maintain the integrity of the time-stamps. The external interface means 1206 provides for the signal connection to other repositories and to a credit server. The external interface means 1206 provides for the exchange of signals via such standard interfaces such as RS-232 or Personal Computer Manufacturers Card Industry Association (PCMCIA) standards, or FDDI. The external interface means 1206 may also provide network connectivity.

The functional embodiment of a repository is described with reference to FIG. 13. Referring to FIG. 13, the functional embodiment is comprised of an operating system 1301, core repository services 1302, usage transaction handlers 1303, repository specific functions, 1304 and a user interface 1305. The operating system 1301 is specific to the repository and would typically depend on the type of processor being used. The operating system 1301 would also

14

provide the basic services for controlling and interfacing between the basic components of the repository.

The core repository services 1302 comprise a set of functions required by each and every repository. The core repository services 1302 include the session initiation transactions which are defined in greater detail below. This set of services also includes a generic ticket agent which is used to "punch" a digital ticket and a generic authorization server for processing authorization specifications. Digital tickets and authorizations are specific mechanisms for controlling the distribution and use of digital works and are described and more detail below. Note that coupled to the core repository services are a plurality of identification certificates 1306. The identification certificates 1306 are required to enable the use of the repository.

The usage transactions handler 1303 comprise functionality for processing access requests to digital works and for billing fees based on access. The usage transactions supported will be different for each repository type. For example, it may not be necessary for some repositories to handle access requests for digital works.

The repository specific functionality 1304 comprises functionality that is unique to a repository. For example, the master repository has special functionality for issuing digital certificates and maintaining encryption keys. The repository specific functionality 1304 would include the user interface implementation for the repository.

Repository Security Classes

For some digital works the losses caused by any individual instance of unauthorized copying is insignificant and the chief economic concern lies in assuring the convenience of access and low-overhead billing. In such cases, simple and inexpensive handheld repositories and network-based workstations may be suitable repositories, even though the measures and guarantees of security are modest.

At the other extreme, some digital works such as a digital copy of a first run movie or a bearer bond or stock certificate would be of very high value so that it is prudent to employ caution and fairly elaborate security measures to ensure that they are not copied or forged. A repository suitable for holding such a digital work could have elaborate measures for ensuring physical integrity and for verifying authorization before use.

By arranging a universal protocol, all kinds of repositories can communicate with each other in principle. However, creators of some works will want to specify that their works will only be transferred to repositories whose level of security is high enough. For this reason, document repositories have a ranking system for classes and levels of security. The security classes in the currently preferred embodiment are described in Table 2.

TABLE 2

REPOSITORY SECURITY LEVELS

Level Description of Security

- 0 Open system. Document transmission is unencrypted. No digital certificate is required for identification. The security of the system depends mostly on user honesty, since only modest knowledge may be needed to circumvent the security measures. The repository has no provisions for preventing unauthorized programs from running and accessing or copying files. The system does not prevent the use of removable storage and does not encrypt stored files.
- 1 Minimal security. Like the previous class except that stored files are minimally encrypted, including ones on removable storage.

US 6,963,859 B2

15

TABLE 2-continued

REPOSITORY SECURITY LEVELS	
Level	Description of Security
2	Basic security. Like the previous class except that special tools and knowledge are required to compromise the programming, the contents of the repository, or the state of the clock. All digital communications are encrypted. A digital certificate is provided as identification. Medium level encryption is used. Repository identification number is unforgeable.
3	General security. Like the previous class plus the requirement of special tools are needed to compromise the physical integrity of the repository and that modest encryption is used on all transmissions. Password protection is required to use the local user interface. The digital clock system cannot be reset without authorization. No works would be stored on removable storage. When executing works as programs, it runs them in their own address space and does not give them direct access to any file storage or other memory containing system code or works. They can access works only through the transmission transaction protocol.
4	Like the previous class except that high level encryption is used on all communications. Sensors are used to record attempts at physical and electronic tampering. After such tampering, the repository will not perform other transactions until it has reported such tampering to a designated server.
5	Like the previous class except that if the physical or digital attempts at tampering exceed some preset thresholds that threaten the physical integrity of the repository or the integrity of digital and cryptographic barriers, then the repository will save only document description records of history but will erase or destroy any digital identifiers that could be misused if released to an unscrupulous party. It also modifies any certificates of authenticity to indicate that the physical system has been compromised. It also erases the contents of designated documents.
6	Like the previous class except that the repository will attempt wireless communication to report tampering and will employ noisy alarms.
10	This would correspond to a very high level of security. This server would maintain constant communications to remote security systems reporting transactions, sensor readings, and attempts to circumvent security.

The characterization of security levels described in Table 2 is not intended to be fixed. More important is the idea of having different security levels for different repositories. It is anticipated that new security classes and requirements will evolve according to social situations and changes in technology.

Repository User Interface

A user interface is broadly defined as the mechanism by which a user interacts with a repository in order to invoke transactions to gain access to a digital work, or exercise usage rights. As described above, a repository may be embodied in various forms. The user interface for a repository will differ depending on the particular embodiment. The user interface may be a graphical user interface having icons representing the digital works and the various transactions that may be performed. The user interface may be a generated dialog in which a user is prompted for information.

The user interface itself need not be part of the repository. As a repository may be embedded in some other device, the user interface may merely be a part of the device in which the repository is embedded. For example, the repository could be embedded in a "card" that is inserted into an available slot in a computer system. The user interface may be combination of a display, keyboard, cursor control device and software executing on the computer system.

At a minimum, the user interface must permit a user to input information such as access requests and alpha numeric data and provide feedback as to transaction status. The user interface will then cause the repository to initiate the suitable transactions to service the request. Other facets of a particu-

16

lar user interface will depend on the functionality that a repository will provide.

Credit Servers

In the present invention, fees may be associated with the exercise of a right. The requirement for payment of fees is described with each version of a usage right in the usage rights language. The recording and reporting of such fees is performed by the credit server. One of the capabilities enabled by associating fees with rights is the possibility of supporting a wide range of charging models. The simplest model, used by conventional software, is that there is a single fee at the time of purchase, after which the purchaser obtains unlimited rights to use the work as often and for as long as he or she wants. Alternative models, include metered use and variable fees. A single work can have different fees for different uses. For example, viewing a photograph on a display could have different fees than making a hardcopy or including it in a newly created work. A key to these alternative charging models is to have a low overhead means of establishing fees and accounting for credit on these transactions.

A credit server is a computational system that reliably authorizes and records these transactions so that fees are billed and paid. The credit server reports fees to a billing clearinghouse. The billing clearinghouse manages the financial transactions as they occur. As a result, bills may be generated and accounts reconciled. Preferably, the credit server would store the fee transactions and periodically communicate via a network with billing clearinghouse for reconciliation. In such an embodiment, communications with the billing clearinghouse would be encrypted for integrity and security reasons. In another embodiment, the credit server acts as a "debit card" where transactions occur in "real-time" against a user account.

A credit server is comprised of memory, a processing means, a clock, and interface means for coupling to a repository and a financial institution (e.g. a modem). The credit server will also need to have security and authentication functionality. These elements are essentially the same elements as those of a repository. Thus, a single device can be both a repository and a credit server, provided that it has the appropriate processing elements for carrying out the corresponding functions and protocols. Typically, however, a credit server would be a card-sized system in the possession of the owner of the credit. The credit server is coupled to a repository and would interact via financial transactions as described below. Interactions with a financial institution may occur via protocols established by the financial institutions themselves.

In the currently preferred embodiment credit servers associated with both the server and the repository report the financial transaction to the billing clearinghouse. For example, when a digital work is copied by one repository to another for a fee, credit servers coupled to each of the repositories will report the transaction to the billing clearinghouse. This is desirable in that it insures that a transaction will be accounted for in the event of some break in the communication between a credit server and the billing clearinghouse. However, some implementations may embody only a single credit server reporting the transaction to minimize transaction processing at the risk of losing some transactions.

Usage Rights Language

The present invention uses statements in a high level "usage rights language" to define rights associated with digital works and their parts. Usage rights statements are interpreted by repositories and are used to determine what

US 6,963,859 B2

17

transactions can be successfully carried out for a digital work and also to determine parameters for those transactions. For example, sentences in the language determine whether a given digital work can be copied, when and how it can be used, and what fees (if any) are to be charged for that use. Once the usage rights statements are generated, they are encoded in a suitable form for accessing during the processing of transactions.

Defining usage rights in terms of a language in combination with the hierarchical representation of a digital work enables the support of a wide variety of distribution and fee schemes. An example is the ability to attach multiple versions of a right to a work. So a creator may attach a PRINT right to make 5 copies for \$10.00 and a PRINT right to make unlimited copies for \$100.00. A purchaser may then choose which option best fits his needs. Another example is that rights and fees are additive. So in the case of a composite work, the rights and fees of each of the components works is used in determining the rights and fees for the work as a whole. Other features and benefits of the usage rights language will become apparent in the description of distribution and use scenarios provided below.

The basic contents of a right are illustrated in FIG. 14. Referring to FIG. 14, a right 1450 has a transactional component 1451 and a specifications component 1452. A right 1450 has a label (e.g. COPY or PRINT) which indicate the use or distribution privileges that are embodied by the right. The transactional component 1451 corresponds to a particular way in which a digital work may be used or distributed. The transactional component 1451 is typically embodied in software instructions in a repository which implement the use or distribution privileges for the right. The specifications components 1452 are used to specify conditions which must be satisfied prior to the right being exercised or to designate various transaction related parameters. In the currently preferred embodiment, these specifications include copy count 1453, Fees and Incentives 1454, Time 1455, Access and Security 1456 and Control 1457. Each of these specifications will be described in greater detail below with respect to the language grammar elements.

The usage rights language is based on the grammar described below. A grammar is a convenient means for defining valid sequence of symbols for a language. In describing the grammar the notation “[a|b|c]” is used to indicate distinct choices among alternatives. In this example, a sentence can have either an “a”, “b” or “c”. It must include exactly one of them. The braces { } are used to indicate optional items. Note that brackets, bars and braces are used to describe the language of usage rights sentences but do not appear in actual sentences in the language.

In contrast, parentheses are part of the usage rights language. Parentheses are used to group items together in lists. The notation (x*) is used to indicate a variable length list, that is, a list containing one or more items of type x. The notation (x)* is used to indicate a variable number of lists containing x.

Keywords in the grammar are words followed by colons. Keywords are a common and very special case in the language. They are often used to indicate a single value, typically an identifier. In many cases, the keyword and the parameter are entirely optional. When a keyword is given, it often takes a single identifier as its value. In some cases, the keyword takes a list of identifiers.

In the usage rights language, time is specified in an hours:minutes:seconds (or hh:mm:ss) representation. Time zone indicators, e.g. PDT for Pacific Daylight Time, may also be specified. Dates are represented as year/month/day

18

(or YYYY/MMM/DD). Note that these time and date representations may specify moments in time or units of time. Money units are specified in terms of dollars.

Finally, in the usage rights language, various “things” will need to interact with each other. For example, an instance of a usage right may specify a bank account, a digital ticket, etc. Such things need to be identified and are specified herein using the suffix “-ID.”

The Usage Rights Grammar is listed in its entirety in FIG. 15 and is described below.

Grammar element 1501 “Digital Work Rights:=(Rights*)” define the digital work rights as a set of rights. The set of rights attached to a digital work define how that digital work may be transferred, used, performed or played. A set of rights will attach to the entire digital work and in the case of compound digital works, each of the components of the digital work. The usage rights of components of a digital may be different.

Grammar element 1502 “Right:=(Right-Code{Copy-Count}{Control-Spec}{Time-Spec}{Access-Spec}{Fee-Spec})” enumerates the content of a right. Each usage right must specify a right code. Each right may also optionally specify conditions which must be satisfied before the right can be exercised. These conditions are copy count, control, time, access and fee conditions. In the currently preferred embodiment, for the optional elements, the following defaults apply: copy count equals 1, no time limit on the use of the right, no access tests or a security level required to use the right and no fee is required. These conditions will each be described in greater detail below.

It is important to note that a digital work may have multiple versions of a right, each having the same right code. The multiple version would provide alternative conditions and fees for accessing the digital work.

Grammar element 1503 “Right-Code:=(Render-Code|Transport-Code|File-Management-Code|Derivative-Works-Code|Configuration-Code)” distinguishes each of the specific rights into a particular right type (although each right is identified by distinct right codes). In this way, the grammar provides a catalog of possible rights that can be associated with parts of digital works. In the following, rights are divided into categories for convenience in describing them.

Grammar element 1504 “Render-Code:=[Play:{Player:Player-ID}|Print:{Printer:Printer-ID}]” lists a category of rights all involving the making of ephemeral, transitory, or non-digital copies of the digital work. After use the copies are erased.

Play A process of rendering or performing a digital work on some processor. This includes such things as playing digital movies, playing digital music, playing a video game, running a computer program, or displaying a document on a display.

Print To render the work in a medium that is not further protected by usage rights, such as printing on paper.

Grammar element 1505 “Transport-Code:=[[Copy|Transfer|Loan]{Remaining-Rights: Next-Set-of-Rights}]{(Next-Copy-Rights:Next-Set of Rights)}” lists a category of rights involving the making of persistent, usable copies of the digital work on other repositories. The optional Next-Copy-Rights determine the rights on the work after it is transported. If this is not specified, then the rights on the transported copy are the same as on the original. The optional Remaining-Rights specify the rights that remain with a digital work when it is loaned out. If this is not specified, then the default is that no rights can be exercised when it is loaned out.

US 6,963,859 B2

19

Copy Make a new copy of a work

Transfer Moving a work from one repository to another.

Loan Temporarily loaning a copy to another repository for a specified period of time.

Grammar element **1506** “File-Management-Code:= Backup{Back-Up-Copy-Rights:Next-Set of Rights}|Restore|Delete|Folder|Directory{Name:Hide-Local|Hide-Remote}{Parts:Hide-Local|Hide-Remote}” lists a category of rights involving operations for file management, such as the making of backup copies to protect the copy owner against catastrophic equipment failure.

Many software licenses and also copyright law give a copy owner the right to make backup copies to protect against catastrophic failure of equipment. However, the making of uncontrolled backup copies is inherently at odds with the ability to control usage, since an uncontrolled backup copy can be kept and then restored even after the authorized copy was sold.

The File management rights enable the making and restoring of backup copies in a way that respects usage rights, honoring the requirements of both the copy owner and the rights grantor and revenue owner. Backup copies of work descriptions (including usage rights and fee data) can be sent under appropriate protocol and usage rights control to other document repositories of sufficiently high security. Further rights permit organization of digital works into folders which themselves are treated as digital works and whose contents may be “hidden” from a party seeking to determine the contents of a repository.

Backup To make a backup copy of a digital work as protection against media failure.

Restore To restore a backup copy of a digital work.

Delete To delete or erase a copy of a digital work.

Folder To create and name folders, and to move files and folders between folders.

Directory To hide a folder or its contents.

Grammar element **1507** “Derivative-Works-Code:[Extract|Embed|Edit{Process:Process-ID}]{Next-Copy-Rights:Next-Set-ofRights}” lists a category of rights involving the use of a digital work to create new works.

Extract To remove a portion of a work, for the purposes of creating a new work.

Embed To include a work in an existing work.

Edit To alter a digital work by copying, selecting and modifying portions of an existing digital work.

Grammar element **1508** “Configuration-Code:= Install|Uninstall” lists a category of rights for installing and uninstalling software on a repository (typically a rendering repository.) This would typically occur for the installation of a new type of player within the rendering repository.

Install: To install new software on a repository.

Uninstall: To remove existing software from a repository.

Grammar element **1509** “Next-Set-of-Rights:={Add:Set-Of-Rights}{Delete:Set-Of-Rights}{Replace:Set-Of-Rights}{Keep:Set-Of-Rights}” defines how rights are carried forward for a copy of a digital work. If the Next-Copy-Rights is not specified, the rights for the next copy are the same as those of the current copy. Otherwise, the set of rights for the next copy can be specified. Versions of rights after Add: are added to the current set of rights. Rights after Delete: are deleted from the current set of rights. If only right codes are listed after Delete:, then all versions of rights with those codes are deleted. Versions of rights after Replace: subsume all versions of rights of the same type in the current set of rights.

20

If Remaining-Rights is not specified, then there are no rights for the original after all Loan copies are loaned out. If Remaining-Rights is specified, then the Keep: token can be used to simplify the expression of what rights to keep behind. A list of right codes following keep means that all of the versions of those listed rights are kept in the remaining copy. This specification can be overridden by subsequent Delete: or Replace: specifications.

Copy Count Specification

For various transactions, it may be desirable to provide some limit as to the number of “copies” of the work which may be exercised simultaneously for the right. For example, it may be desirable to limit the number of copies of a digital work that may be loaned out at a time or viewed at a time.

Grammar element **1510** “Copy-Count:=(Copies:positive-integer|unlimited)” provides a condition which defines the number of “copies” of a work subject to the right. A copy count can be 0, a fixed number, or unlimited. The copy-count is associated with each right, as opposed to there being just a single copy-count for the digital work. The Copy-Count for a right is decremented each time that a right is exercised. When the Copy-Count equals zero, the right can no longer be exercised. If the Copy-Count is not specified, the default is one.

Control Specification

Rights and fees depend in general on rights granted by the creator as well as further restrictions imposed by later distributors. Control specifications deal with interactions between the creators and their distributors governing the imposition of further restrictions and fees. For example, a distributor of a digital work may not want an end consumer of a digital work to add fees or otherwise profit by commercially exploiting the purchased digital work.

Grammar element **1511** “Control-Spec:=(Control:{Restrictable|Unrestrictable}{Unchargeable|Chargeable})” provides a condition to specify the effect of usage rights and fees of parents on the exercise of the right. A digital work is restrictable if higher level d-blocks can impose further restrictions (time specifications and access specifications) on the right. It is unrestrictable if no further restrictions can be imposed. The default setting is restrictable. A right is unchargeable if no more fees can be imposed on the use of the right. It is chargeable if more fees can be imposed. The default is chargeable.

Time Specification

It is often desirable to assign a start date or specify some duration as to when a right may be exercised. Grammar element **1512** “Time-Spec:={Fixed-Interval|Sliding-Interval|Meter-Time}Until:Expiration-Date)” provides for specification of time conditions on the exercise of a right. Rights may be granted for a specified time. Different kinds of time specifications are appropriate for different kinds of rights. Some rights may be exercised during a fixed and predetermined duration. Some rights may be exercised for an interval that starts the first time that the right is invoked by some transaction. Some rights may be exercised or are charged according to some kind of metered time, which may be split into separate intervals. For example, a right to view a picture for an hour might be split into six ten minute viewings or four fifteen minute viewings or twenty three minute viewings.

The terms “time” and “date” are used synonymously to refer to a moment in time. There are several kinds of time specifications. Each specification represents some limitation on the times over which the usage right applies. The Expiration-Date specifies the moment at which the usage right ends. For example, if the Expiration-Date is “Jan. 1,

US 6,963,859 B2

21

1995,” then the right ends at the first moment of 1995. If the Expiration-Date is specified as *forever*, then the rights are interpreted as continuing without end. If only an expiration date is given, then the right can be exercised as often as desired until the expiration date.

Grammar element **1513** “Fixed-Interval:=From:Start-Time” is used to define a predetermined interval that runs from the start time to the expiration date.

Grammar element **1514** “Sliding-Interval:=Interval:Use-Duration” is used to define an indeterminate (or “open”) start time. It sets limits on a continuous period of time over which the contents are accessible. The period starts on the first access and ends after the duration has passed or the expiration date is reached, whichever comes first. For example, if the right gives 10 hours of continuous access, the use-duration would begin when the first access was made and end 10 hours later.

Grammar element **1515** “Meter-Time: Time-Remaining:Remaining-Use” is used to define a “meter time,” that is, a measure of the time that the right is actually exercised. It differs from the Sliding-Interval specification in that the time that the digital work is in use need not be continuous. For example, if the rights guarantee three days of access, those days could be spread out over a month. With this specification, the rights can be exercised until the meter time is exhausted or the expiration date is reached, whichever comes first.

Remaining-Use:=Time-Unit

Start-Time:=Time-Unit

Use-Duration:=Time-Unit

All of the time specifications include time-unit specifications in their ultimate instantiation.

Security Class and Authorization Specification

The present invention provides for various security mechanisms to be introduced into a distribution or use scheme. Grammar element **1516** “Access-Spec:={SC:Security-Class}{Authorization:Authorization-ID*}{Other-Authorization:Authorization-ID*}{Ticket:Ticket-ID}” provides a means for restricting access and transmission. Access specifications can specify a required security class for a repository to exercise a right or a required authorization test that must be satisfied.

The keyword “SC:” is used to specify a minimum security level for the repositories involved in the access. If “SC:” is not specified, the lowest security level is acceptable.

The optional “Authorization:” keyword is used to specify required authorizations on the same repository as the work. The optional “Other-Authorization:” keyword is used to specify required authorizations on the other repository in the transaction.

The optional “Ticket:” keyword specifies the identity of a ticket required for the transaction. A transaction involving digital tickets must locate an appropriate digital ticket agent who can “punch” or otherwise validate the ticket before the transaction can proceed. Tickets are described in greater detail below.

In a transaction involving a repository and a document server, some usage rights may require that the repository have a particular authorization, that the server have some authorization, or that both repositories have (possibly different) authorizations. Authorizations themselves are digital works (hereinafter referred to as an authorization object) that can be moved between repositories in the same manner as other digital works. Their copying and transferring is subject to the same rights and fees as other digital works. A repository is said to have an authorization if that authorization object is contained within the repository.

22

In some cases, an authorization may be required from a source other than the document server and repository. An authorization object referenced by an Authorization-ID can contain digital address information to be used to set up a communications link between a repository and the authorization source. These are analogous to phone numbers. For such access tests, the communication would need to be established and authorization obtained before the right could be exercised.

For one-time usage rights, a variant on this scheme is to have a digital ticket. A ticket is presented to a digital ticket agent, whose type is specified on the ticket. In the simplest case, a certified generic ticket agent, available on all repositories, is available to “punch” the ticket. In other cases, the ticket may contain addressing information for locating a “special-” ticket agent. Once a ticket has been punched, it cannot be used again for the same kind of transaction (unless it is unpunched or refreshed in the manner described below.) Punching includes marking the ticket with a timestamp of the date and time it was used. Tickets are digital works and can be copied or transferred between repositories according to their usage rights.

In the currently preferred embodiment, a “punched” ticket becomes “unpunched” or “refreshed” when it is copied or extracted. The Copy and Extract operations save the date and time as a proper of the digital ticket. When a ticket agent is given a ticket, it can simply check whether the digital copy was made after the last time that it was punched. Of course, the digital ticket must have the copy or extract usage rights attached thereto.

The capability to unpunch a ticket is important in the following cases:

A digital work is circulated at low cost with a limitation that it can be used only once.

A digital work is circulated with a ticket that can be used once to give discounts on purchases of other works.

A digital work is circulated with a ticket (included in the purchase price and possibly embedded in the work) that can be used for a future upgrade.

In each of these cases, if a paid copy is made of the digital work (including the ticket) the new owner would expect to get a fresh (unpunched) ticket, whether the copy seller has used the work or not. In contrast, loaning a work or simply transferring it to another repository should not revitalize the ticket.

Usage Fees and Incentives Specification

The billing for use of a digital work is fundamental to a commercial distribution system. Grammar Element **1517** “Fee-Spec:={Scheduled-Discount}Regular-Fee-Spec|Scheduled-Fee-Spec|Markup-Spec” provides a range of options for billing for the use of digital works.

A key feature of this approach is the development of low-overhead billing for transactions in potentially small amounts. Thus, it becomes feasible to collect fees of only a few cents each for thousands of transactions.

The grammar differentiates between uses where the charge is per use from those where it is metered by the time unit. Transactions can support fees that the user pays for using a digital work as well as incentives paid by the right grantor to users to induce them to use or distribute the digital work.

The optional scheduled discount refers to the rest of the fee specification—discounting it by a percentage over time. If it is not specified, then there is no scheduled discount. Regular fee specifications are constant over time. Scheduled fee specifications give a schedule of dates over which the fee specifications change. Markup specifications are used in d-blocks for adding a percentage to the fees already being charged.

US 6,963,859 B2

23

Grammar Element **1518** “Scheduled-Discount:=(Scheduled-Discount:(Time-Spec Percentage)*)” A Scheduled-Discount is essentially a scheduled modifier of any other fee specification for this version of the right of the digital work. (It does not refer to children or parent digital works or to other versions of rights.) It is a list of pairs of times and percentages. The most recent time in the list that has not yet passed at the time of the transaction is the one in effect. The percentage gives the discount percentage. For example, the number 10 refers to a 10% discount.

Grammar Element **1519** “Regular-Fee-Spec:={Fee:[Incentive:]{Per-Use-Spec[Metered-Rate-Spec]Best-Price-Spec[Call-For-Price-Spec]}{Min:Money-Unit Per:Time-Spec}{Max:Money-Unit Per:Time-Spec}To:Account-ID)” provides for several kinds of fee specifications.

Fees are paid by the copy-owner/user to the revenue-owner if Fee: is specified. Incentives are paid by the revenue-owner to the user if Incentive: is specified. If the Min: specification is given, then there is a minimum fee to be charged per time-spec unit for its use. If the Max: specification is given, then there is a maximum fee to be charged per time-spec for its use. When Fee: is specified, Account-ID identifies the account to which the fee is to be paid. When Incentive: is specified, Account-ID identifies the account from which the fee is to be paid.

Grammar element **1520** “Per-Use-Spec:=Per-Use:Money-unit” defines a simple fee to be paid every time the right is exercised, regardless of how much time the transaction takes.

Grammar element **1521** “Metered-Rate-Spec:=Metered:Money-Unit Per:Time-Spec” defines a metered-rate fee paid according to how long right is exercised. Thus, the time it takes to complete the transaction determines the fee.

Grammar element **1522** “Best-Price-Spec:=Best-Price:Money-unit Max:Money-unit” is used to specify a best-price that is determined when the account is settled. This specification is to accommodate special deals, rebates, and pricing that depends on information that is not available to the repository. All fee specifications can be combined with tickets or authorizations that could indicate that the consumer is a wholesaler or that he is a preferred customer, or that the seller be authorized in some way. The amount of money in the Max: field is the maximum amount that the use will cost. This is the amount that is tentatively debited from the credit server. However, when the transaction is ultimately reconciled, any excess amount will be returned to the consumer in a separate transaction.

Grammar element **1523** “Call-For-Price-Spec:=Call-For-Price” is similar to a “Best-Price-Spec” in that it is intended to accommodate cases where prices are dynamic. A Call-For-Price Spec requires a communication with a dealer to determine the price. This option cannot be exercised if the repository cannot communicate with a dealer at the time that the right is exercised. It is based on a secure transaction whereby the dealer names a price to exercise the right and passes along a deal certificate which is referenced or included in the billing process.

Grammar element **1524** “Scheduled-Fee-Spec:=(Schedule:(Time-Spec Regular-Fee-Spec)*)” is used to provide a schedule of dates over which the fee specifications change. The fee specification with the most recent date not in the future is the one that is in effect. This is similar to but more general than the scheduled discount. It is more general, because it provides a means to vary the fee agreement for each time period.

24

Grammar element **1525** “Markup-Spec:=Markup:percentage To:Account-ID” is provided for adding a percentage to the fees already being charged. For example, a 5% markup means that a fee of 5% of cumulative fee so far will be allocated to the distributor. A markup specification can be applied to all of the other kinds of fee specifications. It is typically used in a shell provided by a distributor. It refers to fees associated with d-blocks that are parts of the current d-block. This might be a convenient specification for use in taxes, or in distributor overhead.

Examples of Sets of Usage Rights

((Play) (Transfer (SC: 3)) (Delete))

This work can be played without requirements for fee or authorization on any rendering system. It can be transferred to any other repository of security level 3 or greater. It can be deleted.

((Play) (Transfer (SC: 3)) (Delete) (Backup) (Restore (Fee: Per-Use: \$5 To: Account-ID-678)))

Same as the previous example plus rights for backup and restore. The work can be backed up without fee. It can be restored for a \$5 fee payable to the account described by Account-ID-678.

((Play) (Transfer (SC: 3))

(Copy (SC:3)(Fee: Per-Use: \$5 To: Account-ID-678))

(Delete (Incentive: Per-Use: \$2.50 To: Account-ID-678)))

This work can be played, transferred, copied, or deleted. Copy or transfer operations can take place only with repositories of security level three or greater. The fee to make a copy is \$5 payable to Account-ID-678. If a copy is deleted, then an incentive of \$2.50 is paid to the former copy owner.

((Play) (Transfer (SC: 3))

Copy (SC: 3) (Fee: Per-Use: \$10 To: Account-ID-678))

Delete) (Backup) (Restore (SC: 3) (Fee: Per-Use: \$5 To: Account-ID-678)))

Same as the previous example plus fees for copying. The work can be copied digitally for a fee of \$10 payable to Account-ID-678. The repository on which the work is copied or restored must be at security level 3 or greater.

((Play) (Transfer (SC: 3))

(Copy Authorization: License-123-ID (SC: 3)))

The digital work can be played, transferred, or copied. Copies or transfers must be on repositories of security level 3 or greater. Copying requires the license License-123-ID issued to the copying repository. None of the rights require fees.

((Play) (Print Printer: Printer-567-ID (Fee: Per-Use: \$1 To: Account-ID-678)))

This work can be played for free. It can be printed on any printer with the identifier Printer-567-ID for a fee of \$1 payable to the account described by Account-ID-678.

((Play Player: Player-876-ID) (From: 94/02/14 Until: 95/02/15) (Fee: Metered: \$0.01 Per: 0:1:0 Min: \$0.25 Per: 0/1/0 To: Account-ID-567))

This work can be played on any player holding the ID Player-876-ID. The time of this right is from Feb. 14, 1994 until Feb. 15, 1995. The fee for use is one cent per minute with a minimum of 25 cents in any day that it is used, payable to the account described by Account-ID-567.

((Play) (Transfer) (Delete)(Loan 2 (Delete: Transfer Loan)))

This work can be played, transferred, deleted, or loaned. Up to two copies can be loaned out at a time. The loaned copy has the same rights except that it cannot be transferred. When both copies are loaned out, no rights can be exercised on the original on the repository.

US 6,963,859 B2

25

((Play) (Transfer) (Delete) (Backup) (Restore (SC:3))
(Loan 2 Remaining-Copy-Rights: (Delete: Play
Transfer)

Next-Set-of-Rights: (Delete: Transfer Loan)))

Similar to previous example. Rights to Backup and Restore the work are added, where restoration requires a repository of at least security level three. When all copies of the work are loaned out, the remaining copy cannot be played or transferred.

((Play) (Transfer) (Copy) (Print) (Backup) (Restore (SC:3))

(Loan 1 Remaining-Copy-Rights: (Add: Play Print
Backup)

Next-Set-of-Rights: (Delete: Transfer Loan)

(Fee: Metered: \$10 Per: 1:0:0 To: Account-ID-567))

(Loan 1 Remaining-Copy-Rights:

Add: ((Play Player: Player-876-ID) 2 (From: 94/02/14
Until: 95/02/15)

(Fee: Metered: \$0.01 Per: 0:1:0 Min: \$0.25 Per: 0/1/0
To: Account-ID-567)))

The original work has rights to Play, Transfer, Copy, Print, Backup, Restore, and Loan. There are two versions of the Loan right. The first version of the loan right costs \$10 per day but allows the original copy owner to exercise free use of the Play, Print and Backup rights. The second version of the Loan right is free. None of the original rights are applicable. However a right to Play the work at the specified metered rate is added.

((Play Player: Player-Small-Screen-123-ID)

(Embed (Fee: Per-Use \$0.01 To: Account-678-ID))

(Copy (Fee: Per-Use \$1.00 To: Account-678-ID)))

The digital work can be played on any player with the identifier Player-Small-Screen-123-ID. It can be embedded in a larger work. The embedding requires a modest one cent registration fee to Account-678-ID. Digital copies can be made for \$1.00.

Repository Transactions

When a user requests access to a digital work, the repository will initiate various transactions. The combination of transactions invoked will depend on the specifications assigned for a usage right. There are three basic types of transactions, Session Initiation Transactions, Financial Transactions and Usage Transactions. Generally, session initiation transactions are initiated first to establish a valid session. When a valid session is established, transactions corresponding to the various usage rights are invoked. Finally, request specific transactions are performed.

Transactions occur between two repositories (one acting as a server), between a repository and a document playback platform (e.g. for executing or viewing), between a repository and a credit server or between a repository and an authorization server. When transactions occur between more than one repository, it is assumed that there is a reliable communication channel between the repositories. For example, this could be a TCP/IP channel or any other commercially available channel that has built-in capabilities for detecting and correcting transmission errors. However, it is not assumed that the communication channel is secure. Provisions for security and privacy are part of the requirements for specifying and implementing repositories and thus form the need for various transactions.

Message Transmission

Transactions require that there be some communication between repositories. Communication between repositories occurs in units termed as messages. Because the communication line is assumed to be unsecure, all communications with repositories that are above the lowest security class are

26

encrypted utilizing a public key encryption technique. Public key encryption is a well known technique in the encryption arts. The term key refers to a numeric code that is used with encryption and decryption algorithms. Keys come in pairs, where "writing keys" are used to encrypt data and "checking keys" are used to decrypt data. Both writing and checking keys may be public or private. Public keys are those that are distributed to others. Private keys are maintained in confidence.

Key management and security is instrumental in the success of a public key encryption system. In the currently preferred embodiment, one or more master repositories maintain the keys and create the identification certificates used by the repositories.

When a sending repository transmits a message to a receiving repository, the sending repository encrypts all of its data using the public writing key of the receiving repository. The sending repository includes its name, the name of the receiving repository, a session identifier such as a nonce (described below), and a message counter in each message.

In this way, the communication can only be read (to a high probability) by the receiving repository, which holds the private checking key for decryption. The auxiliary data is used to guard against various replay attacks to security. If messages ever arrive with the wrong counter or an old nonce, the repositories can assume that someone is interfering with communication and the transaction terminated.

The respective public keys for the repositories to be used for encryption are obtained in the registration transaction described below.

Session Initiation Transactions

A usage transaction is carried out in a session between repositories. For usage transactions involving more than one repository, or for financial transactions between a repository and a credit server, a registration transaction is performed. A second transaction termed a login transaction, may also be needed to initiate the session. The goal of the registration transaction is to establish a secure channel between two repositories who know each others identities. As it is assumed that the communication channel between the repositories is reliable but not secure, there is a risk that a non-repository may mimic the protocol in order to gain illegitimate access to a repository.

The registration transaction between two repositories is described with respect to FIGS. 16 and 17. The steps described are from the perspective of a "repository-1". registering its identity with a "repository-2". The registration must be symmetrical so the same set of steps will be repeated for repository-2 registering its identity with repository-1. Referring to FIG. 16, repository-1 first generates an encrypted registration identifier, step 1601 and then generates a registration message, step 1602. A registration message is comprised of an identifier of a master repository, the identification certificate for the repository-1 and an encrypted random registration identifier. The identification certificate is encrypted by the master repository in its private key and attests to the fact that the repository (here repository-1) is a bona fide repository. The identification certificate also contains a public key for the repository, the repository security level and a timestamp (indicating a time after which the certificate is no longer valid.) The registration identifier is a number generated by the repository for this registration. The registration identifier is unique to the session and is encrypted in repository-1's private key. The registration identifier is used to improve security of authentication by detecting certain kinds of communications based attacks. Repository-1 then transmit the registration message to repository-2, step 1603.

US 6,963,859 B2

27

Upon receiving the registration message, repository-2 determines if it has the needed public key for the master repository, step 1604. If repository-2 does not have the needed public key to decrypt the identification certificate, the registration transaction terminates in an error, step 1618.

Assuming that repository-2 has the proper public key the identification certificate is decrypted, step 1605. Repository-2 saves the encrypted registration identifier, step 1606, and extracts the repository identifier, step 1607. The extracted repository identifier is checked against a "hotlist" of compromised document repositories, step 1608. In the currently preferred embodiment, each repository will contain "hotlists" of compromised repositories. If the repository is on the "hotlist", the registration transaction terminates in an error per step 1618. Repositories can be removed from the hotlist when their certificates expire, so that the list does not need to grow without bound. Also, by keeping a short list of hotlist certificates that it has previously received, a repository can avoid the work of actually going through the list. These lists would be encrypted by a master repository. A minor variation on the approach to improve efficiency would have the repositories first exchange lists of names of hotlist certificates, ultimately exchanging only those lists that they had not previously received. The "hotlists" are maintained and distributed by Master repositories.

Note that rather than terminating in error, the transaction could request that another registration message be sent based on an identification certificate created by another master repository. This may be repeated until a satisfactory identification certificate is found, or it is determined that trust cannot be established.

Assuming that the repository is not on the hotlist, the repository identification needs to be verified. In other words, repository-2 needs to validate that the repository on the other end is really repository-1. This is termed performance testing and is performed in order to avoid invalid access to the repository via a counterfeit repository replaying a recording of a prior session initiation between repository-1 and repository-2. Performance testing is initiated by repository-2 generating a performance message, step 1609. The performance message consists of a nonce, the names of the respective repositories, the time and the registration identifier received from repository-1. A nonce is a generated message based on some random and variable information (e.g. the time or the temperature.) The nonce is used to check whether repository-1 can actually exhibit correct encrypting of a message using the private keys it claims to have, on a message that it has never seen before. The performance message is encrypted using the public key specified in the registration message of repository-1. The performance message is transmitted to repository-1, step 1610, where it is decrypted by repository-1 using its private key, step 1611. Repository-1 then checks to make sure that the names of the two repositories are correct, step 1612, that the time is accurate, step 1613 and that the registration identifier corresponds to the one it sent, step 1614. If any of these tests fails, the transaction is terminated per step 1616. Assuming that the tests are passed, repository-1 transmits the nonce to repository-2 in the clear, step 1615. Repository-2 then compares the received nonce to the original nonce, step 1617. If they are not identical, the registration transaction terminates in an error per step 1618. If they are the same, the registration transaction has successfully completed.

At this point, assuming that the transaction has not terminated, the repositories exchange messages containing session keys to be used in all communications during the session and synchronize their clocks. FIG. 17 illustrates the

28

session information exchange and clock synchronization steps (again from the perspective of repository-1.) Referring to FIG. 17, repository-1 creates a session key pair, step 1701. A first key is kept private and is used by repository-1 to encrypt messages. The second key is a public key used by repository-2 to decrypt messages. The second key is encrypted using the public key of repository-2, step 1702 and is sent to repository-2, step 1703. Upon receipt, repository-2 decrypts the second key, step 1704. The second key is used to decrypt messages in subsequent communications. When each repository has completed this step, they are both convinced that the other repository is bona fide and that they are communicating with the original. Each repository has given the other a key to be used in decrypting further communications during the session. Since that key is itself transmitted in the public key of the receiving repository only it will be able to decrypt the key which is used to decrypt subsequent messages.

After the session information is exchanged, the repositories must synchronize their clocks. Clock synchronization is used by the repositories to establish an agreed upon time base for the financial records of their mutual transactions. Referring back to FIG. 17, repository-2 initiates clock synchronization by generating a time stamp exchange message, step 1705, and transmits it to repository-1, step 1706. Upon receipt, repository-1 generates its own time stamp message, step 1707 and transmits it back to repository-2, step 1708. Repository-2 notes the current time, step 1709 and stores the time received from repository-1, step 1710. The current time is compared to the time received from repository-1, step 1711. The difference is then checked to see if it exceeds a predetermined tolerance (e.g. one minute), step 1712. If it does, repository-2 terminates the transaction as this may indicate tampering with the repository, step 1713. If not repository-2 computes an adjusted time delta, step 1714. The adjusted time delta is the difference between the clock time of repository-2 and the average of the times from repository-1 and repository-2.

To achieve greater accuracy, repository-2 can request the time again up to a fixed number of times (e.g. five times), repeat the clock synchronization steps, and average the results.

A second session initiation transaction is a Login transaction. The Login transaction is used to check the authenticity of a user requesting a transaction. A Login transaction is particularly prudent for the authorization of financial transactions that will be charged to a credit server. The Login transaction involves an interaction between the user at a user interface and the credit server associated with a repository. The information exchanged here is a login string supplied by the repository/credit server to identify itself to the user, and a Personal Identification Number (PIN) provided by the user to identify himself to the credit server. In the event that the user is accessing a credit server on a repository different from the one on which the user interface resides, exchange of the information would be encrypted using the public and private keys of the respective repositories.

Billing Transactions

Billing Transactions are concerned with monetary transaction with a credit server. Billing Transaction are carried out when all other conditions are satisfied and a usage fee is required for granting the request. For the most part, billing transactions are well understood in the state of the art. These transactions are between a repository and a credit server, or between a credit server and a billing clearinghouse. Briefly, the required transactions include the following:

Registration and LOGIN transactions by which the repository and user establish their bona-fides to a credit

US 6,963,859 B2

29

server. These transactions would be entirely internal in cases where the repository and credit server are implemented as a single system.

Registration and LOGIN transactions, by which a credit server establishes its bona fides to a billing clearinghouse.

An Assign-fee transaction to assign a charge. The information in this transaction would include a transaction identifier, the identities of the repositories in the transaction, and a list of charges from the parts of the digital work. If there has been any unusual event in the transaction such as an interruption of communications, that information is included as well.

An Begin-charges transaction to assign a charge. This transaction is much the same as an assign fee transaction except that it is used for metered use. It includes the same information as the assign-fee transaction as well as the usage fee information. The credit-server is then responsible for running a clock.

An End-charges transaction to end a charge for metered use. (In a variation on this approach, the repositories would exchange periodic charge information for each block of time.)

A report-charges transaction between a personal credit server and a billing clearinghouse. This transaction is invoked at least once per billing period. It is used to pass along information about charges. On debit and credit cards, this transaction would also be used to update balance information and credit limits as needed.

All billing transactions are given a transaction ID and are reported to the credit servers by both the server and the client. This reduces possible loss of billing information if one of the parties to a transaction loses a banking card and provides a check against tampering with the system.

Usage Transactions

After the session initiation transactions have been completed, the usage request may then be processed. To simplify the description of the steps carried out in processing a usage request, the term requester is used to refer to a repository in the requester mode which is initiating a request, and the term server is used to refer to a repository in the server mode and which contains the desired digital work. In many cases such as requests to print or view a work, the requester and server may be the same device and the transactions described in the following would be entirely internal. In such instances, certain transaction steps, such as the registration transaction, need not be performed.

There are some common steps that are part of the semantics of all of the usage rights transactions. These steps are referred to as the common transaction steps. There are two sets—the “opening” steps and the “closing” steps. For simplicity, these are listed here rather than repeating them in the descriptions of all of the usage rights transactions.

Transactions can refer to a part of a digital work, a complete digital work, or a Digital work containing other digital works. Although not described in detail herein, a transaction may even refer to a folder comprised of a plurality of digital works. The term “work” is used to refer to what ever portion or set of digital works is being accessed.

Many of the steps here involve determining if certain conditions are satisfied. Recall that each usage right may have one or more conditions which must be satisfied before the right can be exercised. Digital works have parts and parts have parts. Different parts can have different rights and fees. Thus, it is necessary to verify that the requirements are met for ALL of the parts that are involved in a transaction For

30

brevity, when reference is made to checking whether the rights exist and conditions for exercising are satisfied, it is meant that all such checking takes place for each of the relevant parts of the work.

FIG. 18 illustrates the initial common opening and closing steps for a transaction. At this point it is assumed that registration has occurred and that a “trusted” session is in place. General tests are tests on usage rights associated with the folder containing the work or some containing folder higher in the file system hierarchy. These tests correspond to requirements imposed on the work as a consequence of its being on the particular repository, as opposed to being attached to the work itself. Referring to FIG. 18, prior to initiating a usage transaction, the requester performs any general tests that are required before the right associated with the transaction can be exercised, step, 1801. For example, install, uninstall and delete rights may be implemented to require that a requester have an authorization certificate before the right can be exercised. Another example is the requirement that a digital ticket be present and punched before a digital work may be copied to a requester. If any of the general tests fail, the transaction is not initiated, step, 1802. Assuming that such required tests are passed, upon receiving the usage request, the server generates a transaction identifier that is used in records or reports of the transaction, step 1803. The server then checks whether the digital work has been granted the right corresponding to the requested transaction, step 1804. If the digital work has not been granted the right corresponding to the request, the transaction terminates, step 1805. If the digital work has been granted the requested right, the server then determines if the various conditions for exercising the right are satisfied. Time based conditions are examined, step 1806. These conditions are checked by examining the time specification for the the version of the right. If any of the conditions are not satisfied, the transaction terminates per step 1805.

Assuming that the time based conditions are satisfied, the server checks security and access conditions, step 1807. Such security and access conditions are satisfied if: 1) the requester is at the specified security class, or a higher security class, 2) the server satisfies any specified authorization test and 3) the requester satisfies any specified authorization tests and has any required digital tickets. If any of the conditions are not satisfied, the transaction terminates per step 1805.

Assuming that the security and access conditions are all satisfied, the server checks the copy count condition, step 1808. If the copy count equals zero, then the transaction cannot be completed and the transaction terminates per step 1805.

Assuming that the copy count does not equal zero, the server checks if the copies in use for the requested right is greater than or equal to any copy count for the requested right (or relevant parts), step 1809. If the copies in use is greater than or equal to the copy count, this indicates that usage rights for the version of the transaction have been exhausted. Accordingly, the server terminates the transaction, step 1805. If the copy count is less than the copies in use for the transaction the transaction can continue, and the copies in use would be incremented by the number of digital works requested in the transaction, step 1810.

The server then checks if the digital work has a “Loan” access right, step 1811. The “Loan” access right is a special case since remaining rights may be present even though all copies are loaned out. If the digital work has the “Loan” access right, a check is made to see if all copies have been

US 6,963,859 B2

31

loaned out, step **1812**. The number of copies that could be loaned is the sum of the Copy-Counts for all of the versions of the loan right of the digital work. For a composite work, the relevant figure is the minimal such sum of each of the components of the composite work. If all copies have been loaned out, the remaining rights are determined, step **1813**. The remaining-rights is determined from the remaining rights specifications from the versions of the Loan right. If there is only one version of the Loan right, then the determination is simple. The remaining rights are the ones specified in that version of the Loan right, or none if Remaining-Rights: is not specified. If there are multiple versions of the Loan right and all copies of all of the versions are loaned out, then the remaining rights is taken as the minimum set (intersection) of remaining rights across all of the versions of the loan right. The server then determines if the requested right is in the set of remaining rights, step **1814**. If the requested right is not in the set of remaining rights, the server terminates the transaction, step **1805**.

If Loan is not a usage right for the digital work or if all copies have not been loaned out or the requested right is in the set of remaining rights, fee conditions for the right are then checked, step **1815**. This will initiate various financial transactions between the repository and associated credit server. Further, any metering of usage of a digital work will commence. If any financial transaction fails, the transaction terminates per step **1805**.

It should be noted that the order in which the conditions are checked need not follow the order of steps **1806–1815**.

At this point, right specific steps are now performed and are represented here as step **1816**. The right specific steps are described in greater detail below.

The common closing transaction steps are now performed. Each of the closing transaction steps are performed by the server after a successful completion of a transaction. Referring back to FIG. **18**, the copies in use value for the requested right is decremented by the number of copies involved in the transaction, step **1817**. Next, if the right had a metered usage fee specification, the server subtracts the elapsed time from the Remaining-Use-Time associated with the right for every part involved in the transaction, step **1818**. Finally, if there are fee specifications associated with the right, the server initiates End-Charge financial transaction to confirm billing, step **1819**.

Transmission Protocol

An important area to consider is the transmission of the digital work from the server to the requester. The transmission protocol described herein refers to events occurring after a valid session has been created. The transmission protocol must handle the case of disruption in the communications between the repositories. It is assumed that interference such as injecting noise on the communication channel can be detected by the integrity checks (e.g., parity, checksum, etc.) that are built into the transport protocol and are not discussed in detail herein.

The underlying goal in the transmission protocol is to preclude certain failure modes, such as malicious or accidental interference on the communications channel. Suppose, for example, that a user pulls a card with the credit server at a specific time near the end of a transaction. There should not be a vulnerable time at which “pulling the card” causes the repositories to fail to correctly account for the number of copies of the work that have been created. Restated, there should be no time at which a party can break a connection as a means to avoid payment after using a digital work.

If a transaction is interrupted (and fails), both repositories restore the digital works and accounts to their state prior to the failure, modulo records of the failure itself.

32

FIG. **19** is a state diagram showing steps in the process of transmitting information during a transaction. Each box represents a state of a repository in either the server mode (above the central dotted line **1901**) or in the requester mode (below the dotted line **1901**). Solid arrows stand for transitions between states. Dashed arrows stand for message communications between the repositories. A dashed message arrow pointing to a solid transition arrow is interpreted as meaning that the transition takes place when the message is received. Unlabeled transition arrows take place unconditionally. Other labels on state transition arrows describe conditions that trigger the transition.

Referring now to FIG. **19**, the server is initially in a state **1902** where a new transaction is initiated via start message **1903**. This message includes transaction information including a transaction identifier and a count of the blocks of data to be transferred. The requester, initially in a wait state **1904** then enters a data wait state **1905**.

The server enters a data transmit state **1906** and transmits a block of data **1907** and then enters a wait for acknowledgement state **1908**. As the data is received, the requesters enters a data receive state **1909** and when the data blocks is completely received it enters an acknowledgement state **1910** and transmits an Acknowledgement message **1911** to the server.

If there are more blocks to send, the server waits until receiving an Acknowledgement message from the requester. When an Acknowledgement message is received it sends the next block to the requester and again waits for acknowledgement. The requester also repeats the same cycle of states.

If the server detects a communications failure before sending the last block, it enters a cancellation state **1912** wherein the transaction is cancelled. Similarly, if the requester detects a communications failure before receiving the last block it enters a cancellation state **1913**.

If there are no more blocks to send, the server commits to the transaction and waits for the final Acknowledgement in state **1914**. If there is a communications failure before the server receives the final Acknowledgement message, it still commits to the transaction but includes a report about the event to its credit server in state **1915**. This report serves two purposes. It will help legitimize any claims by a user of having been billed for receiving digital works that were not completely received. Also it helps to identify repositories and communications lines that have suspicious patterns of use and interruption. The server then enters its completion state **1916**.

On the requester side, when there are no more blocks to receive, the requester commits to the transaction in state **1917**. If the requester detects a communications failure at this state, it reports the failure to its credit server in state **1918**, but still commits to the transaction. When it has committed, it sends an acknowledgement message to the server. The server then enters its completion state **1919**.

The key property is that both the server and the requester cancel a transaction if it is interrupted before all of the data blocks are delivered, and commits to it if all of the data blocks have been delivered.

There is a possibility that the server will have sent all of the data blocks (and committed) but the requester will not have received all of them and will cancel the transaction. In this case, both repositories will presumably detect a communications failure and report it to their credit server. This case will probably be rare since it depends on very precise timing of the communications failure. The only consequence will be that the user at the requester repository may want to

US 6,963,859 B2

33

request a refund from the credit services—and the case for that refund will be documented by reports by both repositories.

To prevent loss of data, the server should not delete any transferred digital work until receiving the final acknowledgement from the requester. But it also should not use the file. A well known way to deal with this situation is called “two-phase commit” or 2PC.

Two-phase commit works as follows. The first phase works the same as the method described above. The server sends all of the data to the requester. Both repositories mark the transaction (and appropriate files) as uncommitted. The server sends a ready-to-commit message to the requester. The requester sends back an acknowledgement. The server then commits and sends the requester a commit message. When the requester receives the commit message, it commits the file.

If there is a communication failure or other crash, the requester must check back with the server to determine the status of the transaction. The server has the last word on this. The requester may have received all of the data, but if it did not get the final message, it has not committed. The server can go ahead and delete files (except for transaction records) once it commits, since the files are known to have been fully transmitted before starting the 2PC cycle.

There are variations known in the art which can be used to achieve the same effect. For example, the server could use an additional level of encryption when transmitting a work to a client. Only after the client sends a message acknowledging receipt does it send the key. The client then agrees to pay for the digital work. The point of this variation is that it provides a clear audit trail that the client received the work. For trusted systems, however, this variation adds a level of encryption for no real gain in accountability.

The transaction for specific usage rights are now discussed.

The Copy Transaction

A Copy transaction is a request to make one or more independent copies of the work with the same or lesser usage rights. Copy differs from the extraction right discussed later in that it refers to entire digital works or entire folders containing digital works. A copy operation cannot be used to remove a portion of a digital work.

The requester sends the server a message to initiate the Copy Transaction. This message indicates the work to be copied, the version of the copy right to be used for the transaction, the destination address information (location in a folder) for placing the work, the file data for the work (including its size), and the number of copies requested.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the client according to the transmission protocol. If a Next-Set-Of-Rights has been provided in the version of the right, those rights are transmitted as the rights for the work. Otherwise, the rights of the original are transmitted. In any event, the Copy-Count field for the copy of the digital work being sent right is set to the number-of-copies requested.

The requester records the work contents, data, and usage rights and stores the work. It records the date and time that the copy was made in the properties of the digital work.

The repositories perform the common closing transaction steps.

34

The Transfer Transaction

A Transfer transaction is a request to move copies of the work with the same or lesser usage rights to another repository. In contrast with a copy transaction, this results in removing the work copies from the server.

The requester sends the server a message to initiate the Transfer Transaction. This message indicates the work to be transferred, the version of the transfer right to be used in the transaction, the destination address information for placing the work, the file data for the work, and the number of copies involved.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester according to the transmission protocol. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the work. Otherwise, the rights of the original are transmitted. In either case, the Copy-Count field for the transmitted rights are set to the number-of-copies requested.

The requester records the work contents, data, and usage rights and stores the work.

The server decrements its copy count by the number of copies involved in the transaction.

The repositories perform the common closing transaction steps.

If the number of copies remaining in the server is now zero, it erases the digital work from its memory.

The Loan Transaction

A loan transaction is a mechanism for loaning copies of a digital work. The maximum duration of the loan is determined by an internal parameter of the digital work. Works are automatically returned after a predetermined time period.

The requester sends the server a message to initiate the Transfer Transaction. This message indicates the work to be loaned, the version of the loan right to be used in the transaction, the destination address information for placing the work, the number of copies involved, the file data for the work, and the period of the loan.

The server checks the validity of the requested loan period, and ends with an error if the period is not valid. Loans for a loaned copy cannot extend beyond the period of the original loan to the server.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the work. Otherwise, the rights of the original are transmitted, as modified to reflect the loan period.

The requester records the digital work contents, data, usage rights, and loan period and stores the work.

The server updates the usage rights information in the digital work to reflect the number of copies loaned out. The repositories perform the common closing transaction steps.

The server updates the usage rights data for the digital work. This may preclude use of the work until it is returned from the loan. The user on the requester platform can now use the transferred copies of the digital work. A user accessing the original repository cannot use the digital work, unless there are copies remaining. What happens next depends on the order of events in time.

US 6,963,859 B2

35

Case 1. If the time of the loan period is not yet exhausted and the requester sends the repository a Return message.

The return message includes the requester identification, and the transaction ID.

The server decrements the copies-in-use field by the number of copies that were returned. (If the number of digital works returned is greater than the number actually borrowed, this is treated as an error.) This step may now make the work available at the server for other users.

The requester deactivates its copies and removes the contents from its memory.

Case 2. If the time of the loan period is exhausted and the requester has not yet sent a Return message.

The server decrements the copies-in-use field by the number digital works that were borrowed.

The requester automatically deactivates its copies of the digital work. It terminates all current uses and erases the digital work copies from memory. One question is why a requester would ever return a work earlier than the period of the loan, since it would be returned automatically anyway. One reason for early return is that there may be a metered fee which determines the cost of the loan. Returning early may reduce that fee.

The Play Transaction

A play transaction is a request to use the contents of a work. Typically, to “play” a work is to send the digital work through some kind of transducer, such as a speaker or a display device. The request implies the intention that the contents will not be communicated digitally to any other system. For example, they will not be sent to a printer, recorded on any digital medium, retained after the transaction or sent to another repository.

This term “play” is natural for examples like playing music, playing a movie, or playing a video game. The general form of play means that a “player” is used to use the digital work. However, the term play covers all media and kinds of recordings. Thus one would “play” a digital work, meaning, to render it for reading, or play a computer program, meaning to execute it. For a digital ticket the player would be a digital ticket agent.

The requester sends the server a message to initiate the play transaction. This message indicates the work to be played, the version of the play right to be used in the transaction, the identity of the player being used, and the file data for the work.

The server checks the validity of the player identification and the compatibility of the player identification with the player specification in the right. It ends with an error if these are not satisfactory.

The repositories perform the common opening transaction steps.

The server and requester read and write the blocks of data as requested by the player according to the transmission protocol. The requester plays the work contents, using the player.

When the player is finished, the player and the requester remove the contents from their memory.

The repositories perform the common closing transaction steps.

The Print Transaction

A Print transaction is a request to obtain the contents of a work for the purpose of rendering them on a “printer.” We use the term “printer” to include the common case of writing

36

with ink on paper. However, the key aspect of “printing” in our use of the term is that it makes a copy of the digital work in a place outside of the protection of usage rights. As with all rights, this may require particular authorization certificates.

Once a digital work is printed, the publisher and user are bound by whatever copyright laws are in effect. However, printing moves the contents outside the control of repositories. For example, absent any other enforcement mechanisms, once a digital work is printed on paper, it can be copied on ordinary photocopying machines without intervention by a repository to collect usage fees. If the printer to a digital disk is permitted, then that digital copy is outside of the control of usage rights. Both the creator and the user know this, although the creator does not necessarily give tacit consent to such copying, which may violate copyright laws.

The requester sends the server a message to initiate a Print transaction. This message indicates the work to be played, the identity of the printer being used, the file data for the work, and the number of copies in the request.

The server checks the validity of the printer identification and the compatibility of the printer identification with the printer specification in the right. It ends with an error if these are not satisfactory.

The repositories perform the common opening transaction steps.

The server transmits blocks of data according to the transmission protocol.

The requester prints the work contents, using the printer. When the printer is finished, the printer and the requester remove the contents from their memory.

The repositories perform the common closing transaction steps.

The Backup Transaction

A Backup transaction is a request to make a backup copy of a digital work, as a protection against media failure. In the context of repositories, secure backup copies differ from other copies in three ways: (1) they are made under the control of a Backup transaction rather than a Copy transaction, (2) they do not count as regular copies, and (3) they are not usable as regular copies. Generally, backup copies are encrypted.

Although backup copies may be transferred or copied, depending on their assigned rights, the only way to make them useful for playing, printing or embedding is to restore them.

The output of a Backup operation is both an encrypted data file that contains the contents and description of a work, and a restoration file with an encryption key for restoring the encrypted contents. In many cases, the encrypted data file would have rights for “printing” it to a disk outside of the protection system, relying just on its encryption for security. Such files could be stored anywhere that was physically safe and convenient. The restoration file would be held in the repository. This file is necessary for the restoration of a backup copy. It may have rights for transfer between repositories.

The requester sends the server a message to initiate a backup transaction. This message indicates the work to be backed up, the version of the backup right to be used in the transaction, the destination address information for placing the backup copy, the file data for the work.

The repositories perform the common opening transaction steps.

US 6,963,859 B2

37

The server transmits the requested contents and data to the requester. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the work. Otherwise, a set of default rights for backup files of the original are transmitted by the server.

The requester records the work contents, data, and usage rights. It then creates a one-time key and encrypts the contents file. It saves the key information in a restoration file.

The repositories perform the common closing transaction steps.

In some cases, it is convenient to be able to archive the large, encrypted contents file to secure offline storage, such as a magneto-optical storage system or magnetic tape. This creation of a non-repository archive file is as secure as the encryption process. Such non-repository archive storage is considered a form of "printing" and is controlled by a print right with a specified "archive-printer." An archive-printer device is programmed to save the encrypted contents file (but not the description file) offline in such a way that it can be retrieved.

The Restore Transaction

A Restore transaction is a request to convert an encrypted backup copy of a digital work into a usable copy. A restore operation is intended to be used to compensate for catastrophic media failure. Like all usage rights, restoration rights can include fees and access tests including authorization checks.

The requester sends the server a message to initiate a Restore transaction. This message indicates the work to be restored, the version of the restore right for the transaction, the destination address information for placing the work, and the file data for the work.

The server verifies that the contents file is available (i.e. a digital work corresponding to the request has been backed-up.) If it is not, it ends the transaction with an error.

The repositories perform the common opening transaction steps.

The server retrieves the key from the restoration file. It decrypts the work contents, data, and usage rights.

The server transmits the requested contents and data to the requester according to the transmission protocol. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the work. Otherwise, a set of default rights for backup files of the original are transmitted by the server.

The requester stores the digital work.

The repositories perform the common closing transaction steps.

The Delete Transaction

A Delete transaction deletes a digital work or a number of copies of a digital work from a repository. Practically all digital works would have delete rights.

The requester sends the server a message to initiate a delete transaction. This message indicates the work to be deleted, the version of the delete right for the transaction.

The repositories perform the common opening transaction steps.

The server deletes the file, erasing it from the file system.

The repositories perform the common closing transaction steps.

The Directory Transaction

A Directory transaction is a request for information about folders, digital works, and their parts. This amounts to

38

roughly the same idea as protection codes in a conventional file system like TENEX, except that it is generalized to the full power of the access specifications of the usage rights language.

The Directory transaction has the important role of passing along descriptions of the rights and fees associated with a digital work. When a user wants to exercise a right, the user interface of his repository implicitly makes a directory request to determine the versions of the right that are available. Typically these are presented to the user such as with different choices of billing for exercising a right. Thus, many directory transactions are invisible to the user and are exercised as part of the normal process of exercising all rights.

The requester sends the server a message to initiate a Directory transaction. This message indicates the file or folder that is the root of the directory request and the version of the directory right used for the transaction.

The server verifies that the information is accessible to the requester.

In particular, it does not return the names of any files that have a HIDE-NAME status in their directory specifications, and it does not return the parts of any folders or files that have HIDE-PARTS in their specification. If the information is not accessible, the server ends the transaction with an error.

The repositories perform the common opening transaction steps.

The server sends the requested data to the requester according to the transmission protocol.

The requester records the data.

The repositories perform the common closing transaction steps.

The Folder Transaction

A Folder transaction is a request to create or rename a folder, or to move a work between folders. Together with Directory rights, Folder rights control the degree to which organization of a repository can be accessed or modified from another repository.

The requester sends the server a message to initiate a Folder transaction. This message indicates the folder that is the root of the folder request, the version of the folder right for the transaction, an operation, and data.

The operation can be one of create, rename, and move file. The data are the specifications required for the operation, such as a specification of a folder or digital work and a name.

The repositories perform the common opening transaction steps.

The server performs the requested operation—creating a folder, renaming a folder, or moving a work between folders.

The repositories perform the common closing transaction steps.

The Extract Transaction

An extract transaction is a request to copy a part of a digital work and to create a new work containing it. The extraction operation differs from copying in that it can be used to separate a part of a digital work from d-blocks or shells that place additional restrictions or fees on it. The extraction operation differs from the edit operation in that it does not change the contents of a work, only its embedding in d-blocks. Extraction creates a new digital work.

The requester sends the server a message to initiate an Extract transaction. This message indicates the part of

US 6,963,859 B2

39

the work to be extracted, the version of the extract right to be used in the transaction, the destination address information for placing the part as a new work, the file data for the work, and the number of copies involved.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester according to the transmission protocol. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the new work. Otherwise, the rights of the original are transmitted. The Copy-Count field for this right is set to the number-of-copies requested.

The requester records the contents, data, and usage rights and stores the work. It records the date and time that new work was made in the properties of the work.

The repositories perform the common closing transaction steps.

The Embed Transaction

An embed transaction is a request to make a digital work become a part of another digital work or to add a shell d-block to enable the adding of fees by a distributor of the work.

The requester sends the server a message to initiate an Embed transaction. This message indicates the work to be embedded, the version of the embed right to be used in the transaction, the destination address information for placing the part as a work, the file data for the work, and the number of copies involved.

The server checks the control specifications for all of the rights in the part and the destination. If they are incompatible, the server ends the transaction with an error.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester according to the transmission protocol. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the new work. Otherwise, the rights of the original are transmitted. The Copy-Count field for this right is set to the number-of-copies requested.

The requester records the contents, data, and usage rights and embeds the work in the destination file.

The repositories perform the common closing transaction steps.

The Edit Transaction

An Edit transaction is a request to make a new digital work by copying, selecting and modifying portions of an existing digital work. This operation can actually change the contents of a digital work. The kinds of changes that are permitted depend on the process being used. Like the extraction operation, edit operates on portions of a digital work. In contrast with the extract operation, edit does not effect the rights or location of the work. It only changes the contents. The kinds of changes permitted are determined by the type specification of the processor specified in the rights. In the currently preferred embodiment, an edit transaction changes the work itself and does not make a new work. However, it would be a reasonable variation to cause a new copy of the work to be made.

The requester sends the server a message to initiate an Edit transaction. This message indicates the work to be edited, the version of the edit right to be used in the transaction, the file data for the work (including its

40

size), the process-ID for the process, and the number of copies involved.

The server checks the compatibility of the process-ID to be used by the requester against any process-ID specification in the right. If they are incompatible, it ends the transaction with an error.

The repositories perform the common opening transaction steps.

The requester uses the process to change the contents of the digital work as desired. (For example, it can select and duplicate parts of it; combine it with other information; or compute functions based on the information. This can amount to editing text, music, or pictures or taking whatever other steps are useful in creating a derivative work.)

The repositories perform the common closing transaction steps.

The edit transaction is used to cover a wide range of kinds of works. The category describes a process that takes as its input any portion of a digital work and then modifies the input in some way. For example, for text, a process for editing the text would require edit rights. A process for "summarizing" or counting words in the text would also be considered editing. For a music file, processing could involve changing the pitch or tempo, or adding reverberations, or any other audio effect. For digital video works, anything which alters the image would require edit rights. Examples would be colorizing, scaling, extracting still photos, selecting and combining frames into story boards, sharpening with signal processing, and so on.

Some creators may want to protect the authenticity of their works by limiting the kinds of processes that can be performed on them. If there are no edit rights, then no processing is allowed at all. A processor identifier can be included to specify what kind of process is allowed. If no process identifier is specified, then arbitrary processors can be used. For an example of a specific process, a photographer may want to allow use of his photograph but may not want it to be colorized. A musician may want to allow extraction of portions of his work but not changing of the tonality.

Authorization Transactions

There are many ways that authorization transactions can be defined. In the following, our preferred way is to simply define them in terms of other transactions that we already need for repositories. Thus, it is convenient sometimes to speak of "authorization transactions," but they are actually made up of other transactions that repositories already have.

A usage right can specify an authorization-ID, which identifies an authorization object (a digital work in a file of a standard format) that the repository must have and which it must process. The authorization is given to the generic authorization (or ticket) server of the repository which begins to interpret the authorization.

As described earlier, the authorization contains a server identifier, which may just be the generic authorization server or it may be another server. When a remote authorization server is required, it must contain a digital address. It may also contain a digital certificate.

If a remote authorization server is required, then the authorization process first performs the following steps:

The generic authorization server attempts to set up the communications channel. (If the channel cannot be set up, then authorization fails with an error.)

When the channel is set up, it performs a registration process with the remote repository. (If registration fails, then the authorization fails with an error.)

US 6,963,859 B2

41

When registration is complete, the generic authorization server invokes a "Play" transaction with the remote repository, supplying the authorization document as the digital work to be played, and the remote authorization server (a program) as the "player." (If the player cannot be found or has some other error, then the authorization fails with an error.)

The authorization server then "plays" the authorization. This involves decrypting it using either the public key of the master repository that issued the certificate or the session key from the repository that transmitted it. The authorization server then performs various tests. These tests vary according to the authorization server. They include such steps as checking issue and validity dates of the authorization and checking any hot-lists of known invalid authorizations. The authorization server may require carrying out any other transactions on the repository as well, such as checking directories, getting some person to supply a password, or playing some other digital work. It may also invoke some special process for checking information about locations or recent events. The "script" for such steps is contained within the authorization server.

If all of the required steps are completed satisfactorily, the authorization server completes the transaction normally, signaling that authorization is granted.

The Install Transaction

An Install transaction is a request to install a digital work as runnable software on a repository. In a typical case, the requester repository is a rendering repository and the software would be a new kind or new version of a player. Also in a typical case, the software would be copied to file system of the requester repository before it is installed.

The requester sends the server an Install message. This message indicates the work to be installed, the version of the Install right being invoked, and the file data for the work (including its size).

The repositories perform the common opening transaction steps.

The requester extracts a copy of the digital certificate for the software. If the certificate cannot be found or the master repository for the certificate is not known to the requester, the transaction ends with an error.

The requester decrypts the digital certificate using the public key of the master repository, recording the identity of the supplier and creator, a key for decrypting the software, the compatibility information, and a tamper-checking code. (This step certifies the software.)

The requester decrypts the software using the key from the certificate and computes a check code on it using a 1-way hash function. If the check-code does not match the tamper-checking code from the certificate, the installation transaction ends with an error. (This step assures that the contents of the software, including the various scripts, have not been tampered with.)

The requester retrieves the instructions in the compatibility-checking script and follows them. If the software is not compatible with the repository, the installation transaction ends with an error. (This step checks platform compatibility.)

The requester retrieves the instructions in the installation script and follows them. If there is an error in this process (such as insufficient resources), then the transaction ends with an error. Note that the installation process puts the runnable software in a place in the

42

repository where it is no longer accessible as a work for exercising any usage rights other than the execution of the software as part of repository operations in carrying out other transactions.

The repositories perform the common closing transaction steps.

The Uninstall Transaction

An Uninstall transaction is a request to remove software from a repository. Since uncontrolled or incorrect removal of software from a repository could compromise its behavioral integrity, this step is controlled.

The requester sends the server an Uninstall message. This message indicates the work to be uninstalled, the version of the Uninstall right being invoked, and the file data for the work (including its size).

The repositories perform the common opening transaction steps.

The requester extracts a copy of the digital certificate for the software. If the certificate cannot be found or the master repository for the certificate is not known to the requester, the transaction ends with an error.

The requester checks whether the software is installed. If the software is not installed, the transaction ends with an error.

The requester decrypts the digital certificate using the public key of the master repository, recording the identity of the supplier and creator, a key for decrypting the software, the compatibility information, and a tamper-checking code. (This step authenticates the certification of the software, including the script for uninstalling it.)

The requester decrypts the software using the key from the certificate and computes a check code on it using a 1-way hash function. If the check-code does not match the tamper-checking code from the certificate, the installation transaction ends with an error. (This step assures that the contents of the software, including the various scripts, have not been tampered with.)

The requester retrieves the instructions in the uninstallation script and follows them. If there is an error in this process (such as insufficient resources), then the transaction ends with an error.

The repositories perform the common closing transaction steps.

Distribution and Use Scenarios

To appreciate the robustness and flexibility of the present invention, various distribution and use scenarios for digital works are illustrated below. These scenarios are meant to be exemplary rather than exhaustive.

50 Consumers as Unpaid Distributors

In this scenario, a creator distributes copies of his works to various consumers. Each consumer is a potential distributor of the work. If the consumer copies the digital work (usually for a third party), a fee is collected and automatically paid to the creator.

This scenario is a new twist for digital works. It depends on the idea that "manufacturing" is just copying and is essentially free. It also assumes that the consumers as distributors do not require a fee for their time and effort in distributing the work.

This scenario is performed as follows:

A creator creates a digital work. He grants a Copy right with fees paid back to himself. If he does not grant an Embed right, then consumers cannot use the mechanism to act as distributors to cause fees to be paid to themselves on future copies. Of course, they could negotiate side deals or trades to transfer money on their own, outside of the system.

US 6,963,859 B2

43

Paid Distributors

In another scenario, every time a copy of a digital work is sold a fee is paid to the creator and also to the immediate distributor.

This scenario does not give special status to any particular distributor. Anyone who sells a document has the right to add a fee to the sale price. The fee for sale could be established by the consumer. It could also be a fixed nominal amount that is contributed to the account of some charity. This scenario is performed as follows:

A creator creates a digital work. He grants a Copy right with fees to be paid back to himself. He grants an Embed right, so that anyone can add shells to have fees paid to themselves.

A distributor embeds the work in a shell, with fees specified to be paid back to himself. If the distributor is content to receive fees only for copies that he sells himself, he grants an Extract right on the shell.

When a consumer buys a copy from the distributor, fees are paid both to the distributor and to the creator. If he chooses, the consumer can extract the work from the distributor's shell. He cannot extract it from the creator's shell. He can add his own shell with fees to be paid to himself.

Licensed Distribution

In this scenario, a creator wants to protect the reputation and value of his work by making certain requirements on its distributors. He issues licenses to distributors that satisfy the requirements, and in turn, promises to reward their efforts by assuring that the work will not be distributed over competing channels. The distributors incur expenses for selecting the digital work, explaining it to buyers, promoting its sale, and possibly for the license itself. The distributor obtains the right to enclose the digital work in a shell, whose function is to permit the attachment of usage fees to be paid to the distributor in addition to the fees to be paid to the creator.

This differs from the previous scenario in that it precludes the typical copy owner from functioning as a distributor, since the consumer lacks a license to copy the document. Thus, a consumer cannot make copies, even for free. All copies must come initially from authorized distributors. This version makes it possible to hold distributors accountable in some way for the sales and support of the work, by controlling the distribution of certificates that enable distributors to legitimately charge fees and copy owners to make copies. Since licenses are themselves digital works, the same mechanisms give the creators control over distributors by charging for licenses and putting time limits on their validity.

This scenario is performed as follows:

A creator purchases a digital distribution license that he will hand out to his distributors. He puts access requirements (such as a personal license) on the Copy and Transfer rights on the distribution license so that only he can copy or transfer it.

The creator also creates a digital work. He grants an Embed right and a Copy right, both of which require the distribution license to be exercised. He grants a Play right so that the work can be played by anyone. He may optionally add a Transfer or Loan right, so that end consumers can do some non-commercial exchange of the work among friends.

A distributor obtains the distribution license and a number of copies of the work. He makes copies for his customers, using his distribution license.

A customer buys and uses the work. He cannot make new copies because he lacks a distribution license.

44

Super Distributors

This is a variation on the previous scenarios. A distributor can sell to anyone and anyone can sell additional copies, resulting in fees being paid back to the creator. However, only licensed distributors can add fees to be paid to themselves.

This scenario gives distributors the right to add fees to cover their own advertising and promotional costs, without making them be the sole suppliers. Their customers can also make copies, thus broadening the channel without diminishing their revenues. This is because distributors collect fees from copies of any copies that they originally sold. Only distributors can add fees.

This scenario is performed similarly to the previous ones. There are two key differences. (1) The creator only grants Embed rights for people who have a Distribution license. This is done by putting a requirement for a distributor's license on the Embed right. Consequently, non-distributors cannot add their own fees. (2) The Distributor does not grant Extract rights, so that consumers cannot avoid paying fees to the Distributor if they make subsequent copies. Consequently, all subsequent copies result in fees paid to the Distributor and the Creator.

1-Level Distribution Fees

In this scenario, a distributor gets a fee for any copy he sells directly. However, if one of his customers sells further copies, he gets no further fee for those copies.

This scenario pays a distributor only for use of copies that he actually sold.

This scenario is performed similarly to the previous ones. The key feature is that the distributor creates a shell which specifies fees to be paid to him. He puts Extract rights on the shell. When a consumer buys the work, he can extract away the distributor's shell. Copies made after that will not require fees to be paid to the distributor.

Distribution Trees

In another scenario, distributors sell to other distributors and fees are collected at each level. Every copy sold by any distributor—even several d-blocks down in the chain—results in a fee being paid back to all of the previous distributors.

This scenario is like a chain letter or value chain. Every contributor or distributor along the way obtains fees, and is thereby encouraged to promote the sale of copies of the digital work.

This scenario is performed similarly to the previous ones. The key feature is that the distributor creates a shell which specifies fees to be paid to him. He does not grant Extract rights on the shell. Consequently, all future copies that are made will result in fees paid to him.

Weighted Distribution Trees

In this scenario, distributors make money according to a distribution tree. The fee that they make depends on various parameters, such as time since their sale or the number of subsequent distributors.

This is a generalized version of the Distribution Tree scenario, in that it tries to vary the fee to account for the significance of the role of the distributor.

This scenario is similar to the previous one. The difference is that the fee specification on the distributor's shell has provisions for changes in prices. For example, there could be a fee schedule so that copies made after the passage of time will require lower fees to be paid to the distributor. Alternatively, the distributor could employ a "best-price" billing option, using any algorithm he chooses to determine the fee up to the maximum specified in the shell.

US 6,963,859 B2

45

Fees for Reuse

In this scenario, a first creator creates a work. It is distributed by a first distributor and purchased by a second creator. The second creator extracts a portion of the work and embeds in it a new work distributed by a second distributor. A consumer buys the new work from the second distributor. The first creator receives fees from every transaction; the first distributor receives fees only for his sale; the second creator and second distributor receive fees for the final sale.

This scenario shows how that flexible automatic arrangements can be set up to create automatic charging systems that mirror current practice. This scenario is analogous to when an author pays a fee to reuse a figure in some paper. In the most common case, a fee is paid to the creator or publisher, but not to the bookstore that sold the book.

The mechanisms for derived works are the same as those for distribution.

Limited Reuse

In this scenario, several first creators create works. A second creator makes a selection of these, publishing a collection made up of the parts together with some new interstitial material. (For example, the digital work could be a selection of music or a selection of readings.) The second creator wants to continue to allow some of the selected works to be extractable, but not the interstitial material.

This scenario deals with fine grained control of the rights and fees for reuse.

This scenario is performed as follows:

The first creators create their original works. If they grant extraction and embedding rights, then the second creator can include them in a larger collected work. The second creator creates the interstitial material. He does grant an Extract right on the interstitial material. He grants Extract rights on a subset of the reused material. A consumer of the collection can only extract portions that have that right. Fees are automatically collected for all parts of the collection.

Commercial Libraries

Commercial libraries buy works with the right to loan. They limit the loan period and charge their own fees for use. This scenario deals with fees for loaning rather than fees for making copies. The fees are collected by the same automatic mechanisms.

The mechanisms are the same as previous scenarios except that the fees are associated with the Loan usage right rather than the Copy usage right.

Demo Versions

A creator believes that if people try his work that they will want to buy it or use it. Consumers of his work can copy the work for free, and play (or execute) a limited version of the work for free, and can play or use the full featured version for a fee.

This scenario deals with fees for loaning rather than fees for making copies. The fees are collected by the same automatic mechanisms.

This scenario is performed as follows:

The creator creates a digital work and grants various rights and fees. The creator grants Copy and Embed rights without a fee, in order to ensure widespread distribution of the work. Another of the rights is a limited play right with little or no fee attached. For example, this right may be for playing only a portion of the work. The play right can have various restrictions on its use. It could have a ticket that limits the number of times it is used. It could have internal restrictions that limit its functionality. It could have time restrictions that invalidate the right after a period of time or a period of use. Different fees could be associated with other versions of the Play right.

46

Upgrading a Digital Work with a Vendor

A consumer buys a digital work together with an agreement that he can upgrade to a new version at a later date for a modest fee, much less than the usual purchase price. When the new version becomes available, he goes to a qualified vendor to make the transaction.

This scenario deals with a common situation in computer software. It shows how a purchase may include future "rights." Two important features of the scenario are that the transaction must take place at a qualified vendor, and that the transaction can be done only once per copy of the digital work purchased.

This scenario is performed as follows:

The creator creates a digital work, an upgrade ticket, and a distribution license. The upgrade ticket uses the a generic ticket agent that comes with repositories. As usual, the distribution license does not have Copy or Transfer rights. He distributes a bundled copies of the work and the ticket to his distributors as well as distribution licenses.

The distributor sells the old bundled work and ticket to customers.

The customer extracts the work and the ticket. He uses the work according to the agreements until the new version becomes available.

When the new work is ready, the creator gives it to distributors. The new work has a free right to copy from a distributor if a ticket is available.

The consumer goes to distributors and arranges to copy the work. The transaction offers the ticket. The distributor's repository punches the ticket and copies the new version to the consumers repository.

The consumer can now use the new version of the work.

Distributed Upgrading of Digital Works

A consumer buys a digital work together with an agreement that he can upgrade to a new version at a later date for a modest fee, much less than the usual purchase price. When the new version becomes available, he goes to anyone who has the upgraded version and makes the transaction.

This scenario is like the previous one in that the transaction can only be done once per copy of the digital work purchased, but the transaction can be accomplished without the need to connect to a licensed vendor.

This scenario is similar to the previous one except that the Copy right on the new work does not require a distribution license. The consumer can upgrade from any repository having the new version. He cannot upgrade more than once because the ticket cannot work after it has been punched. If desired, the repository can record the upgrade transaction by posting a zero cost bill to alert the creator that the upgrade has taken place.

Limited Printing

A consumer buys a digital work and wants to make a few ephemeral copies. For example, he may want to print out a paper copy of part of a digital newspaper, or he may want to make a (first generation) analog cassette tape for playing in his car. He buys the digital work together with a ticket required for printing rights.

This scenario is like the common practice of people making cassette tapes to play in their car. If a publisher permits the making of cassette tapes, there is nothing to prevent a consumer from further copying the tapes. However, since the tapes are "analog copies," there is a noticeable quality loss with subsequent generations. The new contribution of the present invention is the use of tickets in the access controls for the making of the analog copies.

This scenario is performed as follows:

The creator sells a work together with limited printing rights. The printing rights specify the kind of printer (e.g., a

US 6,963,859 B2

47

kind of cassette recorder or a kind of desktop paper printer) and also the kind of ticket required. The creator either bundles a limited number of tickets or sells them separately. If the tickets use the generic ticket agent, the consumer with the tickets can exercise the right at his convenience.

Demand Publishing

Professors in a business school want to put together course books of readings selected from scenario studies from various sources. The bookstore wants to be able to print the books from digital masters, without negotiating for and waiting for approval of printing of each of the scenarios. The copyright holders of the scenarios want to be sure that they are paid for every copy of their work that is printed.

On many college campuses, the hassle of obtaining copy clearances in a timely way has greatly reduced the viability of preparing course books. Print shops have become much more cautious about copying works in the absence of documented permission.

Demand Publishing is performed as follows: the creator sells a work together with printing rights for a fee. There can be rights to copy (distribute) the work between bookstore repositories, with or without fee. The printing rights specify the kind of printer. Whenever a bookstore prints one of the works (either standalone or embedded in a collection), the fee is credited to the creator automatically. To discourage unauthorized copying of the print outs, it would be possible for the printer to print tracer messages discretely on the pages identifying the printing transaction, the copy number, and any other identifying information. The tracer information could be secretly embedded in the text itself (encoded in the grey scale) or hidden in some other way.

Metered Use and Multiple Price Packages

A consumer does not know what music to purchase until he decides whether he likes it. He would like to be able to take it home and listen to it, and then decide whether to purchase. Furthermore, he would like the flexibility of paying less if he listens to it very infrequently.

This scenario just uses the capability of the approach to have multiple versions of a right on a digital work. Each version of the right has its own billing scheme. In this scenario, the creator of the work can offer the Copy right without fee, and defer billing to the exercise of the Play right. One version of the play right would allow a limited performance without fee—a right to “demo”. Another version of the right could have a metered rate, of say \$0.25 per hour of play. Another version could have a fee of \$15.00 for the first play, but no fee for further playing. When the consumer exercises a play right, he specifies which version of the right is being selected and is billed accordingly.

Fees for Font Usage

A designer of type fonts invests several months in the design of special fonts. The most common way of obtaining revenue for this work is to sell copies of the fonts to publishers for unlimited use over unlimited periods of time. A font designer would like to charge a rate that reflects the amount that the font is used.

This scenario is performed as follows: the font designer creates a font as a digital work. He creates versions of the Play right that bill either for metered use or “per-use”. Each version of the play right would require that the player (a print layout program) be of an approved category. The font designer assigns appropriate fees to exercise the Copy right. When a publisher client wants to use a font, he includes it as input to a layout program, and is billed automatically for its use. In this way, a publisher who makes little use of a font pays less than one who uses it a lot.

48

Rational Database Usage Charges

Online information retrieval services typically charge for access in a way that most clients find unpredictable and uncorrelated to value or information use. The fee depends on which databases are open, dial-up connect time, how long the searches require, and which articles are printed out. There are no provisions for extracting articles or photographs, no method for paying to reuse information in new works, no distinction between having the terminal sit idly versus actively searching for data, no distinction between reading articles on the screen and doing nothing, and higher rates per search when the centralized facility is busy and slow servicing other clients. Articles can not be offloaded to the client’s machine for off-site search and printing. To offer such billing or the expanded services, the service company would need a secure way to account for and bill for how information is used.

This scenario is performed as follows:

The information service bundles its database as files in a repository. The information services company assigns different fees for different rights on the information files. For example, there could be a fee for copying a search database or a source file and a different fee for printing. These fees would be in addition to fees assigned by the original creator for the services. The fees for using information would be different for using them on the information service company’s computers or the client’s computers. This billing distinction would be controlled by having different versions of the rights, where the version for use on the service company’s computer requires a digital certificate held locally. Fees for copying or printing files would be handled in the usual way, by assigning fees to exercising those rights. The distinction between searching and viewing information would be made by having different “players” for the different functions. This distinction would be maintained on the client’s computers as well as the service computers. Articles could be extracted for reuse under the control of Extract and Embed rights. Thus, if a client extracts part of an article or photograph, and then sells copies of a new digital work incorporating it, fees could automatically be collected both by the information service and earlier creators and distributors of the digital work. In this way, the information retrieval service could both offer a wider selection of services and billing that more accurately reflects the client’s use of the information.

Print Spooling with Rights

In the simplest scenario, when a user wants to print a digital document he issues a print command to the user interface. If the document has the appropriate rights and the conditions are satisfied, the user agrees to the fee and the document is printed. In other cases, the printer may be on a remote repository and it is convenient to spool the printing to a later time. This leads to several issues. The user requesting the printing wants to be sure that he is not billed for the printing until the document is actually printed. Restated, if he is billed at the time the print job is spooled but the job is canceled before printing is done, he does not want to pay. Another issue is that when spooling is permitted, there are now two times at which rights, conditions and fees could be checked: the time at which a print job is spooled and the time at which a print is made. As with all usage rights, it is possible to have rights that expire and to have rights whose fee depends on various conditions. What is needed is a means to check rights and conditions at the time that printing is actually done.

This scenario is performed as follows: A printing repository is a repository with the usual repository characteristics

US 6,963,859 B2

49

plus the hardware and software to enable printing. Suppose that a user logs into a home repository and wants to spool print jobs for a digital work at a remote printing repository. The user interface for this could treat this as a request to “spool” prints. Underneath this “spooling” request, however, are standard rights and requests. To support such requests, the creator of the work provides a Copy right, which can be used to copy the work to a printing repository. In the default case, this Copy right would have no fees associated for making the copy. However, the Next-Set-Of-Rights for the copy would only include the Print rights, with the usual fees for each variation of printing. This version of the Copy right could be called the “print spooling” version of the Copy right. The user’s “spool request” is implemented as a Copy transaction to put a copy of the work on the printing repository, followed by Print transactions to create the prints of the work. In this way, the user is only billed for printing that is actually done. Furthermore, the rights, conditions and fees for printing the work are determined when the work is about to be printed.

Thus, a system for enforcing the usage rights of digital works is disclosed. While the embodiments disclosed herein are preferred, it will be appreciated from this teaching that various alternative, modifications, variations or improvements therein may be made by those skilled in the art, which are intended to be encompassed by the following claims.

Appendix A

Glossary

Authorization Repository:

A special type of repository which provides authorization service. An authorization may be specified by a usage right. The authorization must be obtained before the right may be exercised.

Billing Clearinghouse:

A financial institution or the like whose purpose is to reconcile billing information received from credit servers. The billing clearinghouse may generate bills to users or alternatively, credit and debit accounts involved in the commercial transactions.

Billing Transactions:

The protocol used by which a repository reports billing information to a credit server.

Clearinghouse Transactions:

The protocol used between a credit server and a clearinghouse.

Composite Digital Work:

A digital work comprised of distinguishable parts. Each of the distinguishable parts is itself a digital work which have usage rights attached.

Content:

The digital information (i.e. raw bits) representing a digital work.

Copy Owner:

A term which refers to the party who owns a digital work stored in a repository. In the typical case, this party has purchased various rights to the document for printing, viewing, transferring, or specific uses.

Creator:

A term which refers to a party who produces a digital work.

Credit Server:

A device which collects and reports billing information for a repository. In many implementations, this could be built as part of a repository. It requires a means for periodically communicating with a billing clearinghouse.

50

Description Tree:

A structure which describes the location of content and the usage rights and usage fees for a digital work. A description tree is comprised of description blocks. Each description block corresponds to a digital work or to an interest (typically a revenue bearing interest) in a digital work.

Digital Work (Work):

Any encapsulated digital information. Such digital information may represent music, a magazine or book, or a multimedia composition. Usage rights and fees are attached to the digital work.

Distributor:

A term which refers to a party who legitimately obtains a copy of a digital work and offers it for sale.

Identification (Digital) Certificate:

A signed digital message that attests to the identity of the possessor. Typically, digital certificates are encrypted in the private key of a well-known master repository.

Master Repository:

A special type of repository which issues identification certificates and distributes lists of repositories whose integrity have been compromised and which should be denied access to digital works (referred to as repository “hotlists”).

Public Key Encryption:

An encryption technique used for secure transmission of messages on a communication channel. Key pairs are used for the encryption and decryption of messages. Typically one key is referred to as the public key and the other is the private key. The keys are inverses of each other from the perspective of encryption. Restated, a digital work that is encrypted by one key in the pair can be decrypted only by the other.

Registration Transactions:

The protocol used between repositories to establish a trusted session.

Rendering Repository:

A special type of repository which is typically coupled to a rendering system. The rendering repository will be typically be embodied within the secure boundaries of a rendering system.

Rendering System:

The combination of a rendering repository and a rendering device. Examples of rendering systems include printing systems, displaying systems, general purpose computer systems, video systems or audio systems.

Repository:

Conceptually a set of functional specifications defining core functionality in the support of usage rights. A repository is a trusted system in that it maintains physical, communications and behavioral integrity.

Requester Mode:

A mode of repository where it is requesting access to a digital work.

Revenue Owners:

A term which refers to the parties that maintain an interest in collecting fees for document use or who stand to lose revenue if illegitimate copies of the digital work are made.

Server Mode:

A mode of a repository where it is processing an incoming request to access a digital work.

Shell Description Block:

A special type of description block designating an interest in a digital work, but which does not add content. This will typically be added by a distributor of a digital work to add their fees.

US 6,963,859 B2

51

Transactions:

A term used to refer to the protocols by which repositories communicate.

Usage Fees:

A fee charged to a requester for access to a digital work. Usage fees are specified within the usage rights language.

Usage Rights:

A language for defining the manner in which a digital work may be used or distributed, as well as any conditions on which use or distribution is premised.

Usage Transactions:

A set of protocols by which repositories communicate in the exercise of a usage rights. Each usage right has it's own transaction steps.

What is claimed is:

1. A rendering system adapted for use in a distributed system for managing use of content, said rendering system being operative to rendering content in accordance with usage rights associated with the content, said rendering system comprising:

a rendering device configured to render the content; and a distributed repository coupled to said rendering device and including a requester mode of operation and server mode of operation,

wherein the server mode of operation is operative to enforce usage rights associated with the content and permit the rendering device to render the content in accordance with a manner of use specified by the usage rights,

the requester mode of operation is operative to request access to content from another distributed repository, and

said distributed repository is operative to receive a request to render the content and permit the content to be rendered only if a manner of use specified in the request corresponds to a manner of use specified in the usage rights.

2. A rendering system as recited in claim 1, wherein said rendering device is configured to render content into a desired form.

3. A rendering system as recited in claim 1, wherein said repository comprises means for storing the content.

4. A rendering system as recited in claim 3 wherein said means for storing is means for storing ephemeral copies of the content.

5. A rendering system as recited in claim 3 wherein said means for storing comprises means for storing content after rendering.

6. A rendering system as recited in claim 5 wherein the content comprises fonts.

7. A rendering system as recited in claim 5 wherein the content comprises music.

8. A rendering system as recited in claim 5 wherein the content comprises video.

9. A rendering system as recited in claim 3, wherein said repository comprises removable media.

10. A rendering system as recited in claim 1, further comprising means for storing the content.

11. A rendering system as recited in claim 10, wherein said means for storing comprises removable media.

12. A rendering system as recited in claim 1 wherein said rendering device comprises a printer.

13. A rendering system as recited in claim 1, wherein said rendering device comprises a video system.

14. A rendering system as recited in claim 1, wherein said rendering device comprises an audio system.

52

15. A rendering system as recited in claim 1 wherein said rendering device comprises a computer system and said repository comprises software executed on the computer system.

16. A rendering system as recited in claim 1, further comprising an execution device coupled to said repository, said repository being further operative to permit said execution device to execute a computer program only in a manner specified by the usage rights.

17. A rendering system as recited in claim 1, wherein the content is a computer program and the manner of use is a manner of executing the computer program.

18. A rendering system as recited in claim 1, wherein the manner of use is a manner of printing.

19. A rendering system as recited in claim 1, wherein the manner of use is a manner of displaying.

20. A rendering system as recited in claim 1, wherein the manner of use is a manner of playing.

21. A rendering system as recited in claim 1, wherein the rendering device and the repository are integrated into a secure system having a secure boundary.

22. A rendering system as recited in claim 1, wherein the rendering device and the repository are separate devices.

23. A rendering system as recited in claim 1, wherein the usage rights include at least one condition that must be satisfied to exercise the manner of use, and wherein the system further comprises means for communicating with an authorization repository for authorizing a condition.

24. A rendering system as recited in claim 1, further comprising means for communicating with a master repository for obtaining an identification certificate for the repository.

25. A rendering system as recited in claim 1, further comprising a boundary containing said repository and said rendering device in a secure environment.

26. A rendering system as recited in claim 23, wherein the condition is possession of a digital ticket.

27. A rendering as recited in claim 1, wherein the content has plural components having usage rights associated therewith and wherein said repository enforces the usage rights for each component.

28. A rendering system as recited in claim 1, wherein said system is implemented using one or more hardware and/or software devices.

29. A rendering method adapted for use in a distributed system for managing use of content, and operative to render content in accordance with usage rights associated with the content, said method comprising:

configuring a rendering device to render the content;

configuring a distributed repository coupled to said rendering device to include a requester mode of operation and server mode of operation;

enforcing usage rights associated with the content and permitting the rendering device to render the content in accordance with a manner of use specified by the usage rights, when in the server mode of operation;

requesting access to content from another distributed repository, when in the requester mode of operation; and

receiving by said distributed repository a request to render the content and permitting the content to be rendered only if a manner of use specified in the request corresponds to a manner of use specified in the usage rights.

30. A rendering method as recited in claim 29, wherein said rendering device is configured to render content into a desired form.

US 6,963,859 B2

53

31. A rendering method as recited in claim 29, wherein said repository comprises means for storing the content.

32. A rendering method as recited in claim 31, further comprising storing ephemeral copies of the content.

33. A rendering method as recited in claim 31, wherein said means for storing comprises means for storing content after rendering.

34. A rendering method as recited in claim 33, wherein the content comprises fonts.

35. A rendering method as recited in claim 33, wherein the content comprises music.

36. A rendering method as recited in claim 33, wherein the content comprises video.

37. A rendering method as recited in claim 29, further comprising storing the content.

38. A rendering method as recited in claim 37, wherein said means for storing comprises removable media.

39. A rendering method as recited in claim 29, wherein said rendering device comprises a printer.

40. A rendering method as recited in claim 29, wherein said rendering device comprises a video system.

41. A rendering method as recited in claim 29, wherein said rendering device comprises an audio system.

42. A rendering method as recited in claim 29, wherein said rendering device comprises a computer system and said repository comprises software executed on the computer system.

43. A rendering method as recited in claim 29, further comprising:

coupling an execution device to said repository; and permitting by said repository said execution device to execute a computer program only in a manner specified by the usage rights.

44. A rendering method as recited in claim 29, wherein the content is a computer program and the manner of use is a manner of executing the computer program.

45. A rendering method as recited in claim 29, wherein the manner of use is a manner of printing.

46. A rendering method as recited in claim 29, wherein the manner of use is a manner of displaying.

47. A rendering method as recited in claim 29, wherein the manner of use is a manner of playing.

48. A rendering method as recited in claim 29, wherein the rendering device and the repository are integrated into a secure system having a secure boundary.

49. A rendering method as recited in claim 29, wherein the rendering device and the repository are separate devices.

50. A rendering method as recited in claim 29, wherein the usage rights include at least one condition that must be satisfied to exercise the manner of use, and the method further comprises communicating with an authorization repository for authorizing a condition.

51. A rendering method as recited in claim 29, further comprising communicating with a master repository for obtaining an identification certificate for the repository.

52. A rendering method as recited in claim 29, further comprising configuring a boundary containing said repository and said rendering device in a secure environment.

53. A rendering method as recited in claim 50, wherein the condition is possession of a digital ticket.

54. A rendering method as recited in claim 29, wherein the content has plural components having usage rights associated therewith and the method further comprises enforcing by said repository the usage rights for each component.

55. A rendering method as recited in claim 31, wherein said repository comprises removable media.

56. A rendering method as recited in claim 29, wherein said method is implemented using one or more hardware and/or software devices.

54

57. A rendering method as recited in claim 29, wherein said method is implemented using a computer readable medium including one or more computer readable instructions embedded therein and configured to cause one or more computer processors to perform said method.

58. A computer readable medium including one or more computer readable instructions embedded therein for use in a distributed system for managing use of content, and operative to render content in accordance with usage rights associated with the content, said computer readable instructions configured to cause one or more computer processors to perform the steps of:

configuring a rendering device to render the content;

configuring a distributed repository coupled to said rendering device to include a requester mode of operation and server mode of operation;

enforcing usage rights associated with the content and permitting the rendering device to render the content in accordance with a manner of use specified by the usage rights, when in the server mode of operation;

requesting access to content from another distributed repository, when in the requester mode of operation; and

receiving by said distributed repository a request to render the content and permitting the content to be rendered only if a manner of use specified in the request corresponds to a manner of use specified in the usage rights.

59. A computer readable medium as recited in claim 58, wherein said rendering device is configured to render content into a desired form.

60. A computer readable medium as recited in claim 58, wherein said repository comprises means for storing the content.

61. A computer readable medium as recited in claim 60, wherein said computer readable instructions are configured to cause the one or more computer processors to perform the step of storing ephemeral copies of the content.

62. A computer readable medium as recited in claim 60, wherein said means for storing comprises means for storing content after rendering.

63. A computer readable medium as recited in claim 62, wherein the content comprises fonts.

64. A computer readable medium as recited in claim 62, wherein the content comprises music.

65. A computer readable medium as recited in claim 62, wherein the content comprises video.

66. A computer readable medium as recited in claim 60, wherein said repository comprises removable media.

67. A computer readable medium as recited in claim 58, wherein said computer readable instructions are configured to cause the one or more computer processors to perform the step of storing the content.

68. A computer readable medium as recited in claim 58, wherein said rendering device comprises a printer.

69. A computer readable medium as recited in claim 58, wherein said rendering device comprises a video system.

70. A computer readable medium as recited in claim 58, wherein said rendering device comprises an audio system.

71. A computer readable medium as recited in claim 58, wherein said rendering device comprises a computer system and said repository comprises software executed on the computer system.

72. A computer readable medium as recited in claim 58, wherein said computer readable instructions are configured to cause the one or more computer processors to perform the steps of:

US 6,963,859 B2

55

coupling an execution device coupled to said repository;
 and
 permitting by said repository said execution device to
 execute a computer program only in a manner specified
 by the usage rights.

73. A computer readable medium as recited in claim 58,
 wherein the content is a computer program and the manner
 of use is a manner of executing the computer program.

74. A computer readable medium as recited in claim 58,
 wherein the manner of use is a manner of printing.

75. A computer readable medium as recited in claim 58,
 wherein the manner of use is a manner of displaying.

76. A computer readable medium as recited in claim 58,
 wherein the manner of use is a manner of playing.

77. A computer readable medium as recited in claim 58,
 wherein the rendering device and the repository are inte-
 grated into a secure system having a secure boundary.

78. A computer readable medium as recited in claim 58,
 wherein the rendering device and the repository are separate
 devices.

79. A computer readable medium as recited in claim 58,
 wherein the usage rights include at least one condition that
 must be satisfied to exercise the manner of use, and said
 computer readable instructions are configured to cause the
 one or more computer processors to perform the step of

56

communicating with an authorization repository for autho-
 rizing a condition.

80. A computer readable medium as recited in claim 79,
 wherein the condition is possession of a digital ticket.

81. A computer readable medium as recited in claim 58,
 wherein said computer readable instructions are configured
 to cause the one or more computer processors to perform the
 step of communicating with a master repository for obtain-
 ing an identification certificate for the repository.

82. A computer readable medium as recited in claim 58,
 wherein said computer readable instructions are configured
 to cause the one or more computer processors to perform the
 step of configuring a boundary containing said repository
 and said rendering device in a secure environment.

83. A computer readable medium as recited in claim 58,
 wherein the content has plural components having usage
 rights associated therewith and said computer readable
 instructions are configured to cause the one or more com-
 puter processors to perform the step of enforcing by said
 repository the usage rights for each component.

84. A computer readable medium as recited in claim 67,
 wherein said means for storing comprises removable media.

* * * * *



US007523072B2

(12) **United States Patent**
Stefik et al.

(10) **Patent No.:** **US 7,523,072 B2**

(45) **Date of Patent:** ***Apr. 21, 2009**

(54) **SYSTEM FOR CONTROLLING THE DISTRIBUTION AND USE OF DIGITAL WORKS**

(75) Inventors: **Mark J. Stefik**, Portola Valley, CA (US);
Peter L. T. Pirolli, San Francisco, CA (US)

(73) Assignee: **Contentguard Holdings, Inc.**,
 Wilmington, DE (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **11/304,794**

(22) Filed: **Dec. 16, 2005**

(65) **Prior Publication Data**
 US 2006/0149680 A1 Jul. 6, 2006

Related U.S. Application Data

(60) Continuation of application No. 11/198,216, filed on Aug. 8, 2005, which is a continuation of application No. 10/176,608, filed on Jun. 24, 2002, now Pat. No. 6,934,693, which is a continuation of application No. 09/777,845, filed on Feb. 7, 2001, which is a division of application No. 08/967,084, filed on Nov. 10, 1997, now Pat. No. 6,236,971, which is a continuation of application No. 08/344,760, filed on Nov. 23, 1994, now abandoned.

(51) **Int. Cl.**
H04K 1/00 (2006.01)
H04L 9/00 (2006.01)

(52) **U.S. Cl.** **705/59; 705/1; 705/50; 705/51; 726/26; 726/27**

(58) **Field of Classification Search** **705/51, 705/1, 50, 59; 726/26, 27**

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

3,263,158 A 7/1966 Bargaen et al.

(Continued)

FOREIGN PATENT DOCUMENTS

BR 9810967 A 10/2001

(Continued)

OTHER PUBLICATIONS

Henry H. Perritt, Jr. Knowbots, Permissions Headers and Contract Law, Apr. 30, 1993, Villanova Law School, pp. 1-24, <http://www.ifla.org/documents/infopol/copyright/perh2.txt>.*

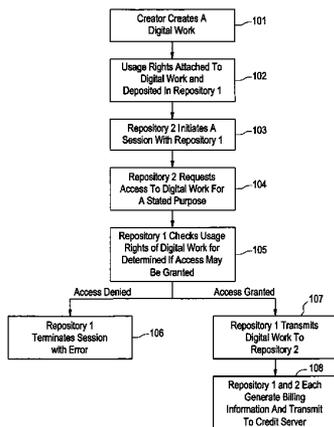
(Continued)

Primary Examiner—Andrew J. Fischer
Assistant Examiner—Joshua Murdough
 (74) *Attorney, Agent, or Firm*—Marc S. Kaufman; Stephen M. Hertzler; Nixon Peabody, LLP

(57) **ABSTRACT**

A method, system and software for securely rendering digital documents, including storing a digital document in a document platform; and storing a usage right associated with the digital document. The usage right specifies a manner of use indicating the manner in which the digital document can be rendered by the document platform. The digital document comprises plural parts of digital content. The usage right includes plural usage rights respectively associated with each of the plural parts of digital content. Whether one of the parts of the digital document may be rendered by the document platform is determined based a respective usage right. If the respective usage right allows the digital document to be rendered on the document platform, the corresponding part of the digital document is rendered by the document platform.

25 Claims, 13 Drawing Sheets



US 7,523,072 B2

Page 2

U.S. PATENT DOCUMENTS						
			5,299,263	A	3/1994	Beller et al.
3,609,697	A	9/1971	5,301,231	A	4/1994	Abraham et al.
3,790,700	A	2/1974	5,311,591	A	5/1994	Fischer
3,798,605	A	3/1974	5,319,705	A	6/1994	Halter et al.
4,159,468	A	6/1979	5,335,275	A	8/1994	Millar et al.
4,200,700	A	4/1980	5,337,357	A	8/1994	Chou et al.
4,220,991	A	9/1980	5,339,091	A	8/1994	Yamazaki et al.
4,278,837	A	7/1981	5,341,429	A	8/1994	Stringer et al.
4,323,921	A	4/1982	5,347,579	A	9/1994	Blandford
4,361,851	A	11/1982	5,381,526	A	1/1995	Ellson
4,423,287	A	12/1983	5,386,369	A	1/1995	Christiano
4,429,385	A	1/1984	5,390,297	A	2/1995	Barber et al.
4,442,486	A	4/1984	5,394,469	A	2/1995	Nagel et al.
4,529,870	A	7/1985	5,410,598	A	4/1995	Shear
4,558,176	A	12/1985	5,412,717	A	5/1995	Fischer
4,593,376	A	6/1986	5,414,852	A	5/1995	Kramer et al.
4,614,861	A	9/1986	5,428,606	A	6/1995	Moskowitz
4,621,321	A	11/1986	5,432,849	A	7/1995	Johnson et al.
4,644,493	A	2/1987	5,438,508	A	8/1995	Wyman
4,658,093	A	4/1987	5,444,779	A	8/1995	Daniele
4,713,753	A	12/1987	5,453,601	A	9/1995	Rosen
4,736,422	A	4/1988	5,455,953	A	10/1995	Russell
4,740,890	A	4/1988	5,457,746	A	10/1995	Dolphin
4,796,220	A	1/1989	5,473,687	A	12/1995	Lipscomb et al.
4,816,655	A	3/1989	5,473,692	A	12/1995	Davis
4,817,140	A	3/1989	5,485,577	A	1/1996	Eyer et al.
4,827,508	A	5/1989	5,499,298	A	3/1996	Narasimhalu et al.
4,868,376	A	9/1989	5,502,766	A	3/1996	Boebert et al.
4,882,752	A	11/1989	5,504,814	A	4/1996	Miyahara
4,888,638	A	12/1989	5,504,816	A	4/1996	Hamilton et al.
4,891,838	A	1/1990	5,504,818	A	4/1996	Okano
4,924,378	A	5/1990	5,504,837	A	4/1996	Griffith et al.
4,932,054	A	6/1990	5,509,070	A	4/1996	Schull
4,937,863	A	6/1990	5,530,235	A	6/1996	Stefik et al.
4,949,187	A	8/1990	5,532,920	A	7/1996	Hartrick et al.
4,953,209	A	8/1990	5,534,975	A	7/1996	Stefik et al.
4,961,142	A	10/1990	5,535,276	A	7/1996	Ganesan
4,975,647	A	12/1990	5,539,735	A	7/1996	Moskowitz
4,977,594	A	12/1990	5,553,143	A	9/1996	Ross et al.
4,999,806	A	3/1991	5,557,678	A	9/1996	Ganesan
5,010,571	A	4/1991	5,563,946	A	10/1996	Cooper et al.
5,014,234	A	5/1991	5,564,038	A	10/1996	Grantz et al.
5,023,907	A	6/1991	5,568,552	A	10/1996	Davis
5,047,928	A	9/1991	5,619,570	A	4/1997	Tsutsui
5,050,213	A	9/1991	5,621,797	A	4/1997	Rosen
5,052,040	A	9/1991	5,625,690	A	4/1997	Michel et al.
5,058,164	A	10/1991	5,629,980	A	5/1997	Stefik et al.
5,103,476	A	4/1992	5,633,932	A	5/1997	Davis et al.
5,113,519	A	5/1992	5,634,012	A	5/1997	Stefik et al.
5,129,083	A	7/1992	5,636,346	A	6/1997	Saxe
5,136,643	A	8/1992	5,638,443	A	6/1997	Stefik et al.
5,138,712	A	8/1992	5,638,494	A	6/1997	Pinard et al.
5,146,499	A	9/1992	5,638,513	A	6/1997	Ananda
5,148,481	A	9/1992	5,649,013	A	7/1997	Stuckey et al.
5,159,182	A	10/1992	5,655,077	A	8/1997	Jones et al.
5,174,641	A	12/1992	5,708,709	A	1/1998	Rose
5,183,404	A	2/1993	5,708,717	A	1/1998	Alasia
5,191,193	A	3/1993	5,715,403	A	2/1998	Stefik
5,204,897	A	4/1993	5,724,425	A	3/1998	Chang et al.
5,222,134	A	6/1993	5,734,823	A	3/1998	Saigh et al.
5,224,163	A	6/1993	5,734,891	A	3/1998	Saigh
5,235,642	A	8/1993	5,737,413	A	4/1998	Akiyama et al.
5,247,575	A	9/1993	5,737,416	A	4/1998	Cooper et al.
5,255,106	A	10/1993	5,745,569	A	4/1998	Moskowitz et al.
5,260,999	A	11/1993	5,745,879	A	4/1998	Wyman
5,263,157	A	11/1993	5,748,783	A	5/1998	Rhoads
5,263,158	A	11/1993	5,757,907	A	5/1998	Cooper et al.
5,276,444	A	1/1994	5,761,686	A	6/1998	Bloomberg
5,276,735	A	1/1994	5,764,807	A	6/1998	Pearlman et al.
5,287,408	A	2/1994	5,765,152	A	6/1998	Erickson
5,291,596	A	3/1994	5,768,426	A	6/1998	Rhoads
5,293,422	A	3/1994	5,787,172	A	7/1998	Arnold
			5,790,677	A	8/1998	Fox et al.

US 7,523,072 B2

Page 3

5,812,664 A	9/1998	Bernobich et al.	2002/0001387 A1	1/2002	Dillon
5,825,876 A	10/1998	Peterson	2002/0035618 A1	3/2002	Mendez et al.
5,825,879 A	10/1998	Davis	2002/0044658 A1	4/2002	Wasilewski et al.
5,825,892 A	10/1998	Braudaway et al.	2002/0056118 A1	5/2002	Hunter et al.
5,838,792 A	11/1998	Ganesan	2002/0069282 A1	6/2002	Reisman
5,848,154 A	12/1998	Nishio et al.	2002/0099948 A1	7/2002	Kocher et al.
5,848,378 A	12/1998	Shelton et al.	2002/0127423 A1	9/2002	Kayanakis
5,850,443 A	12/1998	Van Oorschot et al.	2003/0097567 A1	5/2003	Terao et al.
5,892,900 A	4/1999	Ginter et al.	2004/0052370 A1	3/2004	Katznelson
5,910,987 A	6/1999	Ginter et al.	2004/0064692 A1	4/2004	Kahn et al.
5,915,019 A	6/1999	Ginter et al.	2004/0172552 A1	9/2004	Boyles et al.
5,917,912 A	6/1999	Ginter et al.			
5,920,861 A	7/1999	Hall et al.			
5,933,498 A	8/1999	Schneck et al.			
5,940,504 A	8/1999	Griswold	EP	0 067 556 B1	12/1982
5,943,422 A	8/1999	Van Wie et al.	EP	0 084 441	7/1983
5,949,876 A	9/1999	Ginter et al.	EP	0 180 460	5/1986
5,982,891 A	11/1999	Ginter et al.	EP	0 257 585 A2	3/1988
5,987,134 A	11/1999	Shin et al.	EP	0 262 025 A2	3/1988
5,999,624 A	12/1999	Hopkins	EP	0 332 304 A2	9/1989
5,999,949 A	12/1999	Crandall	EP	0 332 707	9/1989
6,006,332 A	12/1999	Rabne et al.	EP	0 393 806 A2	10/1990
6,020,882 A	2/2000	Kinghorn et al.	EP	0 450 841 A2	10/1991
6,047,067 A	4/2000	Rosen	EP	0 529 261 A2	3/1993
6,073,234 A	6/2000	Kigo et al.	EP	0 538 216 A1	4/1993
6,091,777 A	7/2000	Guetz et al.	EP	0 613 073 A1	8/1994
6,112,181 A	8/2000	Shear et al.	EP	0 651 554	5/1995
6,112,239 A	8/2000	Kenner et al.	EP	0 668 695	8/1995
6,115,471 A	9/2000	Oki et al.	EP	0 678 836 A1	10/1995
6,135,646 A *	10/2000	Kahn et al. 709/217	EP	0 679 977 A1	11/1995
6,138,119 A	10/2000	Hall et al.	EP	0 715 243 A1	6/1996
6,141,754 A	10/2000	Choy	EP	0 715 244 A1	6/1996
6,157,719 A	12/2000	Wasilewski et al.	EP	0 715 245 A1	6/1996
6,157,721 A	12/2000	Shear et al.	EP	0 725 376	8/1996
6,169,976 B1	1/2001	Colosso	EP	0 731 404 A1	9/1996
6,185,683 B1	2/2001	Ginter et al.	EP	0 763 936 A2	3/1997
6,189,037 B1	2/2001	Adams et al.	EP	0 818 748 A2	1/1998
6,189,146 B1	2/2001	Misra et al.	EP	0 840 194 A2	5/1998
6,209,092 B1	3/2001	Linnartz	EP	0 892 521 A2	1/1999
6,216,112 B1	4/2001	Fuller et al.	EP	0 934 765 A1	8/1999
6,219,652 B1	4/2001	Carter et al.	EP	0 946 022 A2	9/1999
6,226,618 B1	5/2001	Downs et al.	EP	0 964 572 A1	12/1999
6,233,684 B1	5/2001	Stefik et al.	EP	1 103 922 A2	5/2001
6,236,971 B1	5/2001	Stefik et al.	GB	1483282	8/1977
6,237,786 B1	5/2001	Ginter et al.	GB	2022969 A	12/1979
6,240,185 B1	5/2001	Van Wie et al.	GB	2 136 175	9/1984
6,253,193 B1	6/2001	Ginter et al.	GB	2 236 604	4/1991
6,266,618 B1	7/2001	Ye et al.	GB	2236604 A	4/1991
6,292,569 B1	9/2001	Shear et al.	GB	2309364 A	7/1997
6,301,660 B1	10/2001	Benson	GB	2316503 A	2/1998
6,307,939 B1	10/2001	Vigarie	GB	2354102 A	3/2001
6,327,652 B1	12/2001	England et al.	JP	62-241061	10/1987
6,330,670 B1	12/2001	England et al.	JP	64-068835	3/1989
6,345,256 B1	2/2002	Milsted et al.	JP	03-014109	1/1991
6,353,888 B1	3/2002	Kakehi et al.	JP	3063717 A	3/1991
6,363,488 B1	3/2002	Ginter et al.	JP	04-369068	12/1992
6,389,402 B1	5/2002	Ginter et al.	JP	5100939	4/1993
6,397,333 B1	5/2002	Söhne et al.	JP	5168039 A2	7/1993
6,401,211 B1	6/2002	Brezak, Jr. et al.	JP	5-268415	10/1993
6,405,369 B1	6/2002	Tsuria	JP	05-298174	11/1993
6,424,717 B1	7/2002	Pinder et al.	JP	06-103286	4/1994
6,424,947 B1	7/2002	Tsuria et al.	JP	6131371 A	5/1994
6,487,659 B1	11/2002	Kigo et al.	JP	6-175794	6/1994
6,516,052 B2	2/2003	Voudouris	JP	06-214862	8/1994
6,516,413 B1	2/2003	Aratani et al.	JP	6-215010	8/1994
6,523,745 B1	2/2003	Tamori	JP	06-230847	8/1994
6,796,555 B1	9/2004	Blahut	JP	06-318167	11/1994
7,130,087 B2 *	10/2006	Rhoads 358/3.28	JP	07-084852	3/1995
2001/0009026 A1	7/2001	Terao et al.	JP	07-200317	8/1995
2001/0011276 A1	8/2001	Durst Jr. et al.	JP	07-244639	9/1995
2001/0014206 A1	8/2001	Artigas et al.	JP	0 715 241	6/1996
2001/0037467 A1	11/2001	O'Toole, Jr. et al.	JP	11031130 A2	2/1999
2001/0039659 A1	11/2001	Simmons et al.	JP		

FOREIGN PATENT DOCUMENTS

US 7,523,072 B2

Page 4

JP	11032037	A2	2/1999
JP	11205306	A2	7/1999
JP	11215121	A2	8/1999
JP	2000215165	A2	8/2000
JP	2005218143	A2	8/2005
JP	2005253109	A2	9/2005
JP	20060180562	A2	7/2006
WO	WO 83/04461	A1	12/1983
WO	WO 92/20022		11/1992
WO	WO 92/20022	A1	11/1992
WO	WO 93/01550		1/1993
WO	WO 93/01550	A1	1/1993
WO	WO 93/11480	A1	6/1993
WO	WO 94/01821		1/1994
WO	WO 94/03003	A1	2/1994
WO	WO 96/13814	A1	5/1996
WO	WO 96/24092		8/1996
WO	WO 96/24092	A2	8/1996
WO	WO 96/27155	A2	9/1996
WO	WO 97/25800	A1	7/1997
WO	WO 97/37492	A1	10/1997
WO	WO 97/41661	A2	11/1997
WO	WO 97/43761	A2	11/1997
WO	WO 97/48203		12/1997
WO	WO 98/09209	A1	3/1998
WO	WO 98/10561	A1	3/1998
WO	WO 98/11690		3/1998
WO	WO 98/11690	A1	3/1998
WO	WO 98/19431	A1	5/1998
WO	WO 98/42098		9/1998
WO	WO 98/43426	A1	10/1998
WO	WO 98/45768	A1	10/1998
WO	WO 99/24928	A2	5/1999
WO	WO 99/34553	A1	7/1999
WO	WO 99/35782	A1	7/1999
WO	WO 99/48296	A1	9/1999
WO	WO 99/49615		9/1999
WO	WO 99/60461	A1	11/1999
WO	WO 99/60750	A2	11/1999
WO	WO 00/04727	A2	1/2000
WO	WO 00/05898	A2	2/2000
WO	WO 00/46994	A1	8/2000
WO	WO 00/59152	A2	10/2000
WO	WO 00/62260	A1	10/2000
WO	WO 00/72118	A1	11/2000
WO	WO 00/73922	A2	12/2000
WO	WO 01/03044	A1	1/2001
WO	WO 01/37209	A1	5/2001
WO	WO 01/63528		8/2001
WO	WO 2004/034223	A2	4/2004
WO	WO 2004/103843		12/2004

OTHER PUBLICATIONS

Henry M. Levy, "Capability-Based Computer Systems", Digital Press, 1984, P1-18 USA.

Kawahara Masaharu, "Consideration of a Method of Charging for Electronic Objects in Superdistribution", Technical Report of the Institute of Electronics, Information and Communication Engineers, EIC Institute of Electronics, Information and Communication Engineers, Sep. 21, 1994, Shingaku Giho vol. 94 No. 240, p. 17-24 Japan—English Translation.

Ryoichi Mori et al., "The Inevitable Way Toward Superdistribution", Information Symposium Papers, Information Processing Society of Japan, Feb. 17, 1994, vol. 94, No. 1, p. 67-76 Japan.

Shinichi Ueki et al., "The Accounting Process in Software Usage Monitor for Superdistribution", Information Processing Society of Japan Technical Report, 90-Is-27, Jan. 16, 1990, vol. 90, No. 1, p. 1-10 Japan English Translation.

Naoya Torii et al., "System Architecture of Superdistribu-

tion", Technical Report of the Institute of Electronics, Information and Communication Engineers, EIC Institute of Electronics, Information and Communication Engineers, Sep. 21, 1994, Shingaku Giho vol. 94 No. 240, p. 59-66 Japan.

Yasushi Kuho et al., "Information Technology System Security for Open Distributed Computing Environment", The Hitachi Hyoron, Nov. 1994, vol. 76, p. 49-52 Japan.

Hiroyoshi Takeda, "Function and Use of Workstation" The Office Management, Nikkan Kogyo Shinbun, Aug. 1, 1989, vol. 28, No. 8, p. 8-17 Japan.

Atsushi Iizawa, "Document Image Database System", Information Processing Society, May 15, 1992, vol. 33, No. 5, p. 497-504 Japan.

"National Semiconductor and EPR Partner for Information Metering/Data Security Cards" Mar. 4, 1994, Press Release from Electronic Publishing Resources, Inc.

Weber, R., "Digital Rights Management Technology" Oct. 1995.

Flasche, U. et al., "Decentralized Processing of Documents", pp. 119-131, 1986, Comput. & Graphics, vol. 10, No. 2.

Mori, R. et al., "Superdistribution: The Concept and the Architecture", pp. 1133-1146, 1990. The Transactions of the IEICE, Vo. E 73, No. 7, Tokyo, JP.

Weber, R., "Metering Technologies for Digital Intellectual Property", pp. 1-29, Oct. 1994, A Report to the International Federation of Reproduction Rights Organizations.

Clark, P.C. et al., "Bits: A Smartcard protected Operating System", pp. 66-70 and 94, Nov. 1994, Communications of the ACM, vol. 37, No. 11.

Ross, P.E., "Data Guard", pp. 101, Jun. 6, 1994, Forbes.

Saigh, W.K., "Knowledge is Sacred", 1992, Video Pocket/Page Reader Systems, Ltd.

Kahn, R.E., "Deposit, Registration and Recordation in an Electronic Copyright Management System", pp. 1-19, Aug. 1992, Corporation for National Research Initiatives, Virginia.

Hilts, P. et al., "Books While U Wait", pp. 48-50, Jan. 3, 1994, Publishers Weekly.

Strattner, A, "Cash Register on a Chip may Revolutionize Software Pricing and Distribution; Wave Systems Corp.", pp. 1-3, Apr. 1994, Computer Shopper, vol. 14, No. 4, ISSN 0886-0556.

O'Conner, M., "New Distribution Option for Electronic Publishers; iOpener Data Encryption and Metering System for CD-ROM use; Column", pp. 1-6, Mar. 1994, CD-ROM Professional, vol. 7, No. 2, ISSN: 1409-0833.

Willett, S., "Metered PCs: Is Your System Watching You? Wave System beta tests new technology", pp. 84, May 2, 1994, InfoWorld.

Linn, R., "Copyright and Information Services in the Context of the National Research and Education Network", pp. 9-20, Jan. 1994, IMA Intellectual Property Project Proceedings, vol. 1, Issue 1.

Perrit, Jr., H., "Permission Headers and Contract Law", pp. 27-48, Jan. 1994, IMA Intellectual Property Project Proceedings, vol. 1, Issue 1.

Uptegrove, L., "Intellectual Property Header Descriptors: A Dynamic Approach", pp. 63-66, Jan. 1994, IMA Intellectual Property Proceedings, vol. 1, Issue 1.

Sirbu, M., "Internet Billing Service Design and prototype Implementation", pp. 67-80, Jan. 1994, IMA Intellectual Property Project Proceedings, vol. 1, Issue 1.

Simmell, S. et al., "Metering and Licensing of Resources: Kala's General Purpose Approach", pp. 81-110, Jan. 1994, IMA Intellectual Property Project Proceedings, vol. 1, Issue 1.

US 7,523,072 B2

Page 5

- Kahn, R., "Deposit, Registration and Recordation in an Electronic Copyright Management System", pp. 111-120, Jan. 1994, IMA Intellectual Property Project Proceedings, vol. 1, Issue 1.
- Tygar, J. et al., "Dyad: A System for Using Physically Secure Coprocessors", pp. 121-152, Jan. 1994, IMA Intellectual Property Project Proceedings, vol. 1, Issue 1.
- Griswold, G., "A Method for Protecting Copyright on Networks", pp. 169-178, Jan. 1994, IMA Intellectual Property Project Proceedings, vol. 1, Issue 1.
- Nelson, T., "A Publishing and Royalty Model for Networked Documents", pp. 257-259, Jan. 1994, IMA Intellectual Property Project Proceedings, vol. 1, Issue 1.
- Robinson, E., "Redefining Mobile Computing", pp. 238-240, 247-248 and 252, Jul. 1993, PC Computing.
- Abadi, M. et al., "Authentication and Delegation with Smart-cards", pp. 1-24, 1990, Research Report DEC Systems Research Center.
- Mark Stefik, "Letting Loose the Light: Igniting Commerce in Electronic Publication", pp. 219-253, 1996, Internet Dreams: Archetypes, Myths, and Metaphors, IDSN 0-262-19373-6.
- Mark Stefik, "Letting Loose the Light: Igniting Commerce in Electronic Publication", pp. 2-35, Feb. 8, 1995, Internet Dreams: Archetypes, Myths and Metaphors.
- Henry H. Perritt, Jr., "Technological Strategies for Protecting Intellectual Property in the Networked Multimedia Environment", Apr. 2-3, 1993, Knowbots, Permissions Headers & Contract Law.
- Perritt, Henry H.; Knowbots, Permissions Headers and Contract Law paper for conference on *Technological Strategies for Protecting Intellectual Property in the Networked Multimedia Environment*, Apr. 2-3 1993, pp. 1-22.
- European Search Report dated Sep. 8, 2004 in corresponding European Application No. EP 03 015 128.6.
- European Search Report dated Sep. 13, 2004 in corresponding European Application No. EP 03 015 127.8.
- European Search Report for Corresponding European Application 95308422.5.
- Blaze et al., "Divertible Protocols and Atomic Proxy Cryptography" 1998 Advances in Cryptography—Euro Crypt International Conference on the Theory and Application of Crypto Techniques, Springer Verlag, DE.
- Blaze et al., "Atomic Proxy Cryptography" Draft (Online) (Nov. 2, 1997) XP002239619 Retrieved from the Internet.
- Cox, "Superdistribution" Wired Magazine (Sep. 1994), XP002233405 URL:http://www.wired.com/wired/archive/2.09/superdis_pr.html>.
- Dunlop et al, Telecommunications Engineering, pp. 346-352 (1984).
- Elgamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," IEEE Transactions on Information Theory IT-31(4):469-472 (Jul. 1985).
- Iannella, ed., Open Digital Rights Language (ODRL), pp. 1-31 (Nov. 21, 2000).
- Kahle, wais.concepts.txt, Wide Area Information Server Concepts, Thinking Machines Version 4, Draft, pp. 1-18 (Nov. 3, 1989).
- Kahn, "Deposit, Registration and Recordation in an Electronic Copyright Management System" Technical Report, Corporation for National Research Initiatives, Reston, Virginia (Aug. 1992) URL:<http://www.cni.org/docs/ima.ip-workshop/kahn.html>.
- Kahn et al, "The Digital Library Project, vol. 1: The World of Knowbots (Draft), An Open Architecture for a Digital Library System and a Plan for its Development," Corporation for National Research Initiatives, pp. 1-48 (Mar. 1988).
- Kohl et al, Network Working Group Request for Comments: 1510, pp. 1-112 (Sep. 1993).
- Lee et al, CDMA Systems Engineering Handbook (1998) [excerpts but not all pages numbered].
- Mambo et al, "Protection of Data and Delegated Keys in Digital Distribution," Information Security and Privacy. Second Australian Conference, ACISP '97 Proceedings, pp. 271-282 (Sydney, NSW, Australia, Jul. 7-9, 1997, 1997 Berlin, Germany, Springer-Verlag, Germany), XP008016393 ISBN: 3-540-63232-8.
- Mambo et al, "Proxy Cryptosystems: Delegation of the Power to Decrypt Ciphertexts," IEICE Trans. Fundamentals vol. E80-A, No. 1:54-63 (Jan. 1997) XP00742245 ISSN: 0916-8508.
- Microsoft Word, Users Guide, Version 6.0, pp. 487-489, 549-555, 560-564, 572-575, 599-613, 616-631 (1993).
- Ojanperä and Prasad, eds., Wideband CDMA for Third Generation Mobile Communications (1998) [excerpts but not all pages numbered].
- Perritt, "Knowbots, Permissions Headers and Contract Law," Paper for the Conference on Technological Strategies for Protecting Intellectual Property in the Networked Multimedia Environment, pp. 1-22 (Apr. 2-3, 1993 with revisions of Apr. 30, 1993).
- Raggett, (Hewlett Packard), "HTML+(Hypertext markup language)," pp. 1-31 (Jul. 12, 1993) URL:<http://citeseer.ist.psu.edu/correct/340709>.
- Samuelson et al, "Intellectual Property Rights for Digital Library and Hypertext Publishing Systems: An Analysis of Xanadu," Hypertext '91 Proceedings, pp. 39-50 (Dec. 1991).
- No Author, "Softlock Services Introduces . . . Softlock Services" Press Release (Jan. 28, 1994).
- Ius Mentis, "The ElGamal Public Key System," pp. 1-2 (Oct. 1, 2005) online at <http://www.iusmentis.com/technology/encryption/elgamal/>.
- Microsoft Word User's Guide, pp. 773-774, 315-316, 487-489, 561-564, 744, 624-633 (1993).
- No Author, "What is the ElGamal Cryptosystem," p. 1 (Nov. 27, 2006) online at <http://www.x5.net/faqs/crypto/q29.html>.
- Johnson et al., "A Secure Distributed Capability Based System," ACM, pp. 392-402 (1985).
- Wikipedia, "El Gamal Encryption," pp. 1-3 (last modified Nov. 2, 2006) online at http://en.wikipedia.org/wiki/ElGamal_encryption.
- Blaze, "Atomic Proxy Cryptography," p. 1 Abstract (Oct. 20, 1998).
- Blaze, "Matt Blaze's Technical Papers," pp. 1-6 (last updated Aug. 6, 2006).
- Online Search Results for "inverted file", "inverted index" from www.techweb.com, www.cryer.co.uk, computing-dictionary.thefreedictionary.com, www.nist.gov, en.wikipedia.org, www.cni.org, www.tiscali.co.uk (Jul. 15-16, 2006).
- Corporation for National Research Initiatives, "Digital Object Architecture Project", <http://www.nnri.reston.va.us/doa.html> (updated Nov. 28, 2006).
- Stefik, Summary and Analysis of A13 (Kahn, Robert E and Vinton G Cerf, "The Digital Library Project, vol. 1: The World of Knowbots (Draft), An Open Architecture for a Digital Library System and a Plan for its Development," Corporation for National Research Initiatives (Mar. 1988)), pp. 1-25 (May 30, 2007).
- Johnson et al., "A Secure Distributed Capability Based System," Proceedings of the 1985 ACM Annual Conference on

US 7,523,072 B2

Page 6

the Range of Computing: MID-80's Perspective: MID-80's Perspective *Association for Computing Machinery* pp. 392-402 (1985).

Perritt, "Technologies Strategies for Protecting IP in the Networked Multimedia Environment", Apr. 2-3, 1993, Knowbot Permissions.

Delaigle, "Digital Watermarking", Spie Conference in Optical Security and Counterfeit Deterrence Techniques, San Jose, CA Feb. 1996, vol. 2659 pp. 99-110.

Delaigle, "Digital Watermarking," Spie Conference in Optical Security and Counterfeit Deterrence Techniques, San Jose, CA (Feb. 1996).

Perritt, "Technologies Strategies for Protecting Intellectual Property in the Networked Multimedia Environment," Knowbots, Permissions Headers and Contract Law (Apr. 2-3, 1993).

* cited by examiner

FIG. 1

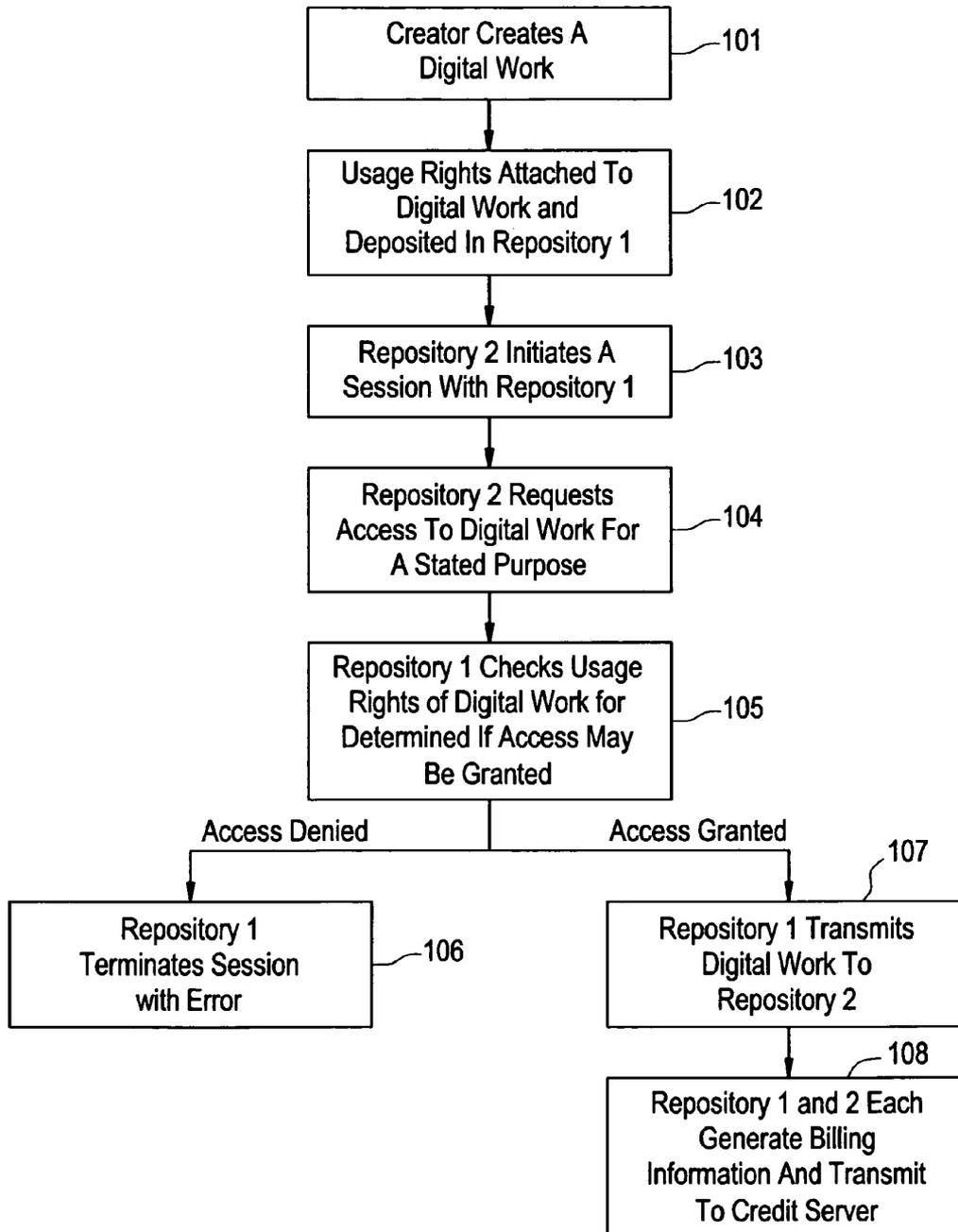


FIG. 2

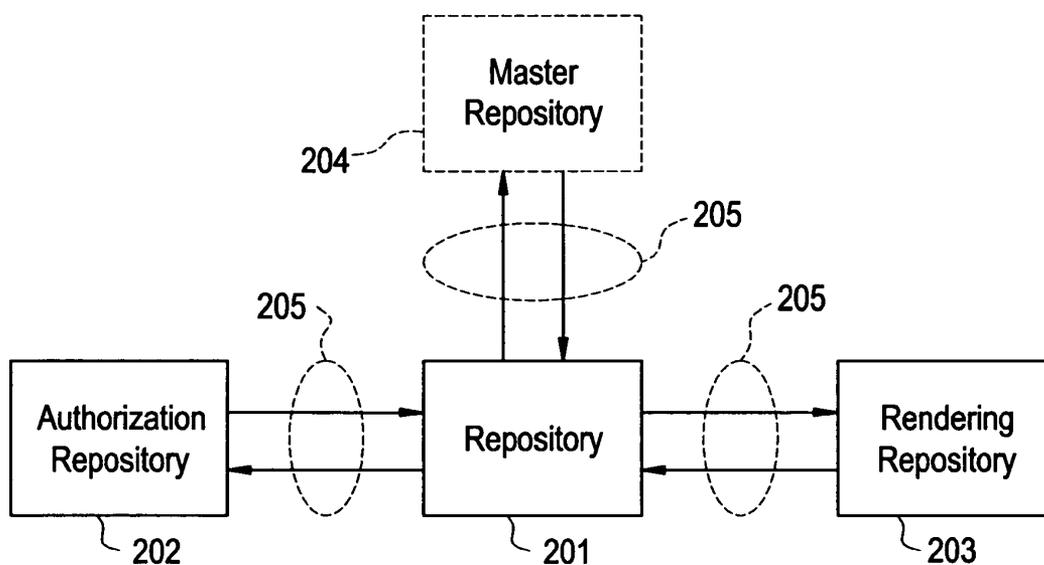


FIG. 3

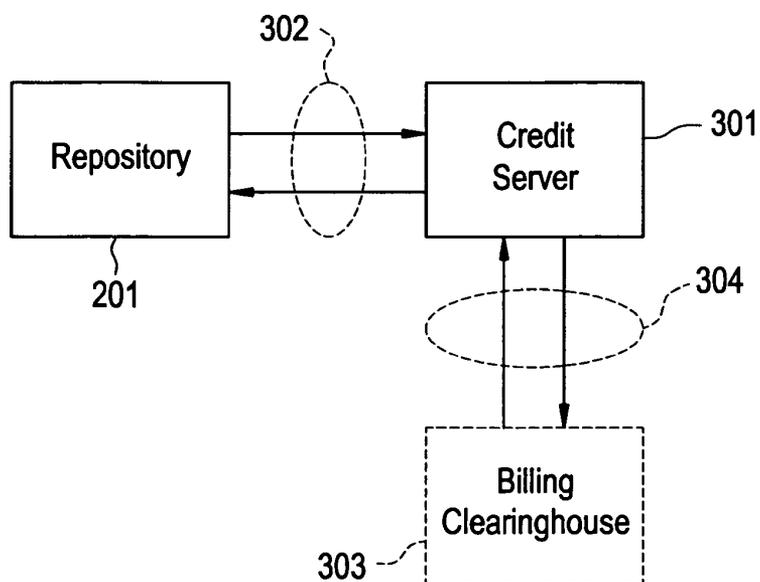


FIG. 4A

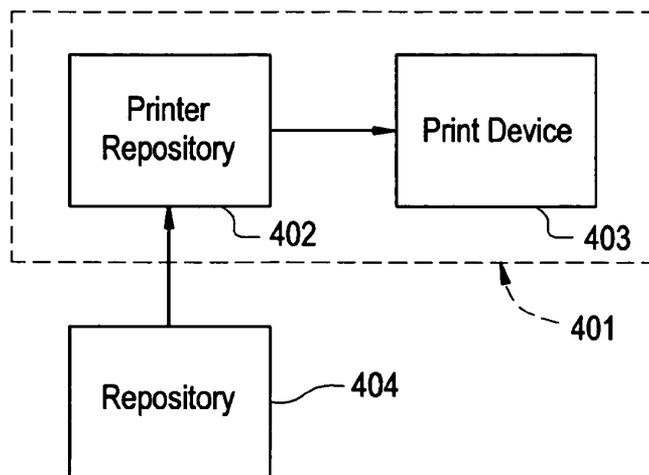


FIG. 4B

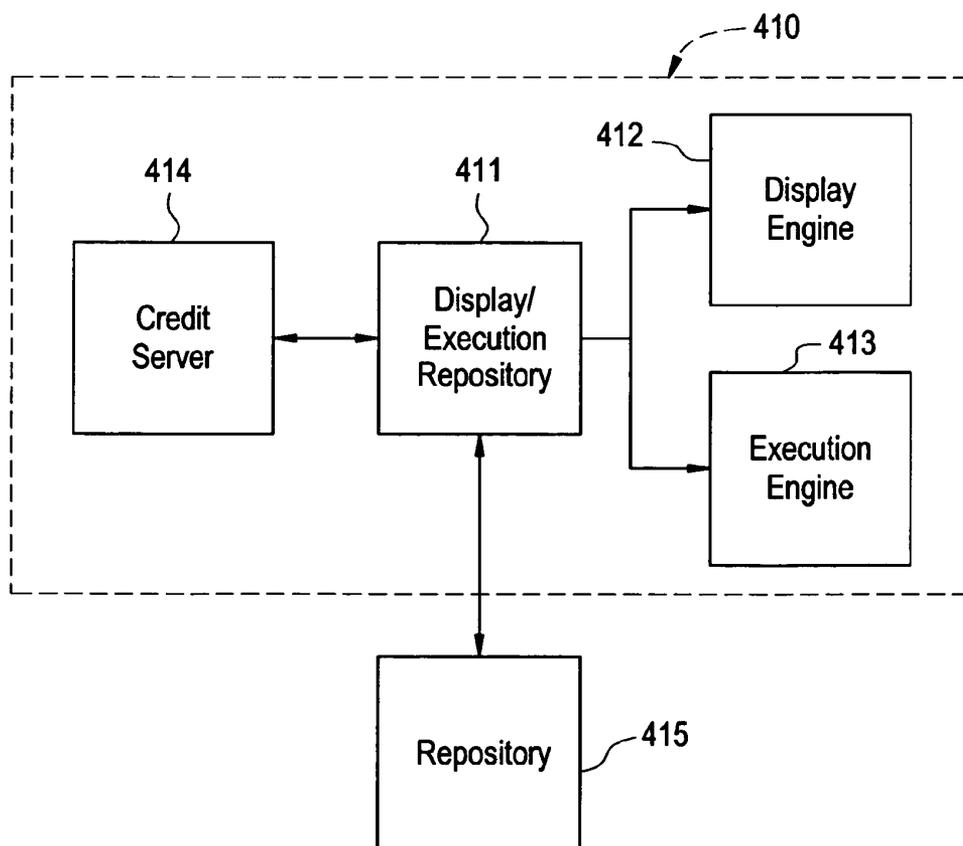


FIG. 5

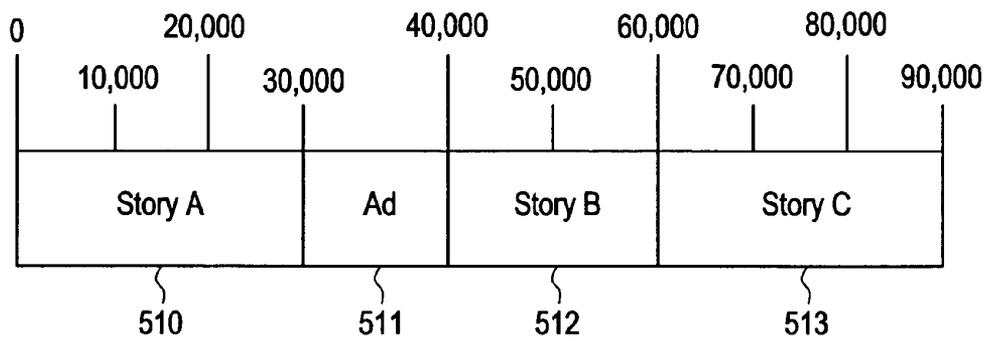


FIG. 6

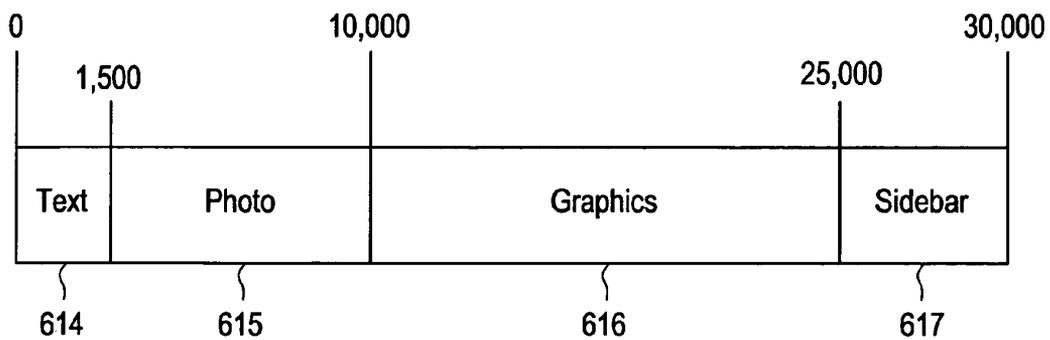


FIG. 7

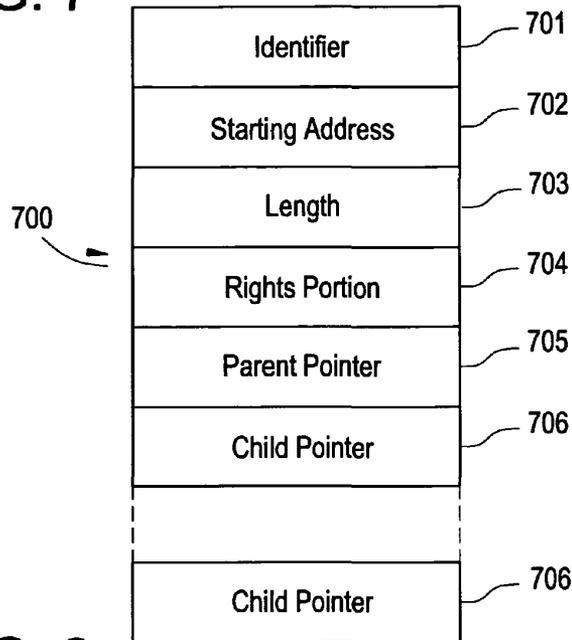


FIG. 8

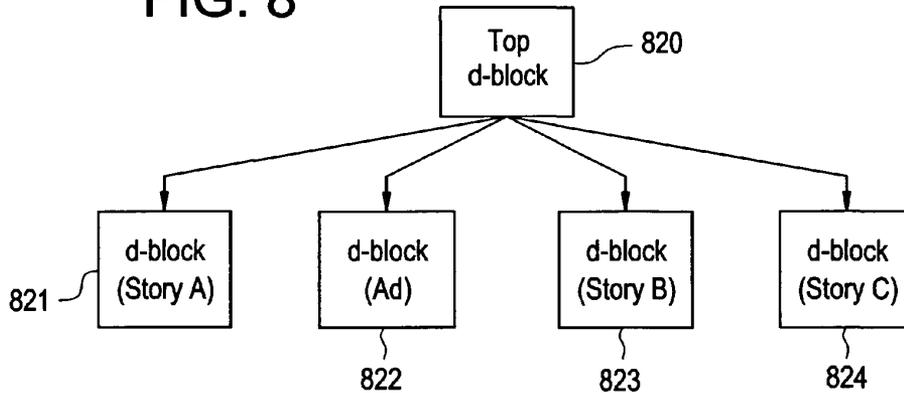


FIG. 9

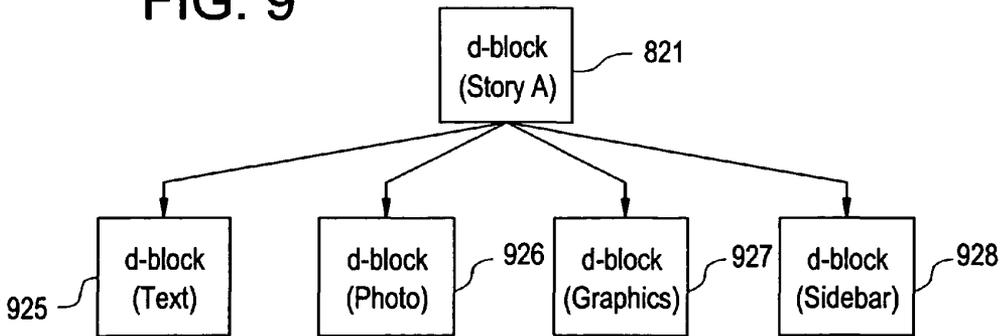


FIG. 10

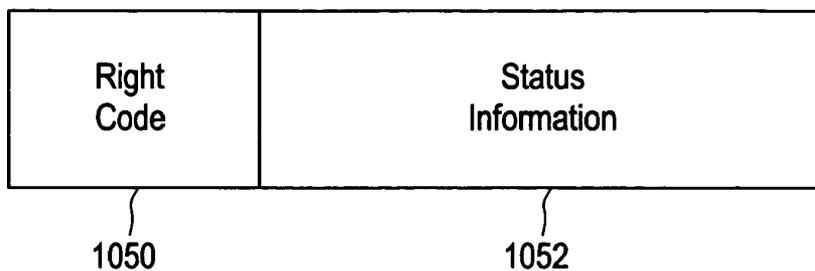


FIG. 14

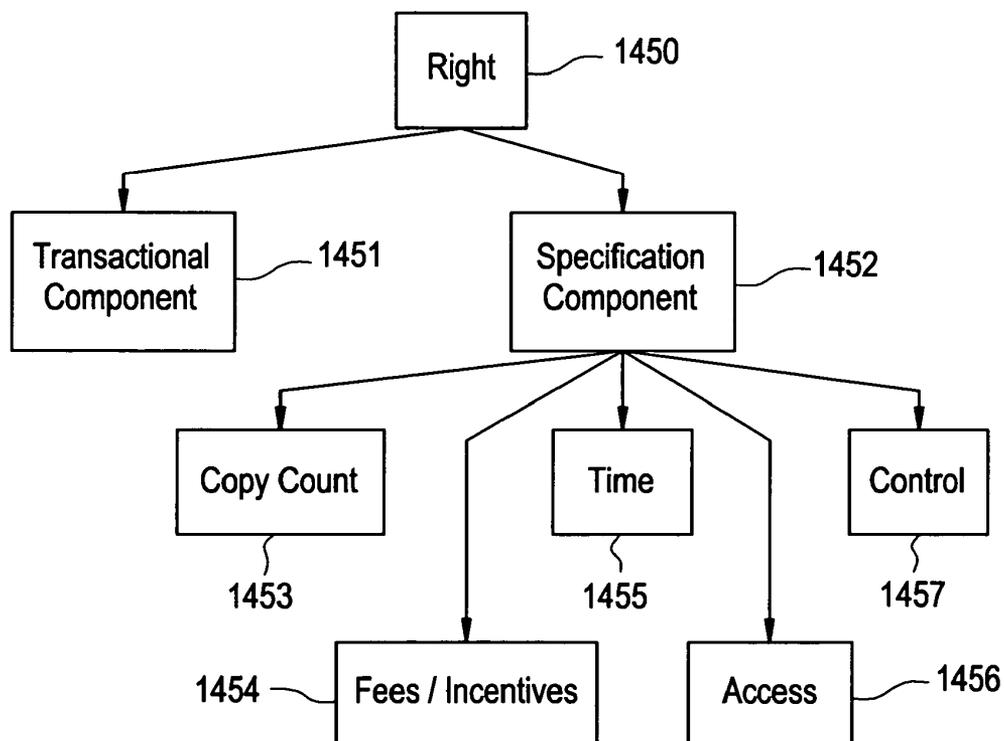


FIG. 11

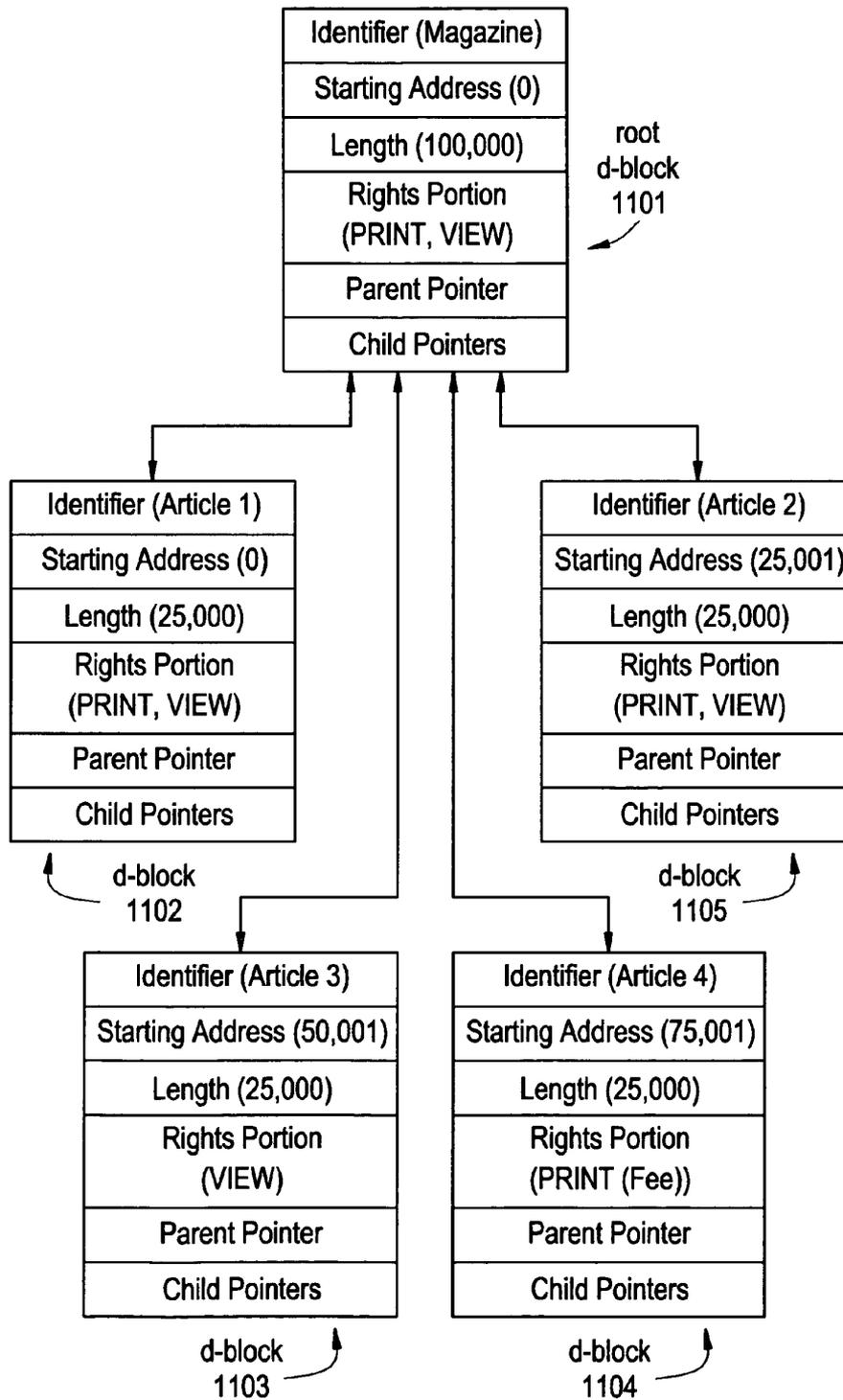


FIG. 12

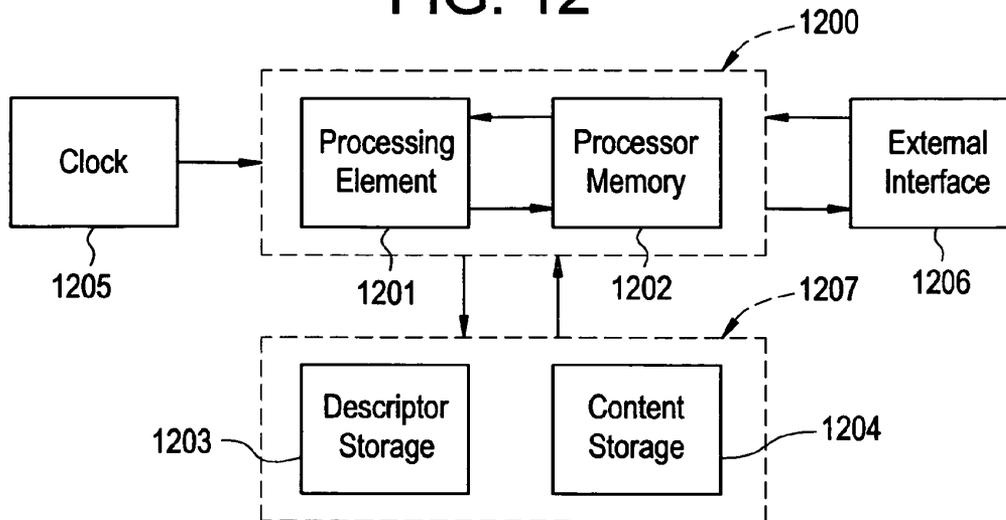
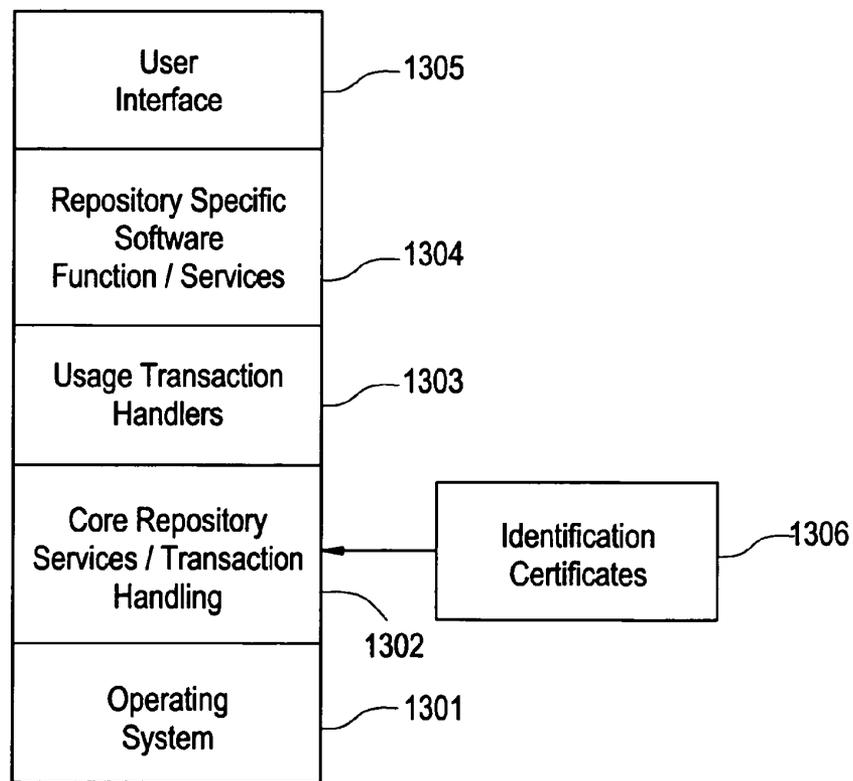


FIG. 13



U.S. Patent

Apr. 21, 2009

Sheet 9 of 13

US 7,523,072 B2

FIG. 15

- 1501 ~ Digital Work Rights: = (Rights*)
- 1502 ~ Right: = (Right-Code {Copy-Count} {Control-Spec} {Time-Spec} {Access-Spec} {Fee-Spec})
- 1503 ~ Right-Code: = Render-Code | Transport-Code | File-Management-Code | Derivative-Works-Code | Configuration-Code
- 1504 ~ Render-Code: = [Play: {Player: Player-ID} | Print: {Printer: Printer-ID}]
- 1505 ~ Transport-Code: = [Copy | Transfer | Loan {Remaining-Rights: Next-Set-of-Rights}] {{Next-Copy-Rights: Next-Set-of-Rights}}
- 1506 ~ File-Management-Code: = Backup {Back-Up-Copy-Rights: Next-Set-of-Rights} | Restore | Delete | Folder | Directory {Name: Hide-Local | Hide-Remote} {Parts: Hide-Local | Hide-Remote}
- 1507 ~ Derivative-Works-Code: = [Extract | Embed | Edit {Process: Process-ID}] {Next-Copy-Rights: Next-Set-of-Rights}
- 1508 ~ Configuration-Code: = Install | Uninstall
- 1509 ~ Next-Set-of-Rights: = {{Add: Set-of-Rights}} {{Delete: Set-of-Rights}} {{Replace: Set-of-Rights}} {{Keep: Set-of-Rights}}
- 1510 ~ Copy-Count: = (Copies: positive-integer | 0 | Unlimited)
- 1511 ~ Control-Spec: = (Control: {Restrictable | Unrestrictable} {Unchargeable | Chargeable})
- 1512 ~ Time-Spec: = {{Fixed-Interval | Sliding-Interval | Meter-Time} Until: Expiration-Date)
- 1513 ~ Fixed-Interval: = From: Start-Time
- 1514 ~ Sliding-Interval: = Interval : Use-Duration
- 1515 ~ Meter-Time: = Time-Remaining: Remaining-Use
- 1516 ~ Access-Spec: = {{SC: Security-Class} {Authorization: Authorization-ID*} {Other-Authorization: Authorization-ID*} {Ticket: Ticket-ID}}
- 1517 ~ Fee-Spec: = {Scheduled-Discount} Regular-Fee-Spec | Scheduled-Fee-Spec | Markup-Spec
- 1518 ~ Scheduled-Discount: = Scheduled-Discount: (Scheduled-Discount: (Time-Spec Percentage)*)
- 1519 ~ Regular-Fee-Spec: = {{Fee: | Incentive:} [Per-Use-Spec | Metered-Rate-Spec | Best-Price-Spec | Call-For-Price-Spec] {Min: Money-Unit Per: Time-Spec} {Max: Money-Unit Per: Time-Spec} To: Account-ID)
- 1520 ~ Per-Use-Spec: = Per-Use: Money-Unit
- 1521 ~ Metered-Rate-Spec: = Metered: Money-Unit Per: Time-Spec
- 1522 ~ Best-Price-Spec: = Best-Price: Money-unit Max: Money-Unit
- 1523 ~ Call-For-Price-Spec: = Call-For-Price
- 1524 ~ Scheduled-Fee-Spec: = (Schedule: (Time-Spec Regular-Fee-Spec)*)
- 1525 ~ Markup-Spec: = Markup: percentage To: Account-ID

FIG. 16

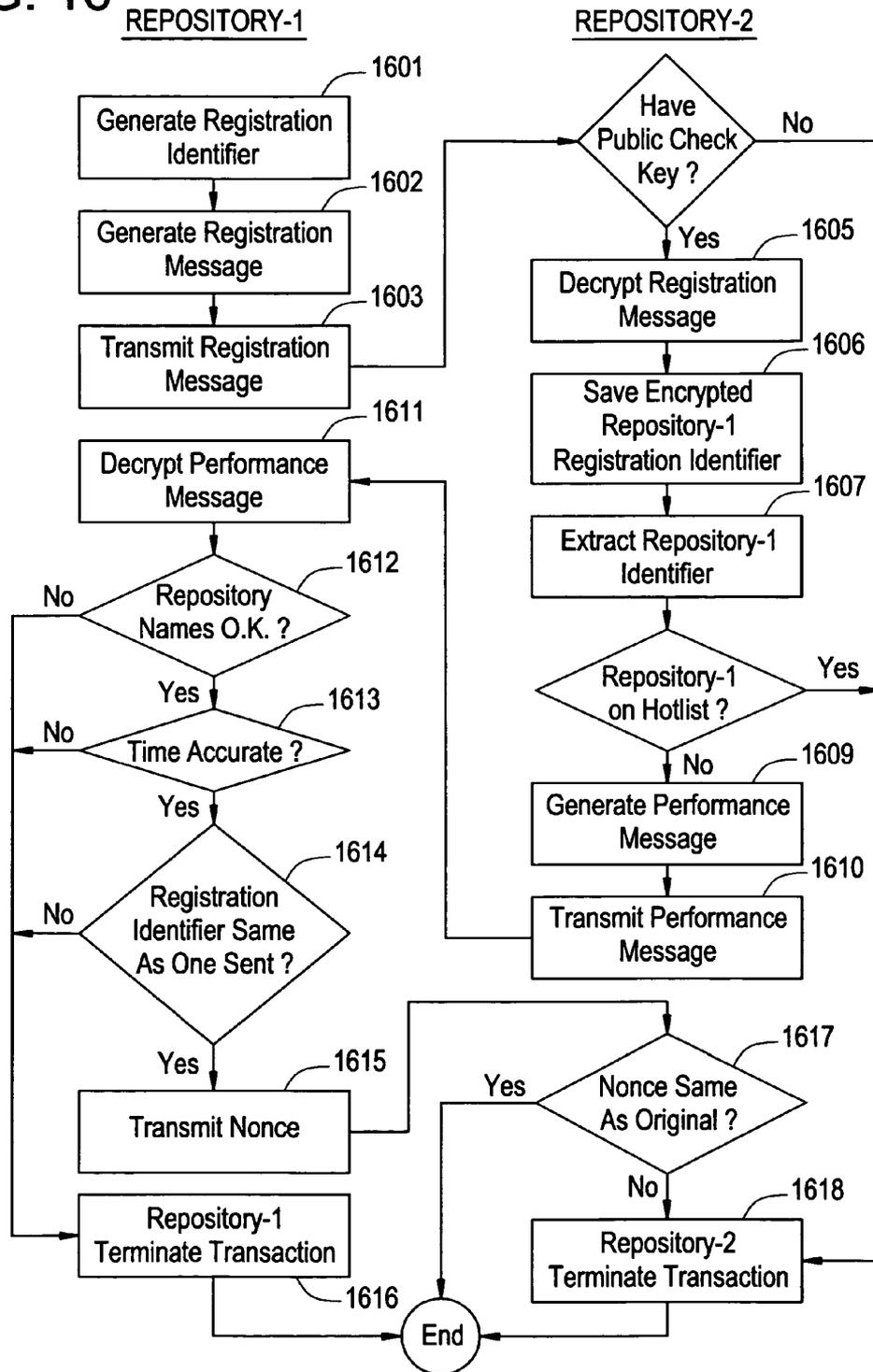


FIG. 17

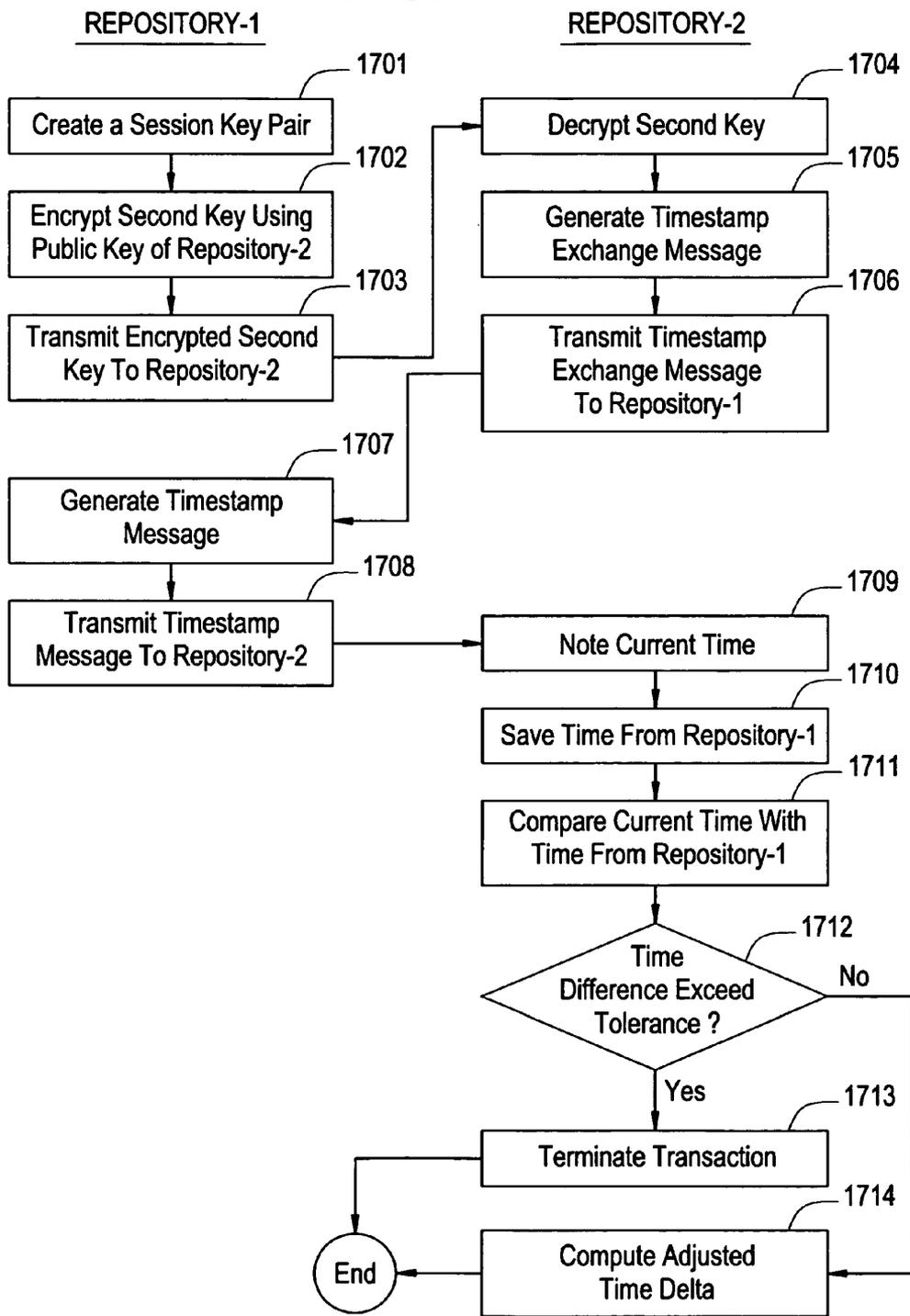


FIG. 18

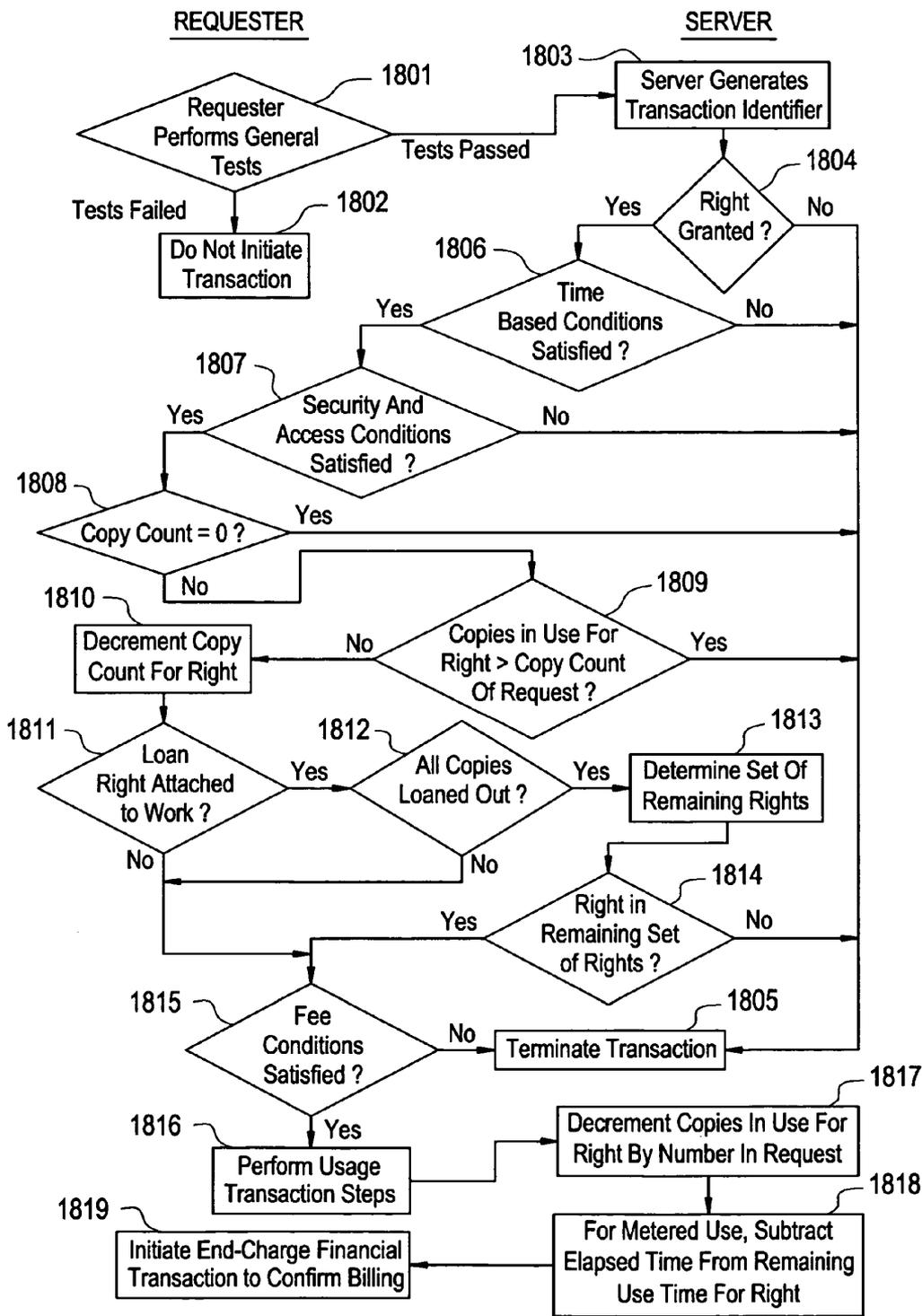
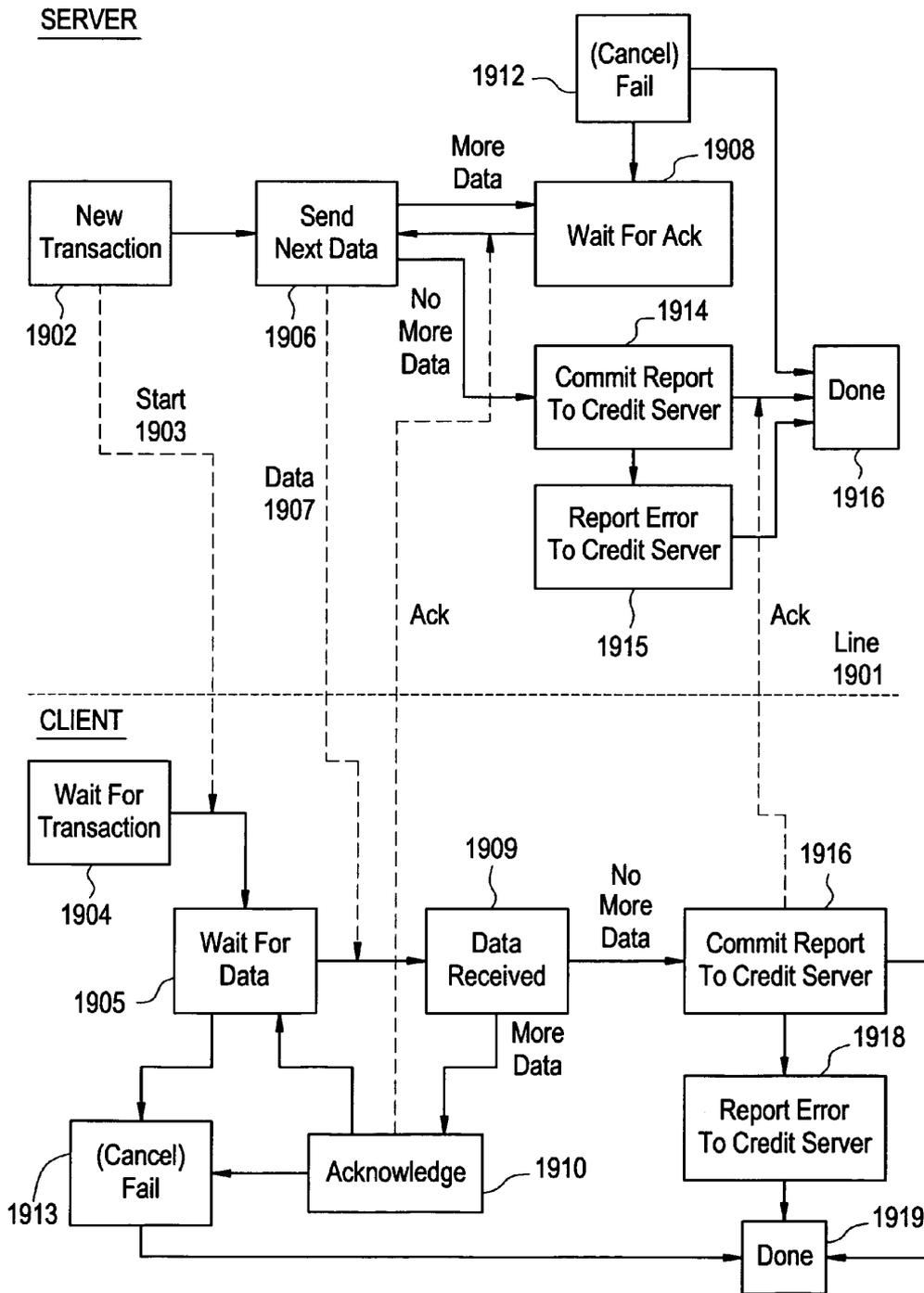


FIG. 19



US 7,523,072 B2

1

**SYSTEM FOR CONTROLLING THE
DISTRIBUTION AND USE OF DIGITAL
WORKS****CROSS REFERENCE TO RELATED
APPLICATIONS**

This application is a continuation of U.S. patent application Ser. No. 11/198,216 of STEFIK, et al., filed on Aug. 8, 2005, entitled "SYSTEM FOR CONTROLLING THE DISTRIBUTION AND USE OF DIGITAL WORKS," now pending, which is a continuation of U.S. patent application Ser. No. 10/176,608 of STEFIK, et al., filed on Jun. 24, 2002, entitled "SYSTEM FOR CONTROLLING THE DISTRIBUTION AND USE OF DIGITAL WORKS," now U.S. Pat. No. 6,934,693, which is a continuation of U.S. patent application Ser. No. 09/777,845, filed on Feb. 7, 2001, now pending, which is a divisional of U.S. patent application Ser. No. 08/967,084, filed on Nov. 10, 1997, now U.S. Pat. No. 6,236,971, which is a continuation of U.S. patent application Ser. No. 08/344,760, filed on Nov. 23, 1994, now abandoned, the disclosures of all of which are hereby incorporated by reference herein.

FIELD OF THE INVENTION

The present invention relates to the field of distribution and usage rights enforcement for digitally encoded works.

BACKGROUND OF THE INVENTION

A fundamental issue facing the publishing and information industries as they consider electronic publishing is how to prevent the unauthorized and unaccounted distribution or usage of electronically published materials. Electronically published materials are typically distributed in a digital form and recreated on a computer based system having the capability to recreate the materials. Audio and video recordings, software, books and multimedia works are all being electronically published. Companies in these industries receive royalties for each accounted for delivery of the materials, e.g. the sale of an audio CD at a retail outlet. Any unaccounted distribution of a work results in an unpaid royalty (e.g. copying the audio recording CD to another digital medium.)

The ease in which electronically published works can be "perfectly" reproduced and distributed is a major concern. The transmission of digital works over networks is commonplace. One such widely used network is the Internet. The Internet is a widespread network facility by which computer users in many universities, corporations and government entities communicate and trade ideas and information. Computer bulletin boards found on the Internet and commercial networks such as CompuServ and Prodigy allow for the posting and retrieving of digital information. Information services such as Dialog and LEXIS/NEXIS provide databases of current information on a wide variety of topics. Another factor which will exacerbate the situation is the development and expansion of the National Information Infrastructure (the NII). It is anticipated that, as the NII grows, the transmission of digital works over networks will increase many times over. It would be desirable to utilize the NII for distribution of digital works without the fear of widespread unauthorized copying.

The most straightforward way to curb unaccounted distribution is to prevent unauthorized copying and transmission. For existing materials that are distributed in digital form, various safeguards are used. In the case of software, copy

2

protection schemes which limit the number of copies that can be made or which corrupt the output when copying is detected have been employed. Another scheme causes software to become disabled after a predetermined period of time has lapsed. A technique used for workstation based software is to require that a special hardware device must be present on the workstation in order for the software to run, e.g., see U.S. Pat. No. 4,932,054 entitled "Method and Apparatus for Protecting Computer Software Utilizing Coded Filter Network in Conjunction with an Active Coded Hardware Device." Such devices are provided with the software and are commonly referred to as dongles.

Yet another scheme is to distribute software, but which requires a "key" to enable its use. This is employed in distribution schemes where "demos" of the software are provided on a medium along with the entire product. The demos can be freely used, but in order to use the actual product, the key must be purchased. These schemes do not hinder copying of the software once the key is initially purchased.

A system for ensuring that licenses are in place for using licensed products is described in PCT Publication WO 93/01550 to Griswold entitled "License Management System and Method." The licensed product may be any electronically published work but is most effective for use with works that are used for extended periods of time such as software programs. Griswold requires that the licensed product contain software to invoke a license check monitor at predetermined time intervals. The license check monitor generates request datagrams which identify the licensee. The request datagrams are sent to a license control system over an appropriate communication facility. The license control system then checks the datagram to determine if the datagram is from a valid licensee. The license control system then sends a reply datagram to the license check monitor indicating denial or approval of usage. The license control system will deny usage in the event that request datagrams go unanswered after a predetermined period of time (which may indicate an unauthorized attempt to use the licensed product). In this system, usage is managed at a central location by the response datagrams. So for example if license fees have not been paid, access to the licensed product is terminated.

It is argued by Griswold that the described system is advantageous because it can be implemented entirely in software. However, the system described by Griswold has limitations. An important limitation is that during the use of the licensed product, the user must always be coupled to an appropriate communication facility in order to send and receive datagrams. This creates a dependency on the communication facility. So if the communication facility is not available, the licensed product cannot be used. Moreover, some party must absorb the cost of communicating with the license server.

A system for controlling the distribution of digitally encoded books is embodied in a system available from VPR Systems, LTD. of St. Louis, Miss. The VPR system is self-contained and is comprised of: (1) point of sale kiosks for storing and downloading of books, (2) personal storage mediums (cartridges) to which the books are downloaded, and (3) readers for viewing the book. In a purchase transaction, a purchaser will purchase a voucher card representing the desired book. The voucher will contain sufficient information to identify the book purchased and perhaps some demographic information relating to the sales transaction. To download the book, the voucher and the cartridge are inserted into the kiosk.

The VPR system may also be used as a library. In such an embodiment, the kiosk manages the number of "copies" that may be checked out at one time. Further, the copy of the book

US 7,523,072 B2

3

is erased from the user's cartridge after a certain check-out time has expired. However, individuals cannot loan books because the cartridges may only be used with the owner's reader.

The foregoing distribution and protection schemes operate in part by preventing subsequent distribution of the work. While this certainly prevents unauthorized distributions, it does so by sacrificing the potential for subsequent revenue bearing uses. For example, it may be desirable to allow the lending of a purchased work to permit exposure of the work to potential buyers. Another example would be to permit the creation of a derivative work for a fee. Yet another example would be to permit copying the work for a fee (essentially purchasing it). Thus, it would be desirable to provide flexibility in how the owner of a digital work may allow it to be distributed.

While flexibility in distribution is a concern, the owners of a work want to make sure they are paid for such distributions. In U.S. Pat. No. 4,977,594 to Shear, entitled "Database Usage Metering and Protection System and Method," a system for metering and billing for usage of information distributed on a CD-ROM is described. The system requires the addition of a billing module to the computer system. The billing module may operate in a number of different ways. First, it may periodically communicate billing data to a central billing facility, whereupon the user may be billed. Second, billing may occur by disconnecting the billing module and the user sending it to a central billing facility where the data is read and a user bill generated.

U.S. Pat. No. 5,247,575, Sprague et al., entitled "Information Distribution System", describes an information distribution system which provides and charges only for user selected information. A plurality of encrypted information packages (IPs) are provided at the user site, via high and/or low density storage media and/or by broadcast transmission. Some of the IPs may be of no interest to the user. The IPs of interest are selected by the user and are decrypted and stored locally. The IPs may be printed, displayed or even copied to other storage media. The charges for the selected IP's are accumulated within a user apparatus and periodically reported by telephone to a central accounting facility. The central accounting facility also issues keys to decrypt the IPs. The keys are changed periodically. If the central accounting facility has not issued a new key for a particular user station, the station is unable to retrieve information from the system when the key is changed.

A system available from Wave Systems Corp. of Princeton, N.Y., provides for metering of software usage on a personal computer. The system is installed onto a computer and collects information on what software is in use, encrypts it and then transmits the information to a transaction center. From the transaction center, a bill is generated and sent to the user. The transaction center also maintains customer accounts so that licensing fees may be forwarded directly to the software providers. Software operating under this system must be modified so that usage can be accounted.

Known techniques for billing do not provide for billing of copies made of the work. For example, if data is copied from the CD-ROM described in Shear, any subsequent use of the copy of the information cannot be metered or billed. In other words, the means for billing runs with the media rather than

4

the underlying work. It would be desirable to have a distribution system where the means for billing is always transported with the work.

SUMMARY OF THE INVENTION

A system for controlling the distribution and use of digital works using digital tickets is disclosed. A ticket is an indicator that the ticket holder has already paid for or is otherwise entitled to some specified right, product or service. In the present invention, a "digital ticket" is used to enable the ticket holder to exercise usage rights specifying the requirement of the digital ticket. Usage rights are used to define how a digital work may be used or distributed. Specific instances of usage rights are used to indicate a particular manner of use or distribution. A usage right may specify a digital ticket which must be present before the right may be exercised. For example, a digital ticket may be specified in a Copy right of a digital work, so that exercise of the Copy right requires the party that desires a copy of the digital work be in possession of the necessary digital ticket. After a copy of the digital work is successfully sent to the requesting party, the digital ticket is "punched" to indicate that a copy of the digital work has been made. When the ticket is "punched" a predetermined number of times, it may no longer be used.

Digital works are stored in repositories. Repositories enforce the usage rights for digital works. Each repository has a "generic ticket agent" which punches tickets. In some instances only the generic ticket agent is necessary. In other instances, punching by a "special ticket agent" residing on another repository may be desired. Punching by a "special ticket agent" enables greater security and control of the digital work. For example, it can help prevent digital ticket forgery. Special ticket agents are also useful in situations where an external database needs to be updated or checked.

A digital ticket is merely an instance of a digital work. Thus, a digital ticket may be distributed among repositories in the same fashion as other digital works.

A digital ticket may be used in many commercial scenarios such as in the purchase of software and prepaid upgrades. A digital ticket may also be used to limit the number of times that a right may be exercised. For example, a user may purchase a copy of a digital work, along with the right to make up to 5 Copies. In this case, the Copy right would have associated therewith a digital ticket that can be punched up to 5 times. Other such commercial scenarios will become apparent from the detailed description.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a flowchart illustrating a simple instantiation of the operation of the currently preferred embodiment of the present invention.

FIG. 2 is a block diagram illustrating the various repository types and the repository transaction flow between them in the currently preferred embodiment of the present invention.

FIG. 3 is a block diagram of a repository coupled with a credit server in the currently preferred embodiment of the present invention.

FIGS. 4a and 4b are examples of rendering systems as may be utilized in the currently preferred embodiment of the present invention.

FIG. 5 illustrates a contents file layout for a digital work as may be utilized in the currently preferred embodiment of the present invention.

US 7,523,072 B2

5

FIG. 6 illustrates a contents file layout for an individual digital work of the digital work of FIG. 5 as may be utilized in the currently preferred embodiment of the present invention.

FIG. 7 illustrates the components of a description block of the currently preferred embodiment of the present invention.

FIG. 8 illustrates a description tree for the contents file layout of the digital work illustrated in FIG. 5.

FIG. 9 illustrates a portion of a description tree corresponding to the individual digital work illustrated in FIG. 6.

FIG. 10 illustrates a layout for the rights portion of a description block as may be utilized in the currently preferred embodiment of the present invention.

FIG. 11 is a description tree wherein certain d-blocks have PRINT usage rights and is used to illustrate "strict" and "lenient" rules for resolving usage rights conflicts.

FIG. 12 is a block diagram of the hardware components of a repository as are utilized in the currently preferred embodiment of the present invention.

FIG. 13 is a block diagram of the functional (logical) components of a repository as are utilized in the currently preferred embodiment of the present invention.

FIG. 14 is diagram illustrating the basic components of a usage right in the currently preferred embodiment of the present invention.

FIG. 15 lists the usage rights grammar of the currently preferred embodiment of the present invention.

FIG. 16 is a flowchart illustrating the steps of certificate delivery, hotlist checking and performance testing as performed in a registration transaction as may be performed in the currently preferred embodiment of the present invention.

FIG. 17 is a flowchart illustrating the steps of session information exchange and clock synchronization as may be performed in the currently preferred embodiment of the present invention, after each repository in the registration transaction has successfully completed the steps described in FIG. 16.

FIG. 18 is a flowchart illustrating the basic flow for a usage transaction, including the common opening and closing step, as may be performed in the currently preferred embodiment of the present invention.

FIG. 19 is a state diagram of server and client repositories in accordance with a transport protocol followed when moving a digital work from the server to the client repositories, as may be performed in the currently preferred embodiment of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

Overview

A system for controlling use and distribution of digital works is disclosed. The present invention is directed to supporting commercial transactions involving digital works. The transition to digital works profoundly and fundamentally changes how creativity and commerce can work. It changes the cost of transporting or storing works because digital property is almost "massless." Digital property can be transported at electronic speeds and requires almost no warehousing. Keeping an unlimited supply of virtual copies on hand requires essentially no more space than keeping one copy on hand. The digital medium also lowers the costs of alteration, reuse and billing.

There is a market for digital works because creators are strongly motivated to reuse portions of digital works from others rather than creating their own completely. This is

6

because it is usually so much easier to use an existing stock photo or music clip than to create a new one from scratch.

Herein the terms "digital work", "work" and "content" refer to any work that has been reduced to a digital representation. This would include any audio, video, text, or multimedia work and any accompanying interpreter (e.g. software) that may be required for recreating the work. The term composite work refers to a digital work comprised of a collection of other digital works. The term "usage rights" or "rights" is a term which refers to rights granted to a recipient of a digital work. Generally, these rights define how a digital work can be used and if it can be further distributed. Each usage right may have one or more specified conditions which must be satisfied before the right may be exercised. Appendix 1 provides a Glossary of the terms used herein.

A key feature of the present invention is that usage rights are permanently "attached" to the digital work. Copies made of a digital work will also have usage rights attached. Thus, the usage rights and any associated fees assigned by a creator and subsequent distributor will always remain with a digital work.

The enforcement elements of the present invention are embodied in repositories. Among other things, repositories are used to store digital works, control access to digital works, bill for access to digital works and maintain the security and integrity of the system.

The combination of attached usage rights and repositories enable distinct advantages over prior systems. As noted in the prior art, payment of fees are primarily for the initial access. In such approaches, once a work has been read, computational control over that copy is gone. Metaphorically, "the content genie is out of the bottle and no more fees can be billed." In contrast, the present invention never separates the fee descriptions from the work. Thus, the digital work genie only moves from one trusted bottle (repository) to another, and all uses of copies are potentially controlled and billable.

FIG. 1 is a high level flowchart omitting various details but which demonstrates the basic operation of the present invention. Referring to FIG. 1, a creator creates a digital work, step 101. The creator will then determine appropriate usage rights and fees, attach them to the digital work, and store them in Repository 1, step 102. The determination of appropriate usage rights and fees will depend on various economic factors. The digital work remains securely in Repository 1 until a request for access is received. The request for access begins with a session initiation by another repository. Here a Repository 2 initiates a session with Repository 1, step 103. As will be described in greater detail below, this session initiation includes steps which help to insure that the respective repositories are trustworthy. Assuming that a session can be established, Repository 2 may then request access to the Digital Work for a stated purpose, step 104. The purpose may be, for example, to print the digital work or to obtain a copy of the digital work. The purpose will correspond to a specific usage right. In any event, Repository 1 checks the usage rights associated with the digital work to determine if the access to the digital work may be granted, step 105. The check of the usage rights essentially involves a determination of whether a right associated with the access request has been attached to the digital work and if all conditions associated with the right are satisfied. If the access is denied, repository 1 terminates the session with an error message, step 106. If access is granted, repository 1 transmits the digital work to repository 2, step 107. Once the digital work has been transmitted to repository 2, repository 1 and 2 each generate billing information for the access which is transmitted to a credit server,

US 7,523,072 B2

7

step 108. Such double billing reporting is done to insure against attempts to circumvent the billing process.

FIG. 2 illustrates the basic interactions between repository types in the present invention. As will become apparent from FIG. 2, the various repository types will serve different functions. It is fundamental that repositories will share a core set of functionality which will enable secure and trusted communications. Referring to FIG. 2, a repository 201 represents the general instance of a repository. The repository 201 has two modes of operation; a server mode and a requester mode. When in the server mode, the repository will be receiving and processing access requests to digital works. When in the requester mode, the repository will be initiating requests to access digital works. Repository 201 is general in the sense that its primary purpose is as an exchange medium for digital works. During the course of operation, the repository 201 may communicate with a plurality of other repositories, namely authorization repository 202, rendering repository 203 and master repository 204. Communication between repositories occurs utilizing a repository transaction protocol 205.

Communication with an authorization repository 202 may occur when a digital work being accessed has a condition requiring an authorization. Conceptually, an authorization is a digital certificate such that possession of the certificate is required to gain access to the digital work. An authorization is itself a digital work that can be moved between repositories and subjected to fees and usage rights conditions. An authorization may be required by both repositories involved in an access to a digital work.

Communication with a rendering repository 203 occurs in connection with the rendering of a digital work. As will be described in greater detail below, a rendering repository is coupled with a rendering device (e.g. a printer device) to comprise a rendering system.

Communication with a master repository 205 occurs in connection with obtaining an identification certificate. Identification certificates are the means by which a repository is identified as "trustworthy". The use of identification certificates is described below with respect to the registration transaction.

FIG. 3 illustrates the repository 201 coupled to a credit server 301. The credit server 301 is a device which accumulates billing information for the repository 201. The credit server 301 communicates with repository 201 via billing transactions 302 to record billing transactions. Billing transactions are reported to a billing clearinghouse 303 by the credit server 301 on a periodic basis. The credit server 301 communicates to the billing clearinghouse 303 via clearinghouse transactions 304. The clearinghouse transactions 304 enable a secure and encrypted transmission of information to the billing clearinghouse 303.

Rendering Systems

A rendering system is generally defined as a system comprising a repository and a rendering device which can render a digital work into its desired form. Examples of a rendering system may be a computer system, a digital audio system, or a printer. A rendering system has the same security features as a repository. The coupling of a rendering repository with the rendering device may occur in a manner suitable for the type of rendering device.

FIG. 4a illustrates a printer as an example of a rendering system. Referring to FIG. 4, printer system 401 has contained therein a printer repository 402 and a print device 403. It should be noted that the dashed line defining printer system

8

401 defines a secure system boundary. Communications within the boundary is assumed to be secure. Depending on the security level, the boundary also represents a barrier intended to provide physical integrity. The printer repository 402 is an instantiation of the rendering repository 205 of FIG. 2. The printer repository 402 will in some instances contain an ephemeral copy of a digital work which remains until it is printed out by the print engine 403. In other instances, the printer repository 402 may contain digital works such as fonts, which will remain and can be billed based on use. This design assures that all communication lines between printers and printing devices are encrypted, unless they are within a physically secure boundary. This design feature eliminates a potential "fault" point through which the digital work could be improperly obtained. The printer device 403 represents the printer components used to create the printed output.

Also illustrated in FIG. 4a is the repository 404. The repository 404 is coupled to the printer repository 402. The repository 404 represents an external repository which contains digital works.

FIG. 4b is an example of a computer system as a rendering system. A computer system may constitute a "multi-function" device since it may execute digital works (e.g. software programs) and display digital works (e.g. a digitized photograph). Logically, each rendering device can be viewed as having its own repository, although only one physical repository is needed. Referring to FIG. 4b, a computer system 410 has contained therein a display/execution repository 411. The display/execution repository 411 is coupled to display device, 412 and execution device 413. The dashed box surrounding the computer system 410 represents a security boundary within which communications are assumed to be secure. The display/execution repository 411 is further coupled to a credit server 414 to report any fees to be billed for access to a digital work and a repository 415 for accessing digital works stored therein.

Structure of Digital Works

Usage rights are attached directly to digital works. Thus, it is important to understand the structure of a digital work. The structure of a digital work, in particular composite digital works, may be naturally organized into an acyclic structure such as a hierarchy. For example, a magazine has various articles and photographs which may have been created and are owned by different persons. Each of the articles and photographs may represent a node in a hierarchical structure. Consequently, controls, i.e. usage rights, may be placed on each node by the creator. By enabling control and fee billing to be associated with each node, a creator of a work can be assured that the rights and fees are not circumvented.

In the currently preferred embodiment, the file information for a digital work is divided into two files: a "contents" file and a "description tree" file. From the perspective of a repository, the "contents" file is a stream of addressable bytes whose format depends completely on the interpreter used to play, display or print the digital work. The description tree file makes it possible to examine the rights and fees for a work without reference to the content of the digital work. It should be noted that the term description tree as used herein refers to any type of acyclic structure used to represent the relationship between the various components of a digital work.

FIG. 5 illustrates the layout of a contents file. Referring to FIG. 5, a digital work 509 is comprised of story A 510, advertisement 511, story B 512 and story C 513. It is assumed that the digital work is stored starting at a relative address of 0. Each of the parts of the digital work are stored linearly so

US 7,523,072 B2

9

that story A 510 is stored at approximately addresses 0-30,000, advertisement 511 at addresses 30,001-40,000, story B 512 at addresses 40,001-60,000 and story C 513 at addresses 60,001-85K. The detail of story A 510 is illustrated in FIG. 6. Referring to FIG. 6, the story A 510 is further broken down to show text 614 stored at address 0-1500, soldier photo 615 at addresses 1501-10,000, graphics 616 stored at addresses 10,001-25,000 and sidebar 617 stored address 25,001-30,000. Note that the data in the contents file may be compressed (for saving storage) or encrypted (for security).

From FIGS. 5 and 6 it is readily observed that a digital work can be represented by its component parts as a hierarchy. The description tree for a digital work is comprised of a set of related descriptor blocks (d-blocks). The contents of each d-block are described with respect to FIG. 7. Referring to FIG. 7, a d-block 700 includes an identifier 701 which is a unique identifier for the work in the repository, a starting address 702 providing the start address of the first byte of the work, a length 703 giving the number of bytes in the work, a rights portion 704 wherein the granted usage rights and their status data are maintained, a parent pointer 705 for pointing to a parent d-block and child pointers 706 for pointing to the child d-blocks. In the currently preferred embodiment, the identifier 701 has two parts. The first part is a unique number assigned to the repository upon manufacture. The second part is a unique number assigned to the work upon creation. The rights portion 704 will contain a data structure, such as a look-up table, wherein the various information associated with a right is maintained. The information required by the respective usage rights is described in more detail below. D-blocks form a strict hierarchy. The top d-block of a work has no parent; all other d-blocks have one parent. The relationship of usage rights between parent and child d-blocks and how conflicts are resolved is described below.

A special type of d-block is a "shell" d-block. A shell d-block adds no new content beyond the content of its parts. A shell d-block is used to add rights and fee information, typically by distributors of digital works.

FIG. 8 illustrates a description tree for the digital work of FIG. 5. Referring to FIG. 8, a top d-block 820 for the digital work points to the various stories and advertisements contained therein. Here, the top d-block 820 points to d-block 821 (representing story A 510), d-block 822 (representing the advertisement 511), d-block 823 (representing story B 512) and d-block 824 (representing story C 513).

The portion of the description tree for Story A 510 is illustrated in FIG. 9. D-block 925 represents text 614, d-block 926 represents photo 615, d-block 927 represents graphics 616 by and d-block 928 represents sidebar 617.

The rights portion 704 of a descriptor block is further illustrated in FIG. 10. FIG. 10 illustrates a structure which is repeated in the rights portion 704 for each right. Referring to FIG. 10, each right will have a right code field 1001 and status information field 1002. The right code field 1001 will contain a unique code assigned to a right. The status information field 1002 will contain information relating to the state of a right and the digital work. Such information is indicated below in Table 1. The rights as stored in the rights portion 304 may typically be in numerical order based on the right code.

The approach for representing digital works by separating description data from content assumes that parts of a file are contiguous but takes no position on the actual representation of content. In particular, it is neutral to the question of whether content representation may take an object oriented approach. It would be natural to represent content as objects. In principle, it may be convenient to have content objects that include the billing structure and rights information that is

10

represented in the d-blocks. Such variations in the design of the representation are possible and are viable alternatives but may introduce processing overhead, e.g. the interpretation of the objects.

TABLE 1

DIGITAL WORK STATE INFORMATION		
Property	Value	Use
Copies-in-Use	Number	A counter of the number of copies of a work that are in use. Incremented when another copy is used; decremented when use is completed.
Loan-Period	Time-Units	Indicator of the maximum number of time-units that a document can be loaned out
Loaner-Copy	Boolean	Indicator that the current work is a loaned out copy of an authorized digital work.
Remaining-Time	Time-Units	Indicator of the remaining time of use on a metered document right.
Document-Descr	String	A string containing various identifying information about a document. The exact format of this is not specified, but it can include information such as a publisher name, author name, ISBN number, and so on.
Revenue-Owner	RO-Descr	A handle identifying a revenue owner for a digital work. This is used for reporting usage fees.
Publication-Date	Date-Descr	The date that the digital work was published.
History-list	History-Rec	A list of events recording the repositories and dates for operations that copy, transfer, backup, or restore a digital work.

Digital works are stored in a repository as part of a hierarchical file system. Folders (also termed directories and sub-directories) contain the digital works as well as other folders. Digital works and folders in a folder are ordered in alphabetical order. The digital works are typed to reflect how the files are used. Usage rights can be attached to folders so that the folder itself is treated as a digital work. Access to the folder would then be handled in the same fashion as any other digital work. As will be described in more detail below, the contents of the folder are subject to their own rights. Moreover, file management rights may be attached to the folder which defines how folder contents can be managed.

Attaching Usage Rights to a Digital Work

It is fundamental to the present invention that the usage rights are treated as part of the digital work. As the digital work is distributed, the scope of the granted usage rights will remain the same or may be narrowed. For example, when a digital work is transferred from a document server to a repository, the usage rights may include the right to loan a copy for a predetermined period of time (called the original rights). When the repository loans out a copy of the digital work, the usage rights in the loaner copy (called the next set of rights) could be set to prohibit any further rights to loan out the copy. The basic idea is that one cannot grant more rights than they have.

The attachment of usage rights into a digital work may occur in a variety of ways. If the usage rights will be the same for an entire digital work, they could be attached when the digital work is processed for deposit in the digital work server. In the case of a digital work having different usage rights for the various components, this can be done as the digital work is being created. An authoring tool or digital work assembling

US 7,523,072 B2

11

tool could be utilized which provides for an automated process of attaching the usage rights.

As will be described below, when a digital work is copied, transferred or loaned, a “next set of rights” can be specified. The “next set of rights” will be attached to the digital work as it is transported.

Resolving Conflicting Rights

Because each part of a digital work may have its own usage rights, there will be instances where the rights of a “contained part” are different from its parent or container part. As a result, conflict rules must be established to dictate when and how a right may be exercised. The hierarchical structure of a digital work facilitates the enforcement of such rules. A “strict” rule would be as follows: a right for a part in a digital work is sanctioned if and only if it is sanctioned for the part, for ancestor d-blocks containing the part and for all descendent d-blocks. By sanctioned, it is meant that (1) each of the respective parts must have the right, and (2) any conditions for exercising the right are satisfied.

It also possible to implement the present invention using a more lenient rule. In the more lenient rule, access to the part may be enabled to the descendent parts which have the right, but access is denied to the descendents which do not.

Example of applying both the strict rule and lenient is illustrated with reference to FIG. 11. Referring to FIG. 11, a root d-block 1101 has child d-blocks 1102-1105. In this case, root d-block represents a magazine, and each of the child d-blocks 1102-1105 represent articles in the magazine. Suppose that a request is made to PRINT the digital work represented by root d-block 1101 wherein the strict rule is followed. The rights for the root d-block 1101 and child d-blocks 1102-1105 are then examined. Root d-block 1101 and child d-blocks 1102 and 1105 have been granted PRINT rights. Child d-block 1103 has not been granted PRINT rights and child d-block 1104 has PRINT rights conditioned on payment of a usage fee.

Under the strict rule the PRINT right cannot be exercised because the child d-block does not have the PRINT right. Under the lenient rule, the result would be different. The digital works represented by child d-blocks 1102 and 1105 could be printed and the digital work represented by d-block 1104 could be printed so long as the usage fee is paid. Only the digital work represented by d-block 1103 could not be printed. This same result would be accomplished under the strict rule if the requests were directed to each of the individual digital works.

The present invention supports various combinations of allowing and disallowing access. Moreover, as will be described below, the usage rights grammar permits the owner of a digital work to specify if constraints may be imposed on the work by a container part. The manner in which digital works may be sanctioned because of usage rights conflicts would be implementation specific and would depend on the nature of the digital works.

Repositories

Many of the powerful functions of repositories—such as their ability to “loan” digital works or automatically handle the commercial reuse of digital works—are possible because they are trusted systems. The systems are trusted because they are able to take responsibility for fairly and reliably carrying out the commercial transactions. That the systems can be responsible (“able to respond”) is fundamentally an issue of integrity. The integrity of repositories has three parts: physical integrity, communications integrity, and behavioral integrity.

12

Physical integrity refers to the integrity of the physical devices themselves. Physical integrity applies both to the repositories and to the protected digital works. Thus, the higher security classes of repositories themselves may have sensors that detect when tampering is attempted on their secure cases. In addition to protection of the repository itself, the repository design protects access to the content of digital works. In contrast with the design of conventional magnetic and optical devices—such as floppy disks, CD-ROMs, and videotapes—repositories never allow non-trusted systems to access the works directly. A maker of generic computer systems cannot guarantee that their platform will not be used to make unauthorized copies. The manufacturer provides generic capabilities for reading and writing information, and the general nature of the functionality of the general computing device depends on it. Thus, a copy program can copy arbitrary data. This copying issue is not limited to general purpose computers. It also arises for the unauthorized duplication of entertainment “software” such as video and audio recordings by magnetic recorders. Again, the functionality of the recorders depends on their ability to copy and they have no means to check whether a copy is authorized. In contrast, repositories prevent access to the raw data by general devices and can test explicit rights and conditions before copying or otherwise granting access. Information is only accessed by protocol between trusted repositories.

Communications integrity refers to the integrity of the communications channels between repositories. Roughly speaking, communications integrity means that repositories cannot be easily fooled by “telling them lies.” Integrity in this case refers to the property that repositories will only communicate with other devices that are able to present proof that they are certified repositories, and furthermore, that the repositories monitor the communications to detect “impostors” and malicious or accidental interference. Thus the security measures involving encryption, exchange of digital certificates, and nonces described below are all security measures aimed at reliable communication in a world known to contain active adversaries.

Behavioral integrity refers to the integrity in what repositories do. What repositories do is determined by the software that they execute. The integrity of the software is generally assured only by knowledge of its source. Restated, a user will trust software purchased at a reputable computer store but not trust software obtained off a random (insecure) server on a network. Behavioral integrity is maintained by requiring that repository software be certified and be distributed with proof of such certification, i.e. a digital certificate. The purpose of the certificate is to authenticate that the software has been tested by an authorized organization, which attests that the software does what it is supposed to do and that it does not compromise the behavioral integrity of a repository. If the digital certificate cannot be found in the digital work or the master repository which generated the certificate is not known to the repository receiving the software, then the software cannot be installed.

In the description of FIG. 2, it was indicated that repositories come in various forms. All repositories provide a core set of services for the transmission of digital works. The manner in which digital works are exchanged is the basis for all transaction between repositories. The various repository types differ in the ultimate functions that they perform. Repositories may be devices themselves, or they may be incorporated into other systems. An example is the rendering repository 205 of FIG. 2.

A repository will have associated with it a repository identifier. Typically, the repository identifier would be a unique

US 7,523,072 B2

13

number assigned to the repository at the time of manufacture. Each repository will also be classified as being in a particular security class. Certain communications and transactions may be conditioned on a repository being in a particular security class. The various security classes are described in greater detail below.

As a prerequisite to operation, a repository will require possession of an identification certificate. Identification certificates are encrypted to prevent forgery and are issued by a Master repository. A master repository plays the role of an authorization agent to enable repositories to receive digital works. Identification certificates must be updated on a periodic basis. Identification certificates are described in greater detail below with respect to the registration transaction.

A repository has both a hardware and functional embodiment. The functional embodiment is typically software executing on the hardware embodiment. Alternatively, the functional embodiment may be embedded in the hardware embodiment such as an Application Specific Integrated Circuit (ASIC) chip.

The hardware embodiment of a repository will be enclosed in a secure housing which if compromised, may cause the repository to be disabled. The basic components of the hardware embodiment of a repository are described with reference to FIG. 12. Referring to FIG. 12, a repository is comprised of a processing means 1200, storage system 1207, clock 1205 and external interface 1206. The processing means 1200 is comprised of a processor element 1201 and processor memory 1202. The processing means 1201 provides controller, repository transaction and usage rights transaction functions for the repository. Various functions in the operation of the repository such as decryption and/or decompression of digital works and transaction messages are also performed by the processing means 1200. The processor element 1201 may be a microprocessor or other suitable computing component. The processor memory 1202 would typically be further comprised of Read Only Memories (ROM) and Random Access Memories (RAM). Such memories would contain the software instructions utilized by the processor element 1201 in performing the functions of the repository.

The storage system 1207 is further comprised of descriptor storage 1203 and content storage 1204. The description tree storage 1203 will store the description tree for the digital work and the content storage will store the associated content. The description tree storage 1203 and content storage 1204 need not be of the same type of storage medium, nor are they necessarily on the same physical device. So for example, the descriptor storage 1203 may be stored on a solid state storage (for rapid retrieval of the description tree information), while the content storage 1204 may be on a high capacity storage such as an optical disk.

The clock 1205 is used to time-stamp various time based conditions for usage rights or for metering usage fees which may be associated with the digital works. The clock 1205 will have an uninterruptible power supply, e.g. a battery, in order to maintain the integrity of the time-stamps. The external interface means 1206 provides for the signal connection to other repositories and to a credit server. The external interface means 1206 provides for the exchange of signals via such standard interfaces such as RS-232 or Personal Computer Manufacturers Card Industry Association (PCMCIA) standards, or FDDI. The external interface means 1206 may also provide network connectivity.

The functional embodiment of a repository is described with reference to FIG. 13. Referring to FIG. 13, the functional embodiment is comprised of an operating system 1301, core

14

repository services 1302, usage transaction handlers 1303, repository specific functions, 1304 and a user interface 1305. The operating system 1301 is specific to the repository and would typically depend on the type of processor being used. The operating system 1301 would also provide the basic services for controlling and interfacing between the basic components of the repository.

The core repository services 1302 comprise a set of functions required by each and every repository. The core repository services 1302 include the session initiation transactions which are defined in greater detail below. This set of services also includes a generic ticket agent which is used to "punch" a digital ticket and a generic authorization server for processing authorization specifications. Digital tickets and authorizations are specific mechanisms for controlling the distribution and use of digital works and are described and more detail below. Note that coupled to the core repository services are a plurality of identification certificates 1306. The identification certificates 1306 are required to enable the use of the repository.

The usage transactions handler 1303 comprise functionality for processing access requests to digital works and for billing fees based on access. The usage transactions supported will be different for each repository type. For example, it may not be necessary for some repositories to handle access requests for digital works.

The repository specific functionality 1304 comprises functionality that is unique to a repository. For example, the master repository has special functionality for issuing digital certificates and maintaining encryption keys. The repository specific functionality 1304 would include the user interface implementation for the repository.

Repository Security Classes

For some digital works the losses caused by any individual instance of unauthorized copying is insignificant and the chief economic concern lies in assuring the convenience of access and low-overhead billing. In such cases, simple and inexpensive handheld repositories and network-based workstations may be suitable repositories, even though the measures and guarantees of security are modest.

At the other extreme, some digital works such as a digital copy of a first run movie or a bearer bond or stock certificate would be of very high value so that it is prudent to employ caution and fairly elaborate security measures to ensure that they are not copied or forged. A repository suitable for holding such a digital work could have elaborate measures for ensuring physical integrity and for verifying authorization before use.

By arranging a universal protocol, all kinds of repositories can communicate with each other in principle. However, creators of some works will want to specify that their works will only be transferred to repositories whose level of security is high enough. For this reason, document repositories have a ranking system for classes and levels of security. The security classes in the currently preferred embodiment are described in Table 2.

TABLE 2

REPOSITORY SECURITY LEVELS	
Level	Description of Security
0	Open system. Document transmission is unencrypted. No digital certificate is required for identification. The security of the system

US 7,523,072 B2

15

TABLE 2-continued

REPOSITORY SECURITY LEVELS	
Level	Description of Security
	depends mostly on user honesty, since only modest knowledge may be needed to circumvent the security measures. The repository has no provisions for preventing unauthorized programs from running and accessing or copying files. The system does not prevent the use of removable storage and does not encrypt stored files.
1	Minimal security. Like the previous class except that stored files are minimally encrypted, including ones on removable storage.
2	Basic security. Like the previous class except that special tools and knowledge are required to compromise the programming, the contents of the repository, or the state of the clock. All digital communications are encrypted. A digital certificate is provided as identification. Medium level encryption is used. Repository identification number is unforgeable.
3	General security. Like the previous class plus the requirement of special tools are needed to compromise the physical integrity of the repository and that modest encryption is used on all transmissions. Password protection is required to use the local user interface. The digital clock system cannot be reset without authorization. No works would be stored on removable storage. When executing works as programs, it runs them in their own address space and does not give them direct access to any file storage or other memory containing system code or works. They can access works only through the transmission transaction protocol.
4	Like the previous class except that high level encryption is used on all communications. Sensors are used to record attempts at physical and electronic tampering. After such tampering, the repository will not perform other transactions until it has reported such tampering to a designated server.
5	Like the previous class except that if the physical or digital attempts at tampering exceed some preset thresholds that threaten the physical integrity of the repository or the integrity of digital and cryptographic barriers, then the repository will save only document description records of history but will erase or destroy any digital identifiers that could be misused if released to an unscrupulous party. It also modifies any certificates of authenticity to indicate that the physical system has been compromised. It also erases the contents of designated documents.
6	Like the previous class except that the repository will attempt

16

TABLE 2-continued

REPOSITORY SECURITY LEVELS	
Level	Description of Security
5	wireless communication to report tampering and will employ noisy alarms.
10	This would correspond to a very high level of security. This server would maintain constant communications to remote security systems reporting transactions, sensor readings, and attempts to circumvent security.

The characterization of security levels described in Table 2 is not intended to be fixed. More important is the idea of having different security levels for different repositories. It is anticipated that new security classes and requirements will evolve according to social situations and changes in technology.

Repository User Interface

A user interface is broadly defined as the mechanism by which a user interacts with a repository in order to invoke transactions to gain access to a digital work, or exercise usage rights. As described above, a repository may be embodied in various forms. The user interface for a repository will differ depending on the particular embodiment. The user interface may be a graphical user interface having icons representing the digital works and the various transactions that may be performed. The user interface may be a generated dialog in which a user is prompted for information.

The user interface itself need not be part of the repository. As a repository may be embedded in some other device, the user interface may merely be a part of the device in which the repository is embedded. For example, the repository could be embedded in a "card" that is inserted into an available slot in a computer system. The user interface may be combination of a display, keyboard, cursor control device and software executing on the computer system.

At a minimum, the user interface must permit a user to input information such as access requests and alpha numeric data and provide feedback as to transaction status. The user interface will then cause the repository to initiate the suitable transactions to service the request. Other facets of a particular user interface will depend on the functionality that a repository will provide.

Credit Servers

In the present invention, fees may be associated with the exercise of a right. The requirement for payment of fees is described with each version of a usage right in the usage rights language. The recording and reporting of such fees is performed by the credit server. One of the capabilities enabled by associating fees with rights is the possibility of supporting a wide range of charging models. The simplest model, used by conventional software, is that there is a single fee at the time of purchase, after which the purchaser obtains unlimited rights to use the work as often and for as long as he or she wants. Alternative models, include metered use and variable fees. A single work can have different fees for different uses. For example, viewing a photograph on a display could have different fees than making a hardcopy or including it in a newly created work. A key to these alternative charging

US 7,523,072 B2

17

models is to have a low overhead means of establishing fees and accounting for credit on these transactions.

A credit server is a computational system that reliably authorizes and records these transactions so that fees are billed and paid. The credit server reports fees to a billing clearinghouse. The billing clearinghouse manages the financial transactions as they occur. As a result, bills may be generated and accounts reconciled. Preferably, the credit server would store the fee transactions and periodically communicate via a network with billing clearinghouse for reconciliation. In such an embodiment, communications with the billing clearinghouse would be encrypted for integrity and security reasons. In another embodiment, the credit server acts as a "debit card" where transactions occur in "real-time" against a user account.

A credit server is comprised of memory, a processing means, a clock, and interface means for coupling to a repository and a financial institution (e.g. a modem). The credit server will also need to have security and authentication functionality. These elements are essentially the same elements as those of a repository. Thus, a single device can be both a repository and a credit server, provided that it has the appropriate processing elements for carrying out the corresponding functions and protocols. Typically, however, a credit server would be a card-sized system in the possession of the owner of the credit. The credit server is coupled to a repository and would interact via financial transactions as described below. Interactions with a financial institution may occur via protocols established by the financial institutions themselves.

In the currently preferred embodiment credit servers associated with both the server and the repository report the financial transaction to the billing clearinghouse. For example, when a digital work is copied by one repository to another for a fee, credit servers coupled to each of the repositories will report the transaction to the billing clearinghouse. This is desirable in that it insures that a transaction will be accounted for in the event of some break in the communication between a credit server and the billing clearinghouse. However, some implementations may embody only a single credit server reporting the transaction to minimize transaction processing at the risk of losing some transactions.

Usage Rights Language

The present invention uses statements in a high level "usage rights language" to define rights associated with digital works and their parts. Usage rights statements are interpreted by repositories and are used to determine what transactions can be successfully carried out for a digital work and also to determine parameters for those transactions. For example, sentences in the language determine whether a given digital work can be copied, when and how it can be used, and what fees (if any) are to be charged for that use. Once the usage rights statements are generated, they are encoded in a suitable form for accessing during the processing of transactions.

Defining usage rights in terms of a language in combination with the hierarchical representation of a digital work enables the support of a wide variety of distribution and fee schemes. An example is the ability to attach multiple versions of a right to a work. So a creator may attach a PRINT right to make 5 copies for \$10.00 and a PRINT right to make unlimited copies for \$100.00. A purchaser may then choose which option best fits his needs. Another example is that rights and fees are additive. So in the case of a composite work, the rights and fees of each of the components works is used in determining the rights and fees for the work as a whole. Other

18

features and benefits of the usage rights language will become apparent in the description of distribution and use scenarios provided below.

The basic contents of a right are illustrated in FIG. 14. Referring to FIG. 14, a right 1450 has a transactional component 1451 and a specifications component 1452. A right 1450 has a label (e.g. COPY or PRINT) which indicate the use or distribution privileges that are embodied by the right. The transactional component 1451 corresponds to a particular way in which a digital work may be used or distributed. The transactional component 1451 is typically embodied in software instructions in a repository which implement the use or distribution privileges for the right. The specifications components 1452 are used to specify conditions which must be satisfied prior to the right being exercised or to designate various transaction related parameters. In the currently preferred embodiment, these specifications include copy count 1453, Fees and Incentives 1454, Time 1455, Access and Security 1456 and Control 1457. Each of these specifications will be described in greater detail below with respect to the language grammar elements.

The usage rights language is based on the grammar described below. A grammar is a convenient means for defining valid sequence of symbols for a language. In describing the grammar the notation "[abc]" is used to indicate distinct choices among alternatives. In this example, a sentence can have either an "a", "b" or "c". It must include exactly one of them. The braces { } are used to indicate optional items. Note that brackets, bars and braces are used to describe the language of usage rights sentences but do not appear in actual sentences in the language.

In contrast, parentheses are part of the usage rights language. Parentheses are used to group items together in lists. The notation (x*) is used to indicate a variable length list, that is, a list containing one or more items of type x. The notation (x)* is used to indicate a variable number of lists containing x.

Keywords in the grammar are words followed by colons. Keywords are a common and very special case in the language. They are often used to indicate a single value, typically an identifier. In many cases, the keyword and the parameter are entirely optional. When a keyword is given, it often takes a single identifier as its value. In some cases, the keyword takes a list of identifiers.

In the usage rights language, time is specified in an hours:minutes:seconds (or hh:mm:ss) representation. Time zone indicators, e.g. PDT for Pacific Daylight Time, may also be specified. Dates are represented as year/month/day (or YYYY/MMM/DD). Note that these time and date representations may specify moments in time or units of time Money units are specified in terms of dollars.

Finally, in the usage rights language, various "things" will need to interact with each other. For example, an instance of a usage right may specify a bank account, a digital ticket, etc. Such things need to be identified and are specified herein using the suffix "-ID."

The Usage Rights Grammar is listed in its entirety in FIG. 15 and is described below.

Grammar element 1501 "Digital Work Rights:=(Rights*)" define the digital work rights as a set of rights. The-set of rights attached to a digital work define how that digital work may be transferred, used, performed or played. A set of rights will attach to the entire digital work and in the case of compound digital works, each of the components of the digital work. The usage rights of components of a digital may be different.

Grammar element 1502 "Right:=(Right-Code {Copy-Count} {Control-Spec} {Time-Spec} {Access-Spec} {Fee-

US 7,523,072 B2

19

Spec})” enumerates the content of a right. Each usage right must specify a right code. Each right may also optionally specify conditions which must be satisfied before the right can be exercised. These conditions are copy count, control, time, access and fee conditions. In the currently preferred embodiment, for the optional elements, the following defaults apply: copy count equals 1, no time limit on the use of the right, no access tests or a security level required to use the right and no fee is required. These conditions will each be described in greater detail below.

It is important to note that a digital work may have multiple versions of a right, each having the same right code. The multiple versions would provide alternative conditions and fees for accessing the digital work.

A Grammar element **1503** “Right-Code:=Render-Code|Transport—Code|File-Management-Code|Derivative-Works-Code Configuration-Code” distinguishes each of the specific rights into a particular right type (although each right is identified by distinct right codes). In this way, the grammar provides a catalog of possible rights that can be associated with parts of digital works. In the following, rights are divided into categories for convenience in describing them.

Grammar element **1504** “Render-Code:=[Play: {Player: Player-ID} |Print: {Printer: Printer-ID}]” lists a category of rights all involving the making of ephemeral, transitory, or non-digital copies of the digital work. After use the copies are erased.

Play: A process of rendering or performing a digital work on some processor. This includes such things as playing digital movies, playing digital music, playing a video game, running a computer program, or displaying a document on a display.

Print: To render the work in a medium that is not further protected by usage rights, such as printing on paper.

Grammar element **1505** “Transport-Code:=[Copy|Transfer|Loan {Remaining-Rights: Next-Set-of-Rights}]{(Next-Copy-Rights: Next-Set of Rights)}” lists a category of rights involving the making of persistent, usable copies of the digital work on other repositories. The optional Next-Copy-Rights determine the rights on the work after it is transported. If this is not specified, then the rights on the transported copy are the same as on the original. The optional Remaining-Rights specify the rights that remain with a digital work when it is loaned out. If this is not specified, then the default is that no rights can be exercised when it is loaned out.

Copy: Make a new copy of a work

Transfer: Moving a work from one repository to another.

Loan: Temporarily loaning a copy to another repository for a specified period of time.

Grammar element **1506** “File-Management-Code:=Backup {Back-Up-Copy-Rights: Next-Set-of Rights} |Restore|Delete|Folder|Directory {Name:Hide-Local|Hide-Remote} {Parts:Hide-Local|Hide-Remote}” lists a category of rights involving operations for file management, such as the making of backup copies to protect the copy owner against catastrophic equipment failure.

Many software licenses and also copyright law give a copy owner the right to make backup copies to protect against catastrophic failure of equipment. However, the making of uncontrolled backup copies is inherently at odds with the ability to control usage, since an uncontrolled backup copy can be kept and then restored even after the authorized copy was sold.

The File management rights enable the making and restoring of backup copies in a way that respects usage rights, honoring the requirements of both the copy owner and the rights grantor and revenue owner. Backup copies of work

20

descriptions (including usage rights and fee data) can be sent under appropriate protocol and usage rights control to other document repositories of sufficiently high security. Further rights permit organization of digital works into folders which themselves are treated as digital works and whose contents may be “hidden” from a party seeking to determine the contents of a repository.

Backup: To make a backup copy of a digital work as protection against media failure.

Restore: To restore a backup copy of a digital work.

Delete: To delete or erase a copy of a digital work.

Folder: To create and name folders, and to move files and folders between folders.

Directory: To hide a folder or it’s contents.

Grammar element **1507** “Derivative-Works-Code: [Extract|Embed|Edit {Process: Process-ID}] {Next-Copy-Rights: Next-Set-of Rights}” lists a category of rights involving the use of a digital work to create new works.

Extract: To remove a portion of a work, for the purposes of creating a new work.

Embed: To include a work in an existing work.

Edit: To alter a digital work by copying, selecting and modifying portions of an existing digital work.

Grammar element **1508** “Configuration-Code:=Install|Uninstall” lists a category of rights for installing and uninstalling software on a repository (typically a rendering repository.) This would typically occur for the installation of a new type of player within the rendering repository.

Install: To install new software on a repository.

Uninstall: To remove existing software from a repository.

Grammar element **1509** “Next-Set-of-Rights:={ (Add: Set-Of-Rights) } { (Delete: Set-Of-Rights) } { (Replace: Set-Of-Rights) } { (Keep: Set-Of-Rights) }” defines how rights are carried forward for a copy of a digital work. If the Next-Copy-Rights is not specified, the rights for the next copy are the same as those of the current copy. Otherwise, the set of rights for the next copy can be specified. Versions of rights after Add: are added to the current set of rights. Rights after Delete: are deleted from the current set of rights. If only right codes are listed after Delete: then all versions of rights with those codes are deleted. Versions of rights after Replace: subsume all versions of rights of the same type in the current set of rights.

If Remaining-Rights is not specified, then there are no rights for the original after all Loan copies are loaned out. If Remaining-Rights is specified, then the Keep: token can be used to simplify the expression of what rights to keep behind. A list of right codes following keep means that all of the versions of those listed rights are kept in the remaining copy. This specification can be overridden by subsequent Delete: or Replace: specifications.

Copy Count Specification

For various transactions, it may be desirable to provide some limit as to the number of “copies” of the work which may be exercised simultaneously for the right. For example, it may be desirable to limit the number of copies of a digital work that may be loaned out at a time or viewed at a time.

Grammar element **1510** “Copy-Count:=(Copies: positive-integer |0| unlimited)” provides a condition which defines the number of “copies” of a work subject to the right. A copy count can be 0, a fixed number, or unlimited. The copy-count is associated with each right, as opposed to there being just a single copy-count for the digital work. The Copy-Count for a right is decremented each time that a right is exercised. When the Copy-Count equals zero, the right can no longer be exercised. If the Copy-Count is not specified, the default is one.

US 7,523,072 B2

21

Control Specification

Rights and fees depend in general on rights granted by the creator as well as further restrictions imposed by later distributors. Control specifications deal with interactions between the creators and their distributors governing the imposition of further restrictions and fees. For example, a distributor of a digital work may not want an end consumer of a digital work to add fees or otherwise profit by commercially exploiting the purchased digital work.

Grammar element **1511** “Control-Spec:=(Control: {Restrictable|Unrestrictable} {Unchargeable|Chargeable}-)” provides a condition to specify the effect of usage rights and fees of parents on the exercise of the right. A digital work is restrictable if higher level d-blocks can impose further restrictions (time specifications and access specifications) on the right. It is unrestrictable if no further restrictions can be imposed. The default setting is restrictable. A right is unchargeable if no more fees can be imposed on the use of the right. It is chargeable if more fees can be imposed. The default is chargeable.

Time Specification

It is often desirable to assign a start date or specify some duration as to when a right may be exercised. Grammar element **1512** “Time-Spec:=({Fixed-Interval|Sliding-Interval|Meter-Time} Until: Expiration-Date)” provides for specification of time conditions on the exercise of a right. Rights may be granted for a specified time. Different kinds of time specifications are appropriate for different kinds of rights. Some rights may be exercised during a fixed and predetermined duration. Some rights may be exercised for an interval that starts the first time that the right is invoked by some transaction. Some rights may be exercised or are charged according to some kind of metered time, which may be split into separate intervals. For example, a right to view a picture for an hour might be split into six ten minute viewings or four fifteen minute viewings or twenty three minute viewings.

The terms “time” and “date” are used synonymously to refer to a moment in time. There are several kinds of time specifications. Each specification represents some limitation on the times over which the usage right applies. The Expiration-Date specifies the moment at which the usage right ends. For example, if the Expiration-Date is “Jan. 1, 1995,” then the right ends at the first moment of 1995. If the Expiration-Date is specified as *forever*, then the rights are interpreted as continuing without end. If only an expiration date is given, then the right can be exercised as often as desired until the expiration date.

Grammar element **1513** “Fixed-Interval:=From: Start-Time” is used to define a predetermined interval that runs from the start time to the expiration date.

Grammar element **1514** “Sliding-Interval:=Interval: Use-Duration” is used to define an indeterminate (or “open”) start time. It sets limits on a continuous period of time over which the contents are accessible. The period starts on the first access and ends after the duration has passed or the expiration date is reached, whichever comes first. For example, if the right gives 10 hours of continuous access, the use-duration would begin when the first access was made and end 10 hours later.

Grammar element **1515** “Meter-Time:=Time-Remaining: Remaining-Use” is used to define a “meter time,” that is, a measure of the time that the right is actually exercised. It differs from the Sliding-Interval specification in that the time that the digital work is in use need not be continuous. For example, if the rights guarantee three days of access, those

22

days could be spread out over a month. With this specification, the rights can be exercised until the meter time is exhausted or the expiration date is reached, whichever comes first.

Remaining-Use:=Time-Unit
Start-Time:=Time-Unit
Use-Duration:=Time-Unit

All of the time specifications include time-unit specifications in their ultimate instantiation.

Security Class and Authorization Specification

The present invention provides for various security mechanisms to be introduced into a distribution or use scheme. Grammar element **1516** “Access-Spec:=({SC: Security-Class} {Authorization: Authorization-ID*} {Other-Authorization: Authorization-ID*} {Ticket: Ticket-ID})” provides a means for restricting access and transmission. Access specifications can specify a required security class for a repository to exercise a right or a required authorization test that must be satisfied.

The keyword “SC:” is used to specify a minimum security level for the repositories involved in the access. If “SC:” is not specified, the lowest security level is acceptable.

The optional “Authorization:” keyword is used to specify required authorizations on the same repository as the work. The optional “Other-Authorization:” keyword is used to specify required authorizations on the other repository in the transaction.

The optional “Ticket:” keyword specifies the identity of a ticket required for the transaction. A transaction involving digital tickets must locate an appropriate digital ticket agent who can “punch” or otherwise validate the ticket before the transaction can proceed. Tickets are described in greater detail below.

In a transaction involving a repository and a document server, some usage rights may require that the repository have a particular authorization, that the server have some authorization, or that both repositories have (possibly different) authorizations. Authorizations themselves are digital works (hereinafter referred to as an authorization object) that can be moved between repositories in the same manner as other digital works. Their copying and transferring is subject to the same rights and fees as other digital works. A repository is said to have an authorization if that authorization object is contained within the repository.

In some cases, an authorization may be required from a source other than the document server and repository. An authorization object referenced by an Authorization-ID can contain digital address information to be used to set up a communications link between a repository and the authorization source. These are analogous to phone numbers. For such access tests, the communication would need to be established and authorization obtained before the right could be exercised.

For one-time usage rights, a variant on this scheme is to have a digital ticket. A ticket is presented to a digital ticket agent, whose type is specified on the ticket. In the simplest case, a certified generic ticket agent, available on all repositories, is available to “punch” the ticket. In other cases, the ticket may contain addressing information for locating a “special” ticket agent. Once a ticket has been punched, it cannot be used again for the same kind of transaction (unless it is unpunched or refreshed in the manner described below.) Punching includes marking the ticket with a timestamp of the date and time it was used. Tickets are digital works and can be copied or transferred between repositories according to their usage rights.

US 7,523,072 B2

23

In the currently preferred embodiment, a “punched” ticket becomes “unpunched” or “refreshed” when it is copied or extracted. The Copy and Extract operations save the date and time as a property of the digital ticket. When a ticket agent is given a ticket, it can simply check whether the digital copy was made after the last time that it was punched. Of course, the digital ticket must have the copy or extract usage rights attached thereto.

The capability to unpunch a ticket is important in the following cases:

A digital work is circulated at low cost with a limitation that it can be used only once.

A digital work is circulated with a ticket that can be used once to give discounts on purchases of other works.

A digital work is circulated with a ticket (included in the purchase price and possibly embedded in the work) that can be used for a future upgrade.

In each of these cases, if a paid copy is made of the digital work (including the ticket) the new owner would expect to get a fresh (unpunched) ticket, whether the copy seller has used the work or not. In contrast, loaning a work or simply transferring it to another repository should not revitalize the ticket.

Usage Fees and Incentives Specification

The billing for use of a digital work is fundamental to a commercial distribution system. Grammar Element 1517 “Fee-Spec:={Scheduled-Discount} Regular-Fee-Spec|Scheduled-Fee-Spec|Markup-Spec” provides a range of options for billing for the use of digital works.

A key feature of this approach is the development of low-overhead billing for transactions in potentially small amounts. Thus, it becomes feasible to collect fees of only a few cents each for thousands of transactions.

The grammar differentiates between uses where the charge is per use from those where it is metered by the time unit. Transactions can support fees that the user pays for using a digital work as well as incentives paid by the right grantor to users to induce them to use or distribute the digital work.

The optional scheduled discount refers to the rest of the fee specification—discounting it by a percentage over time. If it is not specified, then there is no scheduled discount. Regular fee specifications are constant over time. Scheduled fee specifications give a schedule of dates over which the fee specifications change. Markup specifications are used in d-blocks for adding a percentage to the fees already being charged.

Grammar Element 1518 “Scheduled-Discount:=(Scheduled-Discount: (Time-Spec Percentage)*)” A Scheduled-Discount is essentially a scheduled modifier of any other fee specification for this version of the right of the digital work. (It does not refer to children or parent digital works or to other versions of rights.). It is a list of pairs of times and percentages. The most recent time in the list that has not yet passed at the time of the transaction is the one in effect. The percentage gives the discount percentage. For example, the number 10 refers to a 10% discount.

Grammar Element 1519 “Regular-Fee-Spec:={Fee: |Incentive:} [Per-Use-Spec|Metered-Rate-Spec|Best-Price-Spec|Call-For-Price-Spec] {Min: Money-Unit Per: Time-Spec} {Max: Money-Unit Per: Time-Spec} To: Account-ID)” provides for several kinds of fee specifications.

Fees are paid by the copy-owner/user to the revenue-owner if Fee: is specified. Incentives are paid by the revenue-owner to the user if Incentive: is specified. If the Min: specification is given, then there is a minimum fee to be charged per time-spec unit for its use. If the Max: specification is given, then there is a maximum fee to be charged per time-spec for its use. When Fee: is specified, Account-ID identifies the

24

account to which the fee is to be paid. When Incentive: is specified, Account-ID identifies the account from which the fee is to be paid.

Grammar element 1520 “Per-Use-Spec:=Per-Use: Money-unit” defines a simple fee to be paid every time the right is exercised, regardless of how much time the transaction takes.

Grammar element 1521 “Metered-Rate-Spec:=Metered: Money-Unit Per: Time-Spec” defines a metered-rate fee paid according to how long the right is exercised. Thus, the time it takes to complete the transaction determines the fee.

Grammar, element 1522 “Best-Price-Spec:=Best-Price: Money-unit Max: Money-unit” is used to specify a best-price that is determined when the account is settled. This specification is to accommodate special deals, rebates, and pricing that depends on information that is not available to the repository. All fee specifications can be combined with tickets or authorizations that could indicate that the consumer is a wholesaler or that he is a preferred customer, or that the seller be authorized in some way. The amount of money in the Max: field is the maximum amount that the use will cost. This is the amount that is tentatively debited from the credit server. However, when the transaction is ultimately reconciled, any excess amount will be returned to the consumer in a separate transaction.

Grammar element 1523 “Call-For-Price-Spec:=Call-For-Price” is similar to a “Best-Price-Spec” in that it is intended to accommodate cases where prices are dynamic. A Call-For-Price Spec requires a communication with a dealer to determine the price. This option cannot be exercised if the repository cannot communicate with a dealer at the time that the right is exercised. It is based on a secure transaction whereby the dealer names a price to exercise the right and passes along a deal certificate which is referenced or included in the billing process.

Grammar element 1524 “Scheduled-Fee-Spec:=(Schedule: (Time-Spec Regular-Fee-Spec)*)” is used to provide a schedule of dates over which the fee specifications change. The fee specification with the most recent date not in the future is the one that is in effect. This is similar to but more general than the scheduled discount. It is more general, because it provides a means to vary the fee agreement for each time period.

Grammar element 1525 “Markup-Spec:=Markup: percentage To: Account-ID” is provided for adding a percentage to the fees already being charged. For example, a 5% markup means that a fee of 5% of cumulative fee so far will be allocated to the distributor. A markup specification can be applied to all of the other kinds of fee specifications. It is typically used in a shell provided by a distributor. It refers to fees associated with d-blocks that are parts of the current d-block. This might be a convenient specification for use in taxes, or in distributor overhead.

55 Examples of Sets of Usage Rights

((Play) (Transfer (SC: 3)) (Delete))

This work can be played without requirements for fee or authorization on any rendering system. It can be transferred to any other repository of security level 3 or greater. It can be deleted.

((Play) (Transfer (SC: 3)) (Delete) (Backup) (Restore (Fee: Per-Use: \$5 To: Account-ID-678)))

Same as the previous example plus rights for backup and restore. The work can be backed up without fee. It can be restored for a \$5 fee payable to the account described by Account-ID-678.

US 7,523,072 B2

25

((Play) (Transfer (SC: 3))
 (Copy (SC:3)(Fee: Per-Use: \$5 To: Account-ID-678))
 (Delete (Incentive: Per-Use: \$2.50 To: Account-ID-678)))
 This work can be played, transferred, copied, or deleted.
 Copy or transfer operations can take place only with repositories of security level three or greater. The fee to make a copy is \$5 payable to Account-ID-678. If a copy is deleted, then an incentive of \$2.50 is paid to the former copy owner.

((Play) (Transfer (SC: 3))
 Copy (SC: 3) (Fee: Per-Use: \$10 To: Account-ID-678))
 Delete) (Backup) (Restore (SC: 3) (Fee: Per-Use: \$5 To: Account-ID-678)))

Same as the previous example plus fees for copying. The work can be copied digitally for a fee of \$10 payable to Account-ID-678. The repository on which the work is copied or restored must be at security level 3 or greater.

((Play) (Transfer (SC: 3))
 (Copy Authorization: License-123-ID (SC: 3)))

The digital work can be played, transferred, or copied. Copies or transfers must be on repositories of security level 3 or greater. Copying requires the license License-123-ID issued to the copying repository. None of the rights require fees.

((Play) (Print Printer: Printer-567-ID (Fee: Per-Use: \$1 To: Account-ID-678)))

This work can be played for free. It can be printed on any printer with the identifier Printer-567-ID for a fee of \$1 payable to the account described by Account-ID-678.

((Play Player: Player-876-ID) (From: Feb. 2, 1994 Until: Feb. 15, 1995) (Fee: Metered: \$0.01 Per: 0:1:0 Min: \$0.25 Per: 0/1/0 To: Account-ID-567))

This work can be played on any player holding the ID Player-876-ID. The time of this right is from Feb. 14, 1994 until Feb. 15, 1995. The fee for use is one cent per minute with a minimum of 25 cents in any day that it is used, payable to the account described by Account-ID-567.

((Play) (Transfer) (Delete)(Loan 2 (Delete: Transfer Loan)))

This work can be played, transferred, deleted, or loaned. Up to two copies can be loaned out at a time. The loaned copy has the same rights except that it cannot be transferred. When both copies are loaned out, no rights can be exercised on the original on the repository.

((Play) (Transfer) (Delete) (Backup) (Restore (SC:3))
 (Loan 2 Remaining-Copy-Rights: (Delete: Play Transfer)
 Next-Set-of-Rights: (Delete: Transfer Loan)))

Similar to previous example. Rights to Backup and Restore the work are added, where restoration requires a repository of at least security level three. When all copies of the work are loaned out, the remaining copy cannot be played or transferred.

((Play) (Transfer) (Copy) (Print) (Backup) (Restore (SC: 3)))

(Loan 1 Remaining-Copy-Rights: (Add: Play Print Backup)

Next-Set-of-Rights: (Delete: Transfer Loan)
 (Fee: Metered: \$10 Per: 1:0:0 To: Account-ID-567))

(Loan 1 Remaining-Copy-Rights:
 Add: ((Play Player: Player-876-ID) 2 (From: Feb. 14, 1994 Until: Feb. 15, 1995)

(Fee: Metered: \$0.01 Per: 0:1:0 Min: \$0.25 Per: 0/1/0 To: Account-ID-567)))

The original work has rights to Play, Transfer, Copy, Print, Backup, Restore, and Loan. There are two versions of the Loan right. The first version of the loan right costs \$10 per day but allows the original copy owner to exercise free use of the Play, Print and Backup rights. The second version of the Loan

26

right is free. None of the original rights are applicable. However a right to Play the work at the specified metered rate is added.

((Play Player: Player-Small-Screen-123-ID)
 (Embed (Fee: Per-Use \$0.01 To: Account-678-ID))
 (Copy (Fee: Per-Use \$1.00 To: Account-678-ID)))

The digital work can be played on any player with the identifier Player-Small-Screen-123-ID. It can be embedded in a larger work. The embedding requires a modest one cent registration fee to Account-678-ID. Digital copies can be made for \$1.00.

Repository Transactions

When a user requests access to a digital work, the repository will initiate various transactions. The combination of transactions invoked will depend on the specifications assigned for a usage right. There are three basic types of transactions, Session Initiation Transactions, Financial Transactions and Usage Transactions. Generally, session initiation transactions are initiated first to establish a valid session. When a valid session is established, transactions corresponding to the various usage rights are invoked. Finally, request specific transactions are performed.

Transactions occur between two repositories (one acting as a server), between a repository and a document playback platform (e.g. for executing or viewing), between a repository and a credit server or between a repository and an authorization server. When transactions occur between more than one repository, it is assumed that there is a reliable communication channel between the repositories. For example, this could be a TCP/IP channel or any other commercially available channel that has built-in capabilities for detecting and correcting transmission errors. However, it is not assumed that the communication channel is secure. Provisions for security and privacy are part of the requirements for specifying and implementing repositories and thus form the need for various transactions.

Message Transmission

Transactions require that there be some communication between repositories. Communication between repositories occurs in units termed as messages. Because the communication line is assumed to be unsecure, all communications with repositories that are above the lowest security class are encrypted utilizing a public key encryption technique. Public key encryption is a well known technique in the encryption arts. The term key refers to a numeric code that is used with encryption and decryption algorithms. Keys come in pairs, where "writing keys" are used to encrypt data and "checking keys" are used to decrypt data. Both writing and checking keys may be public or private. Public keys are those that are distributed to others. Private keys are maintained in confidence.

Key management and security is instrumental in the success of a public key encryption system. In the currently preferred embodiment, one or more master repositories maintain the keys and create the identification certificates used by the repositories.

When a sending repository transmits a message to a receiving repository, the sending repository encrypts all of its data using the public writing key of the receiving repository. The sending repository includes its name, the name of the receiving repository, a session identifier such as a nonce (described below), and a message counter in each message.

In this way, the communication can only be read (to a high probability) by the receiving repository, which holds the pri-

US 7,523,072 B2

27

vate checking key for decryption. The auxiliary data is used to guard against various replay attacks to security. If messages ever arrive with the wrong counter or an old nonce, the repositories can assume that someone is interfering with communication and the transaction terminated.

The respective public keys for the repositories to be used for encryption are obtained in the registration transaction described below.

Session Initiation Transactions

A usage transaction is carried out in a session between repositories. For usage transactions involving more than one repository, or for financial transactions between a repository and a credit server, a registration transaction is performed. A second transaction termed a login transaction, may also be needed to initiate the session. The goal of the registration transaction is to establish a secure channel between two repositories who know each others identities. As it is assumed that the communication channel between the repositories is reliable but not secure, there is a risk that a non-repository may mimic the protocol in order to gain illegitimate access to a repository.

The registration transaction between two repositories is described with respect to FIGS. 16 and 17. The steps described are from the perspective of a "repository-1" registering its identity with a "repository-2". The registration must be symmetrical so the same set of steps will be repeated for repository-2 registering its identity with repository-1. Referring to FIG. 16, repository-1 first generates an encrypted registration identifier, step 1601 and then generates a registration message, step 1602. A registration message is comprised of an identifier of a master repository, the identification certificate for the repository-1 and an encrypted random registration identifier. The identification certificate is encrypted by the master repository in its private key and attests to the fact that the repository (here repository-1) is a bona fide repository. The identification certificate also contains a public key for the repository, the repository security level and a timestamp (indicating a time after which the certificate is no longer valid.) The registration identifier is a number generated by the repository for this registration. The registration identifier is unique to the session and is encrypted in repository-1's private key. The registration identifier is used to improve security of authentication by detecting certain kinds of communications based attacks. Repository-1 then transmits the registration message to repository-2, step 1603.

Upon receiving the registration message, repository-2 determines if it has the needed public key for the master repository, step 1604. If repository-2 does not have the needed public key to decrypt the identification certificate, the registration transaction terminates in an error, step 1618.

Assuming that repository-2 has the proper public key the identification certificate is decrypted, step 1605. Repository-2 saves the encrypted registration identifier, step 1606, and extracts the repository identifier, step 1607. The extracted repository identifier is checked against a "hotlist" of compromised document repositories, step 1608. In the currently preferred embodiment, each repository will contain "hotlists" of compromised repositories. If the repository is on the "hotlist", the registration transaction terminates in an error per step 1618. Repositories can be removed from the hotlist when their certificates expire, so that the list does not need to grow without bound. Also, by keeping a short list of hotlist certificates that it has previously received, a repository can avoid the work of actually going through the list. These lists would be encrypted by a master repository. A minor variation on the approach to improve efficiency would have the repositories

28

first exchange lists of names of hotlist certificates, ultimately exchanging only those lists that they had not previously received. The "hotlists" are maintained and distributed by Master repositories.

Note that rather than terminating in error, the transaction could request that another registration message be sent based on an identification certificate created by another master repository. This may be repeated until a satisfactory identification certificate is found, or it is determined that trust cannot be established.

Assuming that the repository is not on the hotlist, the repository identification needs to be verified. In other words, repository-2 needs to validate that the repository on the other end is really repository-1. This is termed performance testing and is performed in order to avoid invalid access to the repository via a counterfeit repository replaying a recording of a prior session initiation between repository-1 and repository-2. Performance testing is initiated by repository-2 generating a performance message, step 1609. The performance message consists of a nonce, the names of the respective repositories, the time and the registration identifier received from repository-1. A nonce is a generated message based on some random and variable information (e.g. the time or the temperature.) The nonce is used to check whether repository-1 can actually exhibit correct encrypting of a message using the private keys it claims to have, on a message that it has never seen before. The performance message is encrypted using the public key specified in the registration message of repository-1. The performance message is transmitted to repository-1, step 1610, where it is decrypted by repository-1 using its private key, step 1611. Repository-1 then checks to make sure that the names of the two repositories are correct, step 1612, that the time is accurate, step 1613 and that the registration identifier corresponds to the one it sent, step 1614. If any of these tests fails, the transaction is terminated per step 1616. Assuming that the tests are passed, repository-1 transmits the nonce to repository-2 in the clear, step 1615. Repository-2 then compares the received nonce to the original nonce, step 1617. If they are not identical, the registration transaction terminates in an error per step 1618. If they are the same, the registration transaction has successfully completed.

At this point, assuming that the transaction has not terminated, the repositories exchange messages containing session keys to be used in all communications during the session and synchronize their clocks. FIG. 17 illustrates the session information exchange and clock synchronization steps (again from the perspective of repository-1.) Referring to FIG. 17, repository-1 creates a session key pair, step 1701. A first key is kept private and is used by repository-1 to encrypt messages. The second key is a public key used by repository-2 to decrypt messages. The second key is encrypted using the public key of repository-2, step 1702 and is sent to repository-2, step 1703. Upon receipt, repository-2 decrypts the second key, step 1704. The second key is used to decrypt messages in subsequent communications. When each repository has completed this step, they are both convinced that the other repository is bona fide and that they are communicating with the original. Each repository has given the other a key to be used in decrypting further communications during the session. Since that key is itself transmitted in the public key of the receiving repository only it will be able to decrypt the key which is used to decrypt subsequent messages.

After the session information is exchanged, the repositories must synchronize their clocks. Clock synchronization is used by the repositories to establish an agreed upon time base for the financial records of their mutual transactions. Referring back to FIG. 17, repository-2 initiates clock synchroni-

US 7,523,072 B2

29

zation by generating a time stamp exchange message, step 1705, and transmits it to repository-1, step 1706. Upon receipt, repository-1 generates its own time stamp message, step 1707 and transmits it back to repository-2, step 1708. Repository-2 notes the current time, step 1709 and stores the time received from repository-1, step 1710. The current time is compared to the time received from repository-1, step 1711. The difference is then checked to see if it exceeds a predetermined tolerance (e.g. one minute), step 1712. If it does, repository-2 terminates the transaction as this may indicate tampering with the repository, step 1713. If not repository-2 computes an adjusted time delta, step 1714. The adjusted time delta is the difference between the clock time of repository-2 and the average of the times from repository-1 and repository-2.

To achieve greater accuracy, repository-2 can request the time again up to a fixed number of times (e.g. five times), repeat the clock synchronization steps, and average the results.

A second session initiation transaction is a Login transaction. The Login transaction is used to check the authenticity of a user requesting a transaction. A Login transaction is particularly prudent for the authorization of financial transactions that will be charged to a credit server. The Login transaction involves an interaction between the user at a user interface and the credit server associated with a repository. The information exchanged here is a login string supplied by the repository/credit server to identify itself to the user, and a Personal Identification Number (PIN) provided by the user to identify himself to the credit server. In the event that the user is accessing a credit server on a repository different from the one on which the user interface resides, exchange of the information would be encrypted using the public and private keys of the respective repositories.

Billing Transactions

Billing Transactions are concerned with monetary transaction with a credit server. Billing Transactions are carried out when all other conditions are satisfied and a usage fee is required for granting the request. For the most part, billing transactions are well understood in the state of the art. These transactions are between a repository and a credit server, or between a credit server and a billing clearinghouse. Briefly, the required transactions include the following:

Registration and LOGIN transactions, by which the repository and user establish their bona fides to a credit server. These transactions would be entirely internal in cases where the repository and credit server are implemented as a single system.

Registration and LOGIN transactions, by which a credit server establishes its bona fides to a billing clearinghouse.

An Assign-fee transaction to assign a charge. The information in this transaction would include a transaction identifier, the identities of the repositories in the transaction, and a list of charges from the parts of the digital work. If there has been any unusual event in the transaction such as an interruption of communications, that information is included as well.

A Begin-charges transaction to assign a charge. This transaction is much the same as an assign-fee transaction except that it is used for metered use. It includes the same information as the assign-fee 4, ii transaction as well as the usage fee information. The credit-server is then responsible for running a clock.

30

An End-charges transaction to end a charge for metered use. (In a variation on this approach, the repositories would exchange periodic charge information for each block of time.)

5 A report-charges transaction between a personal credit server and a billing clearinghouse. This transaction is invoked at least once per billing period. It is used to pass along information about charges. On debit and credit cards, this transaction would also be used to update balance information and credit limits as needed.

10 All billing transactions are given a transaction ID and are reported to the credit servers by both the server and the client. This reduces possible loss of billing information if one of the parties to a transaction loses a banking card and provides a check against tampering with the system.

Usage Transactions

After the session initiation transactions have been completed, the usage request may then be processed. To simplify the description of the steps carried out in processing a usage request, the term requester is used to refer to a repository in the requester mode which is initiating a request, and the term server is used to refer to a repository in the server mode and which contains the desired digital work. In many cases such as requests to print or view a work, the requester and server may be the same device and the transactions described in the following would be entirely internal. In such instances, certain transaction steps, such as the registration transaction, need not be performed.

25 There are some common steps that are part of the semantics of all of the usage rights transactions. These steps are referred to as the common transaction steps. There are two sets—the “opening” steps and the “closing” steps. For simplicity, these are listed here rather than repeating them in the descriptions of all of the usage rights transactions.

30 Transactions can refer to a part of a digital work, a complete digital work, or a Digital work containing other digital works. Although not described in detail herein, a transaction may even refer to a folder comprised of a plurality of digital works. The term “work” is used to refer to what ever portion or set of digital works is being accessed.

35 Many of the steps here involve determining if certain conditions are satisfied. Recall that each usage right may have one or more conditions which must be satisfied before the right can be exercised. Digital works have parts and parts have parts. Different parts can have different rights and fees. Thus, it is necessary to verify that the requirements are met for ALL of the parts that are involved in a transaction. For brevity, when reference is made to checking whether the rights exist and conditions for exercising are satisfied, it is meant that all such checking takes place for each of the relevant parts of the work.

FIG. 18 illustrates the initial common opening and closing steps for a transaction. At this point it is assumed that registration has occurred and that a “trusted” session is in place. General tests are tests on usage rights associated with the folder containing the work or some containing folder higher in the file system hierarchy. These tests correspond to requirements imposed on the work as a consequence of its being on the particular repository, as opposed to being attached to the work itself. Referring to FIG. 18, prior to initiating a usage transaction, the requester performs any general tests that are required before the right associated with the transaction can be exercised, step, 1801. For example, install, uninstall and delete rights may be implemented to require that a requester have an authorization certificate before the right can be exercised. Another example is the requirement that a digital ticket be present and punched before a digital work may be copied

US 7,523,072 B2

31

to a requester. If any of the general tests fail, the transaction is not initiated, step **1802**. Assuming that such required tests are passed, upon receiving the usage request, the server generates a transaction identifier that is used in records or reports of the transaction, step **1803**. The server then checks whether the digital work has been granted the right corresponding to the requested transaction, step **1804**. If the digital work has not been granted the right corresponding to the request, the transaction terminates, step **1805**. If the digital work has been granted the requested right, the server then determines if the various conditions for exercising the right are satisfied. Time based conditions are examined, step **1806**. These conditions are checked by examining the time specification for the version of the right. If any of the conditions are not satisfied, the transaction terminates per step **1805**.

Assuming that the time based conditions are satisfied, the server checks security and access conditions, step **1807**. Such security and access conditions are satisfied if: 1) the requester is at the specified security class, or a higher security class, 2) the server satisfies any specified authorization test and 3) the requester satisfies any specified authorization tests and has any required digital tickets. If any of the conditions are not satisfied, the transaction terminates per step **1805**.

Assuming that the security and access conditions are all satisfied, the server checks the copy count condition, step **1808**. If the copy count equals zero, then the transaction cannot be completed and the transaction terminates per step **1805**.

Assuming that the copy count does not equal zero, the server checks if the copies in use for the requested right is greater than or equal to any copy count for the requested right (or relevant parts), step **1809**. If the copies in use are greater than or equal to the copy count, this indicates that usage rights for the version of the transaction have been exhausted. Accordingly, the server terminates the transaction, step **1805**. If the copy count is less than the copies in use for the transaction the transaction can continue, and the copies in use would be incremented by the number of digital works requested in the transaction, step **1810**.

The server then checks if the digital work has a "Loan" access right, step **1811**. The "Loan" access right is a special case since remaining rights may be present even though all copies are loaned out. If the digital work has the "Loan" access right, a check is made to see if all copies have been loaned out, step **1812**. The number of copies that could be loaned is the sum of the Copy-Counts for all of the versions of the loan right of the digital work. For a composite work, the relevant figure is the minimal such sum of each of the components of the composite work. If all copies have been loaned out, the remaining rights are determined, step **1813**. The remaining-rights are determined from the remaining rights specifications from the versions of the Loan right. If there is only one version of the Loan right, then the determination is simple. The remaining rights are the ones specified in that version of the Loan right, or none if Remaining-Rights: is not specified. If there are multiple versions of the Loan right and all copies of all of the versions are loaned out, then the remaining rights is taken as the minimum set (intersection) of remaining rights across all of the versions of the loan right. The server then determines if the requested right is in the set of remaining rights, step **1814**. If the requested right is not in the set of remaining rights, the server terminates the transaction, step **1805**.

If Loan is not a usage right for the digital work or if all copies have not been loaned out or the requested right is in the set of remaining rights, fee conditions for the right are then checked, step **1815**. This will initiate various financial trans-

32

actions between the repository and associated credit server. Further, any metering of usage of a digital work will commence. If any financial transaction fails, the transaction terminates per step **1805**.

It should be noted that the order in which the conditions are checked need not follow the order of steps **1806-1815**.

At this point, right specific steps are now performed and are represented here as step **1816**. The right specific steps are described in greater detail below.

The common closing transaction steps are now performed. Each of the closing transaction steps are performed by the server after a successful completion of a transaction. Referring back to FIG. **18**, the copies in use value for the requested right is decremented by the number of copies involved in the transaction, step **1817**. Next, if the right had a metered usage fee specification, the server subtracts the elapsed time from the Remaining-Use-Time associated with the right for every part involved in the transaction, step **1818**. Finally, if there are fee specifications associated with the right, the server initiates End-Charge financial transaction to confirm billing, step **1819**.

Transmission Protocol

An important area to consider is the transmission of the digital work from the server to the requester. The transmission protocol described herein refers to events occurring after a valid session has been created. The transmission protocol must handle the case of disruption in the communications between the repositories. It is assumed that interference such as injecting noise on the communication channel can be detected by the integrity checks (e.g., parity, checksum, etc.) that are built into the transport protocol and are not discussed in detail herein.

The underlying goal in the transmission protocol is to preclude certain failure modes, such as malicious or accidental interference on the communications channel. Suppose, for example, that a user pulls a card with the credit server at a specific time near the end of a transaction. There should not be a vulnerable time at which "pulling the card" causes the repositories to fail to correctly account for the number of copies of the work that have been created. Restated, there should be no time at which a party can break a connection as a means to avoid payment after using a digital work.

If a transaction is interrupted (and fails), both repositories restore the digital works and accounts to their state prior to the failure, modulo records of the failure itself.

FIG. **19** is a state diagram showing steps in the process of transmitting information during a transaction. Each box represents a state of a repository in either the server mode (above the central dotted line **1901**) or in the requester mode (below the dotted line **1901**). Solid arrows stand for transitions between states. Dashed arrows stand for message communications between the repositories. A dashed message arrow pointing to a solid transition arrow is interpreted as meaning that the transition takes place when the message is received. Unlabeled transition arrows take place unconditionally. Other labels on state transition arrows describe conditions that trigger the transition.

Referring now to FIG. **19**, the server is initially in a state **1902** where a new transaction is initiated via start message **1903**. This message includes transaction information including a transaction identifier and a count of the blocks of data to be transferred. The requester, initially in a wait state **1904** then enters a data wait state **1905**.

The server enters a data transmit state **1906** and transmits a block of data **1907** and then enters a wait for acknowledgement state **1908**. As the data is received, the requesters enters

US 7,523,072 B2

33

a data receive state **1909** and when the data blocks is completely received it enters an acknowledgement state **1910** and transmits an Acknowledgement message **1911** to the server.

If there are more blocks to send, the server waits until receiving an Acknowledgement message from the requester. When an Acknowledgement message is received it sends the next block to the requester and again waits for acknowledgement. The requester also repeats the same cycle of states.

If the server detects a communications failure before sending the last block, it enters a cancellation state **1912** wherein the transaction is cancelled. Similarly, if the requester detects a communications failure before receiving the last block it enters a cancellation state **1913**.

If there are no more blocks to send, the server commits to the transaction and waits for the final Acknowledgement in state **1914**. If there is a communications failure before the server receives the final Acknowledgement message, it still commits to the transaction but includes a report about the event to its credit server in state **1915**. This report serves two purposes. It will help legitimize any claims by a user of having been billed for receiving digital works that were not completely received. Also it helps to identify repositories and communications lines that have suspicious patterns of use and interruption. The server then enters its completion state

On the requester side, when there are no more blocks to receive, the requester commits to the transaction in state **1917**. If the requester detects a communications failure at this state, it reports the failure to its credit server in state **1918**, but still commits to the transaction. When it has committed, it sends an acknowledgement message to the server. The server then enters its completion state **1919**.

The key property is that both the server and the requester cancel a transaction if it is interrupted before all of the data blocks are delivered, and commits to it if all of the data blocks have been delivered.

There is a possibility that the server will have sent all of the data blocks (and committed) but the requester will not have received all of them and will cancel the transaction. In this case, both repositories will presumably detect a communications failure and report it to their credit server. This case will probably be rare since it depends on very precise timing of the communications failure. The only consequence will be that the user at the requester repository may want to request a refund from the credit services—and the case for that refund will be documented by reports by both repositories.

To prevent loss of data, the server should not delete any transferred digital work until receiving the final acknowledgement from the requester. But it also should not use the file. A well known way to deal with this situation is called “two-phase commit” or 2PC.

Two-phase commit works as follows. The first phase works the same as the method described above. The server sends all of the data to the requester. Both repositories mark the transaction (and appropriate files) as uncommitted. The server sends a ready-to-commit message to the requester. The requester sends back an acknowledgement. The server then commits and sends the requester a commit message. When the requester receives the commit message, it commits the file.

If there is a communication failure or other crash, the requester must check back with the server to determine the status of the transaction. The server has the last word on this. The requester may have received all of the data, but if it did not get the final message, it has not committed. The server can go ahead and delete files (except for transaction records) once it commits, since the files are known to have been fully transmitted before starting the 2PC cycle.

34

There are variations known in the art which can be used to achieve the same effect. For example, the server could use an additional level of encryption when transmitting a work to a client. Only after the client sends a message acknowledging receipt does it send the key. The client then agrees to pay for the digital work. The point of this variation is that it provides a clear audit trail that the client received the work. For trusted systems, however, this variation adds a level of encryption for no real gain in accountability.

The transactions for specific usage rights are now discussed.

The Copy Transaction

A Copy transaction is a request to make one or more independent copies of the work with the same or lesser usage rights. Copy differs from the extraction right discussed later in that it refers to entire digital works or entire folders containing digital works. A copy operation cannot be used to remove a portion of a digital work.

The requester sends the server a message to initiate the Copy Transaction. This message indicates the work to be copied, the version of the copy right to be used for the transaction, the destination address information (location in a folder) for placing the work, the file data for the work (including its size), and the number of copies requested.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the client according to the transmission protocol. If a Next-Set-Of-Rights has been provided in the version of the right, those rights are transmitted as the rights for the work. Otherwise, the rights of the original are transmitted. In any event, the Copy-Count field for the copy of the digital work being sent right is set to the number-of-copies requested.

The requester records the work contents, data, and usage rights and stores the work. It records the date and time that the copy was made in the properties of the digital work.

The repositories perform the common closing transaction steps.

The Transfer Transaction

A Transfer transaction is a request to move copies of the work with the same or lesser usage rights to another repository. In contrast with a copy transaction, this results in removing the work copies from the server.

The requester sends the server a message to initiate the Transfer Transaction. This message indicates the work to be transferred, the version of the transfer right to be used in the transaction, the destination address information for placing the work, the file data for the work, and the number of copies involved.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester according to the transmission protocol. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the work. Otherwise, the rights of the original are transmitted. In either case, the Copy-Count field for the transmitted rights is set to the number-of-copies requested.

The requester records the work contents, data, and usage rights and stores the work.

The server decrements its copy count by the number of copies involved in the transaction.

US 7,523,072 B2

35

The repositories perform the common closing transaction steps.

If the number of copies remaining in the server is now zero, it erases the digital work from its memory.

The Loan Transaction

A loan transaction is a mechanism for loaning copies of a digital work. The maximum duration of the loan is determined by an internal parameter of the digital work. Works are automatically returned after a predetermined time period.

The requester sends the server a message to initiate the Transfer Transaction. This message indicates the work to be loaned, the version of the loan right to be used in the transaction, the destination address information for placing the work, the number of copies involved, the file data for the work, and the period of the loan.

The server checks the validity of the requested loan period, and ends with an error if the period is not valid. Loans for a loaned copy cannot extend beyond the period of the original loan to the server.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester.

If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the work. Otherwise, the rights of the original are transmitted, as modified to reflect the loan period.

The requester records the digital work contents, data, usage rights, and loan period and stores the work.

The server updates the usage rights information in the digital work to reflect the number of copies loaned out.

The repositories perform the common closing transaction steps.

The server updates the usage rights data for the digital work. This may preclude use of the work until it is returned from the loan. The user on the requester platform can now use the transferred copies of the digital work. A user accessing the original repository cannot use the digital work, unless there are copies remaining. What happens next depends on the order of events in time.

Case 1. If the time of the loan period is not yet exhausted and the requester sends the repository a Return message.

The return message includes the requester identification, and the transaction ID.

The server decrements the copies-in-use field by the number of copies that were returned. (If the number of digital works returned is greater than the number actually borrowed, this is treated as an error.) This step may now make the work available at the server for other users.

The requester deactivates its copies and removes the contents from its memory.

Case 2. If the time of the loan period is exhausted and the requester has not yet sent a Return message.

The server decrements the copies-in-use field by the number of digital works that were borrowed.

The requester automatically deactivates its copies of the digital work. It terminates all current uses and erases the digital work copies from memory. One question is why a requester would ever return a work earlier than the period of the loan, since it would be returned automatically anyway. One reason for early return is that there may be a metered fee which determines the cost of the loan. Returning early may reduce that fee.

36

The Play Transaction

A play transaction is a request to use the contents of a work. Typically, to "play" a work is to send the digital work through some kind of transducer, such as a speaker or a display device.

5 The request implies the intention that the contents will not be communicated digitally to any other system. For example, they will not be sent to a printer, recorded on any digital medium, retained after the transaction or sent to another repository.

10 This term "play" is natural for examples like playing music, playing a movie, or playing a video game. The general form of play means that a "player" is used to use the digital work. However, the term play covers all media and kinds of recordings. Thus one would "play" a digital work, meaning, 15 to render it for reading, or play a computer program, meaning to execute it. For a digital ticket the player would be a digital ticket agent.

The requester sends the server a message to initiate the play transaction. This message indicates the work to be played, the version of the play right to be used in the transaction, the identity of the player being used, and the file data for the work.

20 The server checks the validity of the player identification and the compatibility of the player identification with the player specification in the right. It ends with an error if these are not satisfactory.

The repositories perform the common opening transaction steps.

25 The server and requester read and write the blocks of data as requested by the player according to the transmission protocol. The requester plays the work contents, using the player.

When the player is finished, the player and the requester remove the contents from their memory.

30 The repositories perform the common closing transaction steps.

The Print Transaction

A Print transaction is a request to obtain the contents of a work for the purpose of rendering them on a "printer." We use the term "printer" to include the common case of writing with ink on paper. However, the key aspect of "printing" in our use of the term is that it makes a copy of the digital work in a place outside of the protection of usage rights. As with all rights, this may require particular authorization certificates.

35 Once a digital work is printed, the publisher and user are bound by whatever copyright laws are in effect. However, printing moves the contents outside the control of repositories. For example, absent any other enforcement mechanisms, once a digital work is printed on paper, it can be copied on ordinary photocopying machines without intervention by a repository to collect usage fees. If the printer to a digital disk is permitted, then that digital copy is outside of the control of usage rights. Both the creator and the user know this, although the creator does not necessarily give tacit consent to such copying, which may violate copyright laws.

The requester sends the server a message to initiate a Print transaction. This message indicates the work to be played, the identity of the printer being used, the file data for the work, and the number of copies in the request.

40 The server checks the validity of the printer identification and the compatibility of the printer identification with the printer specification in the right. It ends with an error if these are not satisfactory.

The repositories perform the common opening transaction steps.

US 7,523,072 B2

37

The server transmits blocks of data according to the transmission protocol.

The requester prints the work contents, using the printer. When the printer is finished, the printer and the requester remove the contents from their memory.

The repositories perform the common closing transaction steps.

The Backup Transaction

A Backup transaction is a request to make a backup copy of a digital work, as a protection against media failure. In the context of repositories, secure backup copies differ from other copies in three ways: (1) they are made under the control of a Backup transaction rather than a Copy transaction, (2) they do not count as regular copies, and (3) they are not usable as regular copies. Generally, backup copies are encrypted.

Although backup copies may be transferred or copied, depending on their assigned rights, the only way to make them useful for playing, printing or embedding is to restore them.

The output of a Backup operation is both an encrypted data file that contains the contents and description of a work, and a restoration file with an encryption key for restoring the encrypted contents. In many cases, the encrypted data file would have rights for "printing" it to a disk outside of the protection system, relying just on its encryption for security. Such files could be stored anywhere that was physically safe and convenient. The restoration file would be held in the repository. This file is necessary for the restoration of a backup copy. It may have rights for transfer between repositories.

The requester sends the server a message to initiate a backup transaction. This message indicates the work to be backed up, the version of the backup right to be used in the transaction, the destination address information for placing the backup copy, the file data for the work.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the work. Otherwise, a set of default rights for backup files of the original are transmitted by the server.

The requester records the work contents, data, and usage rights. It then creates a one-time key and encrypts the contents file. It saves the key information in a restoration file.

The repositories perform the common closing transaction steps.

In some cases, it is convenient to be able to archive the large, encrypted contents file to secure offline storage, such as a magneto-optical storage system or magnetic tape. This creation of a non-repository archive file is as secure as the encryption process. Such non-repository archive storage is considered a form of "printing" and is controlled by a print right with a specified "archive-printer." An archive-printer device is programmed to save the encrypted contents file (but not the description file) offline in such a way that it can be retrieved.

The Restore Transaction

A Restore transaction is a request to convert an encrypted backup copy of a digital work into a usable copy. A restore operation is intended to be used to compensate for catastrophic media failure. Like all usage rights, restoration rights can include fees and access tests including authorization checks.

38

The requester sends the server a message to initiate a Restore transaction. This message indicates the work to be restored, the version of the restore right for the transaction, the destination address information for placing the work, and the file data for the work.

The server verifies that the contents file is available (i.e. a digital work corresponding to the request has been backed-up.) If it is not, it ends the transaction with an error.

The repositories perform the common opening transaction steps.

The server retrieves the key from the restoration file. It decrypts the work contents, data, and usage rights.

The server transmits the requested contents and data to the requester according to the transmission protocol. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the work. Otherwise, a set of default rights for backup files of the original are transmitted by the server.

The requester stores the digital work.

The repositories perform the common closing transaction steps.

The Delete Transaction

A Delete transaction deletes a digital work or a number of copies of a digital work from a repository. Practically all digital works would have delete rights.

The requester sends the server a message to initiate a delete transaction. This message indicates the work to be deleted, the version of the delete right for the transaction.

The repositories perform the common opening transaction steps.

The server deletes the file, erasing it from the file system.

The repositories perform the common closing transaction steps.

The Directory Transaction

A Directory transaction is a request for information about folders, digital works, and their parts. This amounts to roughly the same idea as protection codes in a conventional file system like TENEX, except that it is generalized to the full power of the access specifications of the usage rights language.

The Directory transaction has the important role of passing along descriptions of the rights and fees associated with a digital work. When a user wants to exercise a right, the user interface of his repository implicitly makes a directory request to determine the versions of the right that are available. Typically these are presented to the user—such as with different choices of billing for exercising a right. Thus, many directory transactions are invisible to the user and are exercised as part of the normal process of exercising all rights.

The requester sends the server a message to initiate a Directory transaction. This message indicates the file or folder that is the root of the directory request and the version of the directory right used for the transaction.

The server verifies that the information is accessible to the requester. In particular, it does not return the names of any files that have a HIDE-NAME status in their directory specifications, and it does not return the parts of any folders or files that have HIDE-PARTS in their specification. If the information is not accessible, the server ends the transaction with an error.

The repositories perform the common opening transaction steps.

The server sends the requested data to the requester according to the transmission protocol.

US 7,523,072 B2

39

The requester records the data.

The repositories perform the common closing transaction steps.

The Folder Transaction

A Folder transaction is a request to create or rename a folder, or to move a work between folders. Together with Directory rights, Folder rights control the degree to which organization of a repository can be accessed or modified from another repository.

The requester sends the server a message to initiate a Folder transaction. This message indicates the folder that is the root of the folder request, the version of the folder right for the transaction, an operation, and data. The operation can be one of create, rename, and move file. The data are the specifications required for the operation, such as a specification of a folder or digital work and a name.

The repositories perform the common opening transaction steps.

The server performs the requested operation—creating a folder, renaming a folder, or moving a work between folders.

The repositories perform the common closing transaction steps.

The Extract Transaction

A extract transaction is a request to copy a part of a digital work and to create a new work containing it. The extraction operation differs from copying in that it can be used to separate a part of a digital work from d-blocks or shells that place additional restrictions or fees on it. The extraction operation differs from the edit operation in that it does not change the contents of a work, only its embedding in d-blocks. Extraction creates a new digital work.

The requester sends the server a message to initiate an Extract transaction. This message indicates the part of the work to be extracted, the version of the extract right to be used in the transaction, the destination address information for placing the part as a new work, the file data for the work, and the number of copies involved.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester according to the transmission protocol. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the new work. Otherwise, the rights of the original are transmitted. The Copy-Count field for this right is set to the number-of-copies requested.

The requester records the contents, data, and usage rights and stores the work. It records the date and time that new work was made in the properties of the work.

The repositories perform the common closing transaction steps.

The Embed Transaction

An embed transaction is a request to make a digital work become a part of another digital work or to add a shell d-block to enable the adding of fees by a distributor of the work.

The requester sends the server a message to initiate an Embed transaction. This message indicates the work to be embedded, the version of the embed right to be used in the transaction, the destination address information for placing the part as a work, the file data for the work, and the number of copies involved.

40

The server checks the control specifications for all of the rights in the part and the destination. If they are incompatible, the server ends the transaction with an error.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester according to the transmission protocol. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the new work. Otherwise, the rights of the original are transmitted. The Copy-Count field for this right is set to the number-of-copies requested.

The requester records the contents, data, and usage rights and embeds the work in the destination file.

The repositories perform the common closing transaction steps.

The Edit Transaction

An Edit transaction is a request to make a new digital work by copying, selecting and modifying portions of an existing digital work. This operation can actually change the contents of a digital work. The kinds of changes that are permitted depend on the process being used. Like the extraction operation, edit operates on portions of a digital work. In contrast with the extract operation, edit does not effect the rights or location of the work. It only changes the contents. The kinds of changes permitted are determined by the type specification of the processor specified in the rights. In the currently preferred embodiment, an edit transaction changes the work itself and does not make a new work. However, it would be a reasonable variation to cause a new copy of the work to be made.

The requester sends the server a message to initiate an Edit transaction. This message indicates the work to be edited, the version of the edit right to be used in the transaction, the file data for the work (including its size), the process-ID for the process, and the number of copies involved.

The server checks the compatibility of the process-ID to be used by the requester against any process-ID specification in the right. If they are incompatible, it ends the transaction with an error.

The repositories perform the common opening transaction steps.

The requester uses the process to change the contents of the digital work as desired. (For example, it can select and duplicate parts of it; combine it with other information; or compute functions based on the information. This can amount to editing text, music, or pictures or taking whatever other steps are useful in creating a derivative work.)

The repositories perform the common closing transaction steps.

The edit transaction is used to cover a wide range of kinds of works. The category describes a process that takes as its input any portion of a digital work and then modifies the input in some way. For example, for text, a process for editing the text would require edit rights. A process for “summarizing” or counting words in the text would also be considered editing. For a music file, processing could involve changing the pitch or tempo, or adding reverberations, or any other audio effect. For digital video works, anything which alters the image would require edit rights. Examples would be colorizing, scaling, extracting still photos, selecting and combining frames into story boards, sharpening with signal processing, and so on.

Some creators may want to protect the authenticity of their works by limiting the kinds of processes that can be per-

US 7,523,072 B2

41

formed on them. If there are no edit rights, then no processing is allowed at all. A processor identifier can be included to specify what kind of process is allowed. If no process identifier is specified, then arbitrary processors can be used. For an example of a specific process, a photographer may want to allow use of his photograph but may not want it to be colored. A musician may want to allow extraction of portions of his work but not changing of the tonality.

Authorization Transactions

There are many ways that authorization transactions can be defined. In the following, our preferred way is to simply define them in terms of other transactions that we already need for repositories. Thus, it is convenient sometimes to speak of "authorization transactions," but they are actually made up of other transactions that repositories already have.

A usage right can specify an authorization-ID, which identifies an authorization object (a digital work in a file of a standard format) that the repository must have and which it must process. The authorization is given to the generic authorization (or ticket) server of the repository which begins to interpret the authorization.

As described earlier, the authorization contains a server identifier, which may just be the generic authorization server or it may be another server. When a remote authorization server is required, it must contain a digital address. It may also contain a digital certificate.

If a remote authorization server is required, then the authorization process first performs the following steps:

The generic authorization server attempts to set up the communications channel. (If the channel cannot be set up, then authorization fails with an error.)

When the channel is set up, it performs a registration process with the remote repository. (If registration fails, then the authorization fails with an error.)

When registration is complete, the generic authorization server invokes a "Play" transaction with the remote repository, supplying the authorization document as the digital work to be played, and the remote authorization server (a program) as the "player." (If the player cannot be found or has some other error, then the authorization fails with an error.)

The authorization server then "plays" the authorization.

This involves decrypting it using either the public key of the master repository that issued the certificate or the session key from the repository that transmitted it. The authorization server then performs various tests. These tests vary according to the authorization server. They include such steps as checking issue and validity dates of the authorization and checking any hot-lists of known invalid authorizations. The authorization server may require carrying out any other transactions on the repository as well, such as checking directories, getting some person to supply a password, or playing some other digital work. It may also invoke some special process for checking information about locations or recent events. The "script" for such steps is contained within the authorization server.

If all of the required steps are completed satisfactorily, the authorization server completes the transaction normally, signaling that authorization is granted.

The Install Transaction

An Install transaction is a request to install a digital work as runnable software on a repository. In a typical case, the requester repository is a rendering repository and the software would be a new kind or new version of a player. Also in

42

a typical case, the software would be copied to file system of the requester repository before it is installed.

The requester sends the server an Install message. This message indicates the work to be installed, the version of the Install right being invoked, and the file data for the work (including its size).

The repositories perform the common opening transaction steps.

The requester extracts a copy of the digital certificate for the software. If the certificate cannot be found or the master repository for the certificate is not known to the requester, the transaction ends with an error.

The requester decrypts the digital certificate using the public key of the master repository, recording the identity of the supplier and creator, a key for decrypting the software, the compatibility information, and a tamper-checking code. (This step certifies the software.)

The requester decrypts the software using the key from the certificate and computes a check code on it using a 1-way hash function. If the check-code does not match the tamper-checking code from the certificate, the installation transaction ends with an error. (This step assures that the contents of the software, including the various scripts, have not been tampered with.)

The requester retrieves the instructions in the compatibility-checking script and follows them. If the software is not compatible with the repository, the installation transaction ends with an error. (This step checks platform compatibility.)

The requester retrieves the instructions in the installation script and follows them. If there is an error in this process (such as insufficient resources), then the transaction ends with an error. Note that the installation process puts the runnable software in a place in the repository where it is no longer accessible as a work for exercising any usage rights other than the execution of the software as part of repository operations in carrying out other transactions.

The repositories perform the common closing transaction steps.

The Uninstall Transaction

An Uninstall transaction is a request to remove software from a repository. Since uncontrolled or incorrect removal of software from a repository could compromise its behavioral integrity, this step is controlled.

The requester sends the server an Uninstall message. This message indicates the work to be uninstalled, the version of the Uninstall right being invoked, and the file data for the work (including its size).

The repositories perform the common opening transaction steps.

The requester extracts a copy of the digital certificate for the software. If the certificate cannot be found or the master repository for the certificate is not known to the requester, the transaction ends with an error.

The requester checks whether the software is installed. If the software is not installed, the transaction ends with an error.

The requester decrypts the digital certificate using the public key of the master repository, recording the identity of the supplier and creator, a key for decrypting the software, the compatibility information, and a tamper-checking code. (This step authenticates the certification of the software, including the script for uninstalling it.)

The requester decrypts the software using the key from the certificate and computes a check code on it using a

US 7,523,072 B2

43

1-way hash function. If the check-code does not match the tamper-checking code from the certificate, the installation transaction ends with an error. (This step assures that the contents of the software, including the various scripts, have not been tampered with.)

The requester retrieves the instructions in the uninstallation script and follows them. If there is an error in this process (such as insufficient resources), then the transaction ends with an error.

The repositories perform the common closing transaction steps.

Distribution and Use Scenarios

To appreciate the robustness and flexibility of the present invention, various distribution and use scenarios for digital works are illustrated below. These scenarios are meant to be exemplary rather than exhaustive.

Consumers as Unpaid Distributors

In this scenario, a creator distributes copies of his works to various consumers. Each consumer is a potential distributor of the work. If the consumer copies the digital work (usually for a third party), a fee is collected and automatically paid to the creator.

This scenario is a new twist for digital works. It depends on the idea that “manufacturing” is just copying and is essentially free. It also assumes that the consumers as distributors do not require a fee for their time and effort in distributing the work.

This scenario is performed as follows:

A creator creates a digital work. He grants a Copy right with fees paid back to himself. If he does not grant an Embed right, then consumers cannot use the mechanism to act as distributors to cause fees to be paid to themselves on future copies. Of course, they could negotiate side deals or trades to transfer money on their own, outside of the system.

Paid Distributors

In another scenario, every time a copy of a digital work is sold a fee is paid to the creator and also to the immediate distributor.

This scenario does not give special status to any particular distributor. Anyone who sells a document has the right to add a fee to the sale price. The fee for sale could be established by the consumer. It could also be a fixed nominal amount that is contributed to the account of some charity.

This scenario is performed as follows:

A creator creates a digital work. He grants a Copy right with fees to be paid back to himself. He grants an Embed right, so that anyone can add shells to have fees paid to themselves.

A distributor embeds the work in a shell, with fees specified to be paid back to himself. If the distributor is content to receive fees only for copies that he sells himself, he grants an Extract right on the shell.

When a consumer buys a copy from the distributor, fees are paid both to the distributor and to the creator. If he chooses, the consumer can extract the work from the distributor’s shell. He cannot extract it from the creator’s shell. He can add his own shell with fees to be paid to himself.

Licensed Distribution

In this scenario, a creator wants to protect the reputation and value of his work by making certain requirements on its distributors. He issues licenses to distributors that satisfy the requirements, and in turn, promises to reward their efforts by assuring that the work will not be distributed over competing

44

channels. The distributors incur expenses for selecting the digital work, explaining it to buyers, promoting its sale, and possibly for the license itself. The distributor obtains the right to enclose the digital work in a shell, whose function is to permit the attachment of usage fees to be paid to the distributor in addition to the fees to be paid to the creator.

This differs from the previous scenario in that it precludes the typical copy owner from functioning as a distributor, since the consumer lacks a license to copy the document. Thus, a consumer cannot make copies, even for free. All copies must come initially from authorized distributors. This version makes it possible to hold distributors accountable in some way for the sales and support of the work, by controlling the distribution of certificates that enable distributors to legitimately charge fees and copy owners to make copies. Since licenses are themselves digital works, the same mechanisms give the creators control over distributors by charging for licenses and putting time limits on their validity.

This scenario is performed as follows:

A creator purchases a digital distribution license that he will hand out to his distributors. He puts access requirements (such as a personal license) on the Copy and Transfer rights on the distribution license so that only he can copy or transfer it.

The creator also creates a digital work. He grants an Embed right and a Copy right, both of which require the distribution license to be exercised. He grants a Play right so that the work can be played by anyone. He may optionally add a Transfer or Loan right, so that end consumers can do some non-commercial exchange of the work among friends.

A distributor obtains the distribution license and a number of copies of the work. He makes copies for his customers, using his distribution license.

A customer buys and uses the work. He cannot make new copies because he lacks a distribution license.

Super Distributors

This is a variation on the previous scenarios. A distributor can sell to anyone and anyone can sell additional copies, resulting in fees being paid back to the creator. However, only licensed distributors can add fees to be paid to themselves.

This scenario gives distributors the right to add fees to cover their own advertising and promotional costs, without making them be the sole suppliers. Their customers can also make copies, thus broadening the channel without diminishing their revenues. This is because distributors collect fees from copies of any copies that they originally sold. Only distributors can add fees.

This scenario is performed similarly to the previous ones. There are two key differences. (1) The creator only grants Embed rights for people who have a Distribution license. This is done by putting a requirement for a distributor’s license on the Embed right. Consequently, non-distributors cannot add their own fees. (2) The Distributor does not grant Extract rights, so that consumers cannot avoid paying fees to the Distributor if they make subsequent copies. Consequently, all subsequent copies result in fees paid to the Distributor and the Creator.

1-Level Distribution Fees

In this scenario, a distributor gets a fee for any copy he sells directly. However, if one of his customers sells further copies, he gets no further fee for those copies.

This scenario pays a distributor only for use of copies that he actually sold.

This scenario is performed similarly to the previous ones. The key feature is that the distributor creates a shell which specifies fees to be paid to him. He puts Extract rights on the

US 7,523,072 B2

45

shell. When a consumer buys the work, he can extract away the distributor's shell. Copies made after that will not require fees to be paid to the distributor.

Distribution Trees

In another scenario, distributors sell to other distributors and fees are collected at each level. Every copy sold by any distributor—even several d-blocks down in the chain—results in a fee being paid back to all of the previous distributors.

This scenario is like a chain letter or value chain. Every contributor or distributor along the way obtains fees, and is thereby encouraged to promote the sale of copies of the digital work.

This scenario is performed similarly to the previous ones. The key feature is that the distributor creates a shell which specifies fees to be paid to him. He does not grant Extract rights on the shell. Consequently, all future copies that are made will result in fees paid to him.

Weighted Distribution Trees

In this scenario, distributors make money according to a distribution tree. The fee that they make depends on various parameters, such as time since their sale or the number of subsequent distributors.

This is a generalized version of the Distribution Tree scenario, in that it tries to vary the fee to account for the significance of the role of the distributor.

This scenario is similar to the previous one. The difference is that the fee specification on the distributor's shell has provisions for changes in prices. For example, there could be a fee schedule so that copies made after the passage of time will require lower fees to be paid to the distributor. Alternatively, the distributor could employ a "best-price" billing option, using any algorithm he chooses to determine the fee up to the maximum specified in the shell.

Fees for Reuse

In this scenario, a first creator creates a work. It is distributed by a first distributor and purchased by a second creator. The second creator extracts a portion of the work and embeds in it a new work distributed by a second distributor. A consumer buys the new work from the second distributor. The first creator receives fees from every transaction; the first distributor receives fees only for his sale; the second creator and second distributor receive fees for the final sale.

This scenario shows how that flexible automatic arrangements can be set up to create automatic charging systems that mirror current practice. This scenario is analogous to when an author pays a fee to reuse a figure in some paper. In the most common case, a fee is paid to the creator or publisher, but not to the bookstore that sold the book.

The mechanisms for derived works are the same as those for distribution.

Limited Reuse

In this scenario, several first creators create works. A second creator makes a selection of these, publishing a collection made up of the parts together with some new interstitial material. (For example, the digital work could be a selection of music or a selection of readings.) The second creator wants to continue to allow some of the selected works to be extractable, but not the interstitial material.

This scenario deals with fine grained control of the rights and fees for reuse.

This scenario is performed as follows:

The first creators create their original works. If they grant extraction and embedding rights, then the second creator can include them in a larger collected work. The second creator creates the interstitial material. He does grant an Extract right

46

on the interstitial material. He grants Extract rights on a subset of the reused material. A consumer of the collection can only extract portions that have that right. Fees are automatically collected for all parts of the collection.

Commercial Libraries

Commercial libraries buy works with the right to loan. They limit the loan period and charge their own fees for use. This scenario deals with fees for loaning rather than fees for making copies. The fees are collected by the same automatic mechanisms.

The mechanisms are the same as previous scenarios except that the fees are associated with the Loan usage right rather than the Copy usage right.

Demo Versions

A creator believes that if people try his work that they will want to buy it or use it. Consumers of his work can copy the work for free, and play (or execute) a limited version of the work for free, and can play or use the full featured version for a fee. This scenario deals with fees for loaning rather than fees for making copies. The fees are collected by the same automatic mechanisms.

This scenario is performed as follows:

The creator creates a digital work and grants various rights and fees. The creator grants Copy and Embed rights without a fee, in order to ensure widespread distribution of the work. Another of the rights is a limited play right with little or no fee attached. For example, this right may be for playing only a portion of the work. The play right can have various restrictions on its use. It could have a ticket that limits the number of times it is used. It could have internal restrictions that limit its functionality. It could have time restrictions that invalidate the right after a period of time or a period of use. Different fees could be associated with other versions of the Play right.

Upgrading a Digital Work with a Vendor

A consumer buys a digital work together with an agreement that he can upgrade to a new version at a later date for a modest fee, much less than the usual purchase price. When the new version becomes available, he goes to a qualified vendor to make the transaction.

This scenario deals with a common situation in computer software. It shows how a purchase may include future "rights." Two important features of the scenario are that the transaction must take place at a qualified vendor, and that the transaction can be done only once per copy of the digital work purchased.

This scenario is performed as follows:

The creator creates a digital work, an upgrade ticket, and a distribution license. The upgrade ticket uses the a generic ticket agent that comes with repositories. As usual, the distribution license does not have Copy or Transfer rights. He distributes a bundled copies of the work and the ticket to his distributors as well as distribution licenses.

The distributor sells the old bundled work and ticket to customers.

The customer extracts the work and the ticket. He uses the work according to the agreements until the new version becomes available.

When the new work is ready, the creator gives it to distributors. The new work has a free right to copy from a distributor if a ticket is available.

The consumer goes to distributors and arranges to copy the work. The transaction offers the ticket. The distributor's repository punches the ticket and copies the new version to the consumer's repository.

The consumer can now use the new version of the work.

US 7,523,072 B2

47

Distributed Upgrading of Digital Works

A consumer buys a digital work together with an agreement that he can upgrade to a new version at a later date for a modest fee, much less than the usual purchase price. When the new version becomes available, he goes to anyone who has the upgraded version and makes the transaction.

This scenario is like the previous one in that the transaction can only be done once per copy of the digital work purchased, but the transaction can be accomplished without the need to connect to a licensed vendor.

This scenario is similar to the previous one except that the Copy right on the new work does not require a distribution license. The consumer can upgrade from any repository having the new version. He cannot upgrade more than once because the ticket cannot work after it has been punched. If desired, the repository can record the upgrade transaction by posting a zero cost bill to alert the creator that the upgrade has taken place.

Limited Printing

A consumer buys a digital work and wants to make a few ephemeral copies. For example, he may want to print out a paper copy of part of a digital newspaper, or he may want to make a (first generation) analog cassette tape for playing in his car. He buys the digital work together with a ticket required for printing rights.

This scenario is like the common practice of people making cassette tapes to play in their car. If a publisher permits the making of cassette tapes, there is nothing to prevent a consumer from further copying the tapes. However, since the tapes are "analog copies," there is a noticeable quality loss with subsequent generations. The new contribution of the present invention is the use of tickets in the access controls for the making of the analog copies.

This scenario is performed as follows:

The creator sells a work together with limited printing rights. The printing rights specify the kind of printer (e.g., a kind of cassette recorder or a kind of desktop paper printer) and also the kind of ticket required. The creator either bundles a limited number of tickets or sells them separately. If the tickets use the generic ticket agent, the consumer with the tickets can exercise the right at his convenience.

Demand Publishing

Professors in a business school want to put together course books of readings selected from scenario studies from various sources. The bookstore wants to be able to print the books from digital masters, without negotiating for and waiting for approval of printing of each of the scenarios. The copyright holders of the scenarios want to be sure that they are paid for every copy of their work that is printed.

On many college campuses, the hassle of obtaining copy clearances in a timely way has greatly reduced the viability of preparing course books. Print shops have become much more cautious about copying works in the absence of documented permission.

Demand Publishing is performed as follows: the creator sells a work together with printing rights for a fee. There can be rights to copy (distribute) the work between bookstore repositories, with or without fee. The printing rights specify the kind of printer. Whenever a bookstore prints one of the works (either standalone or embedded in a collection), the fee is credited to the creator automatically. To discourage unauthorized copying of the print outs, it would be possible for the printer to print tracer messages discretely on the pages identifying the printing transaction, the copy number, and any other identifying information. The tracer information could

48

be secretly embedded in the text itself (encoded in the grey scale) or hidden in some other way.

Metered Use and Multiple Price Packages

A consumer does not know what music to purchase until he decides whether he likes it. He would like to be able to take it home and listen to it, and then decide whether to purchase. Furthermore, he would like the flexibility of paying less if he listens to it very infrequently.

This scenario just uses the capability of the approach to have multiple versions of a right on a digital work. Each version of the right has its own billing scheme. In this scenario, the creator of the work can offer the Copy right without fee, and defer billing to the exercise of the Play right. One version of the play right would allow a limited performance without fee—a right to "demo". Another version of the right could have a metered rate, of say \$0.25 per hour of play. Another version could have a fee of \$15.00 for the first play, but no fee for further playing. When the consumer exercises a play right, he specifies which version of the right is being selected and is billed accordingly.

Fees for Font Usage

A designer of type fonts invests several months in the design of special fonts. The most common way of obtaining revenue for this work is to sell copies of the fonts to publishers for unlimited use over unlimited periods of time. A font designer would like to charge a rate that reflects the amount that the font is used.

This scenario is performed as follows: the font designer creates a font as a digital work. He creates versions of the Play right that bill either for metered use or "per-use". Each version of the play right would require that the player (a print layout program) be of an approved category. The font designer assigns appropriate fees to exercise the Copy right. When a publisher client wants to use a font, he includes it as input to a layout program, and is billed automatically for its use. In this way, a publisher who makes little use of a font pays less than one who uses it a lot.

Rational Database Usage Charges

Online information retrieval services typically charge for access in a way that most clients find unpredictable and uncorrelated to value or information use. The fee depends on which databases are open, dial-up connect time, how long the searches require, and which articles are printed out. There are no provisions for extracting articles or photographs, no method for paying to reuse information in new works, no distinction between having the terminal sit idly versus actively searching for data, no distinction between reading articles on the screen and doing nothing, and higher rates per search when the centralized facility is busy and slow servicing other clients. Articles can not be offloaded to the client's machine for off-site search and printing. To offer such billing or the expanded services, the service company would need a secure way to account for and bill for how information is used.

This scenario is performed as follows:

The information service bundles its database as files in a repository. The information services company assigns different fees for different rights on the information files. For example, there could be a fee for copying a search database or a source file and a different fee for printing. These fees would be in addition to fees assigned by the original creator for the services. The fees for using information would be different for using them on the information service company's computers or the client's computers. This billing distinction would be controlled by having different versions of the rights,

US 7,523,072 B2

49

where the version for use on the service company's computer requires a digital certificate held locally. Fees for copying or printing files would be handled in the usual way, by assigning fees to exercising those rights. The distinction between searching and viewing information would be made by having different "players" for the different functions. This distinction would be maintained on the client's computers as well as the service computers. Articles could be extracted for reuse under the control of Extract and Embed rights. Thus, if a client extracts part of an article or photograph, and then sells copies of a new digital work incorporating it, fees could automatically be collected both by the information service and earlier creators and distributors of the digital work. In this way, the information retrieval service could both offer a wider selection of services and billing that more accurately reflects the client's use of the information.

Print Spooling with Rights

In the simplest scenario, when a user wants to print a digital document he issues a print command to the user interface. If the document has the appropriate rights and the conditions are satisfied, the user agrees to the fee and the document is printed. In other cases, the printer may be on a remote repository and it is convenient to spool the printing to a later time. This leads to several issues. The user requesting the printing wants to be sure that he is not billed for the printing until the document is actually printed. Restated, if he is billed at the time the print job is spooled but the job is canceled before printing is done, he does not want to pay. Another issue is that when spooling is permitted, there are now two times at which rights, conditions and fees could be checked: the time at which a print job is spooled and the time at which a print is made. As with all usage rights, it is possible to have rights that expire and to have rights whose fee depends on various conditions. What is needed is a means to check rights and conditions at the time that printing is actually done.

This scenario is performed as follows: A printing repository is a repository with the usual repository characteristics plus the hardware and software to enable printing. Suppose that a user logs into a home repository and wants to spool print jobs for a digital work at a remote printing repository. The user interface for this could treat this as a request to "spool" prints. Underneath this "spooling" request, however, are standard rights and requests. To support such requests, the creator of the work provides a Copy right, which can be used to copy the work to a printing repository. In the default case, this Copy right would have no fees associated for making the copy. However, the Next-Set-Of-Rights for the copy would only include the Print rights, with the usual fees for each variation of printing. This version of the Copy right could be called the "print spooling" version of the Copy right. The user's "spool request" is implemented as a Copy transaction to put a copy of the work on the printing repository, followed by Print transactions to create the prints of the work. In this way, the user is only billed for printing that is actually done. Furthermore, the rights, conditions and fees for printing the work are determined when the work is about to be printed.

Thus, a system for enforcing the usage rights of digital works is disclosed. While the embodiments disclosed herein are preferred, it will be appreciated from this teaching that various alternative, modifications, variations or improve-

50

ments therein may be made by those skilled in the art, which are intended to be encompassed by the following claims.

APPENDIX A

Glossary

Authorization Repository:

A special type of repository which provides an authorization service. An authorization may be specified by a usage right. The authorization must be obtained before the right may be exercised.

Billing Clearinghouse:

A financial institution or the like whose purpose is to reconcile billing information received from credit servers. The billing clearinghouse may generate bills to users or alternatively, credit and debit accounts involved in the commercial transactions.

Billing Transactions:

The protocol used by which a repository reports billing information to a credit server.

Clearinghouse Transactions:

The protocol used between a credit server and a clearinghouse.

Composite Digital Work:

A digital work comprised of distinguishable parts. Each of the distinguishable parts is itself a digital work which has usage rights attached.

Content:

The digital information (i.e. raw bits) representing a digital work.

Copy Owner:

A term which refers to the party who owns a digital work stored in a repository. In the typical case, this party has purchased various rights to the document for printing, viewing, transferring, or other specific uses.

Creator:

A term which refers to a party who produces a digital work.

Credit Server:

A device which collects and reports billing information for a repository. In many implementations, this could be built as part of a repository. It requires a means for periodically communicating with a billing clearinghouse.

Description Tree:

A structure which describes the location of content and the usage rights and usage fees for a digital work. A description tree is comprised of description blocks. Each description block corresponds to a digital work or to an interest (typically a revenue bearing interest) in a digital work.

Digital Work (Work):

Any encapsulated digital information. Such digital information may represent music, a magazine or book, or a multimedia composition. Usage rights and fees are attached to the digital work.

Distributor:

A term which refers to a party who legitimately obtains a copy of a digital work and offers it for sale.

US 7,523,072 B2

51

Identification (Digital) Certificate:

A signed digital message that attests to the identity of the possessor. Typically, digital certificates are encrypted in the private key of a well-known master repository.

Master Repository:

A special type of repository which issues identification certificates and distributes lists of repositories whose integrity have been compromised and which should be denied access to digital works (referred to as repository "hotlists".)

Public Key Encryption:

An encryption technique used for secure transmission of messages on a communication channel. Key pairs are used for the encryption and decryption of messages. Typically one key is referred to as the public key and the other is the private key. The keys are inverses of each other from the perspective of encryption. Restated, a digital work that is encrypted by one key in the pair can be decrypted only by the other.

Registration Transactions:

The protocol used between repositories to establish a trusted session.

Rendering Repository:

A special type of repository which is typically coupled to a rendering system. The rendering repository will typically be embodied within the secure boundaries of a rendering system.

Rendering System:

The combination of a rendering repository and a rendering device. Examples of a rendering systems include printing systems, display systems, general purpose computer systems, video systems or audio systems.

Repository:

Conceptually a set of functional specifications defining core functionality in the support of usage rights. A repository is a trusted system in that it maintains physical, communications and behavioral integrity.

Requester Mode:

A mode of a repository where it is requesting access to a digital work.

Revenue Owners:

A term which refers to the parties that maintain an interest in collecting fees for document use or who stand to lose revenue if illegitimate copies of the digital work are made.

Server Mode:

A mode of a repository where it is processing an incoming request to access a digital work.

Shell Description Block:

A special type of description block designating an interest in a digital work, but which does not add content. This will typically be added by a distributor of a digital work to add their fees.

Transactions:

A term used to refer to the protocols by which repositories communicate.

Usage Fees:

A fee charged to a requester for access to a digital work. Usage fees are specified within the usage rights language.

Usage Rights:

A language for defining the manner in which a digital work may be used or distributed, as well as any conditions on which use or distribution is premised.

52

Usage Transactions:

A set of protocols by which repositories communicate in the exercise of a usage rights. Each usage right has it's own transaction steps.

What is claimed:

1. A method for securely rendering digital documents, comprising:

retrieving, by a document platform, a digital document and at least one usage right associated with the digital document from a document repository, the at least one usage right specifying a manner of use indicating the manner in which the digital document can be rendered;

storing the digital document and the at least one usage right in separate files in the document platform;

determining, by the document platform, whether the digital document may be rendered based on the at least one usage right; and

if the at least one usage right allows the digital document to be rendered on the document platform, rendering the digital document by the document platform.

2. The method as recited in claim 1, wherein the manner of use includes the number of times the digital document can be rendered.

3. The method as recited in claim 1, wherein at least a portion of the digital document is a software program.

4. The method as recited in claim 1, wherein the at least one usage right comprises a revenue identifier for identifying a revenue owner of the digital document.

5. The method as recited in claim 1, wherein the at least one usage right also specifies one or more conditions which must be satisfied before the manner of rendering may be exercised.

6. The method as recited in claim 5, wherein at least one condition includes determining the presence of a digital ticket.

7. The method as recited in claim 1, wherein the at least one usage right or a part of the digital document is stored on a removable storage device.

8. The method as recited in claim 1, wherein at least one part of the digital document and the at least one usage right are stored on a same device.

9. The method as recited in claim 1, wherein at least one part of the digital document and the at least one usage right are stored on different devices.

10. A method for securely rendering digital documents, comprising:

storing a digital document and at least one usage right in separate files in a document repository, wherein the at least one usage right is associated with the digital document;

receiving a request from a document platform for access to the digital document;

determining, by the document platform, whether the request may be granted based on the at least one usage right, the determining step including authenticating the document platform and determining whether the at least one usage right includes a manner of use that allows transfer of the digital document to the document platform;

if the at least one usage right allows the transfer of the digital document to the document platform, transferring the digital document and the at least one usage right associated with the digital document to the document platform;

US 7,523,072 B2

53

storing the digital document and the at least one usage right in the document platform, wherein the at least one usage right is stored in a separate file from the digital document; and

rendering the digital document by the document platform. 5

11. The method as recited in claim 10, wherein at least a portion of the digital document is a software program.

12. The method as recited in claim 10, wherein the at least one usage right comprises a revenue identifier for identifying a revenue owner of the digital document. 10

13. The method as recited in claim 10, wherein the at least one usage right also specifies one or more conditions which must be satisfied before the manner of rendering may be exercised.

14. The method as recited in claim 13, wherein at least one condition includes determining the presence of a digital ticket. 15

15. The method as recited in claim 10, wherein the at least one usage right or a part of the digital document is stored on a removable storage device.

16. The method as recited in claim 10, wherein at least one part of the digital document and the at least one usage right are stored on a same device. 20

17. The method as recited in claim 10, wherein at least one part of the digital document and the at least one usage right are stored on different devices. 25

18. A method for securely rendering digital documents, comprising:

receiving a request from a document platform to transfer a digital document from a document repository to the document platform, the digital document having at least one usage right associated therewith; 30

authenticating the document platform by accessing at least one identifier associated with the document platform or with a user of the document platform and determining whether the identifier is associated with at least one of the document platform and a user authorized to use the digital document; 35

if the authenticating step is successful, determining whether the digital document may be transferred and

54

stored on the document platform based on a manner of use included in the at least one usage right;

if the at least one usage right allows the digital document to be transferred to the document platform, transferring the digital document and the at least one usage right associated with the digital document to the document platform;

storing the digital document and the at least one usage right in the document platform, wherein the at least one usage right is stored in a separate file from the digital document; determining, by the document platform, whether the digital document may be rendered based on the at least one usage right; and

if the at least one usage right allows the digital document to be rendered on the document platform, rendering the digital document by the document platform.

19. The method as recited in claim 18, wherein at least a portion of the digital document is a software program.

20. The method as recited in claim 18, wherein the at least one usage right comprises a revenue identifier for identifying a revenue owner of the digital content.

21. The method as recited in claim 18, wherein the at least one usage right also specifies one or more conditions which must be satisfied before the manner of rendering may be exercised.

22. The method as recited in claim 21, wherein at least one condition includes determining the presence of a digital ticket.

23. The method as recited in claim 18, wherein the at least one usage right or a part of the digital document is stored on a removable storage device.

24. The method as recited in claim 18, wherein at least one part of the digital document and at least one usage right are stored on a same device.

25. The method as recited in claim 18, wherein at least one part of the digital document and the at least one usage right are stored on different devices.

* * * * *



US008370956B2

(12) **United States Patent**
Stefik et al.

(10) **Patent No.:** **US 8,370,956 B2**

(45) **Date of Patent:** **Feb. 5, 2013**

(54) **SYSTEM AND METHOD FOR RENDERING DIGITAL CONTENT IN ACCORDANCE WITH USAGE RIGHTS INFORMATION**

(56) **References Cited**

(75) Inventors: **Mark J. Stefik**, Portola Valley, CA (US);
Peter L. T. Pirolli, San Francisco, CA (US)

U.S. PATENT DOCUMENTS

4,817,140 A 3/1989 Chandra et al.
 5,204,961 A 4/1993 Barlow
 5,390,297 A 2/1995 Barber et al.
 6,135,646 A 10/2000 Kahn et al.

(73) Assignee: **ContentGuard Holdings, Inc.**,
 Wilmington, DE (US)

FOREIGN PATENT DOCUMENTS

EP 0398492 A2 11/1990
 EP 0588415 A1 3/1994

OTHER PUBLICATIONS

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

Non-Final Office Action dated Jun. 12, 2008 cited in U.S. Appl. No. 11/304,793.

Final Office Action dated Nov. 14, 2008 cited in U.S. Appl. No. 11/304,793.

(21) Appl. No.: **13/584,782**

Non-Final Office Action dated May 27, 2009 cited in U.S. Appl. No. 11/304,793.

(22) Filed: **Aug. 13, 2012**

Final Office Action dated Jan. 22, 2010 cited in U.S. Appl. No. 11/304,793.

(65) **Prior Publication Data**

US 2012/0331569 A1 Dec. 27, 2012

Decision on Appeal dated Jun. 13, 2012 cited in U.S. Appl. No. 11/304,793.

Kohl, John T. et al., "The Evolution of the Kerberos Authentication Service", Distributed Open Systems, IEEE, 1994, 18 pages.

Primary Examiner — Brandon Hoffman

Related U.S. Application Data

(60) Continuation of application No. 11/304,793, filed on Dec. 16, 2005, now abandoned, which is a division of application No. 11/135,352, filed on May 24, 2005, now Pat. No. 7,266,529, which is a continuation of application No. 10/322,759, filed on Dec. 19, 2002, now Pat. No. 6,898,576, which is a continuation of application No. 09/778,001, filed on Feb. 7, 2001, now Pat. No. 6,708,157, which is a division of application No. 08/967,084, filed on Nov. 10, 1997, now Pat. No. 6,236,971, which is a continuation of application No. 08/344,760, filed on Nov. 23, 1994, now abandoned.

(74) *Attorney, Agent, or Firm* — Marc S. Kaufman; Stephen M. Hertzler; Reed Smith LLP

(57) **ABSTRACT**

Methods, apparatus, and media for rendering digital content by at least one recipient computing device in accordance with usage rights information. An exemplary method comprises receiving the digital content by the at least one recipient computing device from at least one sending computing device only if the at least one recipient computing device has been determined to be trusted to receive the digital content from the at least one sending computing device, receiving, by the at least one recipient computing device, a request to render the digital content, determining, based on the usage rights information, whether the digital content may be rendered by the at least one recipient computing device, and rendering the digital content, by the at least one recipient computing device, only if it is determined that the content may be rendered by the at least one recipient computing device.

(51) **Int. Cl.**
G06F 7/04 (2006.01)

(52) **U.S. Cl.** **726/29**

(58) **Field of Classification Search** None
 See application file for complete search history.

18 Claims, 13 Drawing Sheets

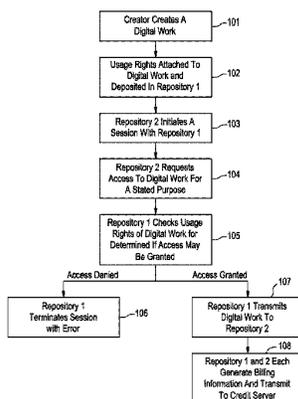


FIG. 1

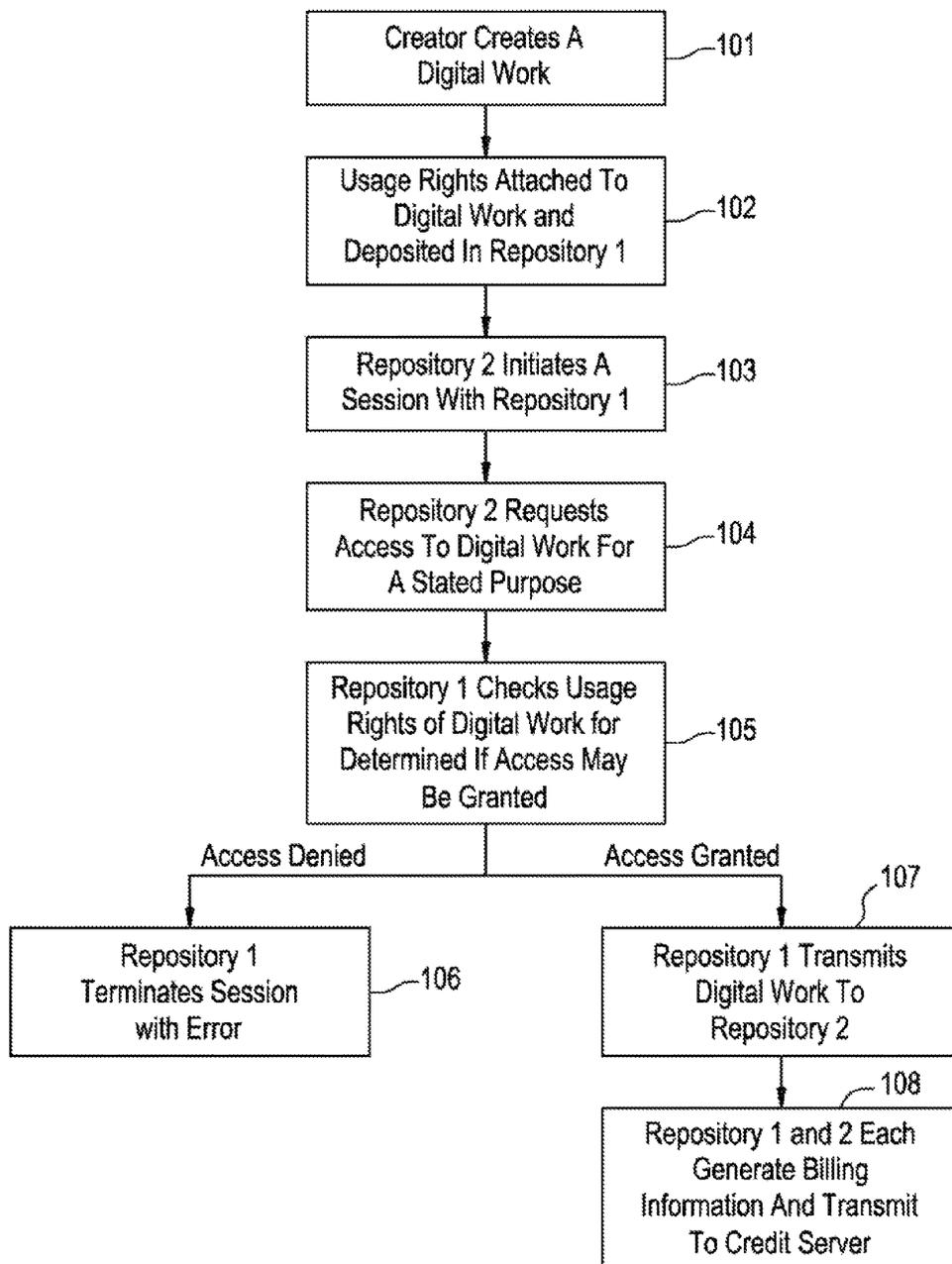


FIG. 2

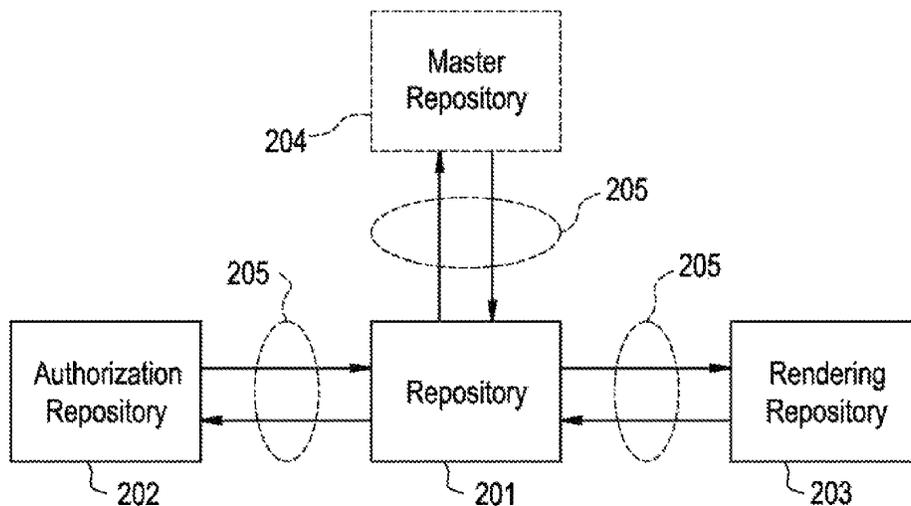


FIG. 3

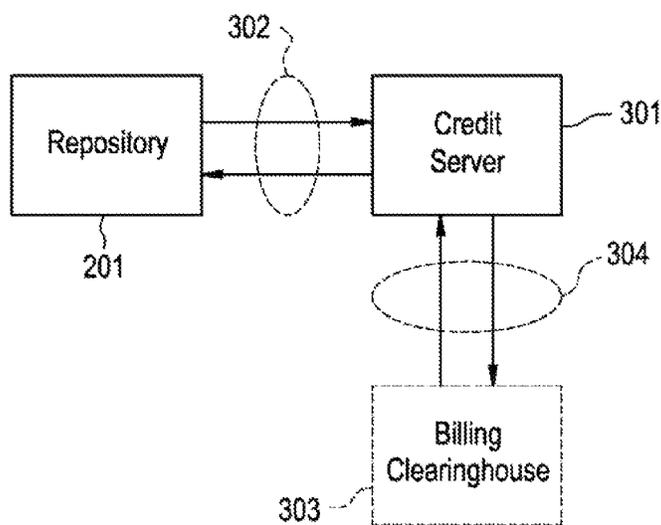


FIG. 4A

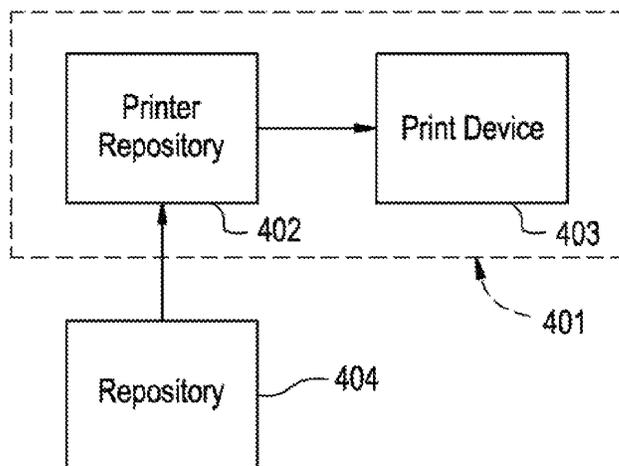


FIG. 4B

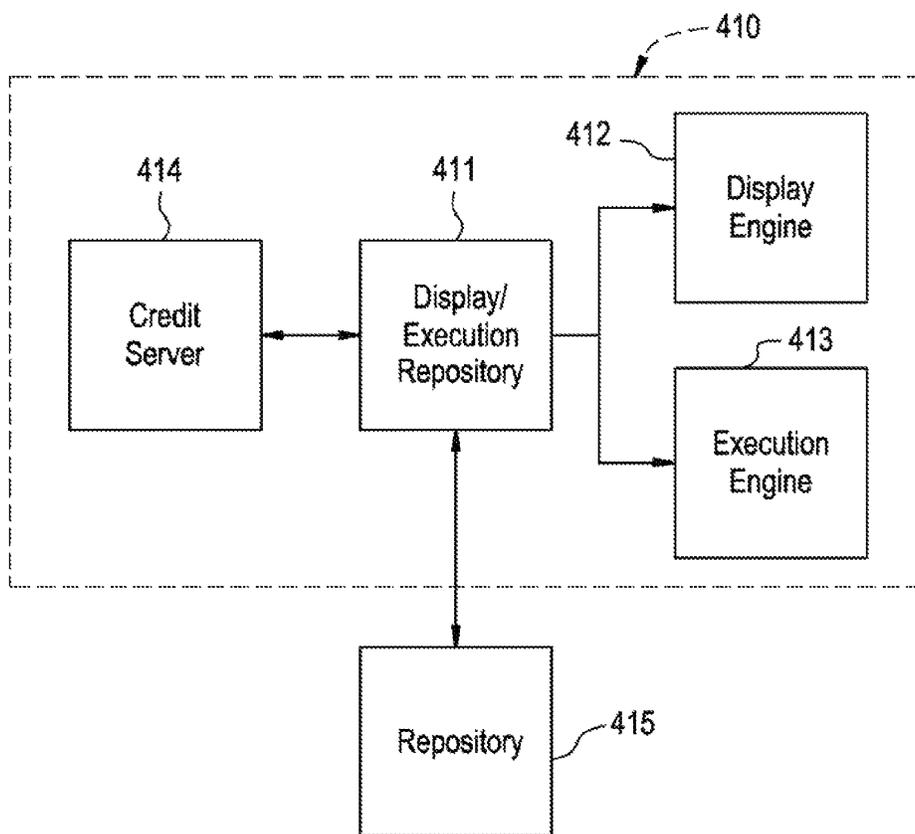


FIG. 5

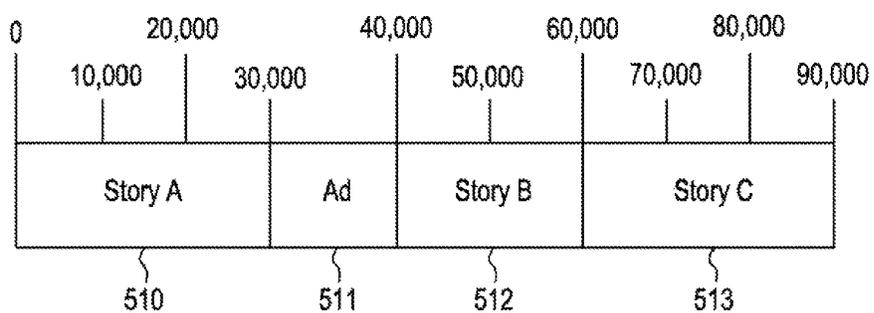


FIG. 6

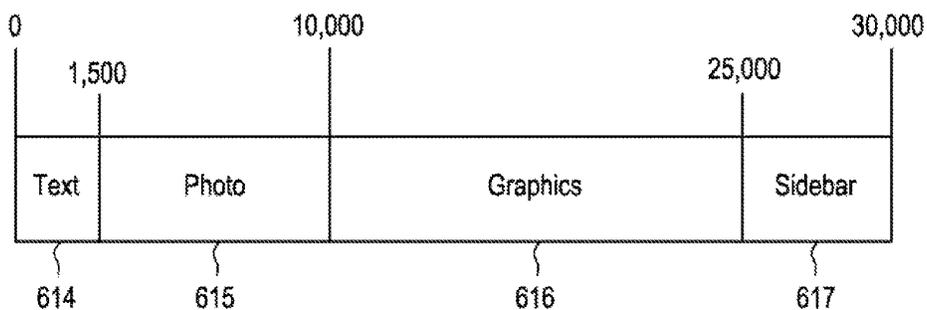


FIG. 7

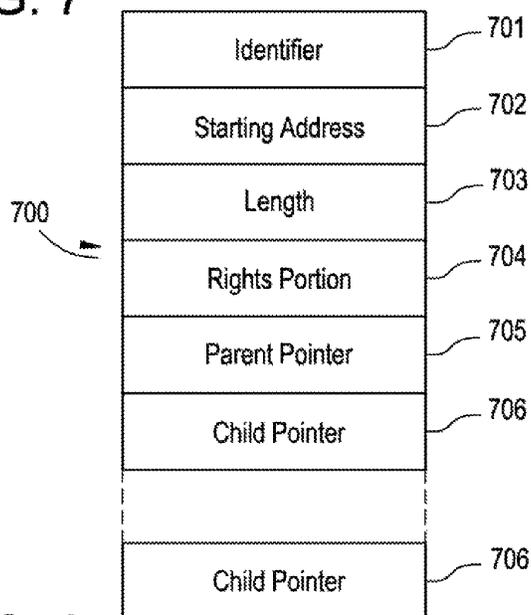


FIG. 8

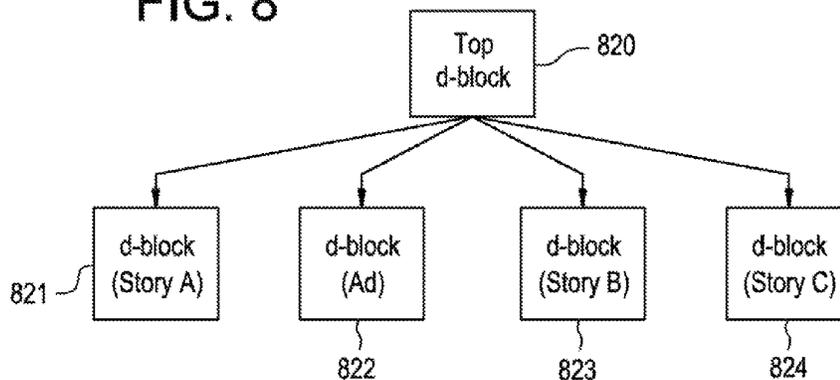


FIG. 9

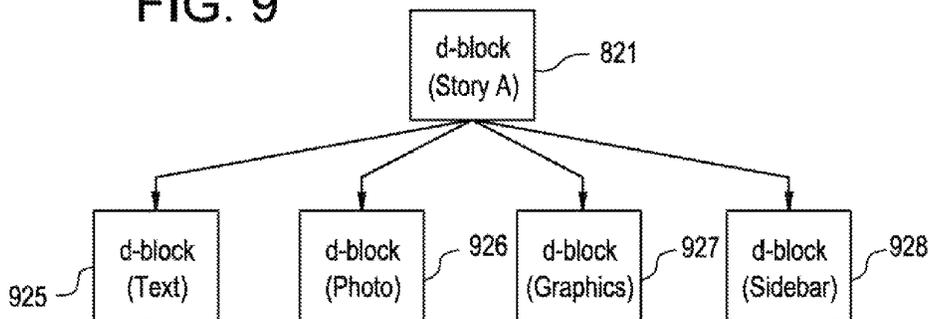


FIG. 10

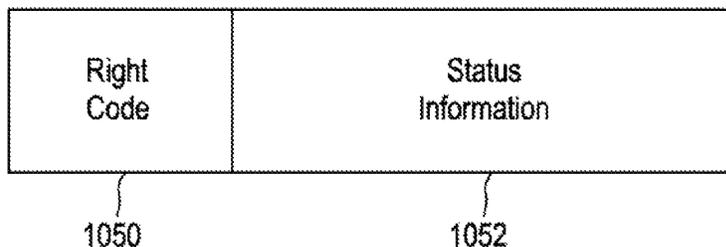


FIG. 14

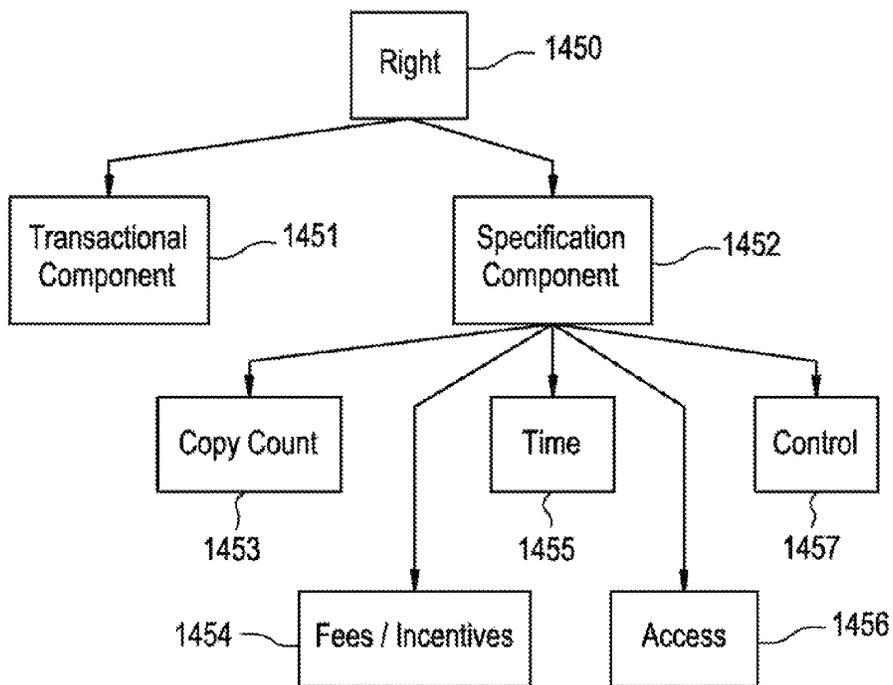


FIG. 11

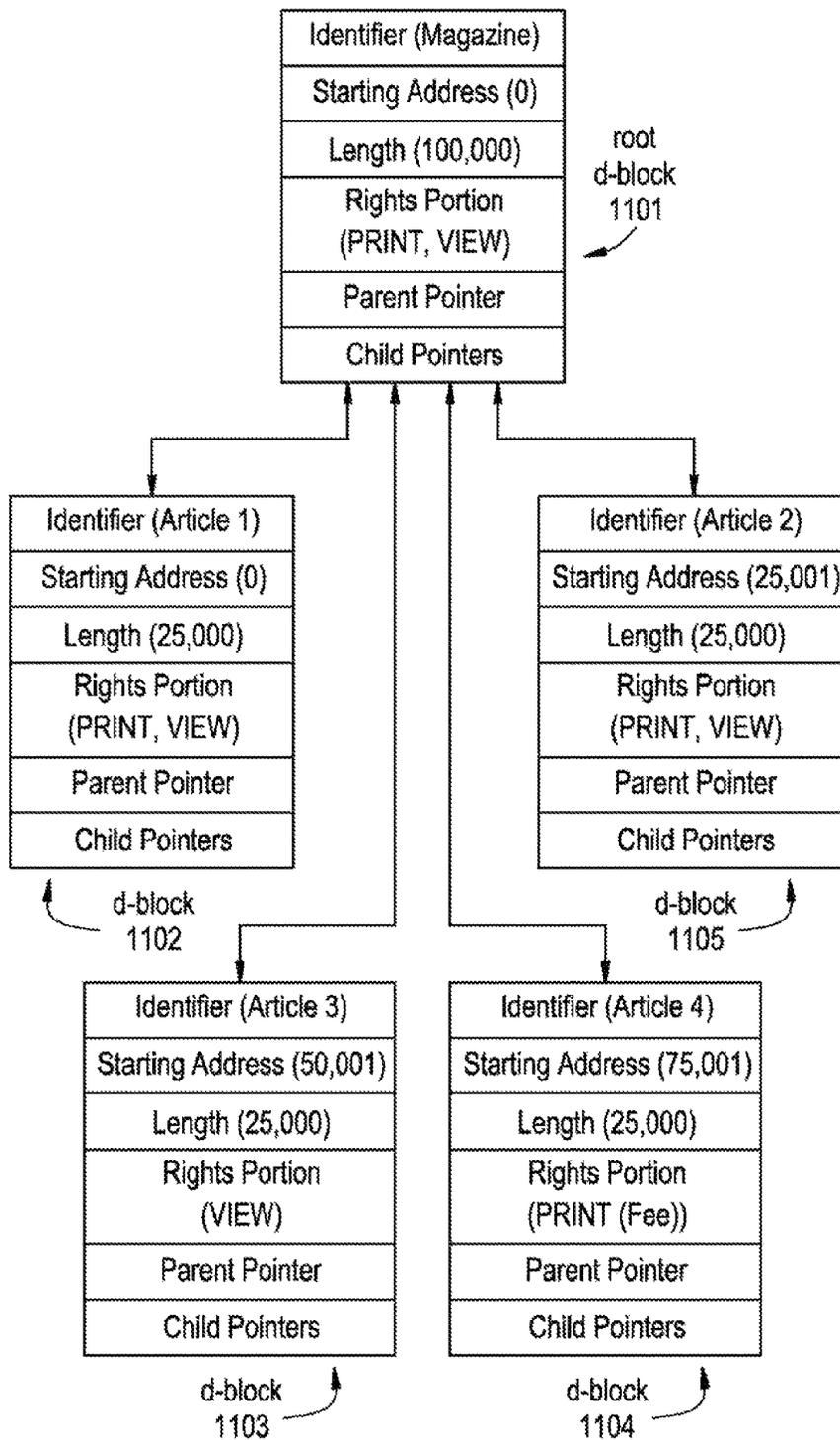


FIG. 12

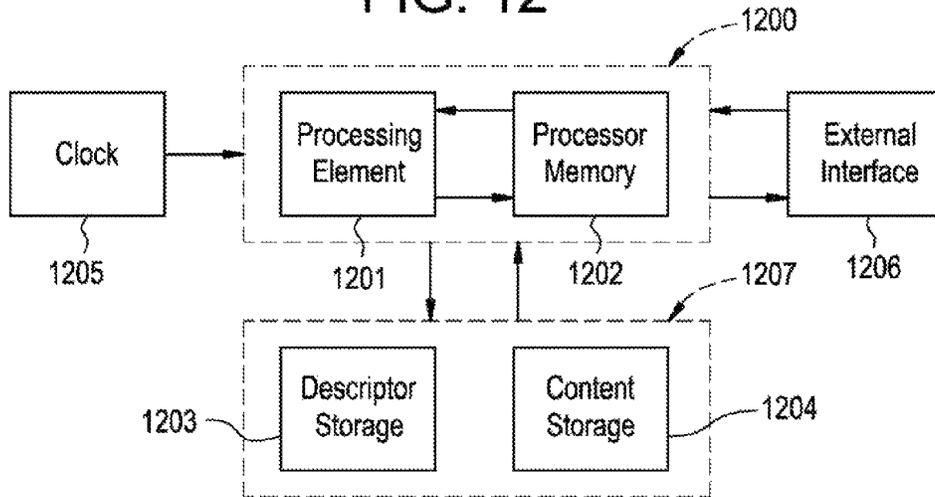


FIG. 13

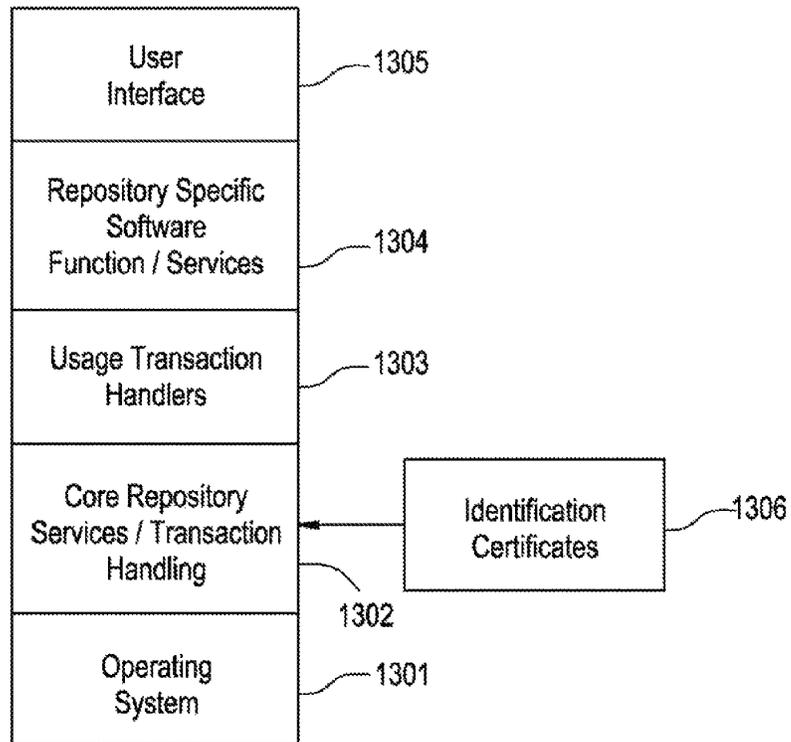


FIG. 15

- 1501 ~ Digital Work Rights: = (Rights*)
- 1502 ~ Right: = (Right-Code {Copy-Count} {Control-Spec} {Time-Spec}
{Access-Spec} {Fee-Spec})
- 1503 ~ Right-Code: = Render-Code | Transport-Code | File-Management-
Code | Derivative-Works-Code | Configuration-Code
- 1504 ~ Render-Code: = [Play: {Player: Player-ID} | Print: {Printer: Printer-ID}]
- 1505 ~ Transport-Code: = [Copy | Transfer | Loan {Remaining-Rights:
Next-Set-of-Rights}] {{Next-Copy-Rights: Next-Set-of-Rights}}
- 1506 ~ File-Management-Code: = Backup {Back-Up-Copy-Rights:
Next-Set-of-Rights} | Restore | Delete | Folder
| Directory {Name: Hide-Local | Hide-Remote}
{Parts: Hide-Local | Hide-Remote}
- 1507 ~ Derivative-Works-Code: = [Extract | Embed | Edit {Process:
Process-ID}] {Next-Copy-Rights:
Next-Set-of-Rights}
- 1508 ~ Configuration-Code: = Install | Uninstall
- 1509 ~ Next-Set-of-Rights: = {{Add: Set-of-Rights}} {{Delete:
Set-of-Rights}} {{Replace: Set-of-Rights}} {{Keep: Set-of-Rights}}
- 1510 ~ Copy-Count: = (Copies: positive-integer | 0 | Unlimited)
- 1511 ~ Control-Spec: = (Control: {Restrictable | Unrestrictable}
{Unchargeable | Chargeable})
- 1512 ~ Time-Spec: = ({Fixed-Interval | Sliding-Interval | Meter-Time}
Until: Expiration-Date)
- 1513 ~ Fixed-Interval: = From: Start-Time
- 1514 ~ Sliding-Interval: = Interval : Use-Duration
- 1515 ~ Meter-Time: = Time-Remaining: Remaining-Use
- 1516 ~ Access-Spec: = ({SC: Security-Class} {Authorization: Authorization-ID*}
{Other-Authorization: Authorization-ID*}) {Ticket: Ticket-ID}
- 1517 ~ Fee-Spec: = (Scheduled-Discount) Regular-Fee-Spec | Scheduled-Fee-Spec |
Markup-Spec
- 1518 ~ Scheduled-Discount: = Scheduled-Discount: (Scheduled-Discount:
(Time-Spec Percentage)*)
- 1519 ~ Regular-Fee-Spec: = ({Fee: | Incentive:} [Per-Use-Spec | Metered-Rate-
Spec | Best-Price-Spec | Call-For-Price-Spec]
{Min: Money-Unit Per: Time-Spec} {Max:
Money-Unit Per: Time-Spec} To: Account-ID)
- 1520 ~ Per-Use-Spec: = Per-Use: Money-Unit
- 1521 ~ Metered-Rate-Spec: = Metered: Money-Unit Per: Time-Spec
- 1522 ~ Best-Price-Spec: = Best-Price: Money-unit Max: Money-Unit
- 1523 ~ Call-For-Price-Spec: = Call-For-Price
- 1524 ~ Scheduled-Fee-Spec: = (Schedule: (Time-Spec Regular-Fee-Spec)*)
- 1525 ~ Markup-Spec: = Markup: percentage To: Account-ID

FIG. 16

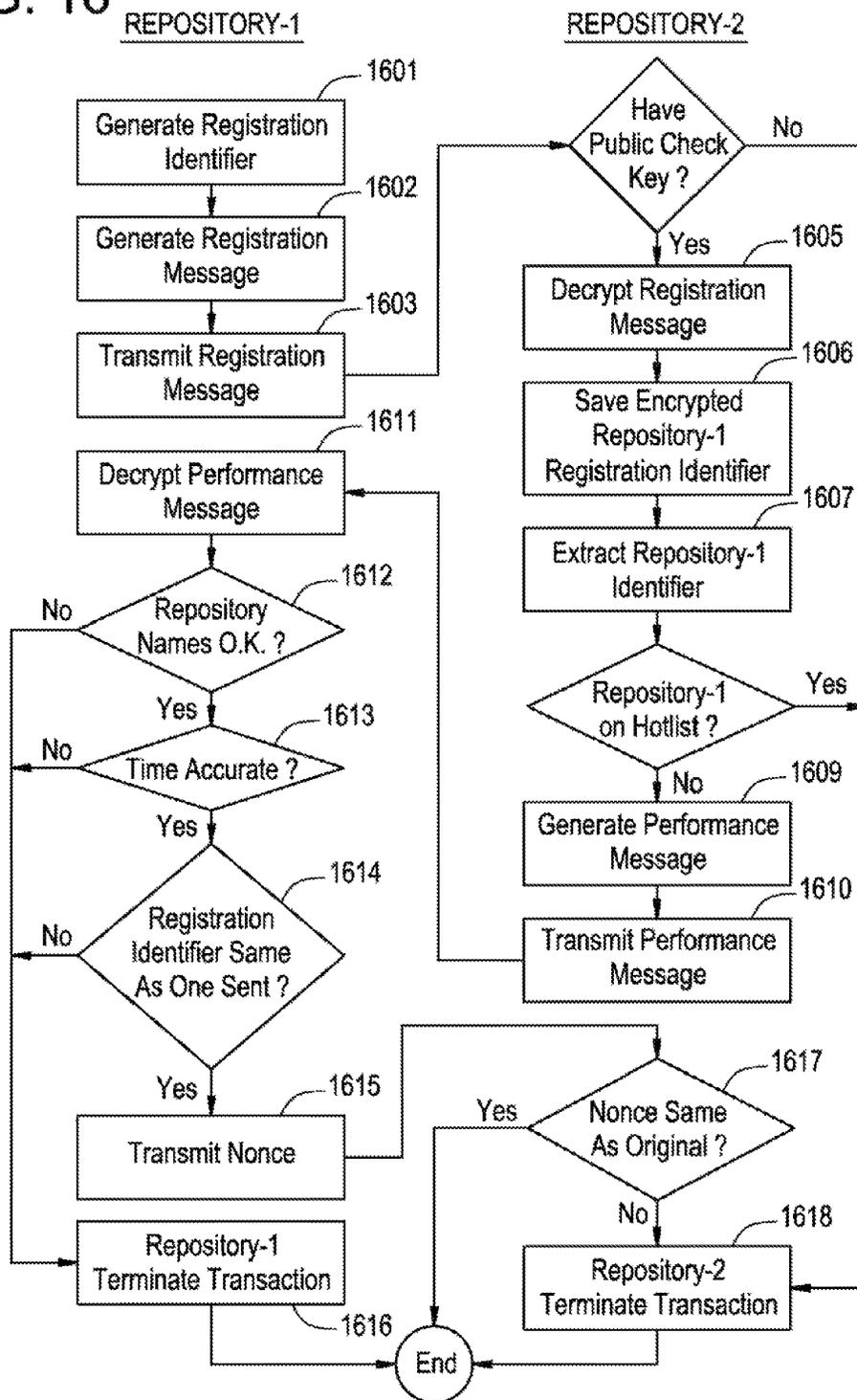


FIG. 17

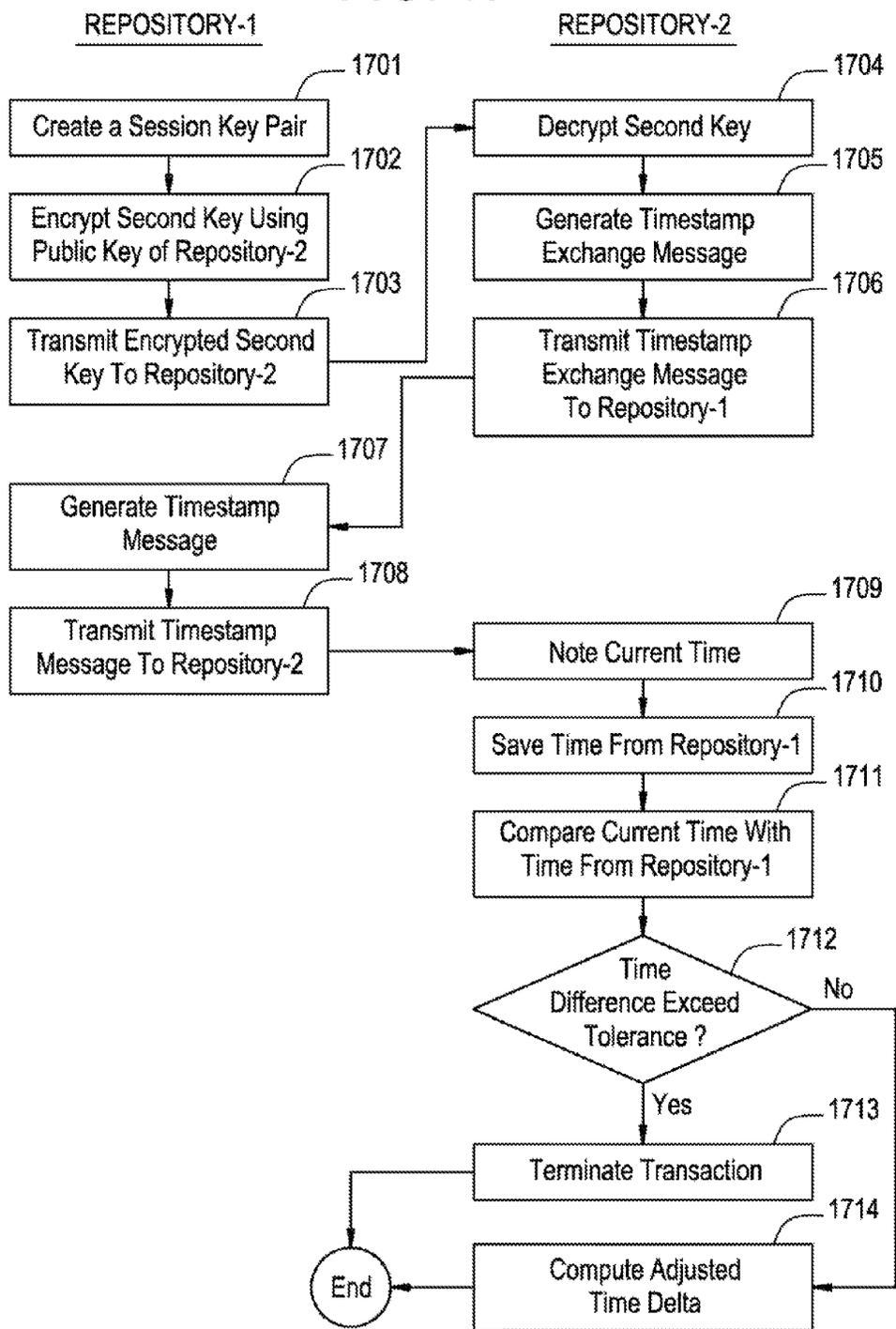
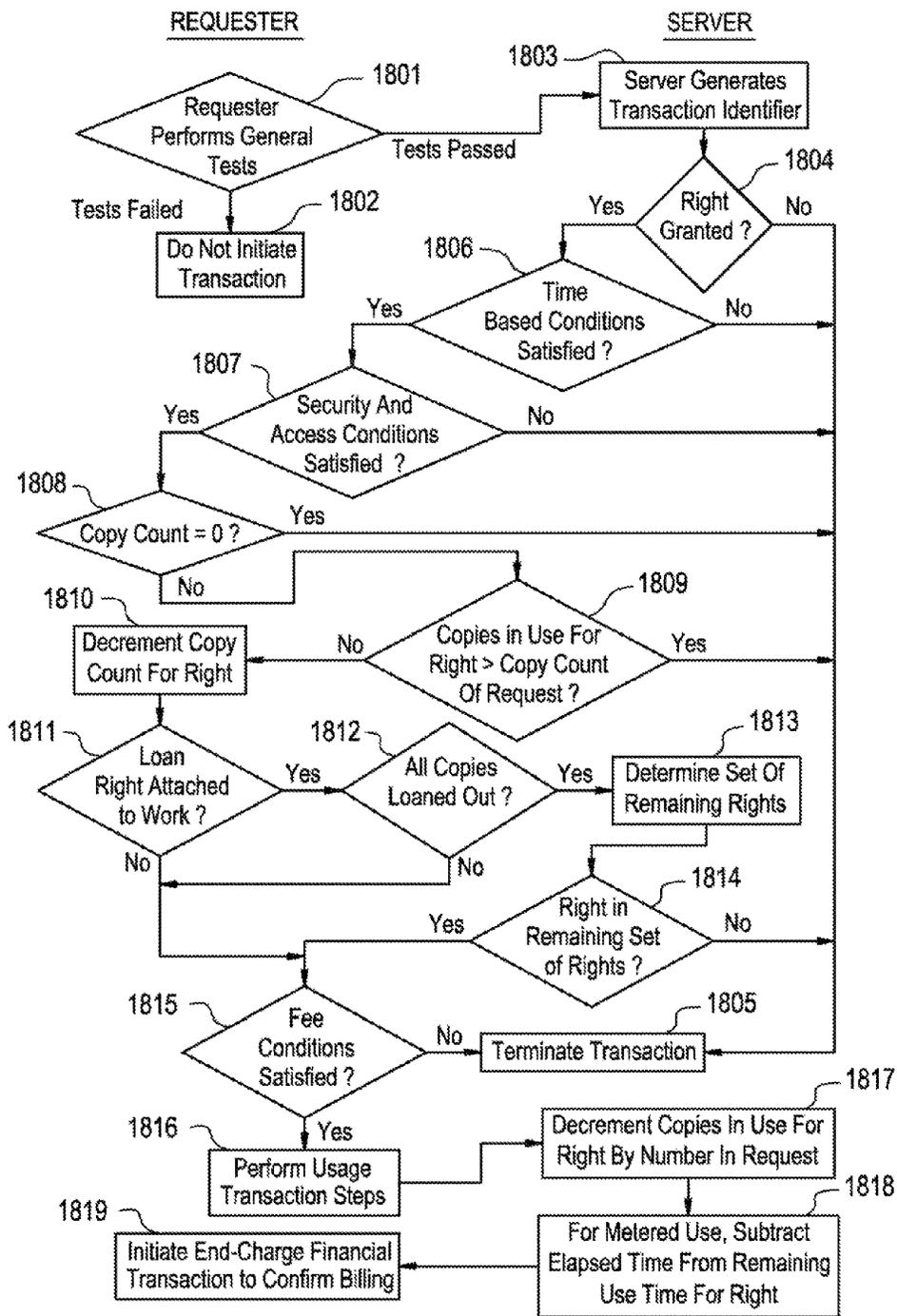


FIG. 18



US 8,370,956 B2

1

SYSTEM AND METHOD FOR RENDERING DIGITAL CONTENT IN ACCORDANCE WITH USAGE RIGHTS INFORMATION

CROSS REFERENCE TO RELATED APPLICATIONS

This application is a continuation of U.S. application Ser. No. 11/304,793, filed Dec. 16, 2005, which is a divisional of U.S. application Ser. No. 11/135,352, filed May 24, 2005, now U.S. Pat. No. 7,266,529, which is a continuation of U.S. application Ser. No. 10/322,759, filed Dec. 19, 2002, now U.S. Pat. No. 6,898,576, which is a continuation of U.S. application Ser. No. 09/778,001, filed Feb. 7, 2001, now U.S. Pat. No. 6,708,157, which is a divisional of U.S. application Ser. No. 08/967,084, filed Nov. 10, 1997, now U.S. Pat. No. 6,236,971, which is a continuation of U.S. application Ser. No. 08/344,760, filed Nov. 23, 1994, now abandoned, the entire disclosures of all of which are hereby incorporated by reference herein.

FIELD OF THE INVENTION

The present invention relates to the field of distribution and usage rights enforcement for digitally encoded works.

BACKGROUND OF THE INVENTION

A fundamental issue facing the publishing and information industries as they consider electronic publishing is how to prevent the unauthorized and unaccounted distribution or usage of electronically published materials. Electronically published materials are typically distributed in a digital form and recreated on a computer based system having the capability to recreate the materials. Audio and video recordings, software, books and multimedia works are all being electronically published. Companies in these industries receive royalties for each accounted for delivery of the materials, e.g. the sale of an audio CD at a retail outlet. Any unaccounted distribution of a work results in an unpaid royalty (e.g. copying the audio recording CD to another digital medium.)

The ease in which electronically published works can be "perfectly" reproduced and distributed is a major concern. The transmission of digital works over networks is commonplace. One such widely used network is the Internet. The Internet is a widespread network facility by which computer users in many universities, corporations and government entities communicate and trade ideas and information. Computer bulletin boards found on the Internet and commercial networks such as CompuServ and Prodigy allow for the posting and retrieving of digital information. Information services such as Dialog and LEXIS/NEXIS provide databases of current information on a wide variety of topics. Another factor which will exacerbate the situation is the development and expansion of the National Information Infrastructure (the NII). It is anticipated that, as the NII grows, the transmission of digital works over networks will increase many times over. It would be desirable to utilize the NII for distribution of digital works without the fear of widespread unauthorized copying.

The most straightforward way to curb unaccounted distribution is to prevent unauthorized copying and transmission. For existing materials that are distributed in digital form, various safeguards are used. In the case of software, copy protection schemes which limit the number of copies that can be made or which corrupt the output when copying is detected have been employed. Another scheme causes software to

2

become disabled after a predetermined period of time has lapsed. A technique used for workstation based software is to require that a special hardware device must be present on the workstation in order for the software to run, e.g., see U.S. Pat. No. 4,932,054 entitled "Method and Apparatus for Protecting Computer Software Utilizing Coded Filter Network in Conjunction with an Active Coded Hardware Device." Such devices are provided with the software and are commonly referred to as dongles.

Yet another scheme is to distribute software, but which requires a "key" to enable its use. This is employed in distribution schemes where "demos" of the software are provided on a medium along with the entire product. The demos can be freely used, but in order to use the actual product, the key must be purchased. These schemes do not hinder copying of the software once the key is initially purchased.

A system for ensuring that licenses are in place for using licensed products is described in PCT Publication WO 93/01550 to Griswold entitled "License Management System and Method." The licensed product may be any electronically published work but is most effective for use with works that are used for extended periods of time such as software programs. Griswold requires that the licensed product contain software to invoke a license check monitor at predetermined time intervals. The license check monitor generates request datagrams which identify the licensee. The request datagrams are sent to a license control system over an appropriate communication facility. The license control system then checks the datagram to determine if the datagram is from a valid licensee. The license control system then sends a reply datagram to the license check monitor indicating denial or approval of usage. The license control system will deny usage in the event that request datagrams go unanswered after a predetermined period of time (which may indicate an unauthorized attempt to use the licensed product). In this system, usage is managed at a central location by the response datagrams. So for example if license fees have not been paid, access to the licensed product is terminated.

It is argued by Griswold that the described system is advantageous because it can be implemented entirely in software. However, the system described by Griswold has limitations. An important limitation is that during the use of the licensed product, the user must always be coupled to an appropriate communication facility in order to send and receive datagrams. This creates a dependency on the communication facility. So if the communication facility is not available, the licensed product cannot be used. Moreover, some party must absorb the cost of communicating with the license server.

A system for controlling the distribution of digitally encoded books is embodied in a system available from VPR Systems, LTD. of St. Louis, Miss. The VPR system is self-contained and is comprised of: (1) point of sale kiosks for storing and downloading of books, (2) personal storage mediums (cartridges) to which the books are downloaded, and (3) readers for viewing the book. In a purchase transaction, a purchaser will purchase a voucher card representing the desired book. The voucher will contain sufficient information to identify the book purchased and perhaps some demographic information relating to the sales transaction. To download the book, the voucher and the cartridge are inserted into the kiosk.

The VPR system may also be used as a library. In such an embodiment, the kiosk manages the number of "copies" that may be checked out at one time. Further, the copy of the book is erased from the user's cartridge after a certain check-out

US 8,370,956 B2

3

time has expired. However, individuals cannot loan books because the cartridges may only be used with the owner's reader.

The foregoing distribution and protection schemes operate in part by preventing subsequent distribution of the work. While this certainly prevents unauthorized distributions, it does so by sacrificing the potential for subsequent revenue bearing uses. For example, it may be desirable to allow the lending of a purchased work to permit exposure of the work to potential buyers. Another example would be to permit the creation of a derivative work for a fee. Yet another example would be to permit copying the work for a fee (essentially purchasing it). Thus, it would be desirable to provide flexibility in how the owner of a digital work may allow it to be distributed.

While flexibility in distribution is a concern, the owners of a work want to make sure they are paid for such distributions. In U.S. Pat. No. 4,977,594 to Shear, entitled "Database Usage Metering and Protection System and Method," a system for metering and billing for usage of information distributed on a CD-ROM is described. The system requires the addition of a billing module to the computer system. The billing module may operate in a number of different ways. First, it may periodically communicate billing data to a central billing facility, whereupon the user may be billed. Second, billing may occur by disconnecting the billing module and the user sending it to a central billing facility where the data is read and a user bill generated.

U.S. Pat. No. 5,247,575, Sprague et al., entitled "Information Distribution System", describes an information distribution system which provides and charges only for user selected information. A plurality of encrypted information packages (IPs) are provided at the user site, via high and/or low density storage media and/or by broadcast transmission. Some of the IPs may be of no interest to the user. The IPs of interest are selected by the user and are decrypted and stored locally. The IPs may be printed, displayed or even copied to other storage media. The charges for the selected IP's are accumulated within a user apparatus and periodically reported by telephone to a central accounting facility. The central accounting facility also issues keys to decrypt the IPs. The keys are changed periodically. If the central accounting facility has not issued a new key for a particular user station, the station is unable to retrieve information from the system when the key is changed.

A system available from Wave Systems Corp. of Princeton, N.Y., provides for metering of software usage on a personal computer. The system is installed onto a computer and collects information on what software is in use, encrypts it and then transmits the information to a transaction center. From the transaction center, a bill is generated and sent to the user. The transaction center also maintains customer accounts so that licensing fees may be forwarded directly to the software providers. Software operating under this system must be modified so that usage can be accounted.

Known techniques for billing do not provide for billing of copies made of the work. For example, if data is copied from the CD-ROM described in Shear, any subsequent use of the copy of the information cannot be metered or billed. In other words, the means for billing runs with the media rather than the underlying work. It would be desirable to have a distribution system where the means for billing is always transported with the work.

SUMMARY OF THE INVENTION

A method, system and software for associating usage rights with digital content is provided, including creating usage

4

rights from a grammar, the usage rights specifying a manner of use indicating purposes for which the digital content is used and/or distributed by an authorized party; associating the usage rights with a digital content; processing a usage transaction specifying the usage rights to determine if access to the digital content is granted; and storing the usage rights in a distributed repository. The usage rights also specify one or more conditions which must be satisfied before the manner of use is exercised. The creating includes selecting symbols from a first set of predetermined symbols to define a valid sequence of symbols to indicate the manner of use, selecting one or more symbols from a second set of predetermined symbols to define a valid sequence of symbols to indicate the conditions.

In further embodiments, a system for controlling the distribution and use of digital works using digital tickets is disclosed. A ticket is an indicator that the ticket holder has already paid for or is otherwise entitled to some specified right, product or service. In the present invention, a "digital ticket" is used to enable the ticket holder to exercise usage rights specifying the requirement of the digital ticket. Usage rights are used to define how a digital work may be used or distributed. Specific instances of usage rights are used to indicate a particular manner of use or distribution. A usage right may specify a digital ticket which must be present before the right may be exercised. For example, a digital ticket may be specified in a Copy right of a digital work, so that exercise of the Copy right requires the party that desires a copy of the digital work be in possession of the necessary digital ticket. After a copy of the digital work is successfully sent to the requesting party, the digital ticket is "punched" to indicate that a copy of the digital work has been made. When the ticket is "punched" a predetermined number of times, it may no longer be used.

Digital works are stored in repositories. Repositories enforce the usage rights for digital works. Each repository has a "generic ticket agent" which punches tickets. In some instances only the generic ticket agent is necessary. In other instances, punching by a "special ticket agent" residing on another repository may be desired. Punching by a "special ticket agent" enables greater security and control of the digital work. For example, it can help prevent digital ticket forgery. Special ticket agents are also useful in situations where an external database needs to be updated or checked.

A digital ticket is merely an instance of a digital work. Thus, a digital ticket may be distributed among repositories in the same fashion as other digital works.

A digital ticket may be used in many commercial scenarios such as in the purchase of software and prepaid upgrades. A digital ticket may also be used to limit the number of times that a right may be exercised. For example, a user may purchase a copy of a digital work, along with the right to make up to 5 Copies. In this case, the Copy right would have associated therewith a digital ticket that can be punched up to 5 times. Other such commercial scenarios will become apparent from the detailed description.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a flowchart illustrating a simple instantiation of the operation of the currently preferred embodiment of the present invention.

FIG. 2 is a block diagram illustrating the various repository types and the repository transaction flow between them in the currently preferred embodiment of the present invention.

US 8,370,956 B2

5

FIG. 3 is a block diagram of a repository coupled with a credit server in the currently preferred embodiment of the present invention.

FIGS. 4a and 4b are examples of rendering systems as may be utilized in the currently preferred embodiment of the present invention.

FIG. 5 illustrates a contents file layout for a digital work as may be utilized in the currently preferred embodiment of the present invention.

FIG. 6 illustrates a contents file layout for an individual digital work of the digital work of FIG. 5 as may be utilized in the currently preferred embodiment of the present invention.

FIG. 7 illustrates the components of a description block of the currently preferred embodiment of the present invention.

FIG. 8 illustrates a description tree for the contents file layout of the digital work illustrated in FIG. 5.

FIG. 9 illustrates a portion of a description tree corresponding to the individual digital work illustrated in FIG. 6.

FIG. 10 illustrates a layout for the rights portion of a description block as may be utilized in the currently preferred embodiment of the present invention.

FIG. 11 is a description tree wherein certain d-blocks have PRINT usage rights and is used to illustrate "strict" and "lenient" rules for resolving usage rights conflicts.

FIG. 12 is a block diagram of the hardware components of a repository as are utilized in the currently preferred embodiment of the present invention.

FIG. 13 is a block diagram of the functional (logical) components of a repository as are utilized in the currently preferred embodiment of the present invention.

FIG. 14 is diagram illustrating the basic components of a usage right in the currently preferred embodiment of the present invention.

FIG. 15 lists the usage rights grammar of the currently preferred embodiment of the present invention.

FIG. 16 is a flowchart illustrating the steps of certificate delivery, hotlist checking and performance testing as performed in a registration transaction as may be performed in the currently preferred embodiment of the present invention.

FIG. 17 is a flowchart illustrating the steps of session information exchange and clock synchronization as may be performed in the currently preferred embodiment of the present invention, after each repository in the registration transaction has successfully completed the steps described in FIG. 16.

FIG. 18 is a flowchart illustrating the basic flow for a usage transaction, including the common opening and closing step, as may be performed in the currently preferred embodiment of the present invention.

FIG. 19 is a state diagram of server and client repositories in accordance with a transport protocol followed when moving a digital work from the server to the client repositories, as may be performed in the currently preferred embodiment of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

Overview

A system for controlling use and distribution of digital works is disclosed. The present invention is directed to supporting commercial transactions involving digital works. The transition to digital works profoundly and fundamentally changes how creativity and commerce can work. It changes the cost of transporting or storing works because digital property is almost "massless." Digital property can be transported

6

at electronic speeds and requires almost no warehousing. Keeping an unlimited supply of virtual copies on hand requires essentially no more space than keeping one copy on hand. The digital medium also lowers the costs of alteration, reuse and billing.

There is a market for digital works because creators are strongly motivated to reuse portions of digital works from others rather than creating their own completely. This is because it is usually so much easier to use an existing stock photo or music clip than to create a new one from scratch.

Herein the terms "digital work", "work" and "content" refer to any work that has been reduced to a digital representation. This would include any audio, video, text, or multimedia work and any accompanying interpreter (e.g. software) that may be required for recreating the work. The term composite work refers to a digital work comprised of a collection of other digital works. The term "usage rights" or "rights" is a term which refers to rights granted to a recipient of a digital work. Generally, these rights define how a digital work can be used and if it can be further distributed. Each usage right may have one or more specified conditions which must be satisfied before the right may be exercised. Appendix 1 provides a Glossary of the terms used herein.

A key feature of the present invention is that usage rights are permanently "attached" to the digital work. Copies made of a digital work will also have usage rights attached. Thus, the usage rights and any associated fees assigned by a creator and subsequent distributor will always remain with a digital work.

The enforcement elements of the present invention are embodied in repositories. Among other things, repositories are used to store digital works, control access to digital works, bill for access to digital works and maintain the security and integrity of the system.

The combination of attached usage rights and repositories enable distinct advantages over prior systems. As noted in the prior art, payment of fees are primarily for the initial access. In such approaches, once a work has been read, computational control over that copy is gone. Metaphorically, "the content genie is out of the bottle and no more fees can be billed." In contrast, the present invention never separates the fee descriptions from the work. Thus, the digital work genie only moves from one trusted bottle (repository) to another, and all uses of copies are potentially controlled and billable.

FIG. 1 is a high level flowchart omitting various details but which demonstrates the basic operation of the present invention. Referring to FIG. 1, a creator creates a digital work, step 101. The creator will then determine appropriate usage rights and fees, attach them to the digital work, and store them in Repository 1, step 102. The determination of appropriate usage rights and fees will depend on various economic factors. The digital work remains securely in Repository 1 until a request for access is received. The request for access begins with a session initiation by another repository. Here a Repository 2 initiates a session with Repository 1, step 103. As will be described in greater detail below, this session initiation includes steps which help to insure that the respective repositories are trustworthy. Assuming that a session can be established, Repository 2 may then request access to the Digital Work for a stated purpose, step 104. The purpose may be, for example, to print the digital work or to obtain a copy of the digital work. The purpose will correspond to a specific usage right. In any event, Repository 1 checks the usage rights associated with the digital work to determine if the access to the digital work may be granted, step 105. The check of the usage rights essentially involves a determination of whether a right associated with the access request has been attached to

US 8,370,956 B2

7

the digital work and if all conditions associated with the right are satisfied. If the access is denied, repository 1 terminates the session with an error message, step 106. If access is granted, repository 1 transmits the digital work to repository 2, step 107. Once the digital work has been transmitted to repository 2, repository 1 and 2 each generate billing information for the access which is transmitted to a credit server, step 108. Such double billing reporting is done to insure against attempts to circumvent the billing process.

FIG. 2 illustrates the basic interactions between repository types in the present invention. As will become apparent from FIG. 2, the various repository types will serve different functions. It is fundamental that repositories will share a core set of functionality which will enable secure and trusted communications. Referring to FIG. 2, a repository 201 represents the general instance of a repository. The repository 201 has two modes of operation; a server mode and a requester mode. When in the server mode, the repository will be receiving and processing access requests to digital works. When in the requester mode, the repository will be initiating requests to access digital works. Repository 201 is general in the sense that its primary purpose is as an exchange medium for digital works. During the course of operation, the repository 201 may communicate with a plurality of other repositories, namely authorization repository 202, rendering repository 203 and master repository 204. Communication between repositories occurs utilizing a repository transaction protocol 205.

Communication with an authorization repository 202 may occur when a digital work being accessed has a condition requiring an authorization. Conceptually, an authorization is a digital certificate such that possession of the certificate is required to gain access to the digital work. An authorization is itself a digital work that can be moved between repositories and subjected to fees and usage rights conditions. An authorization may be required by both repositories involved in an access to a digital work.

Communication with a rendering repository 203 occurs in connection with the rendering of a digital work. As will be described in greater detail below, a rendering repository is coupled with a rendering device (e.g. a printer device) to comprise a rendering system.

Communication with a master repository 205 occurs in connection with obtaining an identification certificate. Identification certificates are the means by which a repository is identified as "trustworthy". The use of identification certificates is described below with respect to the registration transaction.

FIG. 3 illustrates the repository 201 coupled to a credit server 301. The credit server 301 is a device which accumulates billing information for the repository 201. The credit server 301 communicates with repository 201 via billing transactions 302 to record billing transactions. Billing transactions are reported to a billing clearinghouse 303 by the credit server 301 on a periodic basis. The credit server 301 communicates to the billing clearinghouse 303 via clearinghouse transactions 304. The clearinghouse transactions 304 enable a secure and encrypted transmission of information to the billing clearinghouse 303.

Rendering Systems

A rendering system is generally defined as a system comprising a repository and a rendering device which can render a digital work into its desired form. Examples of a rendering system may be a computer system, a digital audio system, or a printer. A rendering system has the same security features as

8

a repository. The coupling of a rendering repository with the rendering device may occur in a manner suitable for the type of rendering device.

FIG. 4a illustrates a printer as an example of a rendering system. Referring to FIG. 4, printer system 401 has contained therein a printer repository 402 and a print device 403. It should be noted that the dashed line defining printer system 401 defines a secure system boundary. Communications within the boundary is assumed to be secure. Depending on the security level, the boundary also represents a barrier intended to provide physical integrity. The printer repository 402 is an instantiation of the rendering repository 205 of FIG. 2. The printer repository 402 will in some instances contain an ephemeral copy of a digital work which remains until it is printed out by the print engine 403. In other instances, the printer repository 402 may contain digital works such as fonts, which will remain and can be billed based on use. This design assures that all communication lines between printers and printing devices are encrypted, unless they are within a physically secure boundary. This design feature eliminates a potential "fault" point through which the digital work could be improperly obtained. The printer device 403 represents the printer components used to create the printed output.

Also illustrated in FIG. 4a is the repository 404. The repository 404 is coupled to the printer repository 402. The repository 404 represents an external repository which contains digital works.

FIG. 4b is an example of a computer system as a rendering system. A computer system may constitute a "multi-function" device since it may execute digital works (e.g. software programs) and display digital works (e.g. a digitized photograph). Logically, each rendering device can be viewed as having its own repository, although only one physical repository is needed. Referring to FIG. 4b, a computer system 410 has contained therein a display/execution repository 411. The display/execution repository 411 is coupled to display device, 412 and execution device 413. The dashed box surrounding the computer system 410 represents a security boundary within which communications are assumed to be secure. The display/execution repository 411 is further coupled to a credit server 414 to report any fees to be billed for access to a digital work and a repository 415 for accessing digital works stored therein.

Structure of Digital Works

Usage rights are attached directly to digital works. Thus, it is important to understand the structure of a digital work. The structure of a digital work, in particular composite digital works, may be naturally organized into an acyclic structure such as a hierarchy. For example, a magazine has various articles and photographs which may have been created and are owned by different persons. Each of the articles and photographs may represent a node in a hierarchical structure. Consequently, controls, i.e. usage rights, may be placed on each node by the creator. By enabling control and fee billing to be associated with each node, a creator of a work can be assured that the rights and fees are not circumvented.

In the currently preferred embodiment, the file information for a digital work is divided into two files: a "contents" file and a "description tree" file. From the perspective of a repository, the "contents" file is a stream of addressable bytes whose format depends completely on the interpreter used to play, display or print the digital work. The description tree file makes it possible to examine the rights and fees for a work without reference to the content of the digital work. It should be noted that the term description tree as used herein refers to

US 8,370,956 B2

9

any type of acyclic structure used to represent the relationship between the various components of a digital work.

FIG. 5 illustrates the layout of a contents file. Referring to FIG. 5, a digital work 509 is comprised of story A 510, advertisement 511, story B 512 and story C 513. It is assumed that the digital work is stored starting at a relative address of 0. Each of the parts of the digital work are stored linearly so that story A 510 is stored at approximately addresses 0-30,000, advertisement 511 at addresses 30,001-40,000, story B 512 at addresses 40,001-60,000 and story C 513 at addresses 60,001-85K. The detail of story A 510 is illustrated in FIG. 6. Referring to FIG. 6, the story A 510 is further broken down to show text 614 stored at address 0-1500, soldier photo 615 at addresses 1501-10,000, graphics 616 stored at addresses 10,001-25,000 and sidebar 617 stored address 25,001-30,000. Note that the data in the contents file may be compressed (for saving storage) or encrypted (for security).

From FIGS. 5 and 6 it is readily observed that a digital work can be represented by its component parts as a hierarchy. The description tree for a digital work is comprised of a set of related descriptor blocks (d-blocks). The contents of each d-block are described with respect to FIG. 7. Referring to FIG. 7, a d-block 700 includes an identifier 701 which is a unique identifier for the work in the repository, a starting address 702 providing the start address of the first byte of the work, a length 703 giving the number of bytes in the work, a rights portion 704 wherein the granted usage rights and their status data are maintained, a parent pointer 705 for pointing to a parent d-block and child pointers 706 for pointing to the child d-blocks. In the currently preferred embodiment, the identifier 701 has two parts. The first part is a unique number assigned to the repository upon manufacture. The second part is a unique number assigned to the work upon creation. The rights portion 704 will contain a data structure, such as a look-up table, wherein the various information associated with a right is maintained. The information required by the respective usage rights is described in more detail below. D-blocks form a strict hierarchy. The top d-block of a work has no parent; all other d-blocks have one parent. The relationship of usage rights between parent and child d-blocks and how conflicts are resolved is described below.

A special type of d-block is a "shell" d-block. A shell d-block adds no new content beyond the content of its parts. A shell d-block is used to add rights and fee information, typically by distributors of digital works.

FIG. 8 illustrates a description tree for the digital work of FIG. 5. Referring to FIG. 8, a top d-block 820 for the digital work points to the various stories and advertisements contained therein. Here, the top d-block 820 points to d-block 821 (representing story A 510), d-block 822 (representing the advertisement 511), d-block 823 (representing story B 512) and d-block 824 (representing story C 513).

The portion of the description tree for Story A 510 is illustrated in FIG. 9. D-block 925 represents text 614, d-block 926 represents photo 615, d-block 927 represents graphics 616 by and d-block 928 represents sidebar 617.

The rights portion 704 of a descriptor block is further illustrated in FIG. 10. FIG. 10 illustrates a structure which is repeated in the rights portion 704 for each right. Referring to FIG. 10, each right will have a right code field 1001 and status information field 1002. The right code field 1001 will contain a unique code assigned to a right. The status information field 1002 will contain information relating to the state of a right and the digital work. Such information is indicated below in Table 1. The rights as stored in the rights portion 304 may typically be in numerical order based on the right code.

10

The approach for representing digital works by separating description data from content assumes that parts of a file are contiguous but takes no position on the actual representation of content. In particular, it is neutral to the question of whether content representation may take an object oriented approach. It would be natural to represent content as objects. In principle, it may be convenient to have content objects that include the billing structure and rights information that is represented in the d-blocks. Such variations in the design of the representation are possible and are viable alternatives but may introduce processing overhead, e.g. the interpretation of the objects.

TABLE 1

DIGITAL WORK STATE INFORMATION		
Property	Value	Use
Copies-in-Use	Number	A counter of the number of copies of a work that are in use. Incremented when another copy is used; decremented when use is completed.
Loan-Period	Time-Units	Indicator of the maximum number of time-units that a document can be loaned out
Loaner-Copy	Boolean	Indicator that the current work is a loaned out copy of an authorized digital work.
Remaining-Time	Time-Units	Indicator of the remaining time of use on a metered document right.
Document-Descr	String	A string containing various identifying information about a document. The exact format of this is not specified, but it can include information such as a publisher name, author name, ISBN number, and so on.
Revenue-Owner	RO-Descr	A handle identifying a revenue owner for a digital work. This is used for reporting usage fees.
Publication-Date	Date-Descr	The date that the digital work was published.
History-list	History-Rec	A list of events recording the repositories and dates for operations that copy, transfer, backup, or restore a digital work.

Digital works are stored in a repository as part of a hierarchical file system. Folders (also termed directories and sub-directories) contain the digital works as well as other folders. Digital works and folders in a folder are ordered in alphabetical order. The digital works are typed to reflect how the files are used. Usage rights can be attached to folders so that the folder itself is treated as a digital work. Access to the folder would then be handled in the same fashion as any other digital work. As will be described in more detail below, the contents of the folder are subject to their own rights. Moreover, file management rights may be attached to the folder which defines how folder contents can be managed.

Attaching Usage Rights to a Digital Work

It is fundamental to the present invention that the usage rights are treated as part of the digital work. As the digital work is distributed, the scope of the granted usage rights will remain the same or may be narrowed. For example, when a digital work is transferred from a document server to a repository, the usage rights may include the right to loan a copy for a predetermined period of time (called the original rights). When the repository loans out a copy of the digital work, the usage rights in the loaner copy (called the next set of rights) could be set to prohibit any further rights to loan out the copy. The basic idea is that one cannot grant more rights than they have.

US 8,370,956 B2

11

The attachment of usage rights into a digital work may occur in a variety of ways. If the usage rights will be the same for an entire digital work, they could be attached when the digital work is processed for deposit in the digital work server. In the case of a digital work having different usage rights for the various components, this can be done as the digital work is being created. An authoring tool or digital work assembling tool could be utilized which provides for an automated process of attaching the usage rights.

As will be described below, when a digital work is copied, transferred or loaned, a “next set of rights” can be specified. The “next set of rights” will be attached to the digital work as it is transported.

Resolving Conflicting Rights

Because each part of a digital work may have its own usage rights, there will be instances where the rights of a “contained part” are different from its parent or container part. As a result, conflict rules must be established to dictate when and how a right may be exercised. The hierarchical structure of a digital work facilitates the enforcement of such rules. A “strict” rule would be as follows: a right for a part in a digital work is sanctioned if and only if it is sanctioned for the part, for ancestor d-blocks containing the part and for all descendent d-blocks. By sanctioned, it is meant that (1) each of the respective parts must have the right, and (2) any conditions for exercising the right are satisfied.

It also possible to implement the present invention using a more lenient rule. In the more lenient rule, access to the part may be enabled to the descendent parts which have the right, but access is denied to the descendents which do not.

Example of applying both the strict rule and lenient is illustrated with reference to FIG. 11. Referring to FIG. 11, a root d-block 1101 has child d-blocks 1102-1105. In this case, root d-block represents a magazine, and each of the child d-blocks 1102-1105 represent articles in the magazine. Suppose that a request is made to PRINT the digital work represented by root d-block 1101 wherein the strict rule is followed. The rights for the root d-block 1101 and child d-blocks 1102-1105 are then examined. Root d-block 1101 and child d-blocks 1102 and 1105 have been granted PRINT rights. Child d-block 1103 has not been granted PRINT rights and child d-block 1104 has PRINT rights conditioned on payment of a usage fee.

Under the strict rule the PRINT right cannot be exercised because the child d-block does not have the PRINT right. Under the lenient rule, the result would be different. The digital works represented by child d-blocks 1102 and 1105 could be printed and the digital work represented by d-block 1104 could be printed so long as the usage fee is paid. Only the digital work represented by d-block 1103 could not be printed. This same result would be accomplished under the strict rule if the requests were directed to each of the individual digital works.

The present invention supports various combinations of allowing and disallowing access. Moreover, as will be described below, the usage rights grammar permits the owner of a digital work to specify if constraints may be imposed on the work by a container part. The manner in which digital works may be sanctioned because of usage rights conflicts would be implementation specific and would depend on the nature of the digital works.

Repositories

Many of the powerful functions of repositories—such as their ability to “loan” digital works or automatically handle the commercial reuse of digital works—are possible because

12

they are trusted systems. The systems are trusted because they are able to take responsibility for fairly and reliably carrying out the commercial transactions. That the systems can be responsible (“able to respond”) is fundamentally an issue of integrity. The integrity of repositories has three parts: physical integrity, communications integrity, and behavioral integrity.

Physical integrity refers to the integrity of the physical devices themselves. Physical integrity applies both to the repositories and to the protected digital works. Thus, the higher security classes of repositories themselves may have sensors that detect when tampering is attempted on their secure cases. In addition to protection of the repository itself, the repository design protects access to the content of digital works. In contrast with the design of conventional magnetic and optical devices—such as floppy disks, CD-ROMs, and videotapes—repositories never allow non-trusted systems to access the works directly. A maker of generic computer systems cannot guarantee that their platform will not be used to make unauthorized copies. The manufacturer provides generic capabilities for reading and writing information, and the general nature of the functionality of the general computing device depends on it. Thus, a copy program can copy arbitrary data. This copying issue is not limited to general purpose computers. It also arises for the unauthorized duplication of entertainment “software” such as video and audio recordings by magnetic recorders. Again, the functionality of the recorders depends on their ability to copy and they have no means to check whether a copy is authorized. In contrast, repositories prevent access to the raw data by general devices and can test explicit rights and conditions before copying or otherwise granting access. Information is only accessed by protocol between trusted repositories.

Communications integrity refers to the integrity of the communications channels between repositories. Roughly speaking, communications integrity means that repositories cannot be easily fooled by “telling them lies.” Integrity in this case refers to the property that repositories will only communicate with other devices that are able to present proof that they are certified repositories, and furthermore, that the repositories monitor the communications to detect “impostors” and malicious or accidental interference. Thus the security measures involving encryption, exchange of digital certificates, and nonces described below are all security measures aimed at reliable communication in a world known to contain active adversaries.

Behavioral integrity refers to the integrity in what repositories do. What repositories do is determined by the software that they execute. The integrity of the software is generally assured only by knowledge of its source. Restated, a user will trust software purchased at a reputable computer store but not trust software obtained off a random (insecure) server on a network. Behavioral integrity is maintained by requiring that repository software be certified and be distributed with proof of such certification, i.e. a digital certificate. The purpose of the certificate is to authenticate that the software has been tested by an authorized organization, which attests that the software does what it is supposed to do and that it does not compromise the behavioral integrity of a repository. If the digital certificate cannot be found in the digital work or the master repository which generated the certificate is not known to the repository receiving the software, then the software cannot be installed.

In the description of FIG. 2, it was indicated that repositories come in various forms. All repositories provide a core set of services for the transmission of digital works. The manner in which digital works are exchanged is the basis for all

US 8,370,956 B2

13

transaction between repositories. The various repository types differ in the ultimate functions that they perform. Repositories may be devices themselves, or they may be incorporated into other systems. An example is the rendering repository 205 of FIG. 2.

A repository will have associated with it a repository identifier. Typically, the repository identifier would be a unique number assigned to the repository at the time of manufacture. Each repository will also be classified as being in a particular security class. Certain communications and transactions may be conditioned on a repository being in a particular security class. The various security classes are described in greater detail below.

As a prerequisite to operation, a repository will require possession of an identification certificate. Identification certificates are encrypted to prevent forgery and are issued by a Master repository. A master repository plays the role of an authorization agent to enable repositories to receive digital works. Identification certificates must be updated on a periodic basis. Identification certificates are described in greater detail below with respect to the registration transaction.

A repository has both a hardware and functional embodiment. The functional embodiment is typically software executing on the hardware embodiment. Alternatively, the functional embodiment may be embedded in the hardware embodiment such as an Application Specific Integrated Circuit (ASIC) chip.

The hardware embodiment of a repository will be enclosed in a secure housing which if compromised, may cause the repository to be disabled. The basic components of the hardware embodiment of a repository are described with reference to FIG. 12. Referring to FIG. 12, a repository is comprised of a processing means 1200, storage system 1207, clock 1205 and external interface 1206. The processing means 1200 is comprised of a processor element 1201 and processor memory 1202. The processing means 1201 provides controller, repository transaction and usage rights transaction functions for the repository. Various functions in the operation of the repository such as decryption and/or decompression of digital works and transaction messages are also performed by the processing means 1200. The processor element 1201 may be a microprocessor or other suitable computing component. The processor memory 1202 would typically be further comprised of Read Only Memories (ROM) and Random Access Memories (RAM). Such memories would contain the software instructions utilized by the processor element 1201 in performing the functions of the repository.

The storage system 1207 is further comprised of descriptor storage 1203 and content storage 1204. The description tree storage 1203 will store the description tree for the digital work and the content storage will store the associated content. The description tree storage 1203 and content storage 1204 need not be of the same type of storage medium, nor are they necessarily on the same physical device. So for example, the descriptor storage 1203 may be stored on a solid state storage (for rapid retrieval of the description tree information), while the content storage 1204 may be on a high capacity storage such as an optical disk.

The clock 1205 is used to time-stamp various time based conditions for usage rights or for metering usage fees which may be associated with the digital works. The clock 1205 will have an uninterruptible power supply, e.g. a battery, in order to maintain the integrity of the time-stamps. The external interface means 1206 provides for the signal connection to other repositories and to a credit server. The external interface means 1206 provides for the exchange of signals via such

14

standard interfaces such as RS-232 or Personal Computer Manufacturers Card Industry Association (PCMCIA) standards, or FDDI. The external interface means 1206 may also provide network connectivity.

The functional embodiment of a repository is described with reference to FIG. 13. Referring to FIG. 13, the functional embodiment is comprised of an operating system 1301, core repository services 1302, usage transaction handlers 1303, repository specific functions, 1304 and a user interface 1305. The operating system 1301 is specific to the repository and would typically depend on the type of processor being used. The operating system 1301 would also provide the basic services for controlling and interfacing between the basic components of the repository.

The core repository services 1302 comprise a set of functions required by each and every repository. The core repository services 1302 include the session initiation transactions which are defined in greater detail below. This set of services also includes a generic ticket agent which is used to "punch" a digital ticket and a generic authorization server for processing authorization specifications. Digital tickets and authorizations are specific mechanisms for controlling the distribution and use of digital works and are described in more detail below. Note that coupled to the core repository services are a plurality of identification certificates 1306. The identification certificates 1306 are required to enable the use of the repository.

The usage transactions handler 1303 comprise functionality for processing access requests to digital works and for billing fees based on access. The usage transactions supported will be different for each repository type. For example, it may not be necessary for some repositories to handle access requests for digital works.

The repository specific functionality 1304 comprises functionality that is unique to a repository. For example, the master repository has special functionality for issuing digital certificates and maintaining encryption keys. The repository specific functionality 1304 would include the user interface implementation for the repository.

Repository Security Classes

For some digital works the losses caused by any individual instance of unauthorized copying is insignificant and the chief economic concern lies in assuring the convenience of access and low-overhead billing. In such cases, simple and inexpensive handheld repositories and network-based workstations may be suitable repositories, even though the measures and guarantees of security are modest.

At the other extreme, some digital works such as a digital copy of a first run movie or a bearer bond or stock certificate would be of very high value so that it is prudent to employ caution and fairly elaborate security measures to ensure that they are not copied or forged. A repository suitable for holding such a digital work could have elaborate measures for ensuring physical integrity and for verifying authorization before use.

By arranging a universal protocol, all kinds of repositories can communicate with each other in principle. However, creators of some works will want to specify that their works will only be transferred to repositories whose level of security is high enough. For this reason, document repositories have a ranking system for classes and levels of security. The security classes in the currently preferred embodiment are described in Table 2.

US 8,370,956 B2

15

TABLE 2

REPOSITORY SECURITY LEVELS	
Level	Description of Security
0	Open system. Document transmission is unencrypted. No digital certificate is required for identification. The security of the system depends mostly on user honesty, since only modest knowledge may be needed to circumvent the security measures. The repository has no provisions for preventing unauthorized programs from running and accessing or copying files. The system does not prevent the use of removable storage and does not encrypt stored files.
1	Minimal security. Like the previous class except that stored files are minimally encrypted, including ones on removable storage.
2	Basic security. Like the previous class except that special tools and knowledge are required to compromise the programming, the contents of the repository, or the state of the clock. All digital communications are encrypted. A digital certificate is provided as identification. Medium level encryption is used. Repository identification number is unforgeable.
3	General security. Like the previous class plus the requirement of special tools are needed to compromise the physical integrity of the repository and that modest encryption is used on all transmissions. Password protection is required to use the local user interface. The digital clock system cannot be reset without authorization. No works would be stored on removable storage. When executing works as programs, it runs them in their own address space and does not give them direct access to any file storage or other memory containing system code or works. They can access works only through the transmission transaction protocol.
4	Like the previous class except that high level encryption is used on all communications. Sensors are used to record attempts at physical and electronic tampering. After such tampering, the repository will not perform other transactions until it has reported such tampering to a designated server.
5	Like the previous class except that if the physical or digital attempts at tampering exceed some preset thresholds that threaten the physical integrity of the repository or the integrity of digital and cryptographic barriers, then the repository will save only document description records of history but will erase or destroy any digital identifiers that could be misused if released to an unscrupulous party. It also modifies any certificates of authenticity to indicate that the physical system has been compromised. It also erases the contents of designated documents.
6	Like the previous class except that the repository will attempt wireless communication to report tampering and will employ noisy alarms.
10	This would correspond to a very high level of security. This server would maintain constant communications to remote security systems reporting transactions, sensor readings, and attempts to circumvent security.

The characterization of security levels described in Table 2 is not intended to be fixed. More important is the idea of having different security levels for different repositories. It is anticipated that new security classes and requirements will evolve according to social situations and changes in technology.

Repository User Interface

A user interface is broadly defined as the mechanism by which a user interacts with a repository in order to invoke transactions to gain access to a digital work, or exercise usage rights. As described above, a repository may be embodied in various forms. The user interface for a repository will differ depending on the particular embodiment. The user interface may be a graphical user interface having icons representing the digital works and the various transactions that may be performed. The user interface may be a generated dialog in which a user is prompted for information.

The user interface itself need not be part of the repository. As a repository may be embedded in some other device, the user interface may merely be a part of the device in which the repository is embedded. For example, the repository could be embedded in a "card" that is inserted into an available slot in

16

a computer system. The user interface may be combination of a display, keyboard, cursor control device and software executing on the computer system.

At a minimum, the user interface must permit a user to input information such as access requests and alpha numeric data and provide feedback as to transaction status. The user interface will then cause the repository to initiate the suitable transactions to service the request. Other facets of a particular user interface will depend on the functionality that a repository will provide.

Credit Servers

In the present invention, fees may be associated with the exercise of a right. The requirement for payment of fees is described with each version of a usage right in the usage rights language. The recording and reporting of such fees is performed by the credit server. One of the capabilities enabled by associating fees with rights is the possibility of supporting a wide range of charging models. The simplest model, used by conventional software, is that there is a single fee at the time of purchase, after which the purchaser obtains unlimited rights to use the work as often and for as long as he or she wants. Alternative models, include metered use and variable fees. A single work can have different fees for different uses. For example, viewing a photograph on a display could have different fees than making a hardcopy or including it in a newly created work. A key to these alternative charging models is to have a low overhead means of establishing fees and accounting for credit on these transactions.

A credit server is a computational system that reliably authorizes and records these transactions so that fees are billed and paid. The credit server reports fees to a billing clearinghouse. The billing clearinghouse manages the financial transactions as they occur. As a result, bills may be generated and accounts reconciled. Preferably, the credit server would store the fee transactions and periodically communicate via a network with billing clearinghouse for reconciliation. In such an embodiment, communications with the billing clearinghouse would be encrypted for integrity and security reasons. In another embodiment, the credit server acts as a "debit card" where transactions occur in "real-time" against a user account.

A credit server is comprised of memory, a processing means, a clock, and interface means for coupling to a repository and a financial institution (e.g. a modem). The credit server will also need to have security and authentication functionality. These elements are essentially the same elements as those of a repository. Thus, a single device can be both a repository and a credit server, provided that it has the appropriate processing elements for carrying out the corresponding functions and protocols. Typically, however, a credit server would be a card-sized system in the possession of the owner of the credit. The credit server is coupled to a repository and would interact via financial transactions as described below. Interactions with a financial institution may occur via protocols established by the financial institutions themselves.

In the currently preferred embodiment credit servers associated with both the server and the repository report the financial transaction to the billing clearinghouse. For example, when a digital work is copied by one repository to another for a fee, credit servers coupled to each of the repositories will report the transaction to the billing clearinghouse. This is desirable in that it insures that a transaction will be accounted for in the event of some break in the communication between a credit server and the billing clearinghouse. However, some implementations may embody only a single credit server

US 8,370,956 B2

17

reporting the transaction to minimize transaction processing at the risk of losing some transactions.

Usage Rights Language

The present invention uses statements in a high level "usage rights language" to define rights associated with digital works and their parts. Usage rights statements are interpreted by repositories and are used to determine what transactions can be successfully carried out for a digital work and also to determine parameters for those transactions. For example, sentences in the language determine whether a given digital work can be copied, when and how it can be used, and what fees (if any) are to be charged for that use. Once the usage rights statements are generated, they are encoded in a suitable form for accessing during the processing of transactions.

Defining usage rights in terms of a language in combination with the hierarchical representation of a digital work enables the support of a wide variety of distribution and fee schemes. An example is the ability to attach multiple versions of a right to a work. So a creator may attach a PRINT right to make 5 copies for \$10.00 and a PRINT right to make unlimited copies for \$100.00. A purchaser may then choose which option best fits his needs. Another example is that rights and fees are additive. So in the case of a composite work, the rights and fees of each of the components works is used in determining the rights and fees for the work as a whole. Other features and benefits of the usage rights language will become apparent in the description of distribution and use scenarios provided below.

The basic contents of a right are illustrated in FIG. 14. Referring to FIG. 14, a right 1450 has a transactional component 1451 and a specifications component 1452. A right 1450 has a label (e.g. COPY or PRINT) which indicate the use or distribution privileges that are embodied by the right. The transactional component 1451 corresponds to a particular way in which a digital work may be used or distributed. The transactional component 1451 is typically embodied in software instructions in a repository which implement the use or distribution privileges for the right. The specifications components 1452 are used to specify conditions which must be satisfied prior to the right being exercised or to designate various transaction related parameters. In the currently preferred embodiment, these specifications include copy count 1453, Fees and Incentives 1454, Time 1455, Access and Security 1456 and Control 1457. Each of these specifications will be described in greater detail below with respect to the language grammar elements.

The usage rights language is based on the grammar described below. A grammar is a convenient means for defining valid sequence of symbols for a language. In describing the grammar the notation "[a|b|c]" is used to indicate distinct choices among alternatives. In this example, a sentence can have either an "a", "b" or "c". It must include exactly one of them. The braces { } are used to indicate optional items. Note that brackets, bars and braces are used to describe the language of usage rights sentences but do not appear in actual sentences in the language.

In contrast, parentheses are part of the usage rights language. Parentheses are used to group items together in lists. The notation (x*) is used to indicate a variable length list, that is, a list containing one or more items of type x. The notation (x)* is used to indicate a variable number of lists containing x.

Keywords in the grammar are words followed by colons. Keywords are a common and very special case in the language. They are often used to indicate a single value, typically

18

an identifier. In many cases, the keyword and the parameter are entirely optional. When a keyword is given, it often takes a single identifier as its value. In some cases, the keyword takes a list of identifiers.

5 In the usage rights language, time is specified in an hours:minutes:seconds (or hh:mm:ss) representation. Time zone indicators, e.g. PDT for Pacific Daylight Time, may also be specified. Dates are represented as year/month/day (or YYYY/MMM/DD). Note that these time and date representations may specify moments in time or units of time Money units are specified in terms of dollars.

10 Finally, in the usage rights language, various "things" will need to interact with each other. For example, an instance of a usage right may specify a bank account, a digital ticket, etc. Such things need to be identified and are specified herein using the suffix "-ID."

15 The Usage Rights Grammar is listed in its entirety in FIG. 15 and is described below.

Grammar element 1501 "Digital Work Rights:=(Rights*)" define the digital work rights as a set of rights. The-set of rights attached to a digital work define how that digital work may be transferred, used, performed or played. A set of rights will attach to the entire digital work and in the case of compound digital works, each of the components of the digital work. The usage rights of components of a digital may be different.

20 Grammar element 1502 "Right:=(Right-Code {Copy-Count} {Control-Spec} {Time-Spec} {Access-Spec} {Fee-Spec})" enumerates the content of a right. Each usage right must specify a right code. Each right may also optionally specify conditions which must be satisfied before the right can be exercised. These conditions are copy count, control, time, access and fee conditions. In the currently preferred embodiment, for the optional elements, the following defaults apply: copy count equals 1, no time limit on the use of the right, no access tests or a security level required to use the right and no fee is required. These conditions will each be described in greater detail below.

25 It is important to note that a digital work may have multiple versions of a right, each having the same right code. The multiple versions would provide alternative conditions and fees for accessing the digital work.

30 A Grammar element 1503 "Right-Code:=Render-Code|Transport-Code|File-Management-Code|Derivative-Works-Code Configuration-Code" distinguishes each of the specific rights into a particular right type (although each right is identified by distinct right codes). In this way, the grammar provides a catalog of possible rights that can be associated with parts of digital works. In the following, rights are divided into categories for convenience in describing them.

35 Grammar element 1504 "Render-Code:={Player: Player-ID}|Print: {Printer: Printer-ID}" lists a category of rights all involving the making of ephemeral, transitory, or non-digital copies of the digital work. After use the copies are erased.

40 Play: A process of rendering or performing a digital work on some processor. This includes such things as playing digital movies, playing digital music, playing a video game, running a computer program, or displaying a document on a display.

45 Print: To render the work in a medium that is not further protected by usage rights, such as printing on paper.

50 Grammar element 1505 "Transport-Code:=[Copy|Transfer|Loan {Remaining-Rights: Next-Set-of-Rights}] {(Next-Copy-Rights: Next-Set of Rights)}" lists a category of rights involving the making of persistent, usable copies of the digital work on other repositories. The optional

US 8,370,956 B2

19

Next-Copy-Rights determine the rights on the work after it is transported. If this is not specified, then the rights on the transported copy are the same as on the original. The optional Remaining-Rights specify the rights that remain with a digital work when it is loaned out. If this is not specified, then the default is that no rights can be exercised when it is loaned out.

Copy: Make a new copy of a work

Transfer: Moving a work from one repository to another.

Loan: Temporarily loaning a copy to another repository for a specified period of time.

Grammar element **1506** “File-Management-Code:=Backup {Back-Up-Copy-Rights: Next-Set-of-Rights}|Restore|Delete|Folder|Directory {Name:Hide-Local|Hide-Remote} {Parts:Hide-Local|Hide-Remote}” lists a category of rights involving operations for file management, such as the making of backup copies to protect the copy owner against catastrophic equipment failure.

Many software licenses and also copyright law give a copy owner the right to make backup copies to protect against catastrophic failure of equipment. However, the making of uncontrolled backup copies is inherently at odds with the ability to control usage, since an uncontrolled backup copy can be kept and then restored even after the authorized copy was sold.

The File management rights enable the making and restoring of backup copies in a way that respects usage rights, honoring the requirements of both the copy owner and the rights grantor and revenue owner. Backup copies of work descriptions (including usage rights and fee data) can be sent under appropriate protocol and usage rights control to other document repositories of sufficiently high security. Further rights permit organization of digital works into folders which themselves are treated as digital works and whose contents may be “hidden” from a party seeking to determine the contents of a repository.

Backup: To make a backup copy of a digital work as protection against media failure.

Restore: To restore a backup copy of a digital work.

Delete: To delete or erase a copy of a digital work.

Folder: To create and name folders, and to move files and folders between folders.

Directory: To hide a folder or it’s contents.

Grammar element **1507** “Derivative-Works-Code: [Extract|Embed|Edit {Process: Process-ID}] {Next-Copy-Rights: Next-Set-of-Rights}” lists a category of rights involving the use of a digital work to create new works.

Extract: To remove a portion of a work, for the purposes of creating a new work.

Embed: To include a work in an existing work.

Edit: To alter a digital work by copying, selecting and modifying portions of an existing digital work.

Grammar element **1508** “Configuration-Code:=Install|Uninstall” lists a category of rights for installing and uninstalling software on a repository (typically a rendering repository.) This would typically occur for the installation of a new type of player within the rendering repository.

Install: To install new software on a repository.

Uninstall: To remove existing software from a repository.

Grammar element **1509** “Next-Set-of-Rights:={Add: Set-Of-Rights} {Delete: Set-Of-Rights} {Replace: Set-Of-Rights} {(Keep: Set-Of-Rights)}” defines how rights are carried forward for a copy of a digital work. If the Next-Copy-Rights is not specified, the rights for the next copy are the same as those of the current copy. Otherwise, the set of rights for the next copy can be specified. Versions of rights after Add: are added to the current set of rights. Rights after Delete: are deleted from the current set of rights. If only right codes

20

are listed after Delete:, then all versions of rights with those codes are deleted. Versions of rights after Replace: subsume all versions of rights of the same type in the current set of rights.

If Remaining-Rights is not specified, then there are no rights for the original after all Loan copies are loaned out. If Remaining-Rights is specified, then the Keep: token can be used to simplify the expression of what rights to keep behind. A list of right codes following keep means that all of the versions of those listed rights are kept in the remaining copy. This specification can be overridden by subsequent Delete: or Replace: specifications.

Copy Count Specification

For various transactions, it may be desirable to provide some limit as to the number of “copies” of the work which may be exercised simultaneously for the right. For example, it may be desirable to limit the number of copies of a digital work that may be loaned out at a time or viewed at a time.

Grammar element **1510** “Copy-Count:=(Copies: positive-integer|0|unlimited)” provides a condition which defines the number of “copies” of a work subject to the right. A copy count can be 0, a fixed number, or unlimited. The copy-count is associated with each right, as opposed to there being just a single copy-count for the digital work. The Copy-Count for a right is decremented each time that a right is exercised. When the Copy-Count equals zero, the right can no longer be exercised. If the Copy-Count is not specified, the default is one.

Control Specification

Rights and fees depend in general on rights granted by the creator as well as further restrictions imposed by later distributors. Control specifications deal with interactions between the creators and their distributors governing the imposition of further restrictions and fees. For example, a distributor of a digital work may not want an end consumer of a digital work to add fees or otherwise profit by commercially exploiting the purchased digital work.

Grammar element **1511** “Control-Spec:=(Control: {Restrictable|Unrestrictable} {Unchargeable|Chargeable}-)” provides a condition to specify the effect of usage rights and fees of parents on the exercise of the right. A digital work is restrictable if higher level d-blocks can impose further restrictions (time specifications and access specifications) on the right. It is unrestrictable if no further restrictions can be imposed. The default setting is restrictable. A right is unchargeable if no more fees can be imposed on the use of the right. It is chargeable if more fees can be imposed. The default is chargeable.

Time Specification

It is often desirable to assign a start date or specify some duration as to when a right may be exercised. Grammar element **1512** “Time-Spec:=(Fixed-Interval|Sliding-Interval|Meter-Time) Until: Expiration-Date)” provides for specification of time conditions on the exercise of a right. Rights may be granted for a specified time. Different kinds of time specifications are appropriate for different kinds of rights. Some rights may be exercised during a fixed and predetermined duration. Some rights may be exercised for an interval that starts the first time that the right is invoked by some transaction. Some rights may be exercised or are charged according to some kind of metered time, which may be split into separate intervals. For example, a right to view a picture for an hour might be split into six ten minute viewings or four fifteen minute viewings or twenty three minute viewings.

The terms “time” and “date” are used synonymously to refer to a moment in time. There are several kinds of time specifications. Each specification represents some limitation

US 8,370,956 B2

21

on the times over which the usage right applies. The Expiration-Date specifies the moment at which the usage right ends. For example, if the Expiration-Date is “Jan. 1, 1995,” then the right ends at the first moment of 1995. If the Expiration-Date is specified as *forever*, then the rights are interpreted as continuing without end. If only an expiration date is given, then the right can be exercised as often as desired until the expiration date.

Grammar element 1513 “Fixed-Interval:=From: Start-Time” is used to define a predetermined interval that runs from the start time to the expiration date.

Grammar element 1514 “Sliding-Interval:=Interval: Use-Duration” is used to define an indeterminate (or “open”) start time. It sets limits on a continuous period of time over which the contents are accessible. The period starts on the first access and ends after the duration has passed or the expiration date is reached, whichever comes first. For example, if the right gives 10 hours of continuous access, the use-duration would begin when the first access was made and end 10 hours later.

Grammar element 1515 “Meter-Time:=Time-Remaining: Remaining-Use” is used to define a “meter time,” that is, a measure of the time that the right is actually exercised. It differs from the Sliding-Interval specification in that the time that the digital work is in use need not be continuous. For example, if the rights guarantee three days of access, those days could be spread out over a month. With this specification, the rights can be exercised until the meter time is exhausted or the expiration date is reached, whichever comes first.

Remaining-Use:=Time-Unit

Start-Time:=Time-Unit

Use-Duration:=Time-Unit

All of the time specifications include time-unit specifications in their ultimate instantiation.

Security Class and Authorization Specification

The present invention provides for various security mechanisms to be introduced into a distribution or use scheme. Grammar element 1516 “Access-Spec:={SC: Security-Class} {Authorization: Authorization-ID*} {Other-Authorization: Authorization-ID*} {Ticket: Ticket-ID}” provides a means for restricting access and transmission. Access specifications can specify a required security class for a repository to exercise a right or a required authorization test that must be satisfied.

The keyword “SC:” is used to specify a minimum security level for the repositories involved in the access. If “SC:” is not specified, the lowest security level is acceptable.

The optional “Authorization:” keyword is used to specify required authorizations on the same repository as the work. The optional “Other-Authorization:” keyword is used to specify required authorizations on the other repository in the transaction.

The optional “Ticket:” keyword specifies the identity of a ticket required for the transaction. A transaction involving digital tickets must locate an appropriate digital ticket agent who can “punch” or otherwise validate the ticket before the transaction can proceed. Tickets are described in greater detail below.

In a transaction involving a repository and a document server, some usage rights may require that the repository have a particular authorization, that the server have some authorization, or that both repositories have (possibly different) authorizations. Authorizations themselves are digital works (hereinafter referred to as an authorization object) that can be moved between repositories in the same manner as other digital works. Their copying and transferring is subject to the

22

same rights and fees as other digital works. A repository is said to have an authorization if that authorization object is contained within the repository.

In some cases, an authorization may be required from a source other than the document server and repository. An authorization object referenced by an Authorization-ID can contain digital address information to be used to set up a communications link between a repository and the authorization source. These are analogous to phone numbers. For such access tests, the communication would need to be established and authorization obtained before the right could be exercised.

For one-time usage rights, a variant on this scheme is to have a digital ticket. A ticket is presented to a digital ticket agent, whose type is specified on the ticket. In the simplest case, a certified generic ticket agent, available on all repositories, is available to “punch” the ticket. In other cases, the ticket may contain addressing information for locating a “special” ticket agent. Once a ticket has been punched, it cannot be used again for the same kind of transaction (unless it is unpunched or refreshed in the manner described below.) Punching includes marking the ticket with a timestamp of the date and time it was used. Tickets are digital works and can be copied or transferred between repositories according to their usage rights.

In the currently preferred embodiment, a “punched” ticket becomes “unpunched” or “refreshed” when it is copied or extracted. The Copy and Extract operations save the date and time as a property of the digital ticket. When a ticket agent is given a ticket, it can simply check whether the digital copy was made after the last time that it was punched. Of course, the digital ticket must have the copy or extract usage rights attached thereto.

The capability to unpunch a ticket is important in the following cases:

A digital work is circulated at low cost with a limitation that it can be used only once.

A digital work is circulated with a ticket that can be used once to give discounts on purchases of other works.

A digital work is circulated with a ticket (included in the purchase price and possibly embedded in the work) that can be used for a future upgrade.

In each of these cases, if a paid copy is made of the digital work (including the ticket) the new owner would expect to get a fresh (unpunched) ticket, whether the copy seller has used the work or not. In contrast, loaning a work or simply transferring it to another repository should not revitalize the ticket.

Usage Fees and Incentives Specification

The billing for use of a digital work is fundamental to a commercial distribution system. Grammar Element 1517 “Fee-Spec:={Scheduled-Discount} Regular-Fee-Spec|Scheduled-Fee-Spec|Markup-Spec” provides a range of options for billing for the use of digital works.

A key feature of this approach is the development of low-overhead billing for transactions in potentially small amounts. Thus, it becomes feasible to collect fees of only a few cents each for thousands of transactions.

The grammar differentiates between uses where the charge is per use from those where it is metered by the time unit. Transactions can support fees that the user pays for using a digital work as well as incentives paid by the right grantor to users to induce them to use or distribute the digital work.

The optional scheduled discount refers to the rest of the fee specification—discounting it by a percentage over time. If it is not specified, then there is no scheduled discount. Regular fee specifications are constant over time. Scheduled fee specifications give a schedule of dates over which the fee speci-

US 8,370,956 B2

23

cations change. Markup specifications are used in d-blocks for adding a percentage to the fees already being charged.

Grammar Element **1518** "Scheduled-Discount:=(Scheduled-Discount: (Time-Spec Percentage)*)" A Scheduled-Discount is essentially a scheduled modifier of any other fee specification for this version of the right of the digital work. (It does not refer to children or parent digital works or to other versions of rights.). It is a list of pairs of times and percentages. The most recent time in the list that has not yet passed at the time of the transaction is the one in effect. The percentage gives the discount percentage. For example, the number 10 refers to a 10% discount.

Grammar Element **1519** "Regular-Fee-Spec:={Fee:|Incentive:} [Per-Use-Spec|Metered-Rate-Spec|Best-Price-Spec|Call-For-Price-Spec] {Min: Money-Unit Per: Time-Spec} {Max: Money-Unit Per: Time-Spec} To: Account-ID)" provides for several kinds of fee specifications.

Fees are paid by the copy-owner/user to the revenue-owner if Fee: is specified. Incentives are paid by the revenue-owner to the user if Incentive: is specified. If the Min: specification is given, then there is a minimum fee to be charged per time-spec unit for its use. If the Max: specification is given, then there is a maximum fee to be charged per time-spec for its use. When Fee: is specified, Account-ID identifies the account to which the fee is to be paid. When Incentive: is specified, Account-ID identifies the account from which the fee is to be paid.

Grammar element **1520** "Per-Use-Spec:=Per-Use: Money-unit" defines a simple fee to be paid every time the right is exercised, regardless of how much time the transaction takes.

Grammar element **1521** "Metered-Rate-Spec:=Metered: Money-Unit Per: Time-Spec" defines a metered-rate fee paid according to how long the right is exercised. Thus, the time it takes to complete the transaction determines the fee.

Grammar, element **1522** "Best-Price-Spec:=Best-Price: Money-unit Max: Money-unit" is used to specify a best-price that is determined when the account is settled. This specification is to accommodate special deals, rebates, and pricing that depends on information that is not available to the repository. All fee specifications can be combined with tickets or authorizations that could indicate that the consumer is a wholesaler or that he is a preferred customer, or that the seller be authorized in some way. The amount of money in the Max: field is the maximum amount that the use will cost. This is the amount that is tentatively debited from the credit server. However, when the transaction is ultimately reconciled, any excess amount will be returned to the consumer in a separate transaction.

Grammar element **1523** "Call-For-Price-Spec:=Call-For-Price" is similar to a "Best-Price-Spec" in that it is intended to accommodate cases where prices are dynamic. A Call-For-Price Spec requires a communication with a dealer to determine the price. This option cannot be exercised if the repository cannot communicate with a dealer at the time that the right is exercised. It is based on a secure transaction whereby the dealer names a price to exercise the right and passes along a deal certificate which is referenced or included in the billing process.

Grammar element **1524** "Scheduled-Fee-Spec:=(Schedule: (Time-Spec Regular-Fee-Spec)*)" is used to provide a schedule of dates over which the fee specifications change. The fee specification with the most recent date not in the future is the one that is in effect. This is similar to but more general than the scheduled discount. It is more general, because it provides a means to vary the fee agreement for each time period.

24

Grammar element **1525** "Markup-Spec:=Markup: percentage To: Account-ID" is provided for adding a percentage to the fees already being charged. For example, a 5% markup means that a fee of 5% of cumulative fee so far will be allocated to the distributor. A markup specification can be applied to all of the other kinds of fee specifications. It is typically used in a shell provided by a distributor. It refers to fees associated with d-blocks that are parts of the current d-block. This might be a convenient specification for use in taxes, or in distributor overhead.

Examples of Sets of Usage Rights

((Play) (Transfer (SC: 3)) (Delete)

This work can be played without requirements for fee or authorization on any rendering system. It can be transferred to any other repository of security level 3 or greater. It can be deleted.

((Play) (Transfer (SC: 3)) (Delete) (Backup) (Restore (Fee: Per-Use: \$5 To: Account-ID-678)))

Same as the previous example plus rights for backup and restore. The work can be backed up without fee. It can be restored for a \$5 fee payable to the account described by Account-ID-678.

((Play) (Transfer (SC: 3))

(Copy (SC:3)(Fee: Per-Use: \$5 To: Account-ID-678))

(Delete (Incentive: Per-Use: \$2.50 To: Account-ID-678)))

This work can be played, transferred, copied, or deleted. Copy or transfer operations can take place only with repositories of security level three or greater. The fee to make a copy is \$5 payable to Account-ID-678. If a copy is deleted, then an incentive of \$2.50 is paid to the former copy owner.

((Play) (Transfer (SC: 3))

Copy (SC: 3) (Fee: Per-Use: \$10 To: Account-ID-678))

Delete) (Backup) (Restore (SC: 3) (Fee: Per-Use: \$5 To: Account-ID-678)))

Same as the previous example plus fees for copying. The work can be copied digitally for a fee of \$10 payable to Account-ID-678. The repository on which the work is copied or restored must be at security level 3 or greater.

((Play) (Transfer (SC: 3))

(Copy Authorization: License-123-ID (SC: 3)))

The digital work can be played, transferred, or copied. Copies or transfers must be on repositories of security level 3 or greater. Copying requires the license License-123-ID issued to the copying repository. None of the rights require fees.

((Play) (Print Printer: Printer-567-ID (Fee: Per-Use: \$1 To: Account-ID-678)))

This work can be played for free. It can be printed on any printer with the identifier Printer-567-ID for a fee of \$1 payable to the account described by Account-ID-678.

((Play Player: Player-876-ID) (From: Feb. 2, 1994 Until: Feb. 15, 1995) (Fee: Metered: \$0.01 Per: 0:1:0 Min: \$0.25 Per: 0/1/0 To: Account-ID-567))

This work can be played on any player holding the ID Player-876-ID. The time of this right is from Feb. 14, 1994 until Feb. 15, 1995. The fee for use is one cent per minute with a minimum of 25 cents in any day that it is used, payable to the account described by Account-ID-567.

((Play) (Transfer) (Delete)(Loan 2 (Delete: Transfer Loan)))

This work can be played, transferred, deleted, or loaned. Up to two copies can be loaned out at a time. The loaned copy has the same rights except that it cannot be transferred. When both copies are loaned out, no rights can be exercised on the original on the repository.

US 8,370,956 B2

25

((Play) (Transfer) (Delete) (Backup) (Restore (SC:3))
 (Loan 2 Remaining-Copy-Rights: (Delete: Play Transfer)
 Next-Set-of-Rights: (Delete: Transfer Loan)))

Similar to previous example. Rights to Backup and Restore
 the work are added, where restoration requires a repository of
 at least security level three. When all copies of the work are
 loaned out, the remaining copy cannot be played or trans-
 ferred.

((Play) (Transfer) (Copy) (Print) (Backup) (Restore (SC:
 3)))

(Loan 1 Remaining-Copy-Rights: (Add: Play Print
 Backup)

Next-Set-of-Rights: (Delete: Transfer Loan)
 (Fee: Metered: \$10 Per: 1:0:0 To: Account-ID-567))

(Loan 1 Remaining-Copy-Rights:

Add: ((Play Player: Player-876-ID) 2 (From: Feb. 14, 1994
 Until: Feb. 15, 1995)

(Fee: Metered: \$0.01 Per: 0:1:0 Min: \$0.25 Per: 0/1/0
 To: Account-ID-567)))

The original work has rights to Play, Transfer, Copy, Print,
 Backup, Restore, and Loan. There are two versions of the
 Loan right. The first version of the loan right costs \$10 per day
 but allows the original copy owner to exercise free use of the
 Play, Print and Backup rights. The second version of the Loan
 right is free. None of the original rights are applicable. How-
 ever a right to Play the work at the specified metered rate is
 added.

((Play Player: Player-Small-Screen-123-ID)
 (Embed (Fee: Per-Use \$0.01 To: Account-678-ID))
 (Copy (Fee: Per-Use \$1.00 To: Account-678-ID)))

The digital work can be played on any player with the
 identifier Player-Small-Screen-123-ID. It can be embedded
 in a larger work. The embedding requires a modest one cent
 registration fee to Account-678-ID. Digital copies can be
 made for \$1.00.

Repository Transactions

When a user requests access to a digital work, the reposi-
 tory will initiate various transactions. The combination of
 transactions invoked will depend on the specifications
 assigned for a usage right. There are three basic types of
 transactions, Session Initiation Transactions, Financial
 Transactions and Usage Transactions. Generally, session ini-
 tiation transactions are initiated first to establish a valid ses-
 sion. When a valid session is established, transactions corre-
 sponding to the various usage rights are invoked. Finally,
 request specific transactions are performed.

Transactions occur between two repositories (one acting as
 a server), between a repository and a document playback
 platform (e.g. for executing or viewing), between a repository
 and a credit server or between a repository and an authoriza-
 tion server. When transactions occur between more than one
 repository, it is assumed that there is a reliable communica-
 tion channel between the repositories. For example, this could
 be a TCP/IP channel or any other commercially available
 channel that has built-in capabilities for detecting and cor-
 recting transmission errors. However, it is not assumed that
 the communication channel is secure. Provisions for security
 and privacy are part of the requirements for specifying and
 implementing repositories and thus form the need for various
 transactions.

Message Transmission

Transactions require that there be some communication
 between repositories. Communication between repositories
 occurs in units termed as messages. Because the communi-
 cation line is assumed to be unsecure, all communications

26

with repositories that are above the lowest security class are
 encrypted utilizing a public key encryption technique. Public
 key encryption is a well known technique in the encryption
 arts. The term key refers to a numeric code that is used with
 encryption and decryption algorithms. Keys come in pairs,
 where "writing keys" are used to encrypt data and "checking
 keys" are used to decrypt data. Both writing and checking
 keys may be public or private. Public keys are those that are
 distributed to others. Private keys are maintained in confi-
 dence.

Key management and security is instrumental in the suc-
 cess of a public key encryption system. In the currently pre-
 ferred embodiment, one or more master repositories maintain
 the keys and create the identification certificates used by the
 repositories.

When a sending repository transmits a message to a receiv-
 ing repository, the sending repository encrypts all of its data
 using the public writing key of the receiving repository. The
 sending repository includes its name, the name of the receiv-
 ing repository, a session identifier such as a nonce (described
 below), and a message counter in each message.

In this way, the communication can only be read (to a high
 probability) by the receiving repository, which holds the pri-
 vate checking key for decryption. The auxiliary data is used to
 guard against various replay attacks to security. If messages
 ever arrive with the wrong counter or an old nonce, the reposi-
 tories can assume that someone is interfering with commu-
 nication and the transaction terminated.

The respective public keys for the repositories to be used
 for encryption are obtained in the registration transaction
 described below.

Session Initiation Transactions

A usage transaction is carried out in a session between
 repositories. For usage transactions involving more than one
 repository, or for financial transactions between a repository
 and a credit server, a registration transaction is performed. A
 second transaction termed a login transaction, may also be
 needed to initiate the session. The goal of the registration
 transaction is to establish a secure channel between two
 repositories who know each others identities. As it is assumed
 that the communication channel between the repositories is
 reliable but not secure, there is a risk that a non-repository
 may mimic the protocol in order to gain illegitimate access to
 a repository.

The registration transaction between two repositories is
 described with respect to FIGS. 16 and 17. The steps
 described are from the perspective of a "repository-1" regis-
 tering its identity with a "repository-2". The registration must
 be symmetrical so the same set of steps will be repeated for
 repository-2 registering its identity with repository-1. Refer-
 ring to FIG. 16, repository-1 first generates an encrypted
 registration identifier, step 1601 and then generates a regis-
 tration message, step 1602. A registration message is com-
 prised of an identifier of a master repository, the identification
 certificate for the repository-1 and an encrypted random regis-
 tration identifier. The identification certificate is encrypted
 by the master repository in its private key and attests to the
 fact that the repository (here repository-1) is a bona fide
 repository. The identification certificate also contains a public
 key for the repository, the repository security level and a
 timestamp (indicating a time after which the certificate is no
 longer valid.) The registration identifier is a number gener-
 ated by the repository for this registration. The registration
 identifier is unique to the session and is encrypted in reposi-
 tory-1's private key. The registration identifier is used to
 improve security of authentication by detecting certain kinds

US 8,370,956 B2

27

of communications based attacks. Repository-1 then transmits the registration message to repository-2, step 1603.

Upon receiving the registration message, repository-2 determines if it has the needed public key for the master repository, step 1604. If repository-2 does not have the needed public key to decrypt the identification certificate, the registration transaction terminates in an error, step 1618.

Assuming that repository-2 has the proper public key the identification certificate is decrypted, step 1605. Repository-2 saves the encrypted registration identifier, step 1606, and extracts the repository identifier, step 1607. The extracted repository identifier is checked against a "hotlist" of compromised document repositories, step 1608. In the currently preferred embodiment, each repository will contain "hotlists" of compromised repositories. If the repository is on the "hotlist", the registration transaction terminates in an error per step 1618. Repositories can be removed from the hotlist when their certificates expire, so that the list does not need to grow without bound. Also, by keeping a short list of hotlist certificates that it has previously received, a repository can avoid the work of actually going through the list. These lists would be encrypted by a master repository. A minor variation on the approach to improve efficiency would have the repositories first exchange lists of names of hotlist certificates, ultimately exchanging only those lists that they had not previously received. The "hotlists" are maintained and distributed by Master repositories.

Note that rather than terminating in error, the transaction could request that another registration message be sent based on an identification certificate created by another master repository. This may be repeated until a satisfactory identification certificate is found, or it is determined that trust cannot be established.

Assuming that the repository is not on the hotlist, the repository identification needs to be verified. In other words, repository-2 needs to validate that the repository on the other end is really repository-1. This is termed performance testing and is performed in order to avoid invalid access to the repository via a counterfeit repository replaying a recording of a prior session initiation between repository-1 and repository-2. Performance testing is initiated by repository-2 generating a performance message, step 1609. The performance message consists of a nonce, the names of the respective repositories, the time and the registration identifier received from repository-1. A nonce is a generated message based on some random and variable information (e.g. the time or the temperature.) The nonce is used to check whether repository-1 can actually exhibit correct encrypting of a message using the private keys it claims to have, on a message that it has never seen before. The performance message is encrypted using the public key specified in the registration message of repository-1. The performance message is transmitted to repository-1, step 1610, where it is decrypted by repository-1 using its private key, step 1611. Repository-1 then checks to make sure that the names of the two repositories are correct, step 1612, that the time is accurate, step 1613 and that the registration identifier corresponds to the one it sent, step 1614. If any of these tests fails, the transaction is terminated per step 1616. Assuming that the tests are passed, repository-1 transmits the nonce to repository-2 in the clear, step 1615. Repository-2 then compares the received nonce to the original nonce, step 1617. If they are not identical, the registration transaction terminates in an error per step 1618. If they are the same, the registration transaction has successfully completed.

At this point, assuming that the transaction has not terminated, the repositories exchange messages containing session keys to be used in all communications during the session and

28

synchronize their clocks. FIG. 17 illustrates the session information exchange and clock synchronization steps (again from the perspective of repository-1.) Referring to FIG. 17, repository-1 creates a session key pair, step 1701. A first key is kept private and is used by repository-1 to encrypt messages. The second key is a public key used by repository-2 to decrypt messages. The second key is encrypted using the public key of repository-2, step 1702 and is sent to repository-2, step 1703. Upon receipt, repository-2 decrypts the second key, step 1704. The second key is used to decrypt messages in subsequent communications. When each repository has completed this step, they are both convinced that the other repository is bona fide and that they are communicating with the original. Each repository has given the other a key to be used in decrypting further communications during the session. Since that key is itself transmitted in the public key of the receiving repository only it will be able to decrypt the key which is used to decrypt subsequent messages.

After the session information is exchanged, the repositories must synchronize their clocks. Clock synchronization is used by the repositories to establish an agreed upon time base for the financial records of their mutual transactions. Referring back to FIG. 17, repository-2 initiates clock synchronization by generating a time stamp exchange message, step 1705, and transmits it to repository-1, step 1706. Upon receipt, repository-1 generates its own time stamp message, step 1707 and transmits it back to repository-2, step 1708. Repository-2 notes the current time, step 1709 and stores the time received from repository-1, step 1710. The current time is compared to the time received from repository-1, step 1711. The difference is then checked to see if it exceeds a predetermined tolerance (e.g. one minute), step 1712. If it does, repository-2 terminates the transaction as this may indicate tampering with the repository, step 1713. If not repository-2 computes an adjusted time delta, step 1714. The adjusted time delta is the difference between the clock time of repository-2 and the average of the times from repository-1 and repository-2.

To achieve greater accuracy, repository-2 can request the time again up to a fixed number of times (e.g. five times), repeat the clock synchronization steps, and average the results.

A second session initiation transaction is a Login transaction. The Login transaction is used to check the authenticity of a user requesting a transaction. A Login transaction is particularly prudent for the authorization of financial transactions that will be charged to a credit server. The Login transaction involves an interaction between the user at a user interface and the credit server associated with a repository. The information exchanged here is a login string supplied by the repository/credit server to identify itself to the user, and a Personal Identification Number (PIN) provided by the user to identify himself to the credit server. In the event that the user is accessing a credit server on a repository different from the one on which the user interface resides, exchange of the information would be encrypted using the public and private keys of the respective repositories.

Billing Transactions

Billing Transactions are concerned with monetary transaction with a credit server. Billing Transactions are carried out when all other conditions are satisfied and a usage fee is required for granting the request. For the most part, billing transactions are well understood in the state of the art. These transactions are between a repository and a credit server, or between a credit server and a billing clearinghouse. Briefly, the required transactions include the following:

US 8,370,956 B2

29

Registration and LOGIN transactions, by which the repository and user establish their bona fides to a credit server. These transactions would be entirely internal in cases where the repository and credit server are implemented as a single system.

Registration and LOGIN transactions, by which a credit server establishes its bona fides to a billing clearinghouse.

An Assign-fee transaction to assign a charge. The information in this transaction would include a transaction identifier, the identities of the repositories in the transaction, and a list of charges from the parts of the digital work. If there has been any unusual event in the transaction such as an interruption of communications, that information is included as well.

A Begin-charges transaction to assign a charge. This transaction is much the same as an assign-fee transaction except that it is used for metered use. It includes the same information as the assign-fee 4, ii transaction as well as the usage fee information. The credit-server is then responsible for running a clock.

An End-charges transaction to end a charge for metered use. (In a variation on this approach, the repositories would exchange periodic charge information for each block of time.)

A report-charges transaction between a personal credit server and a billing clearinghouse. This transaction is invoked at least once per billing period. It is used to pass along information about charges. On debit and credit cards, this transaction would also be used to update balance information and credit limits as needed.

All billing transactions are given a transaction ID and are reported to the credit servers by both the server and the client. This reduces possible loss of billing information if one of the parties to a transaction loses a banking card and provides a check against tampering with the system.

Usage Transactions

After the session initiation transactions have been completed, the usage request may then be processed. To simplify the description of the steps carried out in processing a usage request, the term requester is used to refer to a repository in the requester mode which is initiating a request, and the term server is used to refer to a repository in the server mode and which contains the desired digital work. In many cases such as requests to print or view a work, the requester and server may be the same device and the transactions described in the following would be entirely internal. In such instances, certain transaction steps, such as the registration transaction, need not be performed.

There are some common steps that are part of the semantics of all of the usage rights transactions. These steps are referred to as the common transaction steps. There are two sets—the “opening” steps and the “closing” steps. For simplicity, these are listed here rather than repeating them in the descriptions of all of the usage rights transactions.

Transactions can refer to a part of a digital work, a complete digital work, or a Digital work containing other digital works. Although not described in detail herein, a transaction may even refer to a folder comprised of a plurality of digital works. The term “work” is used to refer to what ever portion or set of digital works is being accessed.

Many of the steps here involve determining if certain conditions are satisfied. Recall that each usage right may have one or more conditions which must be satisfied before the right can be exercised. Digital works have parts and parts have parts. Different parts can have different rights and fees. Thus, it is necessary to verify that the requirements are met for ALL

30

of the parts that are involved in a transaction For brevity, when reference is made to checking whether the rights exist and conditions for exercising are satisfied, it is meant that all such checking takes place for each of the relevant parts of the work.

5 FIG. 18 illustrates the initial common opening and closing steps for a transaction. At this point it is assumed that registration has occurred and that a “trusted” session is in place. General tests are tests on usage rights associated with the folder containing the work or some containing folder higher in the file system hierarchy. These tests correspond to requirements imposed on the work as a consequence of its being on the particular repository, as opposed to being attached to the work itself. Referring to FIG. 18, prior to initiating a usage transaction, the requester performs any general tests that are required before the right associated with the transaction can be exercised, step, 1801. For example, install, uninstall and delete rights may be implemented to require that a requester have an authorization certificate before the right can be exercised. Another example is the requirement that a digital ticket be present and punched before a digital work may be copied to a requester. If any of the general tests fail, the transaction is not initiated, step, 1802. Assuming that such required tests are passed, upon receiving the usage request, the server generates a transaction identifier that is used in records or reports of the transaction, step 1803. The server then checks whether the digital work has been granted the right corresponding to the requested transaction, step 1804. If the digital work has not been granted the right corresponding to the request, the transaction terminates, step 1805. If the digital work has been granted the requested right, the server then determines if the various conditions for exercising the right are satisfied. Time based conditions are examined, step 1806. These conditions are checked by examining the time specification for the version of the right. If any of the conditions are not satisfied, the transaction terminates per step 1805.

Assuming that the time based conditions are satisfied, the server checks security and access conditions, step 1807. Such security and access conditions are satisfied if: 1) the requester is at the specified security class, or a higher security class, 2) the server satisfies any specified authorization test and 3) the requester satisfies any specified authorization tests and has any required digital tickets. If any of the conditions are not satisfied, the transaction terminates per step 1805.

Assuming that the security and access conditions are all satisfied, the server checks the copy count condition, step 1808. If the copy count equals zero, then the transaction cannot be completed and the transaction terminates per step 1805.

Assuming that the copy count does not equal zero, the server checks if the copies in use for the requested right is greater than or equal to any copy count for the requested right (or relevant parts), step 1809. If the copies in use are greater than or equal to the copy count, this indicates that usage rights for the version of the transaction have been exhausted. Accordingly, the server terminates the transaction, step 1805. If the copy count is less than the copies in use for the transaction the transaction can continue, and the copies in use would be incremented by the number of digital works requested in the transaction, step 1810.

60 The server then checks if the digital work has a “Loan” access right, step 1811. The “Loan” access right is a special case since remaining rights may be present even though all copies are loaned out. If the digital work has the “Loan” access right, a check is made to see if all copies have been loaned out, step 1812. The number of copies that could be loaned is the sum of the Copy-Counts for all of the versions of the loan right of the digital work. For a composite work, the

US 8,370,956 B2

31

relevant figure is the minimal such sum of each of the components of the composite work. If all copies have been loaned out, the remaining rights are determined, step **1813**. The remaining-rights are determined from the remaining rights specifications from the versions of the Loan right. If there is only one version of the Loan right, then the determination is simple. The remaining rights are the ones specified in that version of the Loan right, or none if Remaining-Rights: is not specified. If there are multiple versions of the Loan right and all copies of all of the versions are loaned out, then the remaining rights is taken as the minimum set (intersection) of remaining rights across all of the versions of the loan right. The server then determines if the requested right is in the set of remaining rights, step **1814**. If the requested right is not in the set of remaining rights, the server terminates the transaction, step **1805**.

If Loan is not a usage right for the digital work or if all copies have not been loaned out or the requested right is in the set of remaining rights, fee conditions for the right are then checked, step **1815**. This will initiate various financial transactions between the repository and associated credit server. Further, any metering of usage of a digital work will commence. If any financial transaction fails, the transaction terminates per step **1805**.

It should be noted that the order in which the conditions are checked need not follow the order of steps **1806-1815**.

At this point, right specific steps are now performed and are represented here as step **1816**. The right specific steps are described in greater detail below.

The common closing transaction steps are now performed. Each of the closing transaction steps are performed by the server after a successful completion of a transaction. Referring back to FIG. **18**, the copies in use value for the requested right is decremented by the number of copies involved in the transaction, step **1817**. Next, if the right had a metered usage fee specification, the server subtracts the elapsed time from the Remaining-Use-Time associated with the right for every part involved in the transaction, step **1818**. Finally, if there are fee specifications associated with the right, the server initiates End-Charge financial transaction to confirm billing, step **1819**.

Transmission Protocol

An important area to consider is the transmission of the digital work from the server to the requester. The transmission protocol described herein refers to events occurring after a valid session has been created. The transmission protocol must handle the case of disruption in the communications between the repositories. It is assumed that interference such as injecting noise on the communication channel can be detected by the integrity checks (e.g., parity, checksum, etc.) that are built into the transport protocol and are not discussed in detail herein.

The underlying goal in the transmission protocol is to preclude certain failure modes, such as malicious or accidental interference on the communications channel. Suppose, for example, that a user pulls a card with the credit server at a specific time near the end of a transaction. There should not be a vulnerable time at which “pulling the card” causes the repositories to fail to correctly account for the number of copies of the work that have been created. Restated, there should be no time at which a party can break a connection as a means to avoid payment after using a digital work.

If a transaction is interrupted (and fails), both repositories restore the digital works and accounts to their state prior to the failure, modulo records of the failure itself.

FIG. **19** is a state diagram showing steps in the process of transmitting information during a transaction. Each box rep-

32

resents a state of a repository in either the server mode (above the central dotted line **1901**) or in the requester mode (below the dotted line **1901**). Solid arrows stand for transitions between states. Dashed arrows stand for message communications between the repositories. A dashed message arrow pointing to a solid transition arrow is interpreted as meaning that the transition takes place when the message is received. Unlabeled transition arrows take place unconditionally. Other labels on state transition arrows describe conditions that trigger the transition.

Referring now to FIG. **19**, the server is initially in a state **1902** where a new transaction is initiated via start message **1903**. This message includes transaction information including a transaction identifier and a count of the blocks of data to be transferred. The requester, initially in a wait state **1904** then enters a data wait state **1905**.

The server enters a data transmit state **1906** and transmits a block of data **1907** and then enters a wait for acknowledgement state **1908**. As the data is received, the requesters enters a data receive state **1909** and when the data blocks is completely received it enters an acknowledgement state **1910** and transmits an Acknowledgement message **1911** to the server.

If there are more blocks to send, the server waits until receiving an Acknowledgement message from the requester. When an Acknowledgement message is received it sends the next block to the requester and again waits for acknowledgement. The requester also repeats the same cycle of states.

If the server detects a communications failure before sending the last block, it enters a cancellation state **1912** wherein the transaction is cancelled. Similarly, if the requester detects a communications failure before receiving the last block it enters a cancellation state **1913**.

If there are no more blocks to send, the server commits to the transaction and waits for the final Acknowledgement in state **1914**. If there is a communications failure before the server receives the final Acknowledgement message, it still commits to the transaction but includes a report about the event to its credit server in state **1915**. This report serves two purposes. It will help legitimize any claims by a user of having been billed for receiving digital works that were not completely received. Also it helps to identify repositories and communications lines that have suspicious patterns of use and interruption. The server then enters its completion state

On the requester side, when there are no more blocks to receive, the requester commits to the transaction in state **1917**. If the requester detects a communications failure at this state, it reports the failure to its credit server in state **1918**, but still commits to the transaction. When it has committed, it sends an acknowledgement message to the server. The server then enters its completion state **1919**.

The key property is that both the server and the requester cancel a transaction if it is interrupted before all of the data blocks are delivered, and commits to it if all of the data blocks have been delivered.

There is a possibility that the server will have sent all of the data blocks (and committed) but the requester will not have received all of them and will cancel the transaction. In this case, both repositories will presumably detect a communications failure and report it to their credit server. This case will probably be rare since it depends on very precise timing of the communications failure. The only consequence will be that the user at the requester repository may want to request a refund from the credit services—and the case for that refund will be documented by reports by both repositories.

To prevent loss of data, the server should not delete any transferred digital work until receiving the final acknowl-

US 8,370,956 B2

33

edgement from the requester. But it also should not use the file. A well known way to deal with this situation is called “two-phase commit” or 2PC.

Two-phase commit works as follows. The first phase works the same as the method described above. The server sends all of the data to the requester. Both repositories mark the transaction (and appropriate files) as uncommitted. The server sends a ready-to-commit message to the requester. The requester sends back an acknowledgement. The server then commits and sends the requester a commit message. When the requester receives the commit message, it commits the file.

If there is a communication failure or other crash, the requester must check back with the server to determine the status of the transaction. The server has the last word on this. The requester may have received all of the data, but if it did not get the final message, it has not committed. The server can go ahead and delete files (except for transaction records) once it commits, since the files are known to have been fully transmitted before starting the 2PC cycle.

There are variations known in the art which can be used to achieve the same effect. For example, the server could use an additional level of encryption when transmitting a work to a client. Only after the client sends a message acknowledging receipt does it send the key. The client then agrees to pay for the digital work. The point of this variation is that it provides a clear audit trail that the client received the work. For trusted systems, however, this variation adds a level of encryption for no real gain in accountability.

The transactions for specific usage rights are now discussed.

The Copy Transaction

A Copy transaction is a request to make one or more independent copies of the work with the same or lesser usage rights. Copy differs from the extraction right discussed later in that it refers to entire digital works or entire folders containing digital works. A copy operation cannot be used to remove a portion of a digital work.

The requester sends the server a message to initiate the Copy Transaction. This message indicates the work to be copied, the version of the copy right to be used for the transaction, the destination address information (location in a folder) for placing the work, the file data for the work (including its size), and the number of copies requested.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the client according to the transmission protocol. If a Next-Set-Of-Rights has been provided in the version of the right, those rights are transmitted as the rights for the work. Otherwise, the rights of the original are transmitted. In any event, the Copy-Count field for the copy of the digital work being sent right is set to the number-of-copies requested.

The requester records the work contents, data, and usage rights and stores the work. It records the date and time that the copy was made in the properties of the digital work.

The repositories perform the common closing transaction steps.

The Transfer Transaction

A Transfer transaction is a request to move copies of the work with the same or lesser usage rights to another repository. In contrast with a copy transaction, this results in removing the work copies from the server.

34

The requester sends the server a message to initiate the Transfer Transaction. This message indicates the work to be transferred, the version of the transfer right to be used in the transaction, the destination address information for placing the work, the file data for the work, and the number of copies involved.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester according to the transmission protocol. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the work. Otherwise, the rights of the original are transmitted. In either case, the Copy-Count field for the transmitted rights is set to the number-of-copies requested.

The requester records the work contents, data, and usage rights and stores the work.

The server decrements its copy count by the number of copies involved in the transaction.

The repositories perform the common closing transaction steps.

If the number of copies remaining in the server is now zero, it erases the digital work from its memory.

The Loan Transaction

A loan transaction is a mechanism for loaning copies of a digital work. The maximum duration of the loan is determined by an internal parameter of the digital work. Works are automatically returned after a predetermined time period.

The requester sends the server a message to initiate the Transfer Transaction. This message indicates the work to be loaned, the version of the loan right to be used in the transaction, the destination address information for placing the work, the number of copies involved, the file data for the work, and the period of the loan.

The server checks the validity of the requested loan period, and ends with an error if the period is not valid. Loans for a loaned copy cannot extend beyond the period of the original loan to the server.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester.

If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the work. Otherwise, the rights of the original are transmitted, as modified to reflect the loan period.

The requester records the digital work contents, data, usage rights, and loan period and stores the work.

The server updates the usage rights information in the digital work to reflect the number of copies loaned out.

The repositories perform the common closing transaction steps.

The server updates the usage rights data for the digital work. This may preclude use of the work until it is returned from the loan. The user on the requester platform can now use the transferred copies of the digital work. A user accessing the original repository cannot use the digital work, unless there are copies remaining. What happens next depends on the order of events in time.

Case 1. If the time of the loan period is not yet exhausted and the requester sends the repository a Return message.

The return message includes the requester identification, and the transaction ID.

The server decrements the copies-in-use field by the number of copies that were returned. (If the number of digital works returned is greater than the number actually bor-

US 8,370,956 B2

35

rowed, this is treated as an error.) This step may now make the work available at the server for other users.

The requester deactivates its copies and removes the contents from its memory.

Case 2. If the time of the loan period is exhausted and the requester has not yet sent a Return message.

The server decrements the copies-in-use field by the number digital works that were borrowed.

The requester automatically deactivates its copies of the digital work. It terminates all current uses and erases the digital work copies from memory. One question is why a requester would ever return a work earlier than the period of the loan, since it would be returned automatically anyway. One reason for early return is that there may be a metered fee which determines the cost of the loan. Returning early may reduce that fee.

The Play Transaction

A play transaction is a request to use the contents of a work. Typically, to “play” a work is to send the digital work through some kind of transducer, such as a speaker or a display device. The request implies the intention that the contents will not be communicated digitally to any other system. For example, they will not be sent to a printer, recorded on any digital medium, retained after the transaction or sent to another repository.

This term “play” is natural for examples like playing music, playing a movie, or playing a video game. The general form of play means that a “player” is used to use the digital work. However, the term play covers all media and kinds of recordings. Thus one would “play” a digital work, meaning, to render it for reading, or play a computer program, meaning to execute it. For a digital ticket the player would be a digital ticket agent.

The requester sends the server a message to initiate the play transaction. This message indicates the work to be played, the version of the play right to be used in the transaction, the identity of the player being used, and the file data for the work.

The server checks the validity of the player identification and the compatibility of the player identification with the player specification in the right. It ends with an error if these are not satisfactory.

The repositories perform the common opening transaction steps.

The server and requester read and write the blocks of data as requested by the player according to the transmission protocol. The requester plays the work contents, using the player.

When the player is finished, the player and the requester remove the contents from their memory.

The repositories perform the common closing transaction steps.

The Print Transaction

A Print transaction is a request to obtain the contents of a work for the purpose of rendering them on a “printer.” We use the term “printer” to include the common case of writing with ink on paper. However, the key aspect of “printing” in our use of the term is that it makes a copy of the digital work in a place outside of the protection of usage rights. As with all rights, this may require particular authorization certificates.

Once a digital work is printed, the publisher and user are bound by whatever copyright laws are in effect. However, printing moves the contents outside the control of repositories. For example, absent any other enforcement mechanisms, once a digital work is printed on paper, it can be copied on ordinary photocopying machines without intervention by a repository to collect usage fees. If the printer to a digital disk

36

is permitted, then that digital copy is outside of the control of usage rights. Both the creator and the user know this, although the creator does not necessarily give tacit consent to such copying, which may violate copyright laws.

The requester sends the server a message to initiate a Print transaction. This message indicates the work to be played, the identity of the printer being used, the file data for the work, and the number of copies in the request.

The server checks the validity of the printer identification and the compatibility of the printer identification with the printer specification in the right. It ends with an error if these are not satisfactory.

The repositories perform the common opening transaction steps.

The server transmits blocks of data according to the transmission protocol.

The requester prints the work contents, using the printer. When the printer is finished, the printer and the requester remove the contents from their memory.

The repositories perform the common closing transaction steps.

The Backup Transaction

A Backup transaction is a request to make a backup copy of a digital work, as a protection against media failure. In the context of repositories, secure backup copies differ from other copies in three ways: (1) they are made under the control of a Backup transaction rather than a Copy transaction, (2) they do not count as regular copies, and (3) they are not usable as regular copies. Generally, backup copies are encrypted.

Although backup copies may be transferred or copied, depending on their assigned rights, the only way to make them useful for playing, printing or embedding is to restore them.

The output of a Backup operation is both an encrypted data file that contains the contents and description of a work, and a restoration file with an encryption key for restoring the encrypted contents. In many cases, the encrypted data file would have rights for “printing” it to a disk outside of the protection system, relying just on its encryption for security. Such files could be stored anywhere that was physically safe and convenient. The restoration file would be held in the repository. This file is necessary for the restoration of a backup copy. It may have rights for transfer between repositories.

The requester sends the server a message to initiate a backup transaction. This message indicates the work to be backed up, the version of the backup right to be used in the transaction, the destination address information for placing the backup copy, the file data for the work.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the work. Otherwise, a set of default rights for backup files of the original are transmitted by the server.

The requester records the work contents, data, and usage rights. It then creates a one-time key and encrypts the contents file. It saves the key information in a restoration file.

The repositories perform the common closing transaction steps.

In some cases, it is convenient to be able to archive the large, encrypted contents file to secure offline storage, such as a magneto-optical storage system or magnetic tape. This creation of a non-repository archive file is as secure as the encryption process. Such non-repository archive storage is

US 8,370,956 B2

37

considered a form of “printing” and is controlled by a print right with a specified “archive-printer.” An archive-printer device is programmed to save the encrypted contents file (but not the description file) offline in such a way that it can be retrieved.

The Restore Transaction

A Restore transaction is a request to convert an encrypted backup copy of a digital work into a usable copy. A restore operation is intended to be used to compensate for catastrophic media failure. Like all usage rights, restoration rights can include fees and access tests including authorization checks.

The requester sends the server a message to initiate a Restore transaction. This message indicates the work to be restored, the version of the restore right for the transaction, the destination address information for placing the work, and the file data for the work.

The server verifies that the contents file is available (i.e. a digital work corresponding to the request has been backed-up.) If it is not, it ends the transaction with an error.

The repositories perform the common opening transaction steps.

The server retrieves the key from the restoration file. It decrypts the work contents, data, and usage rights.

The server transmits the requested contents and data to the requester according to the transmission protocol. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the work. Otherwise, a set of default rights for backup files of the original are transmitted by the server.

The requester stores the digital work.

The repositories perform the common closing transaction steps.

The Delete Transaction

A Delete transaction deletes a digital work or a number of copies of a digital work from a repository. Practically all digital works would have delete rights.

The requester sends the server a message to initiate a delete transaction. This message indicates the work to be deleted, the version of the delete right for the transaction.

The repositories perform the common opening transaction steps.

The server deletes the file, erasing it from the file system.

The repositories perform the common closing transaction steps.

The Directory Transaction

A Directory transaction is a request for information about folders, digital works, and their parts. This amounts to roughly the same idea as protection codes in a conventional file system like TENEX, except that it is generalized to the full power of the access specifications of the usage rights language.

The Directory transaction has the important role of passing along descriptions of the rights and fees associated with a digital work. When a user wants to exercise a right, the user interface of his repository implicitly makes a directory request to determine the versions of the right that are available. Typically these are presented to the user—such as with different choices of billing for exercising a right. Thus, many directory transactions are invisible to the user and are exercised as part of the normal process of exercising all rights.

The requester sends the server a message to initiate a Directory transaction. This message indicates the file or folder that is the root of the directory request and the version of the directory right used for the transaction.

38

The server verifies that the information is accessible to the requester.

In particular, it does not return the names of any files that have a HIDE-NAME status in their directory specifications, and it does not return the parts of any folders or files that have HIDE-PARTS in their specification. If the information is not accessible, the server ends the transaction with an error.

The repositories perform the common opening transaction steps.

The server sends the requested data to the requester according to the transmission protocol.

The requester records the data.

The repositories perform the common closing transaction steps.

The Folder Transaction

A Folder transaction is a request to create or rename a folder, or to move a work between folders. Together with Directory rights, Folder rights control the degree to which organization of a repository can be accessed or modified from another repository.

The requester sends the server a message to initiate a Folder transaction. This message indicates the folder that is the root of the folder request, the version of the folder right for the transaction, an operation, and data. The operation can be one of create, rename, and move file. The data are the specifications required for the operation, such as a specification of a folder or digital work and a name.

The repositories perform the common opening transaction steps.

The server performs the requested operation—creating a folder, renaming a folder, or moving a work between folders.

The repositories perform the common closing transaction steps.

The Extract Transaction

An extract transaction is a request to copy a part of a digital work and to create a new work containing it. The extraction operation differs from copying in that it can be used to separate a part of a digital work from d-blocks or shells that place additional restrictions or fees on it. The extraction operation differs from the edit operation in that it does not change the contents of a work, only its embedding in d-blocks. Extraction creates a new digital work.

The requester sends the server a message to initiate an Extract transaction. This message indicates the part of the work to be extracted, the version of the extract right to be used in the transaction, the destination address information for placing the part as a new work, the file data for the work, and the number of copies involved. The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester according to the transmission protocol. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the new work. Otherwise, the rights of the original are transmitted. The Copy-Count field for this right is set to the number-of-copies requested.

The requester records the contents, data, and usage rights and stores the work. It records the date and time that new work was made in the properties of the work.

The repositories perform the common closing transaction steps.

US 8,370,956 B2

39

The Embed Transaction

An embed transaction is a request to make a digital work become a part of another digital work or to add a shell d-block to enable the adding of fees by a distributor of the work.

The requester sends the server a message to initiate an Embed transaction. This message indicates the work to be embedded, the version of the embed right to be used in the transaction, the destination address information for placing the part as a work, the file data for the work, and the number of copies involved.

The server checks the control specifications for all of the rights in the part and the destination. If they are incompatible, the server ends the transaction with an error.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester according to the transmission protocol. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the new work. Otherwise, the rights of the original are transmitted. The Copy-Count field for this right is set to the number-of-copies requested.

The requester records the contents, data, and usage rights and embeds the work in the destination file.

The repositories perform the common closing transaction steps.

The Edit Transaction

An Edit transaction is a request to make a new digital work by copying, selecting and modifying portions of an existing digital work. This operation can actually change the contents of a digital work. The kinds of changes that are permitted depend on the process being used. Like the extraction operation, edit operates on portions of a digital work. In contrast with the extract operation, edit does not effect the rights or location of the work. It only changes the contents. The kinds of changes permitted are determined by the type specification of the processor specified in the rights. In the currently preferred embodiment, an edit transaction changes the work itself and does not make a new work. However, it would be a reasonable variation to cause a new copy of the work to be made.

The requester sends the server a message to initiate an Edit transaction. This message indicates the work to be edited, the version of the edit right to be used in the transaction, the file data for the work (including its size), the process-ID for the process, and the number of copies involved.

The server checks the compatibility of the process-ID to be used by the requester against any process-ID specification in the right. If they are incompatible, it ends the transaction with an error.

The repositories perform the common opening transaction steps.

The requester uses the process to change the contents of the digital work as desired. (For example, it can select and duplicate parts of it; combine it with other information; or compute functions based on the information. This can amount to editing text, music, or pictures or taking whatever other steps are useful in creating a derivative work.)

The repositories perform the common closing transaction steps.

The edit transaction is used to cover a wide range of kinds of works. The category describes a process that takes as its input any portion of a digital work and then modifies the input in some way. For example, for text, a process for editing the text would require edit rights. A process for "summarizing" or counting words in the text would also be considered editing.

40

For a music file, processing could involve changing the pitch or tempo, or adding reverberations, or any other audio effect. For digital video works, anything which alters the image would require edit rights. Examples would be colorizing, scaling, extracting still photos, selecting and combining frames into story boards, sharpening with signal processing, and so on.

Some creators may want to protect the authenticity of their works by limiting the kinds of processes that can be performed on them. If there are no edit rights, then no processing is allowed at all. A processor identifier can be included to specify what kind of process is allowed. If no process identifier is specified, then arbitrary processors can be used. For an example of a specific process, a photographer may want to allow use of his photograph but may not want it to be colorized. A musician may want to allow extraction of portions of his work but not changing of the tonality.

Authorization Transactions

There are many ways that authorization transactions can be defined. In the following, our preferred way is to simply define them in terms of other transactions that we already need for repositories. Thus, it is convenient sometimes to speak of "authorization transactions," but they are actually made up of other transactions that repositories already have.

A usage right can specify an authorization-ID, which identifies an authorization object (a digital work in a file of a standard format) that the repository must have and which it must process. The authorization is given to the generic authorization (or ticket) server of the repository which begins to interpret the authorization.

As described earlier, the authorization contains a server identifier, which may just be the generic authorization server or it may be another server. When a remote authorization server is required, it must contain a digital address. It may also contain a digital certificate.

If a remote authorization server is required, then the authorization process first performs the following steps:

The generic authorization server attempts to set up the communications channel. (If the channel cannot be set up, then authorization fails with an error.)

When the channel is set up, it performs a registration process with the remote repository. (If registration fails, then the authorization fails with an error.)

When registration is complete, the generic authorization server invokes a "Play" transaction with the remote repository, supplying the authorization document as the digital work to be played, and the remote authorization server (a program) as the "player." (If the player cannot be found or has some other error, then the authorization fails with an error.)

The authorization server then "plays" the authorization.

This involves decrypting it using either the public key of the master repository that issued the certificate or the session key from the repository that transmitted it. The authorization server then performs various tests. These tests vary according to the authorization server. They include such steps as checking issue and validity dates of the authorization and checking any hot-lists of known invalid authorizations. The authorization server may require carrying out any other transactions on the repository as well, such as checking directories, getting some person to supply a password, or playing some other digital work. It may also invoke some special process for checking information about locations or recent events. The "script" for such steps is contained within the authorization server.

US 8,370,956 B2

41

If all of the required steps are completed satisfactorily, the authorization server completes the transaction normally, signaling that authorization is granted.

The Install Transaction

An Install transaction is a request to install a digital work as runnable software on a repository. In a typical case, the requester repository is a rendering repository and the software would be a new kind or new version of a player. Also in a typical case, the software would be copied to file system of the requester repository before it is installed.

The requester sends the server an Install message. This message indicates the work to be installed, the version of the Install right being invoked, and the file data for the work (including its size).

The repositories perform the common opening transaction steps.

The requester extracts a copy of the digital certificate for the software. If the certificate cannot be found or the master repository for the certificate is not known to the requester, the transaction ends with an error.

The requester decrypts the digital certificate using the public key of the master repository, recording the identity of the supplier and creator, a key for decrypting the software, the compatibility information, and a tamper-checking code. (This step certifies the software.)

The requester decrypts the software using the key from the certificate and computes a check code on it using a 1-way hash function. If the check-code does not match the tamper-checking code from the certificate, the installation transaction ends with an error. (This step assures that the contents of the software, including the various scripts, have not been tampered with.)

The requester retrieves the instructions in the compatibility-checking script and follows them. If the software is not compatible with the repository, the installation transaction ends with an error. (This step checks platform compatibility.)

The requester retrieves the instructions in the installation script and follows them. If there is an error in this process (such as insufficient resources), then the transaction ends with an error. Note that the installation process puts the runnable software in a place in the repository where it is no longer accessible as a work for exercising any usage rights other than the execution of the software as part of repository operations in carrying out other transactions.

The repositories perform the common closing transaction steps.

The Uninstall Transaction

An Uninstall transaction is a request to remove software from a repository. Since uncontrolled or incorrect removal of software from a repository could compromise its behavioral integrity, this step is controlled.

The requester sends the server an Uninstall message. This message indicates the work to be uninstalled, the version of the Uninstall right being invoked, and the file data for the work (including its size).

The repositories perform the common opening transaction steps.

The requester extracts a copy of the digital certificate for the software. If the certificate cannot be found or the master repository for the certificate is not known to the requester, the transaction ends with an error.

The requester checks whether the software is installed. If the software is not installed, the transaction ends with an error.

42

The requester decrypts the digital certificate using the public key of the master repository, recording the identity of the supplier and creator, a key for decrypting the software, the compatibility information, and a tamper-checking code. (This step authenticates the certification of the software, including the script for uninstalling it.)

The requester decrypts the software using the key from the certificate and computes a check code on it using a 1-way hash function. If the check-code does not match the tamper-checking code from the certificate, the installation transaction ends with an error. (This step assures that the contents of the software, including the various scripts, have not been tampered with.)

The requester retrieves the instructions in the uninstallation script and follows them. If there is an error in this process (such as insufficient resources), then the transaction ends with an error.

The repositories perform the common closing transaction steps.

Distribution and Use Scenarios

To appreciate the robustness and flexibility of the present invention, various distribution and use scenarios for digital works are illustrated below. These scenarios are meant to be exemplary rather than exhaustive.

Consumers as Unpaid Distributors

In this scenario, a creator distributes copies of his works to various consumers. Each consumer is a potential distributor of the work. If the consumer copies the digital work (usually for a third party), a fee is collected and automatically paid to the creator.

This scenario is a new twist for digital works. It depends on the idea that "manufacturing" is just copying and is essentially free. It also assumes that the consumers as distributors do not require a fee for their time and effort in distributing the work.

This scenario is performed as follows:

A creator creates a digital work. He grants a Copy right with fees paid back to himself. If he does not grant an Embed right, then consumers cannot use the mechanism to act as distributors to cause fees to be paid to themselves on future copies. Of course, they could negotiate side deals or trades to transfer money on their own, outside of the system.

Paid Distributors

In another scenario, every time a copy of a digital work is sold a fee is paid to the creator and also to the immediate distributor.

This scenario does not give special status to any particular distributor. Anyone who sells a document has the right to add a fee to the sale price. The fee for sale could be established by the consumer. It could also be a fixed nominal amount that is contributed to the account of some charity.

This scenario is performed as follows:

A creator creates a digital work. He grants a Copy right with fees to be paid back to himself. He grants an Embed right, so that anyone can add shells to have fees paid to themselves.

A distributor embeds the work in a shell, with fees specified to be paid back to himself. If the distributor is content to receive fees only for copies that he sells himself, he grants an Extract right on the shell.

When a consumer buys a copy from the distributor, fees are paid both to the distributor and to the creator. If he chooses, the consumer can extract the work from the distributor's shell. He cannot extract it from the creator's shell. He can add his own shell with fees to be paid to himself.

US 8,370,956 B2

43

Licensed Distribution

In this scenario, a creator wants to protect the reputation and value of his work by making certain requirements on its distributors. He issues licenses to distributors that satisfy the requirements, and in turn, promises to reward their efforts by assuring that the work will not be distributed over competing channels. The distributors incur expenses for selecting the digital work, explaining it to buyers, promoting its sale, and possibly for the license itself. The distributor obtains the right to enclose the digital work in a shell, whose function is to permit the attachment of usage fees to be paid to the distributor in addition to the fees to be paid to the creator.

This differs from the previous scenario in that it precludes the typical copy owner from functioning as a distributor, since the consumer lacks a license to copy the document. Thus, a consumer cannot make copies, even for free. All copies must come initially from authorized distributors. This version makes it possible to hold distributors accountable in some way for the sales and support of the work, by controlling the distribution of certificates that enable distributors to legitimately charge fees and copy owners to make copies. Since licenses are themselves digital works, the same mechanisms give the creators control over distributors by charging for licenses and putting time limits on their validity.

This scenario is performed as follows:

A creator purchases a digital distribution license that he will hand out to his distributors. He puts access requirements (such as a personal license) on the Copy and Transfer rights on the distribution license so that only he can copy or transfer it.

The creator also creates a digital work. He grants an Embed right and a Copy right, both of which require the distribution license to be exercised. He grants a Play right so that the work can be played by anyone. He may optionally add a Transfer or Loan right, so that end consumers can do some non-commercial exchange of the work among friends.

A distributor obtains the distribution license and a number of copies of the work. He makes copies for his customers, using his distribution license.

A customer buys and uses the work. He cannot make new copies because he lacks a distribution license.

Super Distributors

This is a variation on the previous scenarios. A distributor can sell to anyone and anyone can sell additional copies, resulting in fees being paid back to the creator. However, only licensed distributors can add fees to be paid to themselves.

This scenario gives distributors the right to add fees to cover their own advertising and promotional costs, without making them be the sole suppliers. Their customers can also make copies, thus broadening the channel without diminishing their revenues. This is because distributors collect fees from copies of any copies that they originally sold. Only distributors can add fees.

This scenario is performed similarly to the previous ones. There are two key differences. (1) The creator only grants Embed rights for people who have a Distribution license. This is done by putting a requirement for a distributor's license on the Embed right. Consequently, non-distributors cannot add their own fees. (2) The Distributor does not grant Extract rights, so that consumers cannot avoid paying fees to the Distributor if they make subsequent copies. Consequently, all subsequent copies result in fees paid to the Distributor and the Creator.

1-Level Distribution Fees

In this scenario, a distributor gets a fee for any copy he sells directly. However, if one of his customers sells further copies, he gets no further fee for those copies.

44

This scenario pays a distributor only for use of copies that he actually sold.

This scenario is performed similarly to the previous ones. The key feature is that the distributor creates a shell which specifies fees to be paid to him. He puts Extract rights on the shell. When a consumer buys the work, he can extract away the distributor's shell. Copies made after that will not require fees to be paid to the distributor.

Distribution Trees

In another scenario, distributors sell to other distributors and fees are collected at each level. Every copy sold by any distributor—even several d-blocks down in the chain—results in a fee being paid back to all of the previous distributors.

This scenario is like a chain letter or value chain. Every contributor or distributor along the way obtains fees, and is thereby encouraged to promote the sale of copies of the digital work.

This scenario is performed similarly to the previous ones. The key feature is that the distributor creates a shell which specifies fees to be paid to him. He does not grant Extract rights on the shell. Consequently, all future copies that are made will result in fees paid to him.

Weighted Distribution Trees

In this scenario, distributors make money according to a distribution tree. The fee that they make depends on various parameters, such as time since their sale or the number of subsequent distributors.

This is a generalized version of the Distribution Tree scenario, in that it tries to vary the fee to account for the significance of the role of the distributor.

This scenario is similar to the previous one. The difference is that the fee specification on the distributor's shell has provisions for changes in prices. For example, there could be a fee schedule so that copies made after the passage of time will require lower fees to be paid to the distributor. Alternatively, the distributor could employ a "best-price" billing option, using any algorithm he chooses to determine the fee up to the maximum specified in the shell.

Fees for Reuse

In this scenario, a first creator creates a work. It is distributed by a first distributor and purchased by a second creator. The second creator extracts a portion of the work and embeds in it a new work distributed by a second distributor. A consumer buys the new work from the second distributor. The first creator receives fees from every transaction; the first distributor receives fees only for his sale; the second creator and second distributor receive fees for the final sale.

This scenario shows how that flexible automatic arrangements can be set up to create automatic charging systems that mirror current practice. This scenario is analogous to when an author pays a fee to reuse a figure in some paper. In the most common case, a fee is paid to the creator or publisher, but not to the bookstore that sold the book.

The mechanisms for derived works are the same as those for distribution.

Limited Reuse

In this scenario, several first creators create works. A second creator makes a selection of these, publishing a collection made up of the parts together with some new interstitial material. (For example, the digital work could be a selection of music or a selection of readings.) The second creator wants to continue to allow some of the selected works to be extractable, but not the interstitial material.

This scenario deals with fine grained control of the rights and fees for reuse.

US 8,370,956 B2

45

This scenario is performed as follows:

The first creators create their original works. If they grant extraction and embedding rights, then the second creator can include them in a larger collected work. The second creator creates the interstitial material. He does grant an Extract right on the interstitial material. He grants Extract rights on a subset of the reused material. A consumer of the collection can only extract portions that have that right. Fees are automatically collected for all parts of the collection.

Commercial Libraries

Commercial libraries buy works with the right to loan. They limit the loan period and charge their own fees for use. This scenario deals with fees for loaning rather than fees for making copies. The fees are collected by the same automatic mechanisms.

The mechanisms are the same as previous scenarios except that the fees are associated with the Loan usage right rather than the Copy usage right.

Demo Versions

A creator believes that if people try his work that they will want to buy it or use it. Consumers of his work can copy the work for free, and play (or execute) a limited version of the work for free, and can play or use the full featured version for a fee. This scenario deals with fees for loaning rather than fees for making copies. The fees are collected by the same automatic mechanisms.

This scenario is performed as follows:

The creator creates a digital work and grants various rights and fees. The creator grants Copy and Embed rights without a fee, in order to ensure widespread distribution of the work. Another of the rights is a limited play right with little or no fee attached. For example, this right may be for playing only a portion of the work. The play right can have various restrictions on its use. It could have a ticket that limits the number of times it is used. It could have internal restrictions that limit its functionality. It could have time restrictions that invalidate the right after a period of time or a period of use. Different fees could be associated with other versions of the Play right.

Upgrading a Digital Work with a Vendor

A consumer buys a digital work together with an agreement that he can upgrade to a new version at a later date for a modest fee, much less than the usual purchase price. When the new version becomes available, he goes to a qualified vendor to make the transaction.

This scenario deals with a common situation in computer software. It shows how a purchase may include future "rights." Two important features of the scenario are that the transaction must take place at a qualified vendor, and that the transaction can be done only once per copy of the digital work purchased.

This scenario is performed as follows:

The creator creates a digital work, an upgrade ticket, and a distribution license. The upgrade ticket uses the a generic ticket agent that comes with repositories. As usual, the distribution license does not have Copy or Transfer rights. He distributes a bundled copies of the work and the ticket to his distributors as well as distribution licenses.

The distributor sells the old bundled work and ticket to customers.

The customer extracts the work and the ticket. He uses the work according to the agreements until the new version becomes available.

When the new work is ready, the creator gives it to distributors. The new work has a free right to copy from a distributor if a ticket is available.

46

The consumer goes to distributors and arranges to copy the work. The transaction offers the ticket. The distributor's repository punches the ticket and copies the new version to the consumer's repository.

5 The consumer can now use the new version of the work. Distributed Upgrading of Digital Works

A consumer buys a digital work together with an agreement that he can upgrade to a new version at a later date for a modest fee, much less than the usual purchase price. When the new version becomes available, he goes to anyone who has the upgraded version and makes the transaction.

10 This scenario is like the previous one in that the transaction can only be done once per copy of the digital work purchased, but the transaction can be accomplished without the need to connect to a licensed vendor.

15 This scenario is similar to the previous one except that the Copy right on the new work does not require a distribution license. The consumer can upgrade from any repository having the new version. He cannot upgrade more than once because the ticket cannot work after it has been punched. If desired, the repository can record the upgrade transaction by posting a zero cost bill to alert the creator that the upgrade has taken place.

Limited Printing

25 A consumer buys a digital work and wants to make a few ephemeral copies. For example, he may want to print out a paper copy of part of a digital newspaper, or he may want to make a (first generation) analog cassette tape for playing in his car. He buys the digital work together with a ticket required for printing rights.

30 This scenario is like the common practice of people making cassette tapes to play in their car. If a publisher permits the making of cassette tapes, there is nothing to prevent a consumer from further copying the tapes. However, since the tapes are "analog copies," there is a noticeable quality loss with subsequent generations. The new contribution of the present invention is the use of tickets in the access controls for the making of the analog copies.

This scenario is performed as follows:

40 The creator sells a work together with limited printing rights. The printing rights specify the kind of printer (e.g., a kind of cassette recorder or a kind of desktop paper printer) and also the kind of ticket required. The creator either bundles a limited number of tickets or sells them separately. If the tickets use the generic ticket agent, the consumer with the tickets can exercise the right at his convenience.

Demand Publishing

Professors in a business school want to put together course books of readings selected from scenario studies from various sources. The bookstore wants to be able to print the books from digital masters, without negotiating for and waiting for approval of printing of each of the scenarios. The copyright holders of the scenarios want to be sure that they are paid for every copy of their work that is printed.

50 On many college campuses, the hassle of obtaining copy clearances in a timely way has greatly reduced the viability of preparing course books. Print shops have become much more cautious about copying works in the absence of documented permission.

60 Demand Publishing is performed as follows: the creator sells a work together with printing rights for a fee. There can be rights to copy (distribute) the work between bookstore repositories, with or without fee. The printing rights specify the kind of printer. Whenever a bookstore prints one of the works (either standalone or embedded in a collection), the fee is credited to the creator automatically. To discourage unauthorized copying of the print outs, it would be possible for the

US 8,370,956 B2

47

printer to print tracer messages discretely on the pages identifying the printing transaction, the copy number, and any other identifying information. The tracer information could be secretly embedded in the text itself (encoded in the grey scale) or hidden in some other way.

Metered Use and Multiple Price Packages

A consumer does not know what music to purchase until he decides whether he likes it. He would like to be able to take it home and listen to it, and then decide whether to purchase. Furthermore, he would like the flexibility of paying less if he listens to it very infrequently.

This scenario just uses the capability of the approach to have multiple versions of a right on a digital work. Each version of the right has its own billing scheme. In this scenario, the creator of the work can offer the Copy right without fee, and defer billing to the exercise of the Play right. One version of the play right would allow a limited performance without fee—a right to “demo”. Another version of the right could have a metered rate, of say \$0.25 per hour of play. Another version could have a fee of \$15.00 for the first play, but no fee for further playing. When the consumer exercises a play right, he specifies which version of the right is being selected and is billed accordingly.

Fees for Font Usage

A designer of type fonts invests several months in the design of special fonts. The most common way of obtaining revenue for this work is to sell copies of the fonts to publishers for unlimited use over unlimited periods of time. A font designer would like to charge a rate that reflects the amount that the font is used.

This scenario is performed as follows: the font designer creates a font as a digital work. He creates versions of the Play right that bill either for metered use or “per-use”. Each version of the play right would require that the player (a print layout program) be of an approved category. The font designer assigns appropriate fees to exercise the Copy right. When a publisher client wants to use a font, he includes it as input to a layout program, and is billed automatically for its use. In this way, a publisher who makes little use of a font pays less than one who uses it a lot.

Rational Database Usage Charges

Online information retrieval services typically charge for access in a way that most clients find unpredictable and uncorrelated to value or information use. The fee depends on which databases are open, dial-up connect time, how long the searches require, and which articles are printed out. There are no provisions for extracting articles or photographs, no method for paying to reuse information in new works, no distinction between having the terminal sit idly versus actively searching for data, no distinction between reading articles on the screen and doing nothing, and higher rates per search when the centralized facility is busy and slow servicing other clients. Articles can not be offloaded to the client’s machine for off-site search and printing. To offer such billing or the expanded services, the service company would need a secure way to account for and bill for how information is used.

This scenario is performed as follows:

The information service bundles its database as files in a repository. The information services company assigns different fees for different rights on the information files. For example, there could be a fee for copying a search database or a source file and a different fee for printing. These fees would be in addition to fees assigned by the original creator for the services. The fees for using information would be different for using them on the information service company’s computers or the client’s computers. This billing distinction

48

would be controlled by having different versions of the rights, where the version for use on the service company’s computer requires a digital certificate held locally. Fees for copying or printing files would be handled in the usual way, by assigning fees to exercising those rights. The distinction between searching and viewing information would be made by having different “players” for the different functions. This distinction would be maintained on the client’s computers as well as the service computers. Articles could be extracted for reuse under the control of Extract and Embed rights. Thus, if a client extracts part of an article or photograph, and then sells copies of a new digital work incorporating it, fees could automatically be collected both by the information service and earlier creators and distributors of the digital work. In this way, the information retrieval service could both offer a wider selection of services and billing that more accurately reflects the client’s use of the information.

Print Spooling with Rights

In the simplest scenario, when a user wants to print a digital document he issues a print command to the user interface. If the document has the appropriate rights and the conditions are satisfied, the user agrees to the fee and the document is printed. In other cases, the printer may be on a remote repository and it is convenient to spool the printing to a later time. This leads to several issues. The user requesting the printing wants to be sure that he is not billed for the printing until the document is actually printed. Restated, if he is billed at the time the print job is spooled but the job is canceled before printing is done, he does not want to pay. Another issue is that when spooling is permitted, there are now two times at which rights, conditions and fees could be checked: the time at which a print job is spooled and the time at which a print is made. As with all usage rights, it is possible to have rights that expire and to have rights whose fee depends on various conditions. What is needed is a means to check rights and conditions at the time that printing is actually done.

This scenario is performed as follows: A printing repository is a repository with the usual repository characteristics plus the hardware and software to enable printing. Suppose that a user logs into a home repository and wants to spool print jobs for a digital work at a remote printing repository. The user interface for this could treat this as a request to “spool” prints. Underneath this “spooling” request, however, are standard rights and requests. To support such requests, the creator of the work provides a Copy right, which can be used to copy the work to a printing repository. In the default case, this Copy right would have no fees associated for making the copy. However, the Next-Set-Of-Rights for the copy would only include the Print rights, with the usual fees for each variation of printing. This version of the Copy right could be called the “print spooling” version of the Copy right. The user’s “spool request” is implemented as a Copy transaction to put a copy of the work on the printing repository, followed by Print transactions to create the prints of the work. In this way, the user is only billed for printing that is actually done. Furthermore, the rights, conditions and fees for printing the work are determined when the work is about to be printed.

Thus, a system for enforcing the usage rights of digital works is disclosed. While the embodiments disclosed herein are preferred, it will be appreciated from this teaching that various alternative, modifications, variations or improvements therein may be made by those skilled in the art, which are intended to be encompassed by the following claims.

US 8,370,956 B2

49

APPENDIX A

Glossary

Authorization Repository:

A special type of repository which provides an authorization service. An authorization may be specified by a usage right. The authorization must be obtained before the right may be exercised.

Billing Clearinghouse:

A financial institution or the like whose purpose is to reconcile billing information received from credit servers. The billing clearinghouse may generate bills to users or alternatively, credit and debit accounts involved in the commercial transactions.

Billing Transactions:

The protocol used by which a repository reports billing information to a credit server.

Clearinghouse Transactions:

The protocol used between a credit server and a clearinghouse.

Composite Digital Work:

A digital work comprised of distinguishable parts. Each of the distinguishable parts is itself a digital work which has usage rights attached.

Content:

The digital information (i.e. raw bits) representing a digital work.

Copy Owner:

A term which refers to the party who owns a digital work stored in a repository. In the typical case, this party has purchased various rights to the document for printing, viewing, transferring, or other specific uses.

Creator:

A term which refers to a party who produces a digital work.

Credit Server:

A device which collects and reports billing information for a repository. In many implementations, this could be built as part of a repository. It requires a means for periodically communicating with a billing clearinghouse.

Description Tree:

A structure which describes the location of content and the usage rights and usage fees for a digital work. A description tree is comprised of description blocks. Each description block corresponds to a digital work or to an interest (typically a revenue bearing interest) in a digital work.

Digital Work (Work):

Any encapsulated digital information. Such digital information may represent music, a magazine or book, or a multimedia composition. Usage rights and fees are attached to the digital work.

Distributor:

A term which refers to a party who legitimately obtains a copy of a digital work and offers it for sale.

Identification (Digital) Certificate:

A signed digital message that attests to the identity of the possessor. Typically, digital certificates are encrypted in the private key of a well-known master repository.

Master Repository:

A special type of repository which issues identification certificates and distributes lists of repositories whose integrity have been compromised and which should be denied access to digital works (referred to as repository "hotlists".)

Public Key Encryption:

An encryption technique used for secure transmission of messages on a communication channel. Key pairs are used for the encryption and decryption of messages. Typically one key

50

is referred to as the public key and the other is the private key. The keys are inverses of each other from the perspective of encryption. Restated, a digital work that is encrypted by one key in the pair can be decrypted only by the other.

5 Registration Transactions:

The protocol used between repositories to establish a trusted session.

Rendering Repository:

10 A special type of repository which is typically coupled to a rendering system. The rendering repository will typically be embodied within the secure boundaries of a rendering system.

Rendering System:

15 The combination of a rendering repository and a rendering device. Examples of a rendering systems include printing systems, display systems, general purpose computer systems, video systems or audio systems.

Repository:

20 Conceptually a set of functional specifications defining core functionality in the support of usage rights. A repository is a trusted system in that it maintains physical, communications and behavioral integrity.

Requester Mode:

25 A mode of a repository where it is requesting access to a digital work.

Revenue Owners:

A term which refers to the parties that maintain an interest in collecting fees for document use or who stand to lose revenue if illegitimate copies of the digital work are made.

30 Server Mode:

A mode of a repository where it is processing an incoming request to access a digital work.

Shell Description Block:

35 A special type of description block designating an interest in a digital work, but which does not add content. This will typically be added by a distributor of a digital work to add their fees.

Transactions:

40 A term used to refer to the protocols by which repositories communicate.

Usage Fees:

A fee charged to a requester for access to a digital work. Usage fees are specified within the usage rights language.

45 Usage Rights:

A language for defining the manner in which a digital work may be used or distributed, as well as any conditions on which use or distribution is premised.

Usage Transactions:

50 A set of protocols by which repositories communicate in the exercise of a usage rights. Each usage right has its own transaction steps.

What is claimed:

55 **1.** A computer-implemented method of rendering digital content by at least one recipient computing device in accordance with usage rights information, the method comprising: receiving the digital content by the at least one recipient computing device from at least one sending computing device only if the at least one recipient computing device has been determined to be trusted to receive the digital content from the at least one sending computing device; receiving, by the at least one recipient computing device, a request to render the digital content; determining, based on the usage rights information, whether the digital content may be rendered by the at least one recipient computing device; and

US 8,370,956 B2

51

rendering the digital content, by the at least one recipient computing device, only if it is determined that the content may be rendered by the at least one recipient computing device.

2. The method of claim 1, further comprising denying the request and preventing rendering of the digital content by the at least one recipient computing device if it is determined that the digital content may not be rendered by the at least one recipient computing device.

3. The method of claim 1, wherein the usage rights information further includes a condition under which the content can be rendered, and the determining step further includes determining whether the condition is satisfied.

4. The method of claim 1, wherein the receiving the digital content comprises:

- requesting an authorization object for the at least one recipient computing device to make the digital content available for use, the authorization object being required to receive the digital content and to use the digital content; and
- receiving the authorization object if it is determined that the request for the authorization object should be granted.

5. The method of claim 1, wherein the receiving the digital content comprises:

- generating a registration message, the registration message including an identification certificate of the recipient computing device and a random registration identifier, the identification certificate being certified by a master device;
- exchanging messages including at least one session key with at least one provider computing device, the session key to be used in communications during a session; and
- conducting a secure transaction using the session key, wherein the secure transaction includes receiving the digital content.

6. The method of claim 5, further comprising:

- receiving a message to test the authenticity of the at least one recipient computing device, the generated message including a nonce; and
- processing the generated message to indicate authenticity.

7. A recipient apparatus for rendering digital content in accordance with usage rights information, the recipient apparatus comprising:

- one or more processors; and
- one or more memories operatively coupled to at least one of the one or more processors and having instructions stored thereon that, when executed by at least one of the one or more processors, cause at least one of the one or more processors to:
 - enable the receipt of the digital content by the recipient apparatus from at least one sending computing device only if the recipient apparatus has been determined to be trusted to receive the digital content from the at least one sending computing device;
 - receive a request to render the digital content;
 - determine, based on the usage rights information, whether the digital content may be rendered by the recipient apparatus; and
 - render the digital content only if it is determined that the content may be rendered by the recipient apparatus.

8. The recipient apparatus of claim 7, further comprising denying the request and preventing rendering of the digital content by the at least one recipient computing device if it is determined that the digital content may not be rendered by the at least one recipient computing device.

52

9. The recipient apparatus of claim 7, wherein the usage rights information further includes a condition under which the content can be rendered, and the determining step further includes determining whether the condition is satisfied.

10. The recipient apparatus of claim 7, wherein enabling the receipt of the digital content comprises:

- requesting an authorization object for the recipient apparatus to make the digital content available for use, the authorization object being required to receive the digital content and to use the digital content; and
- receiving the authorization object if it is determined that the request for the authorization object should be granted.

11. The recipient apparatus of claim 7, wherein enabling the receipt of the digital content comprises:

- generating a registration message, the registration message including an identification certificate of the recipient apparatus and a random registration identifier, the identification certificate being certified by a master device;
- exchanging messages including at least one session key with at least one provider computing device, the session key to be used in communications during a session; and
- conducting a secure transaction using the session key, wherein the secure transaction includes receiving the digital content.

12. The recipient apparatus of claim 11, wherein at least one of the one or more memories has further instructions stored thereon that, when executed by at least one of the one or more processors, cause at least one of the one or more processors to:

- receive a message to test the authenticity of the recipient apparatus, the generated message including a nonce; and
- process the generated message to indicate authenticity.

13. At least one non-transitory computer-readable medium storing computer-readable instructions that, when executed by at least one recipient computing device, cause the at least one recipient computing device to:

- receive the digital content from at least one sending computing device only if the at least one recipient computing device has been determined to be trusted to receive the digital content from the at least one sending computing device;
- receive a request to render the digital content;
- determine, based on the usage rights information, whether the digital content may be rendered by the at least one recipient computing device; and
- render the digital content only if it is determined that the content may be rendered by the at least one recipient computing device.

14. The at least one non-transitory computer-readable medium of claim 13, further storing computer-readable instructions that, when executed by at least one recipient computing device, cause the at least one recipient computing device to deny the request and prevent rendering of the digital content by the at least one recipient computing device if it is determined that the digital content may not be rendered by the at least one recipient computing device.

15. The at least one non-transitory computer-readable medium of claim 13, wherein the usage rights information further includes a condition under which the content can be rendered, and the determining step further includes determining whether the condition is satisfied.

16. The at least one non-transitory computer-readable medium of claim 13, wherein receiving the digital content comprises:

- requesting an authorization object for the at least one recipient computing device to make the digital content

US 8,370,956 B2

53

available for use, the authorization object being required to receive the digital content and to use the digital content; and receiving the authorization object if it is determined that the request for the authorization object should be granted.

17. The at least one non-transitory computer-readable medium of claim 13, wherein receiving the digital content comprises:

generating a registration message, the registration message including an identification certificate of the recipient computing device and a random registration identifier, the identification certificate being certified by a master device;

exchanging messages including at least one session key with at least one provider computing device, the session key to be used in communications during a session; and

54

conducting a secure transaction using the session key, wherein the secure transaction includes receiving the digital content.

18. The at least one non-transitory computer-readable medium of claim 17, further storing computer-readable instructions that, when executed by at least one recipient computing device, cause the at least one recipient computing device to:

receive a message to test the authenticity of the at least one recipient computing device, the generated message including a nonce; and

process the generated message to indicate authenticity.

* * * * *



US008393007B2

(12) **United States Patent**
Stefik et al.

(10) **Patent No.:** **US 8,393,007 B2**

(45) **Date of Patent:** ***Mar. 5, 2013**

(54) **SYSTEM AND METHOD FOR DISTRIBUTING DIGITAL CONTENT TO BE RENDERED IN ACCORDANCE WITH USAGE RIGHTS INFORMATION**

(75) Inventors: **Mark J. Stefik**, Portola Valley, CA (US);
Peter L. T. Pirolli, San Francisco, CA (US)

(73) Assignee: **ContentGuard Holdings, Inc.**,
 Wilmington, DE (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.
 This patent is subject to a terminal disclaimer.

(21) Appl. No.: **13/585,408**

(22) Filed: **Aug. 14, 2012**

(65) **Prior Publication Data**
 US 2012/0317658 A1 Dec. 13, 2012

Related U.S. Application Data

(60) Continuation of application No. 13/584,782, filed on Aug. 13, 2012, which is a continuation of application No. 11/304,793, filed on Dec. 16, 2005, now abandoned, which is a division of application No. 11/135,352, filed on May 24, 2005, now Pat. No.

(Continued)

(51) **Int. Cl.**
G06F 7/04 (2006.01)

(52) **U.S. Cl.** **726/29**

(58) **Field of Classification Search** None
 See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,817,140 A 3/1989 Chandra et al.
 5,204,961 A 4/1993 Barlow
 5,390,297 A 2/1995 Barber et al.
 6,135,646 A 10/2000 Kahn et al.

FOREIGN PATENT DOCUMENTS

EP 0398492 A2 11/1990
 EP 0588415 A1 3/1994

OTHER PUBLICATIONS

Kohl, John T. et al., "The Evolution of the Kerberos Authentication Service", Distributed Open Systems, IEEE, 1994, 18 pages.
 Non-Final Office Action dated Jun. 12, 2008 cited in U.S. Appl. No. 11/304,793.
 Final Office Action dated Nov. 14, 2008 cited in U.S. Appl. No. 11/304,793.

(Continued)

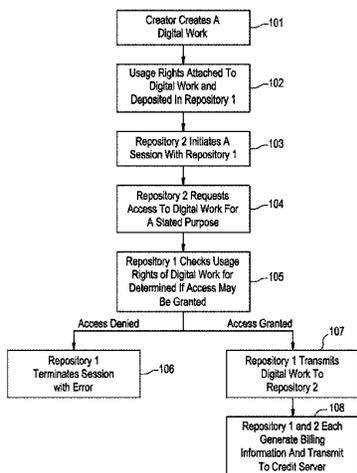
Primary Examiner — Brandon Hoffman

(74) *Attorney, Agent, or Firm* — Marc S. Kaufman; Stephen M. Hertzler; Reed Smith LLP

(57) **ABSTRACT**

Methods, apparatus, and media for distributing digital content to at least one recipient computing device to be rendered by the at least one recipient computing device in accordance with usage rights information. An exemplary method comprises determining, by at least one sending computing device, if the at least one recipient computing device is trusted to receive the digital content from the at least one sending computing device, sending the digital content, by the at least one sending computing device, to the at least one recipient computing device only if the at least one recipient computing device has been determined to be trusted to receive the digital content from the at least one sending computing device, and sending usage rights information indicating how the digital content may be rendered by the at least one recipient computing device, the usage rights information being enforceable by the at least on recipient computing device.

15 Claims, 13 Drawing Sheets



US 8,393,007 B2

Page 2

Related U.S. Application Data

7,266,529, which is a continuation of application No. 10/322,759, filed on Dec. 19, 2002, now Pat. No. 6,898,576, which is a continuation of application No. 09/778,001, filed on Feb. 7, 2001, now Pat. No. 6,708,157, which is a division of application No. 08/967,084, filed on Nov. 10, 1997, now Pat. No. 6,236,971, which is a continuation of application No. 08/344,760, filed on Nov. 23, 1994, now abandoned.

(56)

References Cited

OTHER PUBLICATIONS

Non-Final Office Action dated May 27, 2009 cited in U.S. Appl. No. 11/304,793.
Final Office Action dated Jan. 22, 2010 cited in U.S. Appl. No. 11/304,793.
Decision on Appeal dated Jun. 13, 2012 cited in U.S. Appl. No. 11/304,793.

FIG. 1

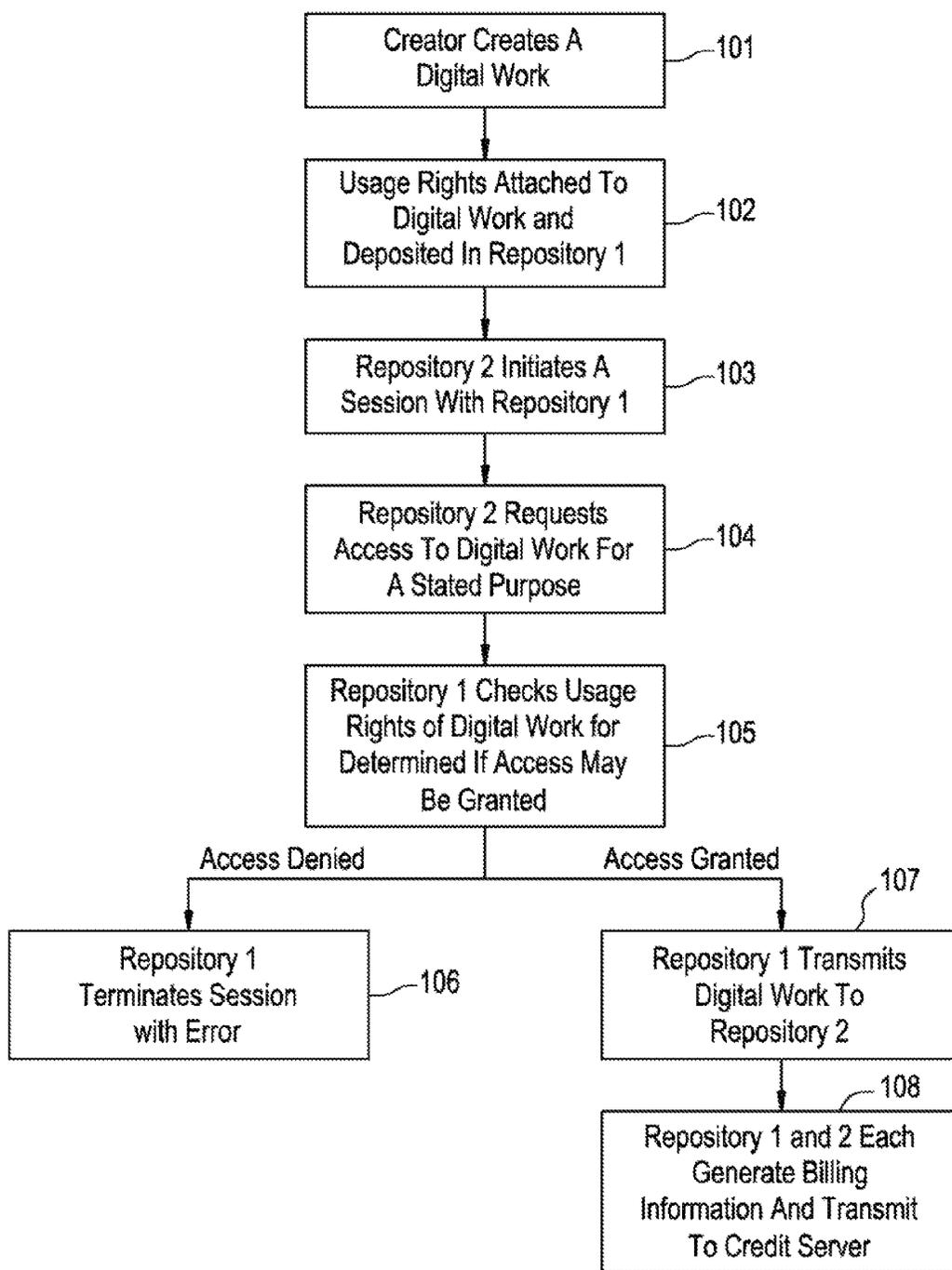


FIG. 2

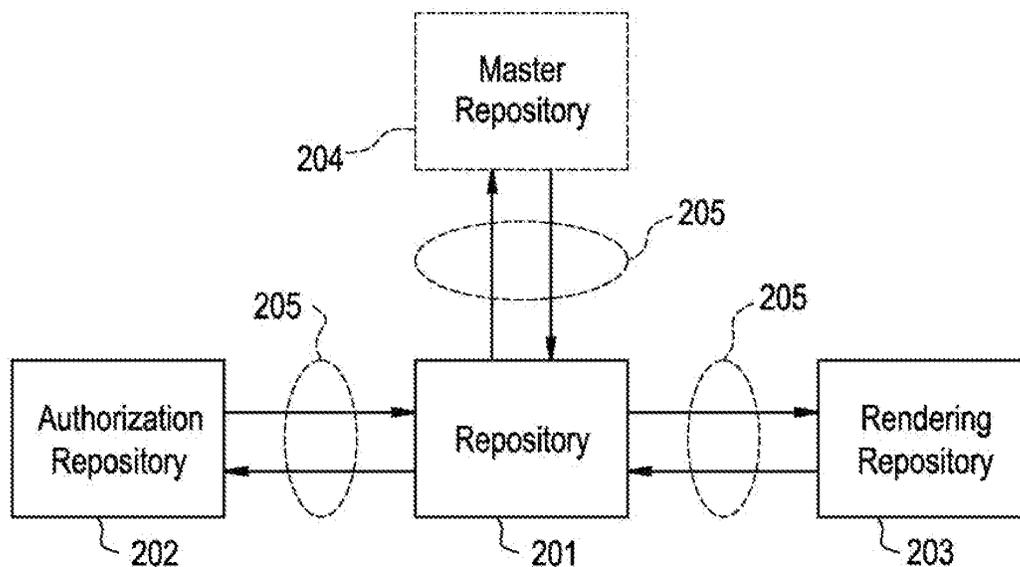


FIG. 3

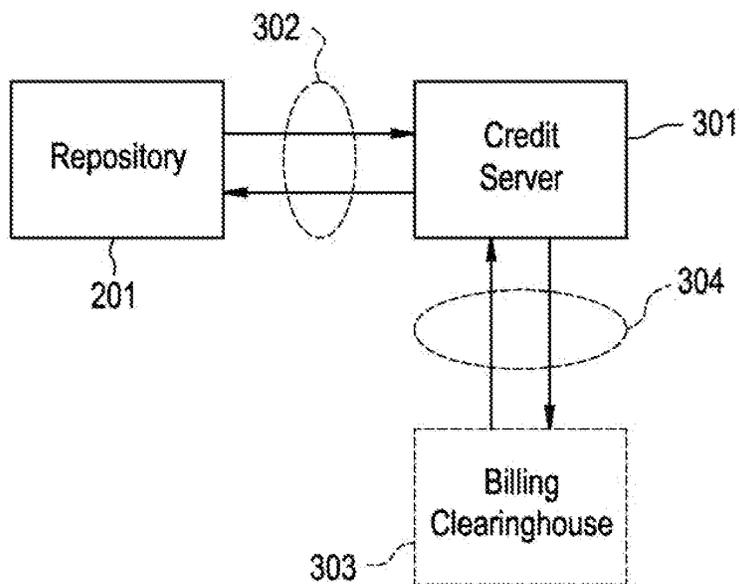


FIG. 4A

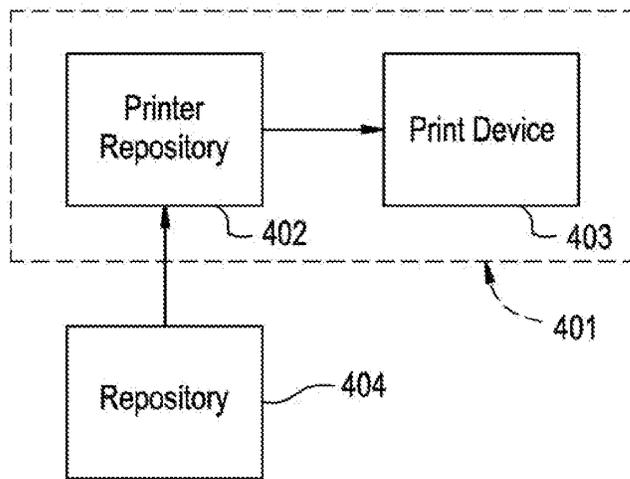


FIG. 4B

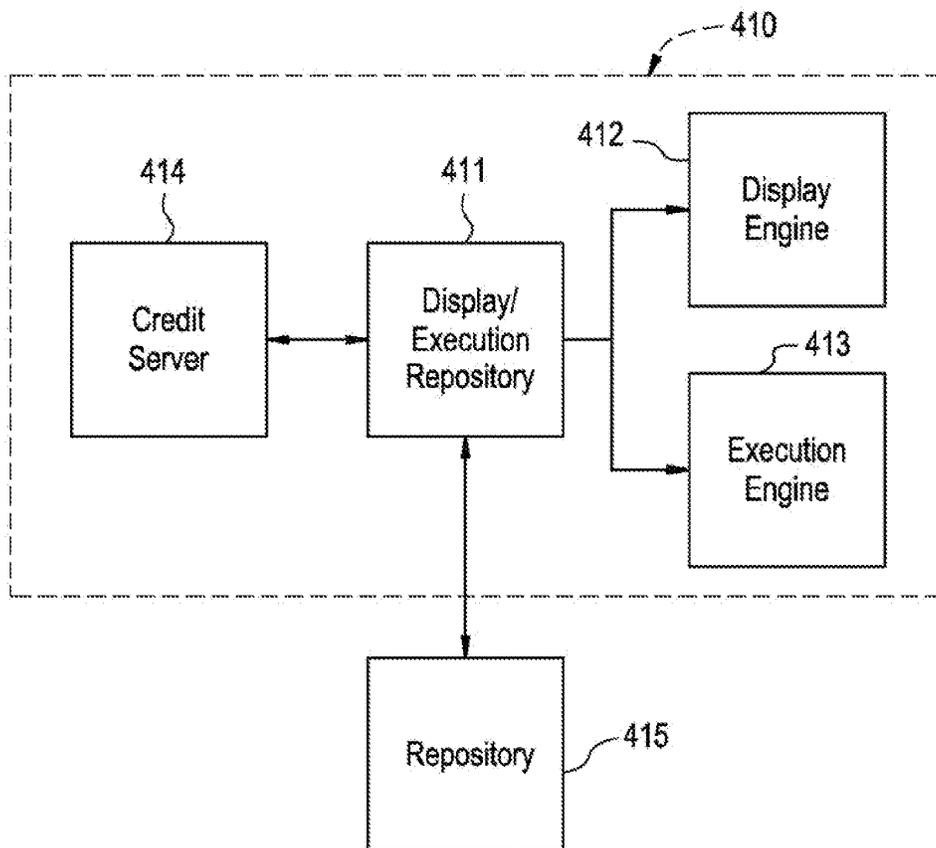


FIG. 5

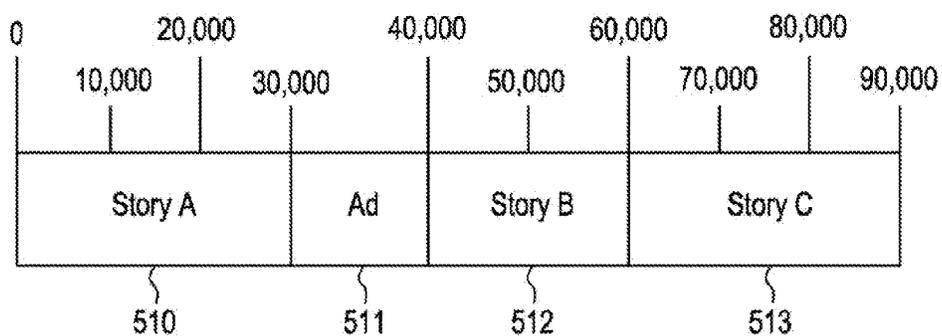


FIG. 6

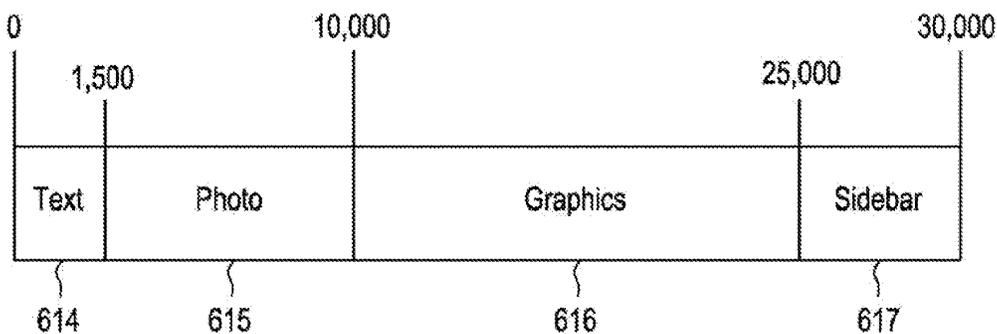


FIG. 7

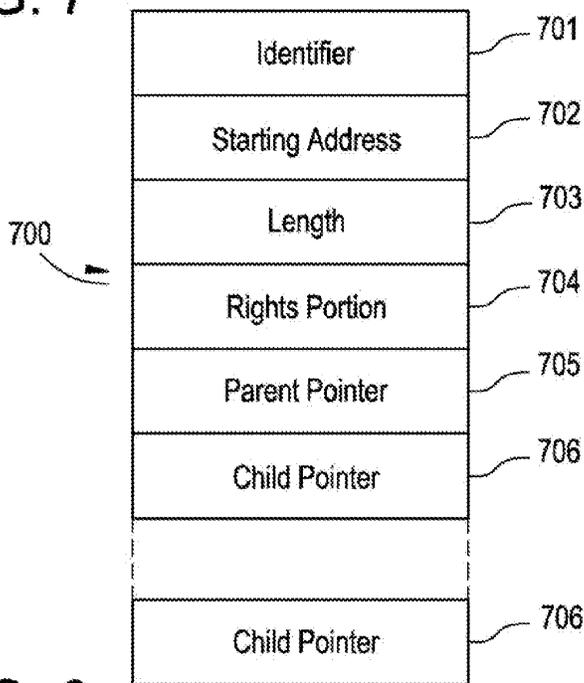


FIG. 8

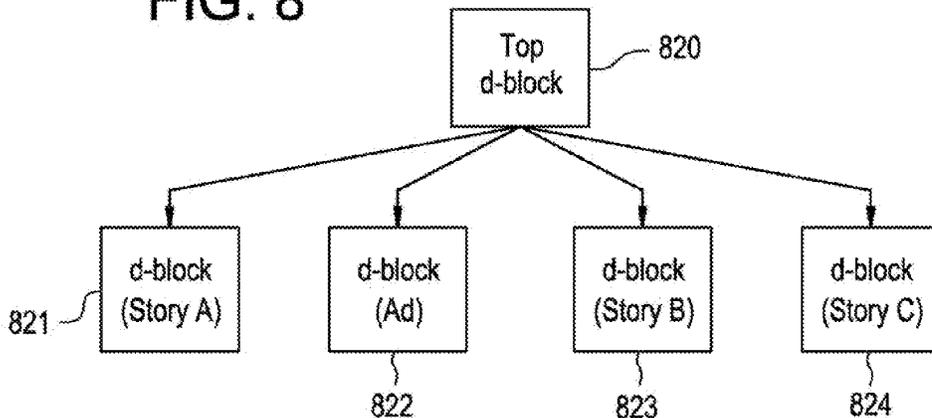


FIG. 9

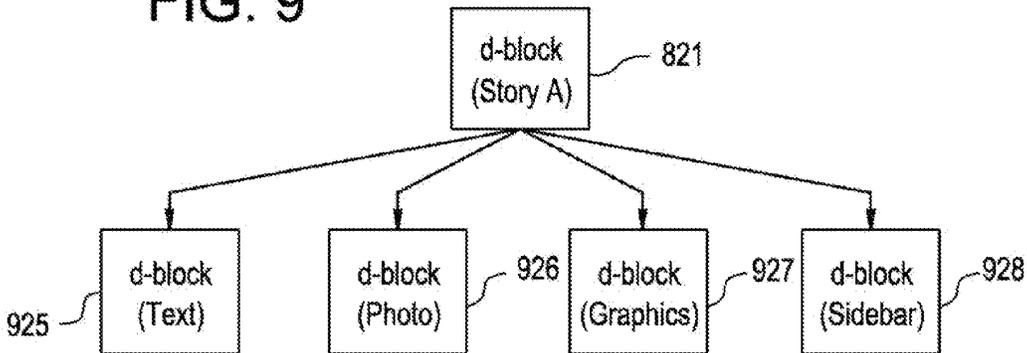


FIG. 10

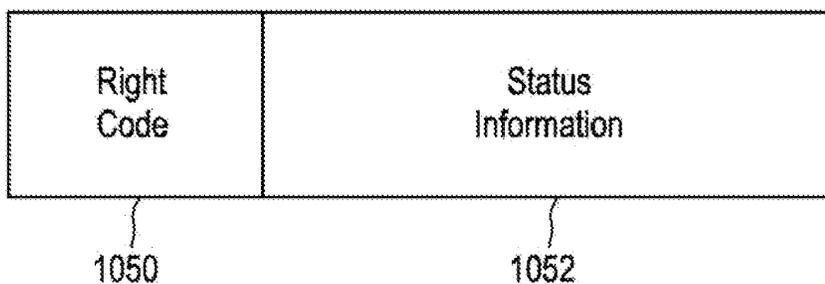


FIG. 14

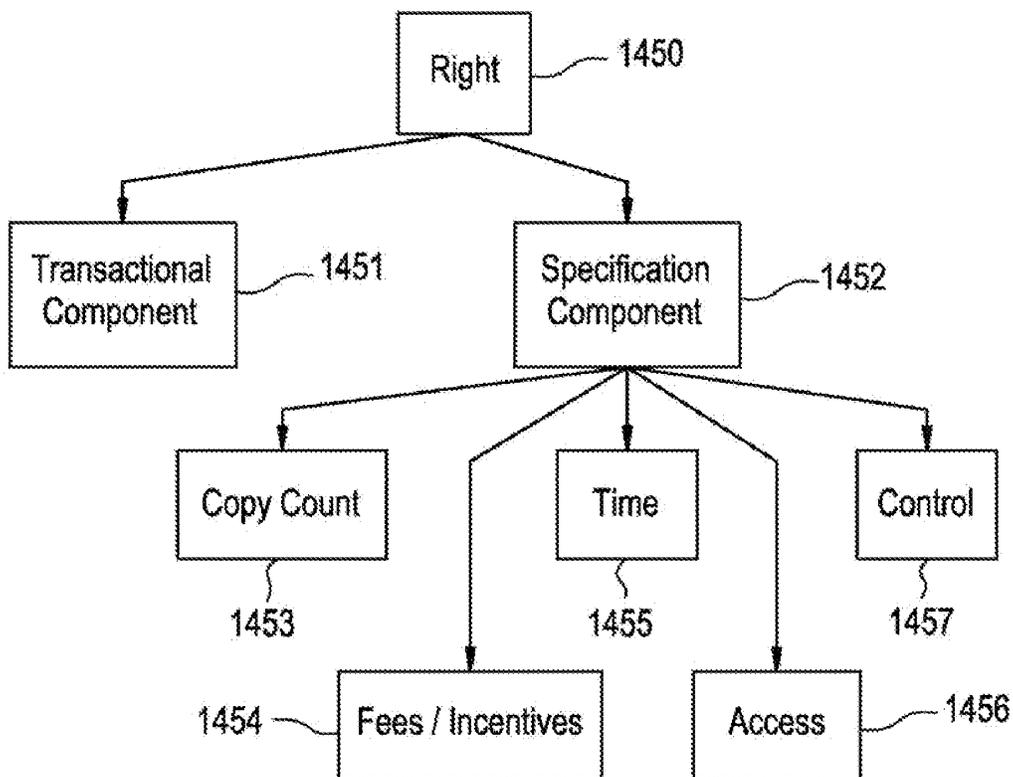


FIG. 11

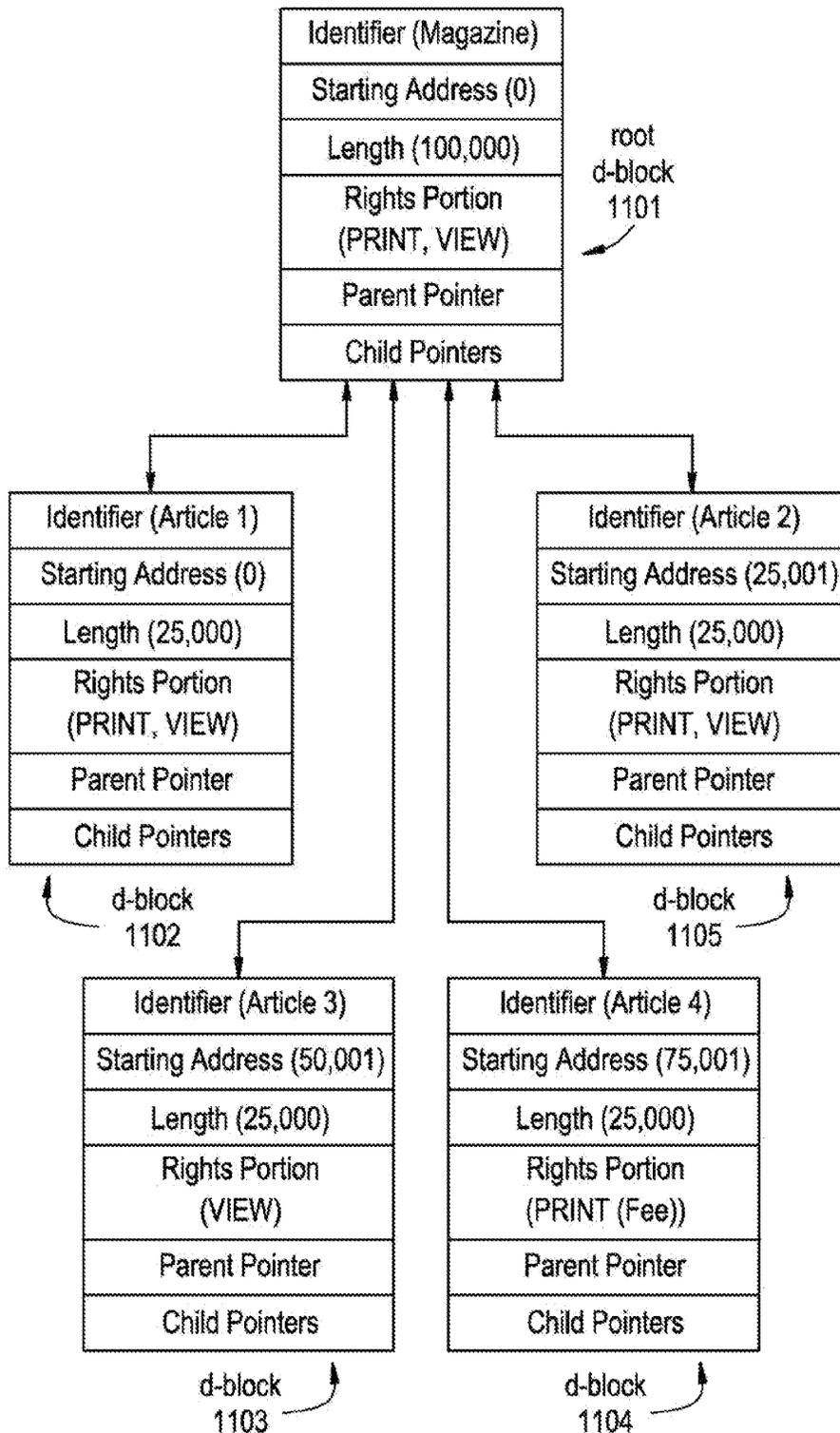


FIG. 12

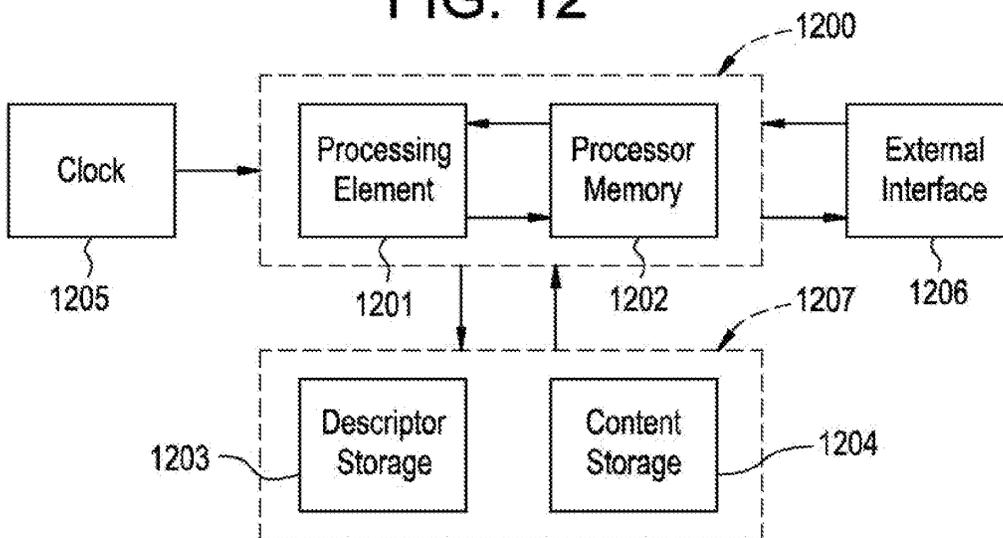


FIG. 13

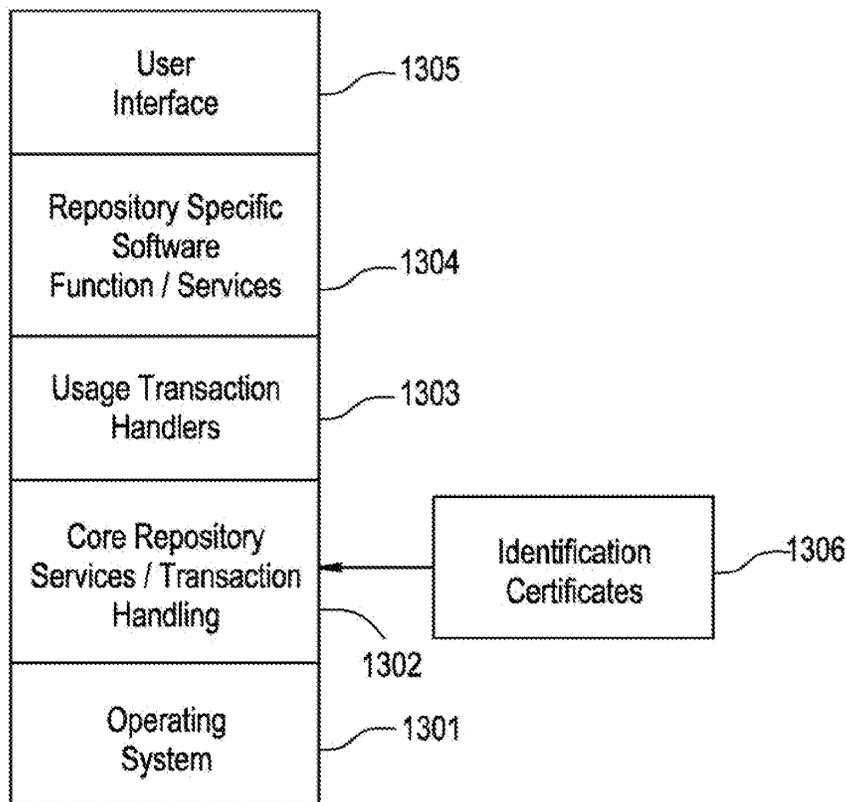


FIG. 15

- 1501 ~ Digital Work Rights: = (Rights*)
- 1502 ~ Right: = (Right-Code {Copy-Count} {Control-Spec} {Time-Spec}
{Access-Spec} {Fee-Spec})
- 1503 ~ Right-Code: = Render-Code | Transport-Code | File-Management-
Code | Derivative-Works-Code | Configuration-Code
- 1504 ~ Render-Code: = [Play: {Player: Player-ID} | Print: {Printer: Printer-ID}]
- 1505 ~ Transport-Code: = [Copy | Transfer | Loan {Remaining-Rights:
Next-Set-of-Rights}] {(Next-Copy-Rights: Next-Set-of-Rights)}
- 1506 ~ File-Management-Code: = Backup {Back-Up-Copy-Rights:
Next-Set-of-Rights} | Restore | Delete | Folder
| Directory {Name: Hide-Local | Hide-Remote}
{Parts: Hide-Local | Hide-Remote}
- 1507 ~ Derivative-Works-Code: = [Extract | Embed | Edit {Process:
Process-ID}] {Next-Copy-Rights:
Next-Set-of-Rights}
- 1508 ~ Configuration-Code: = Install | Uninstall
- 1509 ~ Next-Set-of-Rights: = {(Add: Set-of-Rights)} {(Delete:
Set-of-Rights)} {(Replace: Set-of-Rights)} {(Keep: Set-of-Rights)}
- 1510 ~ Copy-Count: = (Copies: positive-integer | 0 | Unlimited)
- 1511 ~ Control-Spec: = (Control: {Restrictable | Unrestrictable}
{Unchargeable | Chargeable})
- 1512 ~ Time-Spec: = ({Fixed-Interval | Sliding-Interval | Meter-Time}
Until: Expiration-Date)
- 1513 ~ Fixed-Interval: = From: Start-Time
- 1514 ~ Sliding-Interval: = Interval : Use-Duration
- 1515 ~ Meter-Time: = Time-Remaining: Remaining-Use
- 1516 ~ Access-Spec: = ({SC: Security-Class} {Authorization: Authorization-ID*}
{Other-Authorization: Authorization-ID*} {Ticket: Ticket-ID})
- 1517 ~ Fee-Spec: = {Scheduled-Discount} Regular-Fee-Spec | Scheduled-Fee-Spec |
Markup-Spec
- 1518 ~ Scheduled-Discount: = Scheduled-Discount: (Scheduled-Discount:
(Time-Spec Percentage)*)
- 1519 ~ Regular-Fee-Spec: = ({Fee: | Incentive:} [Per-Use-Spec | Metered-Rate-
Spec | Best-Price-Spec | Call-For-Price-Spec]
{Min: Money-Unit Per: Time-Spec} {Max:
Money-Unit Per: Time-Spec} To: Account-ID)
- 1520 ~ Per-Use-Spec: = Per-Use: Money-Unit
- 1521 ~ Metered-Rate-Spec: = Metered: Money-Unit Per: Time-Spec
- 1522 ~ Best-Price-Spec: = Best-Price: Money-unit Max: Money-Unit
- 1523 ~ Call-For-Price-Spec: = Call-For-Price
- 1524 ~ Scheduled-Fee-Spec: = (Schedule: (Time-Spec Regular-Fee-Spec)*)
- 1525 ~ Markup-Spec: = Markup: percentage To: Account-ID

FIG. 16

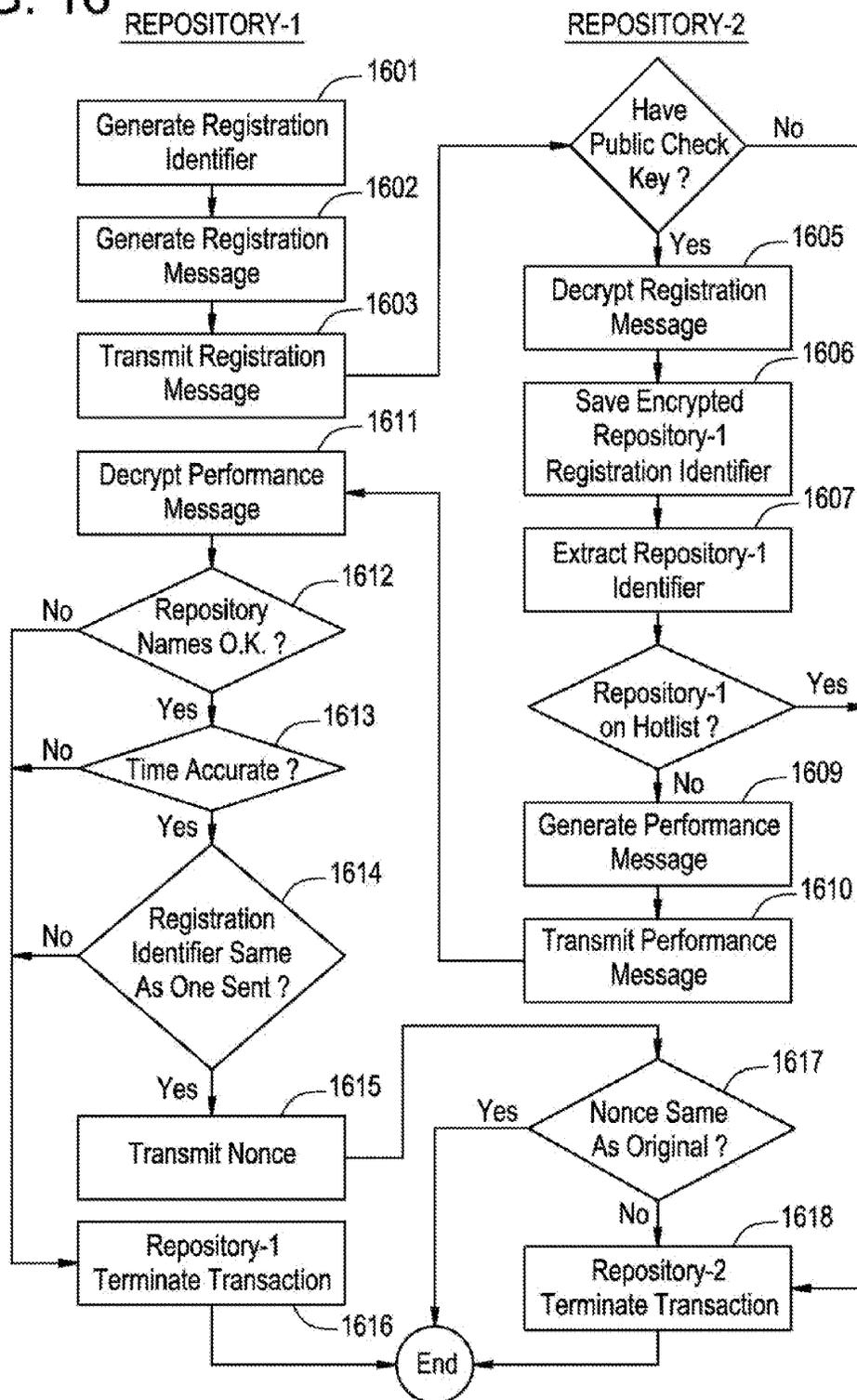


FIG. 17

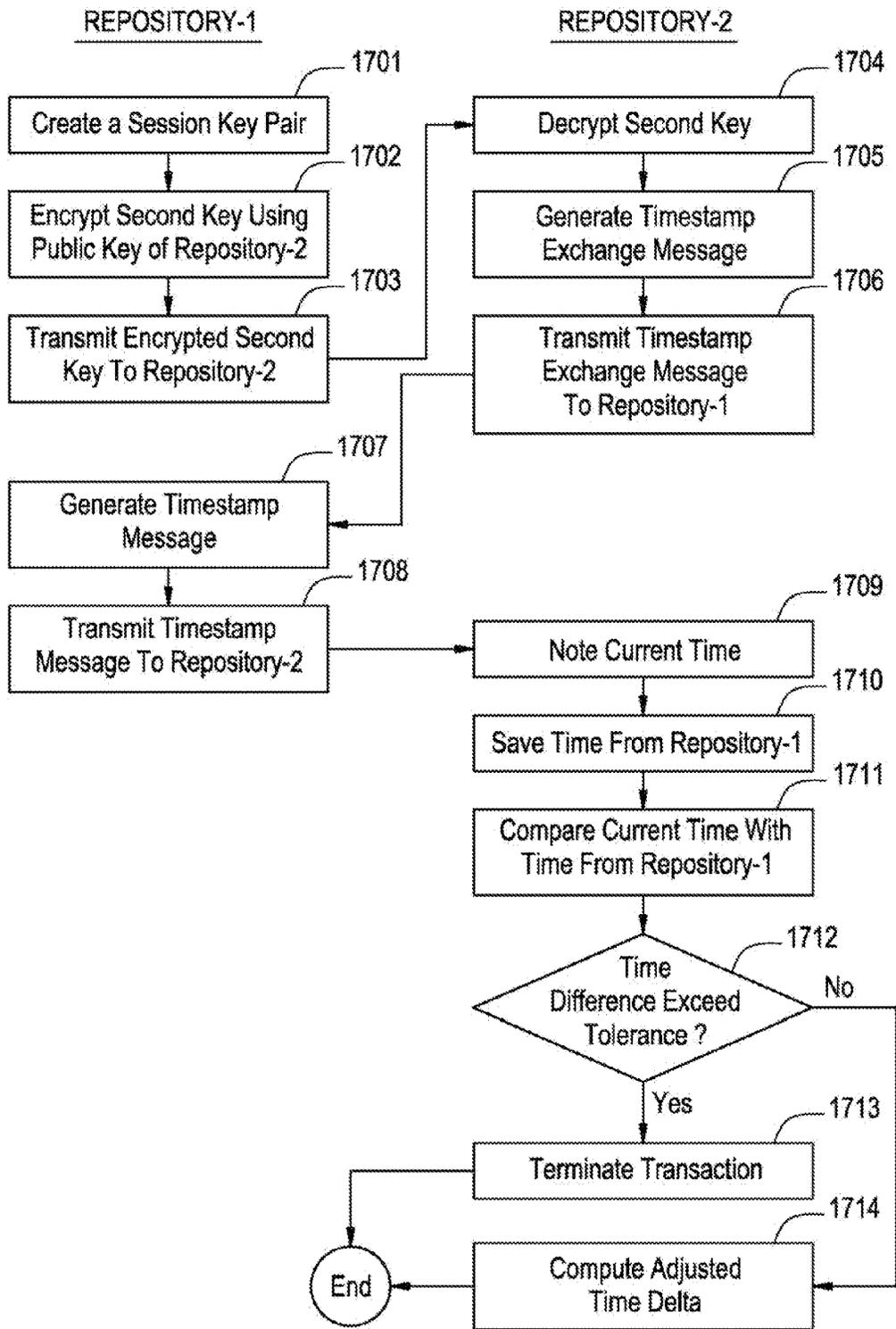


FIG. 18

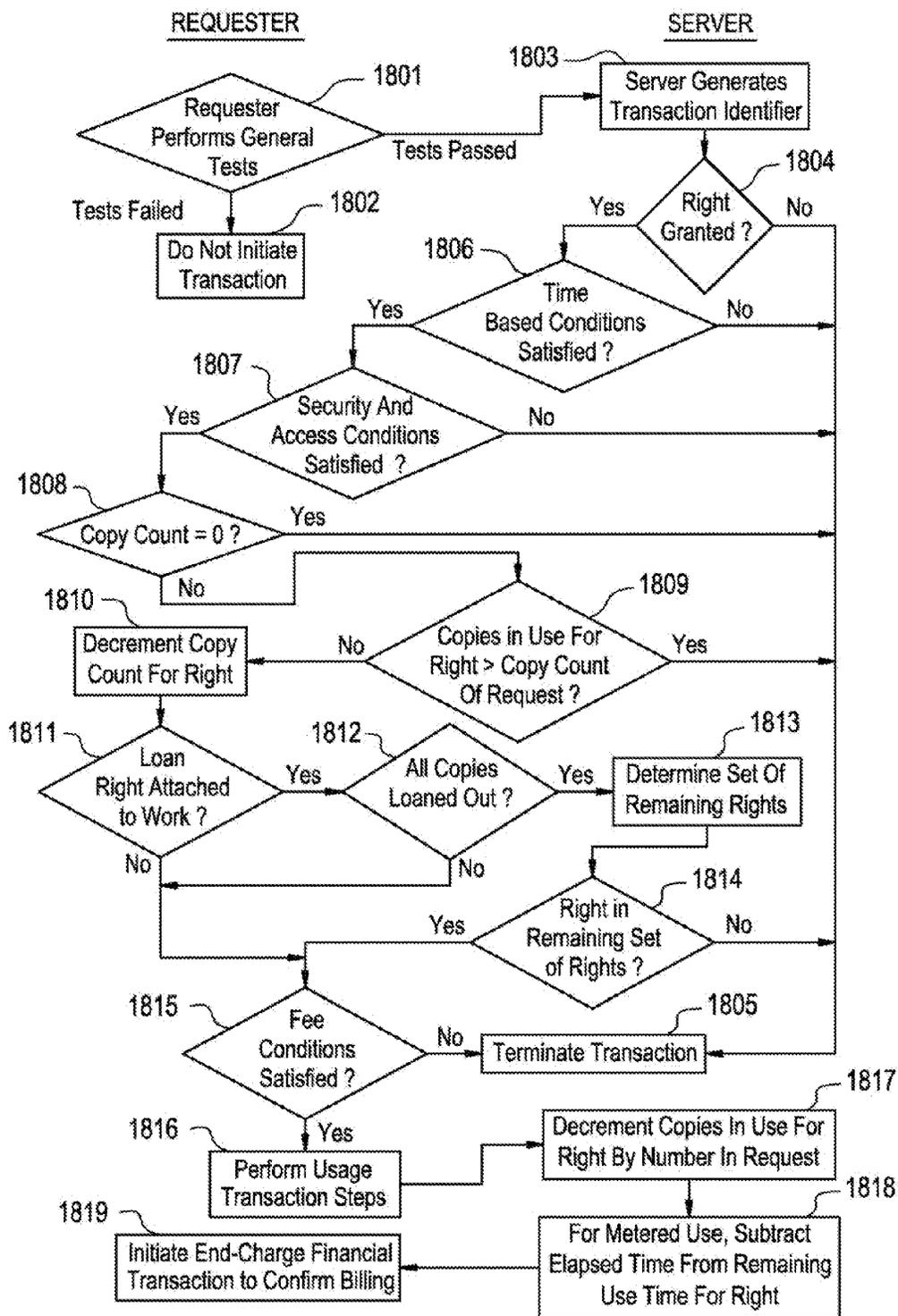
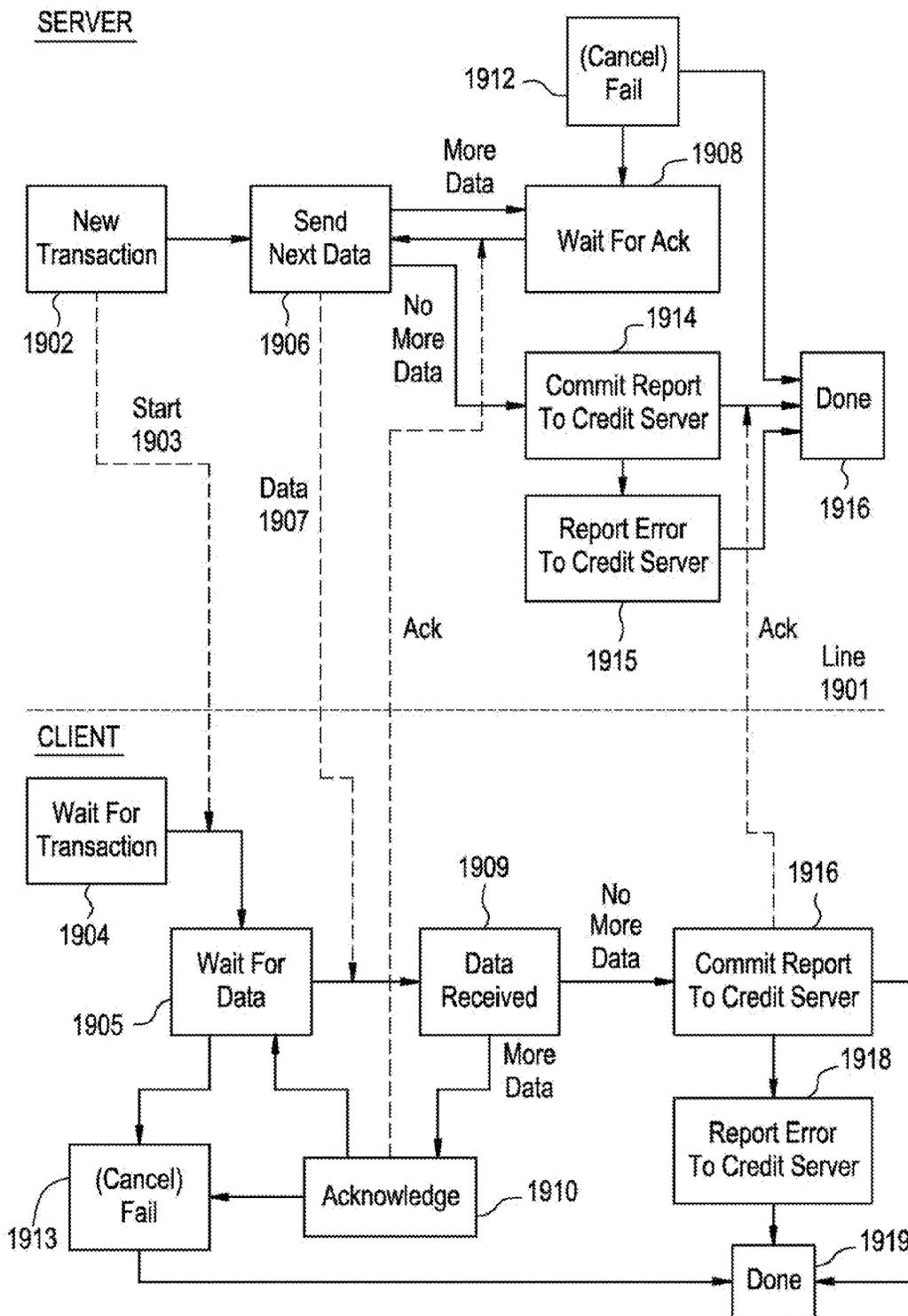


FIG. 19



US 8,393,007 B2

1

**SYSTEM AND METHOD FOR DISTRIBUTING
DIGITAL CONTENT TO BE RENDERED IN
ACCORDANCE WITH USAGE RIGHTS
INFORMATION**

**CROSS REFERENCE TO RELATED
APPLICATIONS**

This application is a continuation of U.S. application Ser. No. 13/584,782, filed Aug. 13, 2012, which is a continuation of U.S. application Ser. No. 11/304,793, filed Dec. 16, 2005, which is a divisional of U.S. application Ser. No. 11/135,352, filed May 24, 2005, now U.S. Pat. No. 7,266,529, which is a continuation of U.S. application Ser. No. 10/322,759, filed Dec. 19, 2002, now U.S. Pat. No. 6,898,576, which is a continuation of U.S. application Ser. No. 09/778,001, filed Feb. 7, 2001, now U.S. Pat. No. 6,708,157, which is a divisional of U.S. application Ser. No. 08/967,084, filed Nov. 10, 1997, now U.S. Pat. No. 6,236,971, which is a continuation of U.S. application Ser. No. 08/344,760, filed Nov. 23, 1994, now abandoned, the entire disclosures of all of which are hereby incorporated by reference herein.

FIELD OF THE INVENTION

The present invention relates to the field of distribution and usage rights enforcement for digitally encoded works.

BACKGROUND OF THE INVENTION

A fundamental issue facing the publishing and information industries as they consider electronic publishing is how to prevent the unauthorized and unaccounted distribution or usage of electronically published materials. Electronically published materials are typically distributed in a digital form and recreated on a computer based system having the capability to recreate the materials. Audio and video recordings, software, books and multimedia works are all being electronically published. Companies in these industries receive royalties for each accounted for delivery of the materials, e.g. the sale of an audio CD at a retail outlet. Any unaccounted distribution of a work results in an unpaid royalty (e.g. copying the audio recording CD to another digital medium.)

The ease in which electronically published works can be "perfectly" reproduced and distributed is a major concern. The transmission of digital works over networks is commonplace. One such widely used network is the Internet. The Internet is a widespread network facility by which computer users in many universities, corporations and government entities communicate and trade ideas and information. Computer bulletin boards found on the Internet and commercial networks such as CompuServ and Prodigy allow for the posting and retrieving of digital information. Information services such as Dialog and LEXIS/NEXIS provide databases of current information on a wide variety of topics. Another factor which will exacerbate the situation is the development and expansion of the National Information Infrastructure (the NII). It is anticipated that, as the NII grows, the transmission of digital works over networks will increase many times over. It would be desirable to utilize the NII for distribution of digital works without the fear of widespread unauthorized copying.

The most straightforward way to curb unaccounted distribution is to prevent unauthorized copying and transmission. For existing materials that are distributed in digital form, various safeguards are used. In the case of software, copy protection schemes which limit the number of copies that can

2

be made or which corrupt the output when copying is detected have been employed. Another scheme causes software to become disabled after a predetermined period of time has lapsed. A technique used for workstation based software is to require that a special hardware device must be present on the workstation in order for the software to run, e.g., see U.S. Pat. No. 4,932,054 entitled "Method and Apparatus for Protecting Computer Software Utilizing Coded Filter Network in Conjunction with an Active Coded Hardware Device." Such devices are provided with the software and are commonly referred to as dongles.

Yet another scheme is to distribute software, but which requires a "key" to enable its use. This is employed in distribution schemes where "demos" of the software are provided on a medium along with the entire product. The demos can be freely used, but in order to use the actual product, the key must be purchased. These schemes do not hinder copying of the software once the key is initially purchased.

A system for ensuring that licenses are in place for using licensed products is described in PCT Publication WO 93/01550 to Griswold entitled "License Management System and Method." The licensed product may be any electronically published work but is most effective for use with works that are used for extended periods of time such as software programs. Griswold requires that the licensed product contain software to invoke a license check monitor at predetermined time intervals. The license check monitor generates request datagrams which identify the licensee. The request datagrams are sent to a license control system over an appropriate communication facility. The license control system then checks the datagram to determine if the datagram is from a valid licensee. The license control system then sends a reply datagram to the license check monitor indicating denial or approval of usage. The license control system will deny usage in the event that request datagrams go unanswered after a predetermined period of time (which may indicate an unauthorized attempt to use the licensed product). In this system, usage is managed at a central location by the response datagrams. So for example if license fees have not been paid, access to the licensed product is terminated.

It is argued by Griswold that the described system is advantageous because it can be implemented entirely in software. However, the system described by Griswold has limitations. An important limitation is that during the use of the licensed product, the user must always be coupled to an appropriate communication facility in order to send and receive datagrams. This creates a dependency on the communication facility. So if the communication facility is not available, the licensed product cannot be used. Moreover, some party must absorb the cost of communicating with the license server.

A system for controlling the distribution of digitally encoded books is embodied in a system available from VPR Systems, LTD. of St. Louis, Miss. The VPR system is self-contained and is comprised of: (1) point of sale kiosks for storing and downloading of books, (2) personal storage mediums (cartridges) to which the books are downloaded, and (3) readers for viewing the book. In a purchase transaction, a purchaser will purchase a voucher card representing the desired book. The voucher will contain sufficient information to identify the book purchased and perhaps some demographic information relating to the sales transaction. To download the book, the voucher and the cartridge are inserted into the kiosk.

The VPR system may also be used as a library. In such an embodiment, the kiosk manages the number of "copies" that may be checked out at one time. Further, the copy of the book is erased from the user's cartridge after a certain check-out

US 8,393,007 B2

3

time has expired. However, individuals cannot loan books because the cartridges may only be used with the owner's reader.

The foregoing distribution and protection schemes operate in part by preventing subsequent distribution of the work. While this certainly prevents unauthorized distributions, it does so by sacrificing the potential for subsequent revenue bearing uses. For example, it may be desirable to allow the lending of a purchased work to permit exposure of the work to potential buyers. Another example would be to permit the creation of a derivative work for a fee. Yet another example would be to permit copying the work for a fee (essentially purchasing it). Thus, it would be desirable to provide flexibility in how the owner of a digital work may allow it to be distributed.

While flexibility in distribution is a concern, the owners of a work want to make sure they are paid for such distributions. In U.S. Pat. No. 4,977,594 to Shear, entitled "Database Usage Metering and Protection System and Method," a system for metering and billing for usage of information distributed on a CD-ROM is described. The system requires the addition of a billing module to the computer system. The billing module may operate in a number of different ways. First, it may periodically communicate billing data to a central billing facility, whereupon the user may be billed. Second, billing may occur by disconnecting the billing module and the user sending it to a central billing facility where the data is read and a user bill generated.

U.S. Pat. No. 5,247,575, Sprague et al., entitled "Information Distribution System", describes an information distribution system which provides and charges only for user selected information. A plurality of encrypted information packages (IPs) are provided at the user site, via high and/or low density storage media and/or by broadcast transmission. Some of the IPs may be of no interest to the user. The IPs of interest are selected by the user and are decrypted and stored locally. The IPs may be printed, displayed or even copied to other storage media. The charges for the selected IP's are accumulated within a user apparatus and periodically reported by telephone to a central accounting facility. The central accounting facility also issues keys to decrypt the IPs. The keys are changed periodically. If the central accounting facility has not issued a new key for a particular user station, the station is unable to retrieve information from the system when the key is changed.

A system available from Wave Systems Corp. of Princeton, N.Y., provides for metering of software usage on a personal computer. The system is installed onto a computer and collects information on what software is in use, encrypts it and then transmits the information to a transaction center. From the transaction center, a bill is generated and sent to the user. The transaction center also maintains customer accounts so that licensing fees may be forwarded directly to the software providers. Software operating under this system must be modified so that usage can be accounted.

Known techniques for billing do not provide for billing of copies made of the work. For example, if data is copied from the CD-ROM described in Shear, any subsequent use of the copy of the information cannot be metered or billed. In other words, the means for billing runs with the media rather than the underlying work. It would be desirable to have a distribution system where the means for billing is always transported with the work.

SUMMARY OF THE INVENTION

A method, system and software for associating usage rights with digital content is provided, including creating usage

4

rights from a grammar, the usage rights specifying a manner of use indicating purposes for which the digital content is used and/or distributed by an authorized party; associating the usage rights with a digital content; processing a usage transaction specifying the usage rights to determine if access to the digital content is granted; and storing the usage rights in a distributed repository. The usage rights also specify one or more conditions which must be satisfied before the manner of use is exercised. The creating includes selecting symbols from a first set of predetermined symbols to define a valid sequence of symbols to indicate the manner of use, selecting one or more symbols from a second set of predetermined symbols to define a valid sequence of symbols to indicate the conditions.

In further embodiments, a system for controlling the distribution and use of digital works using digital tickets is disclosed. A ticket is an indicator that the ticket holder has already paid for or is otherwise entitled to some specified right, product or service. In the present invention, a "digital ticket" is used to enable the ticket holder to exercise usage rights specifying the requirement of the digital ticket. Usage rights are used to define how a digital work may be used or distributed. Specific instances of usage rights are used to indicate a particular manner of use or distribution. A usage right may specify a digital ticket which must be present before the right may be exercised. For example, a digital ticket may be specified in a Copy right of a digital work, so that exercise of the Copy right requires the party that desires a copy of the digital work be in possession of the necessary digital ticket. After a copy of the digital work is successfully sent to the requesting party, the digital ticket is "punched" to indicate that a copy of the digital work has been made. When the ticket is "punched" a predetermined number of times, it may no longer be used.

Digital works are stored in repositories. Repositories enforce the usage rights for digital works. Each repository has a "generic ticket agent" which punches tickets. In some instances only the generic ticket agent is necessary. In other instances, punching by a "special ticket agent" residing on another repository may be desired. Punching by a "special ticket agent" enables greater security and control of the digital work. For example, it can help prevent digital ticket forgery. Special ticket agents are also useful in situations where an external database needs to be updated or checked.

A digital ticket is merely an instance of a digital work. Thus, a digital ticket may be distributed among repositories in the same fashion as other digital works.

A digital ticket may be used in many commercial scenarios such as in the purchase of software and prepaid upgrades. A digital ticket may also be used to limit the number of times that a right may be exercised. For example, a user may purchase a copy of a digital work, along with the right to make up to 5 Copies. In this case, the Copy right would have associated therewith a digital ticket that can be punched up to 5 times. Other such commercial scenarios will become apparent from the detailed description.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a flowchart illustrating a simple instantiation of the operation of the currently preferred embodiment of the present invention.

FIG. 2 is a block diagram illustrating the various repository types and the repository transaction flow between them in the currently preferred embodiment of the present invention.

US 8,393,007 B2

5

FIG. 3 is a block diagram of a repository coupled with a credit server in the currently preferred embodiment of the present invention.

FIGS. 4a and 4b are examples of rendering systems as may be utilized in the currently preferred embodiment of the present invention.

FIG. 5 illustrates a contents file layout for a digital work as may be utilized in the currently preferred embodiment of the present invention.

FIG. 6 illustrates a contents file layout for an individual digital work of the digital work of FIG. 5 as may be utilized in the currently preferred embodiment of the present invention.

FIG. 7 illustrates the components of a description block of the currently preferred embodiment of the present invention.

FIG. 8 illustrates a description tree for the contents file layout of the digital work illustrated in FIG. 5.

FIG. 9 illustrates a portion of a description tree corresponding to the individual digital work illustrated in FIG. 6.

FIG. 10 illustrates a layout for the rights portion of a description block as may be utilized in the currently preferred embodiment of the present invention.

FIG. 11 is a description tree wherein certain d-blocks have PRINT usage rights and is used to illustrate "strict" and "lenient" rules for resolving usage rights conflicts.

FIG. 12 is a block diagram of the hardware components of a repository as are utilized in the currently preferred embodiment of the present invention.

FIG. 13 is a block diagram of the functional (logical) components of a repository as are utilized in the currently preferred embodiment of the present invention.

FIG. 14 is diagram illustrating the basic components of a usage right in the currently preferred embodiment of the present invention.

FIG. 15 lists the usage rights grammar of the currently preferred embodiment of the present invention.

FIG. 16 is a flowchart illustrating the steps of certificate delivery, hotlist checking and performance testing as performed in a registration transaction as may be performed in the currently preferred embodiment of the present invention.

FIG. 17 is a flowchart illustrating the steps of session information exchange and clock synchronization as may be performed in the currently preferred embodiment of the present invention, after each repository in the registration transaction has successfully completed the steps described in FIG. 16.

FIG. 18 is a flowchart illustrating the basic flow for a usage transaction, including the common opening and closing step, as may be performed in the currently preferred embodiment of the present invention.

FIG. 19 is a state diagram of server and client repositories in accordance with a transport protocol followed when moving a digital work from the server to the client repositories, as may be performed in the currently preferred embodiment of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

Overview

A system for controlling use and distribution of digital works is disclosed. The present invention is directed to supporting commercial transactions involving digital works. The transition to digital works profoundly and fundamentally changes how creativity and commerce can work. It changes the cost of transporting or storing works because digital property is almost "massless." Digital property can be transported at electronic speeds and requires almost no warehousing.

6

Keeping an unlimited supply of virtual copies on hand requires essentially no more space than keeping one copy on hand. The digital medium also lowers the costs of alteration, reuse and billing.

There is a market for digital works because creators are strongly motivated to reuse portions of digital works from others rather than creating their own completely. This is because it is usually so much easier to use an existing stock photo or music clip than to create a new one from scratch.

Herein the terms "digital work", "work" and "content" refer to any work that has been reduced to a digital representation. This would include any audio, video, text, or multimedia work and any accompanying interpreter (e.g. software) that may be required for recreating the work. The term composite work refers to a digital work comprised of a collection of other digital works. The term "usage rights" or "rights" is a term which refers to rights granted to a recipient of a digital work. Generally, these rights define how a digital work can be used and if it can be further distributed. Each usage right may have one or more specified conditions which must be satisfied before the right may be exercised. Appendix 1 provides a Glossary of the terms used herein.

A key feature of the present invention is that usage rights are permanently "attached" to the digital work. Copies made of a digital work will also have usage rights attached. Thus, the usage rights and any associated fees assigned by a creator and subsequent distributor will always remain with a digital work.

The enforcement elements of the present invention are embodied in repositories. Among other things, repositories are used to store digital works, control access to digital works, bill for access to digital works and maintain the security and integrity of the system.

The combination of attached usage rights and repositories enable distinct advantages over prior systems. As noted in the prior art, payment of fees are primarily for the initial access. In such approaches, once a work has been read, computational control over that copy is gone. Metaphorically, "the content genie is out of the bottle and no more fees can be billed." In contrast, the present invention never separates the fee descriptions from the work. Thus, the digital work genie only moves from one trusted bottle (repository) to another, and all uses of copies are potentially controlled and billable.

FIG. 1 is a high level flowchart omitting various details but which demonstrates the basic operation of the present invention. Referring to FIG. 1, a creator creates a digital work, step 101. The creator will then determine appropriate usage rights and fees, attach them to the digital work, and store them in Repository 1, step 102. The determination of appropriate usage rights and fees will depend on various economic factors. The digital work remains securely in Repository 1 until a request for access is received. The request for access begins with a session initiation by another repository. Here a Repository 2 initiates a session with Repository 1, step 103. As will be described in greater detail below, this session initiation includes steps which help to insure that the respective repositories are trustworthy. Assuming that a session can be established, Repository 2 may then request access to the Digital Work for a stated purpose, step 104. The purpose may be, for example, to print the digital work or to obtain a copy of the digital work. The purpose will correspond to a specific usage right. In any event, Repository 1 checks the usage rights associated with the digital work to determine if the access to the digital work may be granted, step 105. The check of the usage rights essentially involves a determination of whether a right associated with the access request has been attached to

US 8,393,007 B2

7

the digital work and if all conditions associated with the right are satisfied. If the access is denied, repository 1 terminates the session with an error message, step 106. If access is granted, repository 1 transmits the digital work to repository 2, step 107. Once the digital work has been transmitted to repository 2, repository 1 and 2 each generate billing information for the access which is transmitted to a credit server, step 108. Such double billing reporting is done to insure against attempts to circumvent the billing process.

FIG. 2 illustrates the basic interactions between repository types in the present invention. As will become apparent from FIG. 2, the various repository types will serve different functions. It is fundamental that repositories will share a core set of functionality which will enable secure and trusted communications. Referring to FIG. 2, a repository 201 represents the general instance of a repository. The repository 201 has two modes of operation; a server mode and a requester mode. When in the server mode, the repository will be receiving and processing access requests to digital works. When in the requester mode, the repository will be initiating requests to access digital works. Repository 201 is general in the sense that its primary purpose is as an exchange medium for digital works. During the course of operation, the repository 201 may communicate with a plurality of other repositories, namely authorization repository 202, rendering repository 203 and master repository 204. Communication between repositories occurs utilizing a repository transaction protocol 205.

Communication with an authorization repository 202 may occur when a digital work being accessed has a condition requiring an authorization. Conceptually, an authorization is a digital certificate such that possession of the certificate is required to gain access to the digital work. An authorization is itself a digital work that can be moved between repositories and subjected to fees and usage rights conditions. An authorization may be required by both repositories involved in an access to a digital work.

Communication with a rendering repository 203 occurs in connection with the rendering of a digital work. As will be described in greater detail below, a rendering repository is coupled with a rendering device (e.g. a printer device) to comprise a rendering system.

Communication with a master repository 205 occurs in connection with obtaining an identification certificate. Identification certificates are the means by which a repository is identified as "trustworthy". The use of identification certificates is described below with respect to the registration transaction.

FIG. 3 illustrates the repository 201 coupled to a credit server 301. The credit server 301 is a device which accumulates billing information for the repository 201. The credit server 301 communicates with repository 201 via billing transactions 302 to record billing transactions. Billing transactions are reported to a billing clearinghouse 303 by the credit server 301 on a periodic basis. The credit server 301 communicates to the billing clearinghouse 303 via clearinghouse transactions 304. The clearinghouse transactions 304 enable a secure and encrypted transmission of information to the billing clearinghouse 303.

Rendering Systems

A rendering system is generally defined as a system comprising a repository and a rendering device which can render a digital work into its desired form. Examples of a rendering system may be a computer system, a digital audio system, or a printer. A rendering system has the same security features as

8

a repository. The coupling of a rendering repository with the rendering device may occur in a manner suitable for the type of rendering device.

FIG. 4a illustrates a printer as an example of a rendering system. Referring to FIG. 4, printer system 401 has contained therein a printer repository 402 and a print device 403. It should be noted that the dashed line defining printer system 401 defines a secure system boundary. Communications within the boundary is assumed to be secure. Depending on the security level, the boundary also represents a barrier intended to provide physical integrity. The printer repository 402 is an instantiation of the rendering repository 205 of FIG. 2. The printer repository 402 will in some instances contain an ephemeral copy of a digital work which remains until it is printed out by the print engine 403. In other instances, the printer repository 402 may contain digital works such as fonts, which will remain and can be billed based on use. This design assures that all communication lines between printers and printing devices are encrypted, unless they are within a physically secure boundary. This design feature eliminates a potential "fault" point through which the digital work could be improperly obtained. The printer device 403 represents the printer components used to create the printed output.

Also illustrated in FIG. 4a is the repository 404. The repository 404 is coupled to the printer repository 402. The repository 404 represents an external repository which contains digital works.

FIG. 4b is an example of a computer system as a rendering system. A computer system may constitute a "multi-function" device since it may execute digital works (e.g. software programs) and display digital works (e.g. a digitized photograph). Logically, each rendering device can be viewed as having its own repository, although only one physical repository is needed. Referring to FIG. 4b, a computer system 410 has contained therein a display/execution repository 411. The display/execution repository 411 is coupled to display device, 412 and execution device 413. The dashed box surrounding the computer system 410 represents a security boundary within which communications are assumed to be secure. The display/execution repository 411 is further coupled to a credit server 414 to report any fees to be billed for access to a digital work and a repository 415 for accessing digital works stored therein.

Structure of Digital Works

Usage rights are attached directly to digital works. Thus, it is important to understand the structure of a digital work. The structure of a digital work, in particular composite digital works, may be naturally organized into an acyclic structure such as a hierarchy. For example, a magazine has various articles and photographs which may have been created and are owned by different persons. Each of the articles and photographs may represent a node in a hierarchical structure. Consequently, controls, i.e. usage rights, may be placed on each node by the creator. By enabling control and fee billing to be associated with each node, a creator of a work can be assured that the rights and fees are not circumvented.

In the currently preferred embodiment, the file information for a digital work is divided into two files: a "contents" file and a "description tree" file. From the perspective of a repository, the "contents" file is a stream of addressable bytes whose format depends completely on the interpreter used to play, display or print the digital work. The description tree file makes it possible to examine the rights and fees for a work without reference to the content of the digital work. It should be noted that the term description tree as used herein refers to

US 8,393,007 B2

9

any type of acyclic structure used to represent the relationship between the various components of a digital work.

FIG. 5 illustrates the layout of a contents file. Referring to FIG. 5, a digital work 509 is comprised of story A 510, advertisement 511, story B 512 and story C 513. It is assumed that the digital work is stored starting at a relative address of 0. Each of the parts of the digital work are stored linearly so that story A 510 is stored at approximately addresses 0-30,000, advertisement 511 at addresses 30,001-40,000, story B 512 at addresses 40,001-60,000 and story C 513 at addresses 60,001-85K. The detail of story A 510 is illustrated in FIG. 6. Referring to FIG. 6, the story A 510 is further broken down to show text 614 stored at address 0-1500, soldier photo 615 at addresses 1501-10,000, graphics 616 stored at addresses 10,001-25,000 and sidebar 617 stored address 25,001-30,000. Note that the data in the contents file may be compressed (for saving storage) or encrypted (for security).

From FIGS. 5 and 6 it is readily observed that a digital work can be represented by its component parts as a hierarchy. The description tree for a digital work is comprised of a set of related descriptor blocks (d-blocks). The contents of each d-block are described with respect to FIG. 7. Referring to FIG. 7, a d-block 700 includes an identifier 701 which is a unique identifier for the work in the repository, a starting address 702 providing the start address of the first byte of the work, a length 703 giving the number of bytes in the work, a rights portion 704 wherein the granted usage rights and their status data are maintained, a parent pointer 705 for pointing to a parent d-block and child pointers 706 for pointing to the child d-blocks. In the currently preferred embodiment, the identifier 701 has two parts. The first part is a unique number assigned to the repository upon manufacture. The second part is a unique number assigned to the work upon creation. The rights portion 704 will contain a data structure, such as a look-up table, wherein the various information associated with a right is maintained. The information required by the respective usage rights is described in more detail below. D-blocks form a strict hierarchy. The top d-block of a work has no parent; all other d-blocks have one parent. The relationship of usage rights between parent and child d-blocks and how conflicts are resolved is described below.

A special type of d-block is a "shell" d-block. A shell d-block adds no new content beyond the content of its parts. A shell d-block is used to add rights and fee information, typically by distributors of digital works.

FIG. 8 illustrates a description tree for the digital work of FIG. 5. Referring to FIG. 8, a top d-block 820 for the digital work points to the various stories and advertisements contained therein. Here, the top d-block 820 points to d-block 821 (representing story A 510), d-block 822 (representing the advertisement 511), d-block 823 (representing story B 512) and d-block 824 (representing story C 513).

The portion of the description tree for Story A 510 is illustrated in FIG. 9. D-block 925 represents text 614, d-block 926 represents photo 615, d-block 927 represents graphics 616 by and d-block 928 represents sidebar 617.

The rights portion 704 of a descriptor block is further illustrated in FIG. 10. FIG. 10 illustrates a structure which is repeated in the rights portion 704 for each right. Referring to FIG. 10, each right will have a right code field 1001 and status information field 1002. The right code field 1001 will contain a unique code assigned to a right. The status information field 1002 will contain information relating to the state of a right and the digital work. Such information is indicated below in Table 1. The rights as stored in the rights portion 304 may typically be in numerical order based on the right code.

10

The approach for representing digital works by separating description data from content assumes that parts of a file are contiguous but takes no position on the actual representation of content. In particular, it is neutral to the question of whether content representation may take an object oriented approach. It would be natural to represent content as objects. In principle, it may be convenient to have content objects that include the billing structure and rights information that is represented in the d-blocks. Such variations in the design of the representation are possible and are viable alternatives but may introduce processing overhead, e.g. the interpretation of the objects.

TABLE 1

DIGITAL WORK STATE INFORMATION		
Property	Value	Use
Copies-in-Use	Number	A counter of the number of copies of a work that are in use. Incremented when another copy is used; decremented when use is completed.
Loan-Period	Time-Units	Indicator of the maximum number of time-units that a document can be loaned out
Loaner-Copy	Boolean	Indicator that the current work is a loaned out copy of an authorized digital work.
Remaining-Time	Time-Units	Indicator of the remaining time of use on a metered document right.
Document-Descr	String	A string containing various identifying information about a document. The exact format of this is not specified, but it can include information such as a publisher name, author name, ISBN number, and so on.
Revenue-Owner	RO-Descr	A handle identifying a revenue owner for a digital work. This is used for reporting usage fees.
Publication-Date	Date-Descr	The date that the digital work was published.
History-list	History-Rec	A list of events recording the repositories and dates for operations that copy, transfer, backup, or restore a digital work.

Digital works are stored in a repository as part of a hierarchical file system. Folders (also termed directories and sub-directories) contain the digital works as well as other folders. Digital works and folders in a folder are ordered in alphabetical order. The digital works are typed to reflect how the files are used. Usage rights can be attached to folders so that the folder itself is treated as a digital work. Access to the folder would then be handled in the same fashion as any other digital work. As will be described in more detail below, the contents of the folder are subject to their own rights. Moreover, file management rights may be attached to the folder which defines how folder contents can be managed.

Attaching Usage Rights to a Digital Work

It is fundamental to the present invention that the usage rights are treated as part of the digital work. As the digital work is distributed, the scope of the granted usage rights will remain the same or may be narrowed. For example, when a digital work is transferred from a document server to a repository, the usage rights may include the right to loan a copy for a predetermined period of time (called the original rights). When the repository loans out a copy of the digital work, the usage rights in the loaner copy (called the next set of rights) could be set to prohibit any further rights to loan out the copy. The basic idea is that one cannot grant more rights than they have.

US 8,393,007 B2

11

The attachment of usage rights into a digital work may occur in a variety of ways. If the usage rights will be the same for an entire digital work, they could be attached when the digital work is processed for deposit in the digital work server. In the case of a digital work having different usage rights for the various components, this can be done as the digital work is being created. An authoring tool or digital work assembling tool could be utilized which provides for an automated process of attaching the usage rights.

As will be described below, when a digital work is copied, transferred or loaned, a “next set of rights” can be specified. The “next set of rights” will be attached to the digital work as it is transported.

Resolving Conflicting Rights

Because each part of a digital work may have its own usage rights, there will be instances where the rights of a “contained part” are different from its parent or container part. As a result, conflict rules must be established to dictate when and how a right may be exercised. The hierarchical structure of a digital work facilitates the enforcement of such rules. A “strict” rule would be as follows: a right for a part in a digital work is sanctioned if and only if it is sanctioned for the part, for ancestor d-blocks containing the part and for all descendent d-blocks. By sanctioned, it is meant that (1) each of the respective parts must have the right, and (2) any conditions for exercising the right are satisfied.

It also possible to implement the present invention using a more lenient rule. In the more lenient rule, access to the part may be enabled to the descendent parts which have the right, but access is denied to the descendents which do not.

Example of applying both the strict rule and lenient is illustrated with reference to FIG. 11. Referring to FIG. 11, a root d-block 1101 has child d-blocks 1102-1105. In this case, root d-block represents a magazine, and each of the child d-blocks 1102-1105 represent articles in the magazine. Suppose that a request is made to PRINT the digital work represented by root d-block 1101 wherein the strict rule is followed. The rights for the root d-block 1101 and child d-blocks 1102-1105 are then examined. Root d-block 1101 and child d-blocks 1102 and 1105 have been granted PRINT rights. Child d-block 1103 has not been granted PRINT rights and child d-block 1104 has PRINT rights conditioned on payment of a usage fee.

Under the strict rule the PRINT right cannot be exercised because the child d-block does not have the PRINT right. Under the lenient rule, the result would be different. The digital works represented by child d-blocks 1102 and 1105 could be printed and the digital work represented by d-block 1104 could be printed so long as the usage fee is paid. Only the digital work represented by d-block 1103 could not be printed. This same result would be accomplished under the strict rule if the requests were directed to each of the individual digital works.

The present invention supports various combinations of allowing and disallowing access. Moreover, as will be described below, the usage rights grammar permits the owner of a digital work to specify if constraints may be imposed on the work by a container part. The manner in which digital works may be sanctioned because of usage rights conflicts would be implementation specific and would depend on the nature of the digital works.

Repositories

Many of the powerful functions of repositories—such as their ability to “loan” digital works or automatically handle the commercial reuse of digital works—are possible because

12

they are trusted systems. The systems are trusted because they are able to take responsibility for fairly and reliably carrying out the commercial transactions. That the systems can be responsible (“able to respond”) is fundamentally an issue of integrity. The integrity of repositories has three parts: physical integrity, communications integrity, and behavioral integrity.

Physical integrity refers to the integrity of the physical devices themselves. Physical integrity applies both to the repositories and to the protected digital works. Thus, the higher security classes of repositories themselves may have sensors that detect when tampering is attempted on their secure cases. In addition to protection of the repository itself, the repository design protects access to the content of digital works. In contrast with the design of conventional magnetic and optical devices—such as floppy disks, CD-ROMs, and videotapes—repositories never allow non-trusted systems to access the works directly. A maker of generic computer systems cannot guarantee that their platform will not be used to make unauthorized copies. The manufacturer provides generic capabilities for reading and writing information, and the general nature of the functionality of the general computing device depends on it. Thus, a copy program can copy arbitrary data. This copying issue is not limited to general purpose computers. It also arises for the unauthorized duplication of entertainment “software” such as video and audio recordings by magnetic recorders. Again, the functionality of the recorders depends on their ability to copy and they have no means to check whether a copy is authorized. In contrast, repositories prevent access to the raw data by general devices and can test explicit rights and conditions before copying or otherwise granting access. Information is only accessed by protocol between trusted repositories.

Communications integrity refers to the integrity of the communications channels between repositories. Roughly speaking, communications integrity means that repositories cannot be easily fooled by “telling them lies.” Integrity in this case refers to the property that repositories will only communicate with other devices that are able to present proof that they are certified repositories, and furthermore, that the repositories monitor the communications to detect “impostors” and malicious or accidental interference. Thus the security measures involving encryption, exchange of digital certificates, and nonces described below are all security measures aimed at reliable communication in a world known to contain active adversaries.

Behavioral integrity refers to the integrity in what repositories do. What repositories do is determined by the software that they execute. The integrity of the software is generally assured only by knowledge of its source. Restated, a user will trust software purchased at a reputable computer store but not trust software obtained off a random (insecure) server on a network. Behavioral integrity is maintained by requiring that repository software be certified and be distributed with proof of such certification, i.e. a digital certificate. The purpose of the certificate is to authenticate that the software has been tested by an authorized organization, which attests that the software does what it is supposed to do and that it does not compromise the behavioral integrity of a repository. If the digital certificate cannot be found in the digital work or the master repository which generated the certificate is not known to the repository receiving the software, then the software cannot be installed.

In the description of FIG. 2, it was indicated that repositories come in various forms. All repositories provide a core set of services for the transmission of digital works. The manner in which digital works are exchanged is the basis for all

US 8,393,007 B2

13

transaction between repositories. The various repository types differ in the ultimate functions that they perform. Repositories may be devices themselves, or they may be incorporated into other systems. An example is the rendering repository 205 of FIG. 2.

A repository will have associated with it a repository identifier. Typically, the repository identifier would be a unique number assigned to the repository at the time of manufacture. Each repository will also be classified as being in a particular security class. Certain communications and transactions may be conditioned on a repository being in a particular security class. The various security classes are described in greater detail below.

As a prerequisite to operation, a repository will require possession of an identification certificate. Identification certificates are encrypted to prevent forgery and are issued by a Master repository. A master repository plays the role of an authorization agent to enable repositories to receive digital works. Identification certificates must be updated on a periodic basis. Identification certificates are described in greater detail below with respect to the registration transaction.

A repository has both a hardware and functional embodiment. The functional embodiment is typically software executing on the hardware embodiment. Alternatively, the functional embodiment may be embedded in the hardware embodiment such as an Application Specific Integrated Circuit (ASIC) chip.

The hardware embodiment of a repository will be enclosed in a secure housing which if compromised, may cause the repository to be disabled. The basic components of the hardware embodiment of a repository are described with reference to FIG. 12. Referring to FIG. 12, a repository is comprised of a processing means 1200, storage system 1207, clock 1205 and external interface 1206. The processing means 1200 is comprised of a processor element 1201 and processor memory 1202. The processing means 1201 provides controller, repository transaction and usage rights transaction functions for the repository. Various functions in the operation of the repository such as decryption and/or decompression of digital works and transaction messages are also performed by the processing means 1200. The processor element 1201 may be a microprocessor or other suitable computing component. The processor memory 1202 would typically be further comprised of Read Only Memories (ROM) and Random Access Memories (RAM). Such memories would contain the software instructions utilized by the processor element 1201 in performing the functions of the repository.

The storage system 1207 is further comprised of descriptor storage 1203 and content storage 1204. The description tree storage 1203 will store the description tree for the digital work and the content storage will store the associated content. The description tree storage 1203 and content storage 1204 need not be of the same type of storage medium, nor are they necessarily on the same physical device. So for example, the descriptor storage 1203 may be stored on a solid state storage (for rapid retrieval of the description tree information), while the content storage 1204 may be on a high capacity storage such as an optical disk.

The clock 1205 is used to time-stamp various time based conditions for usage rights or for metering usage fees which may be associated with the digital works. The clock 1205 will have an uninterruptible power supply, e.g. a battery, in order to maintain the integrity of the time-stamps. The external interface means 1206 provides for the signal connection to other repositories and to a credit server. The external interface means 1206 provides for the exchange of signals via such

14

standard interfaces such as RS-232 or Personal Computer Manufacturers Card Industry Association (PCMCIA) standards, or FDDI. The external interface means 1206 may also provide network connectivity.

The functional embodiment of a repository is described with reference to FIG. 13. Referring to FIG. 13, the functional embodiment is comprised of an operating system 1301, core repository services 1302, usage transaction handlers 1303, repository specific functions, 1304 and a user interface 1305. The operating system 1301 is specific to the repository and would typically depend on the type of processor being used. The operating system 1301 would also provide the basic services for controlling and interfacing between the basic components of the repository.

The core repository services 1302 comprise a set of functions required by each and every repository. The core repository services 1302 include the session initiation transactions which are defined in greater detail below. This set of services also includes a generic ticket agent which is used to "punch" a digital ticket and a generic authorization server for processing authorization specifications. Digital tickets and authorizations are specific mechanisms for controlling the distribution and use of digital works and are described in more detail below. Note that coupled to the core repository services are a plurality of identification certificates 1306. The identification certificates 1306 are required to enable the use of the repository.

The usage transactions handler 1303 comprise functionality for processing access requests to digital works and for billing fees based on access. The usage transactions supported will be different for each repository type. For example, it may not be necessary for some repositories to handle access requests for digital works.

The repository specific functionality 1304 comprises functionality that is unique to a repository. For example, the master repository has special functionality for issuing digital certificates and maintaining encryption keys. The repository specific functionality 1304 would include the user interface implementation for the repository.

Repository Security Classes

For some digital works the losses caused by any individual instance of unauthorized copying is insignificant and the chief economic concern lies in assuring the convenience of access and low-overhead billing. In such cases, simple and inexpensive handheld repositories and network-based workstations may be suitable repositories, even though the measures and guarantees of security are modest.

At the other extreme, some digital works such as a digital copy of a first run movie or a bearer bond or stock certificate would be of very high value so that it is prudent to employ caution and fairly elaborate security measures to ensure that they are not copied or forged. A repository suitable for holding such a digital work could have elaborate measures for ensuring physical integrity and for verifying authorization before use.

By arranging a universal protocol, all kinds of repositories can communicate with each other in principle. However, creators of some works will want to specify that their works will only be transferred to repositories whose level of security is high enough. For this reason, document repositories have a ranking system for classes and levels of security. The security classes in the currently preferred embodiment are described in Table 2.

US 8,393,007 B2

15

TABLE 2

REPOSITORY SECURITY LEVELS	
Level	Description of Security
0	Open system. Document transmission is unencrypted. No digital certificate is required for identification. The security of the system depends mostly on user honesty, since only modest knowledge may be needed to circumvent the security measures. The repository has no provisions for preventing unauthorized programs from running and accessing or copying files. The system does not prevent the use of removable storage and does not encrypt stored files.
1	Minimal security. Like the previous class except that stored files are minimally encrypted, including ones on removable storage.
2	Basic security. Like the previous class except that special tools and knowledge are required to compromise the programming, the contents of the repository, or the state of the clock. All digital communications are encrypted. A digital certificate is provided as identification. Medium level encryption is used. Repository identification number is unforgeable.
3	General security. Like the previous class plus the requirement of special tools are needed to compromise the physical integrity of the repository and that modest encryption is used on all transmissions. Password protection is required to use the local user interface. The digital clock system cannot be reset without authorization. No works would be stored on removable storage. When executing works as programs, it runs them in their own address space and does not give them direct access to any file storage or other memory containing system code or works. They can access works only through the transmission transaction protocol.
4	Like the previous class except that high level encryption is used on all communications. Sensors are used to record attempts at physical and electronic tampering. After such tampering, the repository will not perform other transactions until it has reported such tampering to a designated server.
5	Like the previous class except that if the physical or digital attempts at tampering exceed some preset thresholds that threaten the physical integrity of the repository or the integrity of digital and cryptographic barriers, then the repository will save only document description records of history but will erase or destroy any digital identifiers that could be misused if released to an unscrupulous party. It also modifies any certificates of authenticity to indicate that the physical system has been compromised. It also erases the contents of designated documents.
6	Like the previous class except that the repository will attempt wireless communication to report tampering and will employ noisy alarms.
10	This would correspond to a very high level of security. This server would maintain constant communications to remote security systems reporting transactions, sensor readings, and attempts to circumvent security.

The characterization of security levels described in Table 2 is not intended to be fixed. More important is the idea of having different security levels for different repositories. It is anticipated that new security classes and requirements will evolve according to social situations and changes in technology.

Repository User Interface

A user interface is broadly defined as the mechanism by which a user interacts with a repository in order to invoke transactions to gain access to a digital work, or exercise usage rights. As described above, a repository may be embodied in various forms. The user interface for a repository will differ depending on the particular embodiment. The user interface may be a graphical user interface having icons representing the digital works and the various transactions that may be performed. The user interface may be a generated dialog in which a user is prompted for information.

The user interface itself need not be part of the repository. As a repository may be embedded in some other device, the user interface may merely be a part of the device in which the repository is embedded. For example, the repository could be embedded in a "card" that is inserted into an available slot in

16

a computer system. The user interface may be combination of a display, keyboard, cursor control device and software executing on the computer system.

At a minimum, the user interface must permit a user to input information such as access requests and alpha numeric data and provide feedback as to transaction status. The user interface will then cause the repository to initiate the suitable transactions to service the request. Other facets of a particular user interface will depend on the functionality that a repository will provide.

Credit Servers

In the present invention, fees may be associated with the exercise of a right. The requirement for payment of fees is described with each version of a usage right in the usage rights language. The recording and reporting of such fees is performed by the credit server. One of the capabilities enabled by associating fees with rights is the possibility of supporting a wide range of charging models. The simplest model, used by conventional software, is that there is a single fee at the time of purchase, after which the purchaser obtains unlimited rights to use the work as often and for as long as he or she wants. Alternative models, include metered use and variable fees. A single work can have different fees for different uses. For example, viewing a photograph on a display could have different fees than making a hardcopy or including it in a newly created work. A key to these alternative charging models is to have a low overhead means of establishing fees and accounting for credit on these transactions.

A credit server is a computational system that reliably authorizes and records these transactions so that fees are billed and paid. The credit server reports fees to a billing clearinghouse. The billing clearinghouse manages the financial transactions as they occur. As a result, bills may be generated and accounts reconciled. Preferably, the credit server would store the fee transactions and periodically communicate via a network with billing clearinghouse for reconciliation. In such an embodiment, communications with the billing clearinghouse would be encrypted for integrity and security reasons. In another embodiment, the credit server acts as a "debit card" where transactions occur in "real-time" against a user account.

A credit server is comprised of memory, a processing means, a clock, and interface means for coupling to a repository and a financial institution (e.g. a modem). The credit server will also need to have security and authentication functionality. These elements are essentially the same elements as those of a repository. Thus, a single device can be both a repository and a credit server, provided that it has the appropriate processing elements for carrying out the corresponding functions and protocols. Typically, however, a credit server would be a card-sized system in the possession of the owner of the credit. The credit server is coupled to a repository and would interact via financial transactions as described below. Interactions with a financial institution may occur via protocols established by the financial institutions themselves.

In the currently preferred embodiment credit servers associated with both the server and the repository report the financial transaction to the billing clearinghouse. For example, when a digital work is copied by one repository to another for a fee, credit servers coupled to each of the repositories will report the transaction to the billing clearinghouse. This is desirable in that it insures that a transaction will be accounted for in the event of some break in the communication between a credit server and the billing clearinghouse. However, some implementations may embody only a single credit server

US 8,393,007 B2

17

reporting the transaction to minimize transaction processing at the risk of losing some transactions.

Usage Rights Language

The present invention uses statements in a high level "usage rights language" to define rights associated with digital works and their parts. Usage rights statements are interpreted by repositories and are used to determine what transactions can be successfully carried out for a digital work and also to determine parameters for those transactions. For example, sentences in the language determine whether a given digital work can be copied, when and how it can be used, and what fees (if any) are to be charged for that use. Once the usage rights statements are generated, they are encoded in a suitable form for accessing during the processing of transactions.

Defining usage rights in terms of a language in combination with the hierarchical representation of a digital work enables the support of a wide variety of distribution and fee schemes. An example is the ability to attach multiple versions of a right to a work. So a creator may attach a PRINT right to make 5 copies for \$10.00 and a PRINT right to make unlimited copies for \$100.00. A purchaser may then choose which option best fits his needs. Another example is that rights and fees are additive. So in the case of a composite work, the rights and fees of each of the components works is used in determining the rights and fees for the work as a whole. Other features and benefits of the usage rights language will become apparent in the description of distribution and use scenarios provided below.

The basic contents of a right are illustrated in FIG. 14. Referring to FIG. 14, a right 1450 has a transactional component 1451 and a specifications component 1452. A right 1450 has a label (e.g. COPY or PRINT) which indicate the use or distribution privileges that are embodied by the right. The transactional component 1451 corresponds to a particular way in which a digital work may be used or distributed. The transactional component 1451 is typically embodied in software instructions in a repository which implement the use or distribution privileges for the right. The specifications components 1452 are used to specify conditions which must be satisfied prior to the right being exercised or to designate various transaction related parameters. In the currently preferred embodiment, these specifications include copy count 1453, Fees and Incentives 1454, Time 1455, Access and Security 1456 and Control 1457. Each of these specifications will be described in greater detail below with respect to the language grammar elements.

The usage rights language is based on the grammar described below. A grammar is a convenient means for defining valid sequence of symbols for a language. In describing the grammar the notation "[a|b|c]" is used to indicate distinct choices among alternatives. In this example, a sentence can have either an "a", "b" or "c". It must include exactly one of them. The braces { } are used to indicate optional items. Note that brackets, bars and braces are used to describe the language of usage rights sentences but do not appear in actual sentences in the language.

In contrast, parentheses are part of the usage rights language. Parentheses are used to group items together in lists. The notation (x*) is used to indicate a variable length list, that is, a list containing one or more items of type x. The notation (x)* is used to indicate a variable number of lists containing x.

Keywords in the grammar are words followed by colons. Keywords are a common and very special case in the language. They are often used to indicate a single value, typically

18

an identifier. In many cases, the keyword and the parameter are entirely optional. When a keyword is given, it often takes a single identifier as its value. In some cases, the keyword takes a list of identifiers.

5 In the usage rights language, time is specified in an hours:minutes:seconds (or hh:mm:ss) representation. Time zone indicators, e.g. PDT for Pacific Daylight Time, may also be specified. Dates are represented as year/month/day (or YYYY/MM/DD). Note that these time and date representations may specify moments in time or units of time Money units are specified in terms of dollars.

10 Finally, in the usage rights language, various "things" will need to interact with each other. For example, an instance of a usage right may specify a bank account, a digital ticket, etc. Such things need to be identified and are specified herein using the suffix "-ID."

15 The Usage Rights Grammar is listed in its entirety in FIG. 15 and is described below.

Grammar element 1501 "Digital Work Rights:=(Rights*)" define the digital work rights as a set of rights. The-set of rights attached to a digital work define how that digital work may be transferred, used, performed or played. A set of rights will attach to the entire digital work and in the case of compound digital works, each of the components of the digital work. The usage rights of components of a digital may be different.

Grammar element 1502 "Right:=(Right-Code {Copy-Count} {Control-Spec} {Time-Spec} {Access-Spec} {Fee-Spec})" enumerates the content of a right. Each usage right must specify a right code. Each right may also optionally specify conditions which must be satisfied before the right can be exercised. These conditions are copy count, control, time, access and fee conditions. In the currently preferred embodiment, for the optional elements, the following defaults apply: copy count equals 1, no time limit on the use of the right, no access tests or a security level required to use the right and no fee is required. These conditions will each be described in greater detail below.

20 It is important to note that a digital work may have multiple versions of a right, each having the same right code. The multiple versions would provide alternative conditions and fees for accessing the digital work.

A Grammar element 1503 "Right-Code:=Render-Code|Transport-Code|File-Management-Code|Derivative-Works-Code Configuration-Code" distinguishes each of the specific rights into a particular right type (although each right is identified by distinct right codes). In this way, the grammar provides a catalog of possible rights that can be associated with parts of digital works. In the following, rights are divided into categories for convenience in describing them.

Grammar element 1504 "Render-Code:={Player: Player-ID}|Print: {Printer: Printer-ID}" lists a category of rights all involving the making of ephemeral, transitory, or non-digital copies of the digital work. After use the copies are erased.

50 Play: A process of rendering or performing a digital work on some processor. This includes such things as playing digital movies, playing digital music, playing a video game, running a computer program, or displaying a document on a display.

Print: To render the work in a medium that is not further protected by usage rights, such as printing on paper.

Grammar element 1505 "Transport-Code:=[Copy|Transfer|Loan {Remaining-Rights: Next-Set-of-Rights}] {(Next-Copy-Rights: Next-Set of Rights)}" lists a category of rights involving the making of persistent, usable copies of the digital work on other repositories. The optional

US 8,393,007 B2

19

Next-Copy-Rights determine the rights on the work after it is transported. If this is not specified, then the rights on the transported copy are the same as on the original. The optional Remaining-Rights specify the rights that remain with a digital work when it is loaned out. If this is not specified, then the default is that no rights can be exercised when it is loaned out.

Copy: Make a new copy of a work

Transfer: Moving a work from one repository to another.

Loan: Temporarily loaning a copy to another repository for a specified period of time.

Grammar element **1506** “File-Management-Code:=Backup {Back-Up-Copy-Rights: Next-Set-of Rights}|Restore|Delete|Folder|Directory {Name:Hide-Local|Hide-Remote}|{Parts:Hide-Local|Hide-Remote}”

lists a category of rights involving operations for file management, such as the making of backup copies to protect the copy owner against catastrophic equipment failure.

Many software licenses and also copyright law give a copy owner the right to make backup copies to protect against catastrophic failure of equipment. However, the making of uncontrolled backup copies is inherently at odds with the ability to control usage, since an uncontrolled backup copy can be kept and then restored even after the authorized copy was sold.

The File management rights enable the making and restoring of backup copies in a way that respects usage rights, honoring the requirements of both the copy owner and the rights grantor and revenue owner. Backup copies of work descriptions (including usage rights and fee data) can be sent under appropriate protocol and usage rights control to other document repositories of sufficiently high security. Further rights permit organization of digital works into folders which themselves are treated as digital works and whose contents may be “hidden” from a party seeking to determine the contents of a repository.

Backup: To make a backup copy of a digital work as protection against media failure.

Restore: To restore a backup copy of a digital work.

Delete: To delete or erase a copy of a digital work.

Folder: To create and name folders, and to move files and folders between folders.

Directory: To hide a folder or it’s contents.

Grammar element **1507** “Derivative-Works-Code: [Extract|Embed|Edit {Process: Process-ID}] {Next-Copy-Rights: Next-Set-of Rights}” lists a category of rights involving the use of a digital work to create new works.

Extract: To remove a portion of a work, for the purposes of creating a new work.

Embed: To include a work in an existing work.

Edit: To alter a digital work by copying, selecting and modifying portions of an existing digital work.

Grammar element **1508** “Configuration-Code:=Install|Uninstall” lists a category of rights for installing and uninstalling software on a repository (typically a rendering repository.) This would typically occur for the installation of a new type of player within the rendering repository.

Install: To install new software on a repository.

Uninstall: To remove existing software from a repository.

Grammar element **1509** “Next-Set-of-Rights:={ (Add: Set-Of-Rights) } { (Delete: Set-Of-Rights) } { (Replace: Set-Of-Rights) } { (Keep: Set-Of-Rights) }” defines how rights are carried forward for a copy of a digital work. If the Next-Copy-Rights is not specified, the rights for the next copy are the same as those of the current copy. Otherwise, the set of rights for the next copy can be specified. Versions of rights after Add: are added to the current set of rights. Rights after Delete: are deleted from the current set of rights. If only right codes

20

are listed after Delete:, then all versions of rights with those codes are deleted. Versions of rights after Replace: subsume all versions of rights of the same type in the current set of rights.

If Remaining-Rights is not specified, then there are no rights for the original after all Loan copies are loaned out. If Remaining-Rights is specified, then the Keep: token can be used to simplify the expression of what rights to keep behind. A list of right codes following keep means that all of the versions of those listed rights are kept in the remaining copy. This specification can be overridden by subsequent Delete: or Replace: specifications.

Copy Count Specification

For various transactions, it may be desirable to provide some limit as to the number of “copies” of the work which may be exercised simultaneously for the right. For example, it may be desirable to limit the number of copies of a digital work that may be loaned out at a time or viewed at a time.

Grammar element **1510** “Copy-Count:=(Copies: positive-integer|0|unlimited)” provides a condition which defines the number of “copies” of a work subject to the right. A copy count can be 0, a fixed number, or unlimited. The copy-count is associated with each right, as opposed to there being just a single copy-count for the digital work. The Copy-Count for a right is decremented each time that a right is exercised. When the Copy-Count equals zero, the right can no longer be exercised. If the Copy-Count is not specified, the default is one.

Control Specification

Rights and fees depend in general on rights granted by the creator as well as further restrictions imposed by later distributors. Control specifications deal with interactions between the creators and their distributors governing the imposition of further restrictions and fees. For example, a distributor of a digital work may not want an end consumer of a digital work to add fees or otherwise profit by commercially exploiting the purchased digital work.

Grammar element **1511** “Control-Spec:=(Control: {Restrictable|Unrestrictable} {Unchargeable|Chargeable}-)” provides a condition to specify the effect of usage rights and fees of parents on the exercise of the right. A digital work is restrictable if higher level d-blocks can impose further restrictions (time specifications and access specifications) on the right. It is unrestrictable if no further restrictions can be imposed. The default setting is restrictable. A right is unchargeable if no more fees can be imposed on the use of the right. It is chargeable if more fees can be imposed. The default is chargeable.

Time Specification

It is often desirable to assign a start date or specify some duration as to when a right may be exercised. Grammar element **1512** “Time-Spec:=({Fixed-Interval|Sliding-Interval|Meter-Time} Until: Expiration-Date)” provides for specification of time conditions on the exercise of a right. Rights may be granted for a specified time. Different kinds of time specifications are appropriate for different kinds of rights. Some rights may be exercised during a fixed and predetermined duration. Some rights may be exercised for an interval that starts the first time that the right is invoked by some transaction. Some rights may be exercised or are charged according to some kind of metered time, which may be split into separate intervals. For example, a right to view a picture for an hour might be split into six ten minute viewings or four fifteen minute viewings or twenty three minute viewings.

The terms “time” and “date” are used synonymously to refer to a moment in time. There are several kinds of time specifications. Each specification represents some limitation

US 8,393,007 B2

21

on the times over which the usage right applies. The Expiration-Date specifies the moment at which the usage right ends. For example, if the Expiration-Date is “Jan. 1, 1995,” then the right ends at the first moment of 1995. If the Expiration-Date is specified as *forever*, then the rights are interpreted as continuing without end. If only an expiration date is given, then the right can be exercised as often as desired until the expiration date.

Grammar element 1513 “Fixed-Interval:=From: Start-Time” is used to define a predetermined interval that runs from the start time to the expiration date.

Grammar element 1514 “Sliding-Interval:=Interval: Use-Duration” is used to define an indeterminate (or “open”) start time. It sets limits on a continuous period of time over which the contents are accessible. The period starts on the first access and ends after the duration has passed or the expiration date is reached, whichever comes first. For example, if the right gives 10 hours of continuous access, the use-duration would begin when the first access was made and end 10 hours later.

Grammar element 1515 “Meter-Time:=Time-Remaining: Remaining-Use” is used to define a “meter time,” that is, a measure of the time that the right is actually exercised. It differs from the Sliding-Interval specification in that the time that the digital work is in use need not be continuous. For example, if the rights guarantee three days of access, those days could be spread out over a month. With this specification, the rights can be exercised until the meter time is exhausted or the expiration date is reached, whichever comes first.

Remaining-Use:=Time-Unit

Start-Time:=Time-Unit

Use-Duration:=Time-Unit

All of the time specifications include time-unit specifications in their ultimate instantiation.

Security Class and Authorization Specification

The present invention provides for various security mechanisms to be introduced into a distribution or use scheme. Grammar element 1516 “Access-Spec:={SC: Security-Class} {Authorization: Authorization-ID*} {Other-Authorization: Authorization-ID*} {Ticket: Ticket-ID}” provides a means for restricting access and transmission. Access specifications can specify a required security class for a repository to exercise a right or a required authorization test that must be satisfied.

The keyword “SC:” is used to specify a minimum security level for the repositories involved in the access. If “SC:” is not specified, the lowest security level is acceptable.

The optional “Authorization:” keyword is used to specify required authorizations on the same repository as the work. The optional “Other-Authorization:” keyword is used to specify required authorizations on the other repository in the transaction.

The optional “Ticket:” keyword specifies the identity of a ticket required for the transaction. A transaction involving digital tickets must locate an appropriate digital ticket agent who can “punch” or otherwise validate the ticket before the transaction can proceed. Tickets are described in greater detail below.

In a transaction involving a repository and a document server, some usage rights may require that the repository have a particular authorization, that the server have some authorization, or that both repositories have (possibly different) authorizations. Authorizations themselves are digital works (hereinafter referred to as an authorization object) that can be moved between repositories in the same manner as other digital works. Their copying and transferring is subject to the

22

same rights and fees as other digital works. A repository is said to have an authorization if that authorization object is contained within the repository.

In some cases, an authorization may be required from a source other than the document server and repository. An authorization object referenced by an Authorization-ID can contain digital address information to be used to set up a communications link between a repository and the authorization source. These are analogous to phone numbers. For such access tests, the communication would need to be established and authorization obtained before the right could be exercised.

For one-time usage rights, a variant on this scheme is to have a digital ticket. A ticket is presented to a digital ticket agent, whose type is specified on the ticket. In the simplest case, a certified generic ticket agent, available on all repositories, is available to “punch” the ticket. In other cases, the ticket may contain addressing information for locating a “special” ticket agent. Once a ticket has been punched, it cannot be used again for the same kind of transaction (unless it is unpunched or refreshed in the manner described below.) Punching includes marking the ticket with a timestamp of the date and time it was used. Tickets are digital works and can be copied or transferred between repositories according to their usage rights.

In the currently preferred embodiment, a “punched” ticket becomes “unpunched” or “refreshed” when it is copied or extracted. The Copy and Extract operations save the date and time as a property of the digital ticket. When a ticket agent is given a ticket, it can simply check whether the digital copy was made after the last time that it was punched. Of course, the digital ticket must have the copy or extract usage rights attached thereto.

The capability to unpunch a ticket is important in the following cases:

A digital work is circulated at low cost with a limitation that it can be used only once.

A digital work is circulated with a ticket that can be used once to give discounts on purchases of other works.

A digital work is circulated with a ticket (included in the purchase price and possibly embedded in the work) that can be used for a future upgrade.

In each of these cases, if a paid copy is made of the digital work (including the ticket) the new owner would expect to get a fresh (unpunched) ticket, whether the copy seller has used the work or not. In contrast, loaning a work or simply transferring it to another repository should not revitalize the ticket.

Usage Fees and Incentives Specification

The billing for use of a digital work is fundamental to a commercial distribution system. Grammar Element 1517 “Fee-Spec:={Scheduled-Discount} Regular-Fee-Spec|Scheduled-Fee-Spec|Markup-Spec” provides a range of options for billing for the use of digital works.

A key feature of this approach is the development of low-overhead billing for transactions in potentially small amounts. Thus, it becomes feasible to collect fees of only a few cents each for thousands of transactions.

The grammar differentiates between uses where the charge is per use from those where it is metered by the time unit. Transactions can support fees that the user pays for using a digital work as well as incentives paid by the right grantor to users to induce them to use or distribute the digital work.

The optional scheduled discount refers to the rest of the fee specification—discounting it by a percentage over time. If it is not specified, then there is no scheduled discount. Regular fee specifications are constant over time. Scheduled fee specifications give a schedule of dates over which the fee specifi-

US 8,393,007 B2

23

cations change. Markup specifications are used in d-blocks for adding a percentage to the fees already being charged.

Grammar Element **1518** "Scheduled-Discount:=(Scheduled-Discount: (Time-Spec Percentage)*)" A Scheduled-Discount is essentially a scheduled modifier of any other fee specification for this version of the right of the digital work. (It does not refer to children or parent digital works or to other versions of rights.). It is a list of pairs of times and percentages. The most recent time in the list that has not yet passed at the time of the transaction is the one in effect. The percentage gives the discount percentage. For example, the number 10 refers to a 10% discount.

Grammar Element **1519** "Regular-Fee-Spec:=({Fee: |Incentive: } [Per-Use-Spec|Metered-Rate-Spec|Best-Price-Spec|Call-For-Price-Spec] {Min: Money-Unit Per: Time-Spec} {Max: Money-Unit Per: Time-Spec} To: Account-ID)" provides for several kinds of fee specifications.

Fees are paid by the copy-owner/user to the revenue-owner if Fee: is specified. Incentives are paid by the revenue-owner to the user if Incentive: is specified. If the Min: specification is given, then there is a minimum fee to be charged per time-spec unit for its use. If the Max: specification is given, then there is a maximum fee to be charged per time-spec for its use. When Fee: is specified, Account-ID identifies the account to which the fee is to be paid. When Incentive: is specified, Account-ID identifies the account from which the fee is to be paid.

Grammar element **1520** "Per-Use-Spec:=Per-Use: Money-unit" defines a simple fee to be paid every time the right is exercised, regardless of how much time the transaction takes.

Grammar element **1521** "Metered-Rate-Spec:=Metered: Money-Unit Per: Time-Spec" defines a metered-rate fee paid according to how long the right is exercised. Thus, the time it takes to complete the transaction determines the fee.

Grammar, element **1522** "Best-Price-Spec:=Best-Price: Money-unit Max: Money-unit" is used to specify a best-price that is determined when the account is settled. This specification is to accommodate special deals, rebates, and pricing that depends on information that is not available to the repository. All fee specifications can be combined with tickets or authorizations that could indicate that the consumer is a wholesaler or that he is a preferred customer, or that the seller be authorized in some way. The amount of money in the Max: field is the maximum amount that the use will cost. This is the amount that is tentatively debited from the credit server. However, when the transaction is ultimately reconciled, any excess amount will be returned to the consumer in a separate transaction.

Grammar element **1523** "Call-For-Price-Spec:=Call-For-Price" is similar to a "Best-Price-Spec" in that it is intended to accommodate cases where prices are dynamic. A Call-For-Price Spec requires a communication with a dealer to determine the price. This option cannot be exercised if the repository cannot communicate with a dealer at the time that the right is exercised. It is based on a secure transaction whereby the dealer names a price to exercise the right and passes along a deal certificate which is referenced or included in the billing process.

Grammar element **1524** "Scheduled-Fee-Spec:=(Schedule: (Time-Spec Regular-Fee-Spec)*)" is used to provide a schedule of dates over which the fee specifications change. The fee specification with the most recent date not in the future is the one that is in effect. This is similar to but more general than the scheduled discount. It is more general, because it provides a means to vary the fee agreement for each time period.

24

Grammar element **1525** "Markup-Spec:=Markup: percentage To: Account-ID" is provided for adding a percentage to the fees already being charged. For example, a 5% markup means that a fee of 5% of cumulative fee so far will be allocated to the distributor. A markup specification can be applied to all of the other kinds of fee specifications. It is typically used in a shell provided by a distributor. It refers to fees associated with d-blocks that are parts of the current d-block. This might be a convenient specification for use in taxes, or in distributor overhead.

Examples of Sets of Usage Rights

((Play) (Transfer (SC: 3)) (Delete)

This work can be played without requirements for fee or authorization on any rendering system. It can be transferred to any other repository of security level 3 or greater. It can be deleted.

((Play) (Transfer (SC: 3)) (Delete) (Backup) (Restore (Fee: Per-Use: \$5 To: Account-ID-678)))

Same as the previous example plus rights for backup and restore. The work can be backed up without fee. It can be restored for a \$5 fee payable to the account described by Account-ID-678.

((Play) (Transfer (SC: 3))

(Copy (SC:3)(Fee: Per-Use: \$5 To: Account-ID-678))

(Delete (Incentive: Per-Use: \$2.50 To: Account-ID-678)))

This work can be played, transferred, copied, or deleted. Copy or transfer operations can take place only with repositories of security level three or greater. The fee to make a copy is \$5 payable to Account-ID-678. If a copy is deleted, then an incentive of \$2.50 is paid to the former copy owner.

((Play) (Transfer (SC: 3))

Copy (SC: 3) (Fee: Per-Use: \$10 To: Account-ID-678))

Delete) (Backup) (Restore (SC: 3) (Fee: Per-Use: \$5 To: Account-ID-678)))

Same as the previous example plus fees for copying. The work can be copied digitally for a fee of \$10 payable to Account-ID-678. The repository on which the work is copied or restored must be at security level 3 or greater.

((Play) (Transfer (SC: 3))

(Copy Authorization: License-123-ID (SC: 3)))

The digital work can be played, transferred, or copied. Copies or transfers must be on repositories of security level 3 or greater. Copying requires the license License-123-ID issued to the copying repository. None of the rights require fees.

((Play) (Print Printer: Printer-567-ID (Fee: Per-Use: \$1 To: Account-ID-678)))

This work can be played for free. It can be printed on any printer with the identifier Printer-567-ID for a fee of \$1 payable to the account described by Account-ID-678.

((Play Player: Player-876-ID) (From: Feb. 2, 1994 Until: Feb. 15, 1995) (Fee: Metered: \$0.01 Per: 0:1:0 Min: \$0.25 Per: 0/1/0 To: Account-ID-567))

This work can be played on any player holding the ID Player-876-ID. The time of this right is from Feb. 14, 1994 until Feb. 15, 1995. The fee for use is one cent per minute with a minimum of 25 cents in any day that it is used, payable to the account described by Account-ID-567.

((Play) (Transfer) (Delete)(Loan 2 (Delete: Transfer Loan)))

This work can be played, transferred, deleted, or loaned. Up to two copies can be loaned out at a time. The loaned copy has the same rights except that it cannot be transferred. When both copies are loaned out, no rights can be exercised on the original on the repository.

US 8,393,007 B2

25

((Play) (Transfer) (Delete) (Backup) (Restore (SC:3))
 (Loan 2 Remaining-Copy-Rights: (Delete: Play Transfer)
 Next-Set-of-Rights: (Delete: Transfer Loan)))

Similar to previous example. Rights to Backup and Restore
 the work are added, where restoration requires a repository of
 at least security level three. When all copies of the work are
 loaned out, the remaining copy cannot be played or trans-
 ferred.

((Play) (Transfer) (Copy) (Print) (Backup) (Restore (SC:
 3)))

(Loan 1 Remaining-Copy-Rights: (Add: Play Print
 Backup)

Next-Set-of-Rights: (Delete: Transfer Loan)
 (Fee: Metered: \$10 Per: 1:0:0 To: Account-ID-567))

(Loan 1 Remaining-Copy-Rights:

Add: ((Play Player: Player-876-ID) 2 (From: Feb. 14, 1994
 Until: Feb. 15, 1995)

(Fee: Metered: \$0.01 Per: 0:1:0 Min: \$0.25 Per: 0/1/0
 To: Account-ID-567)))

The original work has rights to Play, Transfer, Copy, Print,
 Backup, Restore, and Loan. There are two versions of the
 Loan right. The first version of the loan right costs \$10 per day
 but allows the original copy owner to exercise free use of the
 Play, Print and Backup rights. The second version of the Loan
 right is free. None of the original rights are applicable. How-
 ever a right to Play the work at the specified metered rate is
 added.

((Play Player: Player-Small-Screen-123-ID)
 (Embed (Fee: Per-Use \$0.01 To: Account-678-ID))
 (Copy (Fee: Per-Use \$1.00 To: Account-678-ID)))

The digital work can be played on any player with the
 identifier Player-Small-Screen-123-ID. It can be embedded
 in a larger work. The embedding requires a modest one cent
 registration fee to Account-678-ID. Digital copies can be
 made for \$1.00.

Repository Transactions

When a user requests access to a digital work, the reposi-
 tory will initiate various transactions. The combination of
 transactions invoked will depend on the specifications
 assigned for a usage right. There are three basic types of
 transactions, Session Initiation Transactions, Financial
 Transactions and Usage Transactions. Generally, session ini-
 tiation transactions are initiated first to establish a valid ses-
 sion. When a valid session is established, transactions corre-
 sponding to the various usage rights are invoked. Finally,
 request specific transactions are performed.

Transactions occur between two repositories (one acting as
 a server), between a repository and a document playback
 platform (e.g. for executing or viewing), between a repository
 and a credit server or between a repository and an authoriza-
 tion server. When transactions occur between more than one
 repository, it is assumed that there is a reliable communica-
 tion channel between the repositories. For example, this could
 be a TCP/IP channel or any other commercially available
 channel that has built-in capabilities for detecting and cor-
 recting transmission errors. However, it is not assumed that
 the communication channel is secure. Provisions for security
 and privacy are part of the requirements for specifying and
 implementing repositories and thus form the need for various
 transactions.

Message Transmission

Transactions require that there be some communication
 between repositories. Communication between repositories
 occurs in units termed as messages. Because the communi-
 cation line is assumed to be unsecure, all communications

26

with repositories that are above the lowest security class are
 encrypted utilizing a public key encryption technique. Public
 key encryption is a well known technique in the encryption
 arts. The term key refers to a numeric code that is used with
 encryption and decryption algorithms. Keys come in pairs,
 where "writing keys" are used to encrypt data and "checking
 keys" are used to decrypt data. Both writing and checking
 keys may be public or private. Public keys are those that are
 distributed to others. Private keys are maintained in confi-
 dence.

Key management and security is instrumental in the suc-
 cess of a public key encryption system. In the currently pre-
 ferred embodiment, one or more master repositories maintain
 the keys and create the identification certificates used by the
 repositories.

When a sending repository transmits a message to a receiv-
 ing repository, the sending repository encrypts all of its data
 using the public writing key of the receiving repository. The
 sending repository includes its name, the name of the receiv-
 ing repository, a session identifier such as a nonce (described
 below), and a message counter in each message.

In this way, the communication can only be read (to a high
 probability) by the receiving repository, which holds the pri-
 vate checking key for decryption. The auxiliary data is used to
 guard against various replay attacks to security. If messages
 ever arrive with the wrong counter or an old nonce, the reposi-
 tories can assume that someone is interfering with commu-
 nication and the transaction terminated.

The respective public keys for the repositories to be used
 for encryption are obtained in the registration transaction
 described below.

Session Initiation Transactions

A usage transaction is carried out in a session between
 repositories. For usage transactions involving more than one
 repository, or for financial transactions between a repository
 and a credit server, a registration transaction is performed. A
 second transaction termed a login transaction, may also be
 needed to initiate the session. The goal of the registration
 transaction is to establish a secure channel between two
 repositories who know each others identities. As it is assumed
 that the communication channel between the repositories is
 reliable but not secure, there is a risk that a non-repository
 may mimic the protocol in order to gain illegitimate access to
 a repository.

The registration transaction between two repositories is
 described with respect to FIGS. 16 and 17. The steps
 described are from the perspective of a "repository-1" regis-
 tering its identity with a "repository-2". The registration must
 be symmetrical so the same set of steps will be repeated for
 repository-2 registering its identity with repository-1. Refer-
 ring to FIG. 16, repository-1 first generates an encrypted
 registration identifier, step 1601 and then generates a regis-
 tration message, step 1602. A registration message is com-
 prised of an identifier of a master repository, the identification
 certificate for the repository-1 and an encrypted random regis-
 tration identifier. The identification certificate is encrypted
 by the master repository in its private key and attests to the
 fact that the repository (here repository-1) is a bona fide
 repository. The identification certificate also contains a public
 key for the repository, the repository security level and a
 timestamp (indicating a time after which the certificate is no
 longer valid.) The registration identifier is a number gener-
 ated by the repository for this registration. The registration
 identifier is unique to the session and is encrypted in reposi-
 tory-1's private key. The registration identifier is used to
 improve security of authentication by detecting certain kinds

US 8,393,007 B2

27

of communications based attacks. Repository-1 then transmits the registration message to repository-2, step 1603.

Upon receiving the registration message, repository-2 determines if it has the needed public key for the master repository, step 1604. If repository-2 does not have the needed public key to decrypt the identification certificate, the registration transaction terminates in an error, step 1618.

Assuming that repository-2 has the proper public key the identification certificate is decrypted, step 1605. Repository-2 saves the encrypted registration identifier, step 1606, and extracts the repository identifier, step 1607. The extracted repository identifier is checked against a "hotlist" of compromised document repositories, step 1608. In the currently preferred embodiment, each repository will contain "hotlists" of compromised repositories. If the repository is on the "hotlist", the registration transaction terminates in an error per step 1618. Repositories can be removed from the hotlist when their certificates expire, so that the list does not need to grow without bound. Also, by keeping a short list of hotlist certificates that it has previously received, a repository can avoid the work of actually going through the list. These lists would be encrypted by a master repository. A minor variation on the approach to improve efficiency would have the repositories first exchange lists of names of hotlist certificates, ultimately exchanging only those lists that they had not previously received. The "hotlists" are maintained and distributed by Master repositories.

Note that rather than terminating in error, the transaction could request that another registration message be sent based on an identification certificate created by another master repository. This may be repeated until a satisfactory identification certificate is found, or it is determined that trust cannot be established.

Assuming that the repository is not on the hotlist, the repository identification needs to be verified. In other words, repository-2 needs to validate that the repository on the other end is really repository-1. This is termed performance testing and is performed in order to avoid invalid access to the repository via a counterfeit repository replaying a recording of a prior session initiation between repository-1 and repository-2. Performance testing is initiated by repository-2 generating a performance message, step 1609. The performance message consists of a nonce, the names of the respective repositories, the time and the registration identifier received from repository-1. A nonce is a generated message based on some random and variable information (e.g. the time or the temperature.) The nonce is used to check whether repository-1 can actually exhibit correct encrypting of a message using the private keys it claims to have, on a message that it has never seen before. The performance message is encrypted using the public key specified in the registration message of repository-1. The performance message is transmitted to repository-1, step 1610, where it is decrypted by repository-1 using its private key, step 1611. Repository-1 then checks to make sure that the names of the two repositories are correct, step 1612, that the time is accurate, step 1613 and that the registration identifier corresponds to the one it sent, step 1614. If any of these tests fails, the transaction is terminated per step 1616. Assuming that the tests are passed, repository-1 transmits the nonce to repository-2 in the clear, step 1615. Repository-2 then compares the received nonce to the original nonce, step 1617. If they are not identical, the registration transaction terminates in an error per step 1618. If they are the same, the registration transaction has successfully completed.

At this point, assuming that the transaction has not terminated, the repositories exchange messages containing session keys to be used in all communications during the session and

28

synchronize their clocks. FIG. 17 illustrates the session information exchange and clock synchronization steps (again from the perspective of repository-1.) Referring to FIG. 17, repository-1 creates a session key pair, step 1701. A first key is kept private and is used by repository-1 to encrypt messages. The second key is a public key used by repository-2 to decrypt messages. The second key is encrypted using the public key of repository-2, step 1702 and is sent to repository-2, step 1703. Upon receipt, repository-2 decrypts the second key, step 1704. The second key is used to decrypt messages in subsequent communications. When each repository has completed this step, they are both convinced that the other repository is bona fide and that they are communicating with the original. Each repository has given the other a key to be used in decrypting further communications during the session. Since that key is itself transmitted in the public key of the receiving repository only it will be able to decrypt the key which is used to decrypt subsequent messages.

After the session information is exchanged, the repositories must synchronize their clocks. Clock synchronization is used by the repositories to establish an agreed upon time base for the financial records of their mutual transactions. Referring back to FIG. 17, repository-2 initiates clock synchronization by generating a time stamp exchange message, step 1705, and transmits it to repository-1, step 1706. Upon receipt, repository-1 generates its own time stamp message, step 1707 and transmits it back to repository-2, step 1708. Repository-2 notes the current time, step 1709 and stores the time received from repository-1, step 1710. The current time is compared to the time received from repository-1, step 1711. The difference is then checked to see if it exceeds a predetermined tolerance (e.g. one minute), step 1712. If it does, repository-2 terminates the transaction as this may indicate tampering with the repository, step 1713. If not repository-2 computes an adjusted time delta, step 1714. The adjusted time delta is the difference between the clock time of repository-2 and the average of the times from repository-1 and repository-2.

To achieve greater accuracy, repository-2 can request the time again up to a fixed number of times (e.g. five times), repeat the clock synchronization steps, and average the results.

A second session initiation transaction is a Login transaction. The Login transaction is used to check the authenticity of a user requesting a transaction. A Login transaction is particularly prudent for the authorization of financial transactions that will be charged to a credit server. The Login transaction involves an interaction between the user at a user interface and the credit server associated with a repository. The information exchanged here is a login string supplied by the repository/credit server to identify itself to the user, and a Personal Identification Number (PIN) provided by the user to identify himself to the credit server. In the event that the user is accessing a credit server on a repository different from the one on which the user interface resides, exchange of the information would be encrypted using the public and private keys of the respective repositories.

Billing Transactions

Billing Transactions are concerned with monetary transaction with a credit server. Billing Transactions are carried out when all other conditions are satisfied and a usage fee is required for granting the request. For the most part, billing transactions are well understood in the state of the art. These transactions are between a repository and a credit server, or between a credit server and a billing clearinghouse. Briefly, the required transactions include the following:

US 8,393,007 B2

29

Registration and LOGIN transactions, by which the repository and user establish their bona fides to a credit server. These transactions would be entirely internal in cases where the repository and credit server are implemented as a single system.

Registration and LOGIN transactions, by which a credit server establishes its bona fides to a billing clearinghouse.

An Assign-fee transaction to assign a charge. The information in this transaction would include a transaction identifier, the identities of the repositories in the transaction, and a list of charges from the parts of the digital work. If there has been any unusual event in the transaction such as an interruption of communications, that information is included as well.

A Begin-charges transaction to assign a charge. This transaction is much the same as an assign-fee transaction except that it is used for metered use. It includes the same information as the assign-fee 4, ii transaction as well as the usage fee information. The credit-server is then responsible for running a clock.

An End-charges transaction to end a charge for metered use. (In a variation on this approach, the repositories would exchange periodic charge information for each block of time.)

A report-charges transaction between a personal credit server and a billing clearinghouse. This transaction is invoked at least once per billing period. It is used to pass along information about charges. On debit and credit cards, this transaction would also be used to update balance information and credit limits as needed.

All billing transactions are given a transaction ID and are reported to the credit servers by both the server and the client. This reduces possible loss of billing information if one of the parties to a transaction loses a banking card and provides a check against tampering with the system.

Usage Transactions

After the session initiation transactions have been completed, the usage request may then be processed. To simplify the description of the steps carried out in processing a usage request, the term requester is used to refer to a repository in the requester mode which is initiating a request, and the term server is used to refer to a repository in the server mode and which contains the desired digital work. In many cases such as requests to print or view a work, the requester and server may be the same device and the transactions described in the following would be entirely internal. In such instances, certain transaction steps, such as the registration transaction, need not be performed.

There are some common steps that are part of the semantics of all of the usage rights transactions. These steps are referred to as the common transaction steps. There are two sets—the “opening” steps and the “closing” steps. For simplicity, these are listed here rather than repeating them in the descriptions of all of the usage rights transactions.

Transactions can refer to a part of a digital work, a complete digital work, or a Digital work containing other digital works. Although not described in detail herein, a transaction may even refer to a folder comprised of a plurality of digital works. The term “work” is used to refer to what ever portion or set of digital works is being accessed.

Many of the steps here involve determining if certain conditions are satisfied. Recall that each usage right may have one or more conditions which must be satisfied before the right can be exercised. Digital works have parts and parts have parts. Different parts can have different rights and fees. Thus, it is necessary to verify that the requirements are met for ALL of the parts that are involved in a transaction For brevity, when reference is made to checking whether the rights exist and

30

conditions for exercising are satisfied, it is meant that all such checking takes place for each of the relevant parts of the work.

FIG. 18 illustrates the initial common opening and closing steps for a transaction. At this point it is assumed that registration has occurred and that a “trusted” session is in place. General tests are tests on usage rights associated with the folder containing the work or some containing folder higher in the file system hierarchy. These tests correspond to requirements imposed on the work as a consequence of its being on the particular repository, as opposed to being attached to the work itself. Referring to FIG. 18, prior to initiating a usage transaction, the requester performs any general tests that are required before the right associated with the transaction can be exercised, step, 1801. For example, install, uninstall and delete rights may be implemented to require that a requester have an authorization certificate before the right can be exercised. Another example is the requirement that a digital ticket be present and punched before a digital work may be copied to a requester. If any of the general tests fail, the transaction is not initiated, step, 1802. Assuming that such required tests are passed, upon receiving the usage request, the server generates a transaction identifier that is used in records or reports of the transaction, step 1803. The server then checks whether the digital work has been granted the right corresponding to the requested transaction, step 1804. If the digital work has not been granted the right corresponding to the request, the transaction terminates, step 1805. If the digital work has been granted the requested right, the server then determines if the various conditions for exercising the right are satisfied. Time based conditions are examined, step 1806. These conditions are checked by examining the time specification for the version of the right. If any of the conditions are not satisfied, the transaction terminates per step 1805.

Assuming that the time based conditions are satisfied, the server checks security and access conditions, step 1807. Such security and access conditions are satisfied if: 1) the requester is at the specified security class, or a higher security class, 2) the server satisfies any specified authorization test and 3) the requester satisfies any specified authorization tests and has any required digital tickets. If any of the conditions are not satisfied, the transaction terminates per step 1805.

Assuming that the security and access conditions are all satisfied, the server checks the copy count condition, step 1808. If the copy count equals zero, then the transaction cannot be completed and the transaction terminates per step 1805.

Assuming that the copy count does not equal zero, the server checks if the copies in use for the requested right is greater than or equal to any copy count for the requested right (or relevant parts), step 1809. If the copies in use are greater than or equal to the copy count, this indicates that usage rights for the version of the transaction have been exhausted. Accordingly, the server terminates the transaction, step 1805. If the copy count is less than the copies in use for the transaction the transaction can continue, and the copies in use would be incremented by the number of digital works requested in the transaction, step 1810.

The server then checks if the digital work has a “Loan” access right, step 1811. The “Loan” access right is a special case since remaining rights may be present even though all copies are loaned out. If the digital work has the “Loan” access right, a check is made to see if all copies have been loaned out, step 1812. The number of copies that could be loaned is the sum of the Copy-Counts for all of the versions of the loan right of the digital work. For a composite work, the relevant figure is the minimal such sum of each of the components of the composite work. If all copies have been loaned

US 8,393,007 B2

31

out, the remaining rights are determined, step **1813**. The remaining-rights are determined from the remaining rights specifications from the versions of the Loan right. If there is only one version of the Loan right, then the determination is simple. The remaining rights are the ones specified in that version of the Loan right, or none if Remaining-Rights: is not specified. If there are multiple versions of the Loan right and all copies of all of the versions are loaned out, then the remaining rights is taken as the minimum set (intersection) of remaining rights across all of the versions of the loan right. The server then determines if the requested right is in the set of remaining rights, step **1814**. If the requested right is not in the set of remaining rights, the server terminates the transaction, step **1805**.

If Loan is not a usage right for the digital work or if all copies have not been loaned out or the requested right is in the set of remaining rights, fee conditions for the right are then checked, step **1815**. This will initiate various financial transactions between the repository and associated credit server. Further, any metering of usage of a digital work will commence. If any financial transaction fails, the transaction terminates per step **1805**.

It should be noted that the order in which the conditions are checked need not follow the order of steps **1806-1815**.

At this point, right specific steps are now performed and are represented here as step **1816**. The right specific steps are described in greater detail below.

The common closing transaction steps are now performed. Each of the closing transaction steps are performed by the server after a successful completion of a transaction. Referring back to FIG. **18**, the copies in use value for the requested right is decremented by the number of copies involved in the transaction, step **1817**. Next, if the right had a metered usage fee specification, the server subtracts the elapsed time from the Remaining-Use-Time associated with the right for every part involved in the transaction, step **1818**. Finally, if there are fee specifications associated with the right, the server initiates End-Charge financial transaction to confirm billing, step **1819**.

Transmission Protocol

An important area to consider is the transmission of the digital work from the server to the requester. The transmission protocol described herein refers to events occurring after a valid session has been created. The transmission protocol must handle the case of disruption in the communications between the repositories. It is assumed that interference such as injecting noise on the communication channel can be detected by the integrity checks (e.g., parity, checksum, etc.) that are built into the transport protocol and are not discussed in detail herein.

The underlying goal in the transmission protocol is to preclude certain failure modes, such as malicious or accidental interference on the communications channel. Suppose, for example, that a user pulls a card with the credit server at a specific time near the end of a transaction. There should not be a vulnerable time at which "pulling the card" causes the repositories to fail to correctly account for the number of copies of the work that have been created. Restated, there should be no time at which a party can break a connection as a means to avoid payment after using a digital work.

If a transaction is interrupted (and fails), both repositories restore the digital works and accounts to their state prior to the failure, modulo records of the failure itself.

FIG. **19** is a state diagram showing steps in the process of transmitting information during a transaction. Each box represents a state of a repository in either the server mode (above the central dotted line **1901**) or in the requester mode (below

32

the dotted line **1901**). Solid arrows stand for transitions between states. Dashed arrows stand for message communications between the repositories. A dashed message arrow pointing to a solid transition arrow is interpreted as meaning that the transition takes place when the message is received. Unlabeled transition arrows take place unconditionally. Other labels on state transition arrows describe conditions that trigger the transition.

Referring now to FIG. **19**, the server is initially in a state **1902** where a new transaction is initiated via start message **1903**. This message includes transaction information including a transaction identifier and a count of the blocks of data to be transferred. The requester, initially in a wait state **1904** then enters a data wait state **1905**.

The server enters a data transmit state **1906** and transmits a block of data **1907** and then enters a wait for acknowledgement state **1908**. As the data is received, the requesters enters a data receive state **1909** and when the data blocks is completely received it enters an acknowledgement state **1910** and transmits an Acknowledgement message **1911** to the server.

If there are more blocks to send, the server waits until receiving an Acknowledgement message from the requester. When an Acknowledgement message is received it sends the next block to the requester and again waits for acknowledgement. The requester also repeats the same cycle of states.

If the server detects a communications failure before sending the last block, it enters a cancellation state **1912** wherein the transaction is cancelled. Similarly, if the requester detects a communications failure before receiving the last block it enters a cancellation state **1913**.

If there are no more blocks to send, the server commits to the transaction and waits for the final Acknowledgement in state **1914**. If there is a communications failure before the server receives the final Acknowledgement message, it still commits to the transaction but includes a report about the event to its credit server in state **1915**. This report serves two purposes. It will help legitimize any claims by a user of having been billed for receiving digital works that were not completely received. Also it helps to identify repositories and communications lines that have suspicious patterns of use and interruption. The server then enters its completion state

On the requester side, when there are no more blocks to receive, the requester commits to the transaction in state **1917**. If the requester detects a communications failure at this state, it reports the failure to its credit server in state **1918**, but still commits to the transaction. When it has committed, it sends an acknowledgement message to the server. The server then enters its completion state **1919**.

The key property is that both the server and the requester cancel a transaction if it is interrupted before all of the data blocks are delivered, and commits to it if all of the data blocks have been delivered.

There is a possibility that the server will have sent all of the data blocks (and committed) but the requester will not have received all of them and will cancel the transaction. In this case, both repositories will presumably detect a communications failure and report it to their credit server. This case will probably be rare since it depends on very precise timing of the communications failure. The only consequence will be that the user at the requester repository may want to request a refund from the credit services—and the case for that refund will be documented by reports by both repositories.

To prevent loss of data, the server should not delete any transferred digital work until receiving the final acknowledgement from the requester. But it also should not use the file. A well known way to deal with this situation is called "two-phase commit" or 2PC.

US 8,393,007 B2

33

Two-phase commit works as follows. The first phase works the same as the method described above. The server sends all of the data to the requester. Both repositories mark the transaction (and appropriate files) as uncommitted. The server sends a ready-to-commit message to the requester. The requester sends back an acknowledgement. The server then commits and sends the requester a commit message. When the requester receives the commit message, it commits the file.

If there is a communication failure or other crash, the requester must check back with the server to determine the status of the transaction. The server has the last word on this. The requester may have received all of the data, but if it did not get the final message, it has not committed. The server can go ahead and delete files (except for transaction records) once it commits, since the files are known to have been fully transmitted before starting the 2PC cycle.

There are variations known in the art which can be used to achieve the same effect. For example, the server could use an additional level of encryption when transmitting a work to a client. Only after the client sends a message acknowledging receipt does it send the key. The client then agrees to pay for the digital work. The point of this variation is that it provides a clear audit trail that the client received the work. For trusted systems, however, this variation adds a level of encryption for no real gain in accountability.

The transactions for specific usage rights are now discussed.

The Copy Transaction

A Copy transaction is a request to make one or more independent copies of the work with the same or lesser usage rights. Copy differs from the extraction right discussed later in that it refers to entire digital works or entire folders containing digital works. A copy operation cannot be used to remove a portion of a digital work.

The requester sends the server a message to initiate the Copy Transaction. This message indicates the work to be copied, the version of the copy right to be used for the transaction, the destination address information (location in a folder) for placing the work, the file data for the work (including its size), and the number of copies requested.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the client according to the transmission protocol. If a Next-Set-Of-Rights has been provided in the version of the right, those rights are transmitted as the rights for the work. Otherwise, the rights of the original are transmitted. In any event, the Copy-Count field for the copy of the digital work being sent right is set to the number-of-copies requested.

The requester records the work contents, data, and usage rights and stores the work. It records the date and time that the copy was made in the properties of the digital work.

The repositories perform the common closing transaction steps.

The Transfer Transaction

A Transfer transaction is a request to move copies of the work with the same or lesser usage rights to another repository. In contrast with a copy transaction, this results in removing the work copies from the server.

The requester sends the server a message to initiate the Transfer Transaction. This message indicates the work to be transferred, the version of the transfer right to be used in the transaction, the destination address information for placing the work, the file data for the work, and the number of copies involved.

34

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester according to the transmission protocol. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the work. Otherwise, the rights of the original are transmitted. In either case, the Copy-Count field for the transmitted rights is set to the number-of-copies requested.

The requester records the work contents, data, and usage rights and stores the work.

The server decrements its copy count by the number of copies involved in the transaction.

The repositories perform the common closing transaction steps.

If the number of copies remaining in the server is now zero, it erases the digital work from its memory.

The Loan Transaction

A loan transaction is a mechanism for loaning copies of a digital work. The maximum duration of the loan is determined by an internal parameter of the digital work. Works are automatically returned after a predetermined time period.

The requester sends the server a message to initiate the Transfer Transaction. This message indicates the work to be loaned, the version of the loan right to be used in the transaction, the destination address information for placing the work, the number of copies involved, the file data for the work, and the period of the loan.

The server checks the validity of the requested loan period, and ends with an error if the period is not valid. Loans for a loaned copy cannot extend beyond the period of the original loan to the server.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester.

If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the work. Otherwise, the rights of the original are transmitted, as modified to reflect the loan period.

The requester records the digital work contents, data, usage rights, and loan period and stores the work.

The server updates the usage rights information in the digital work to reflect the number of copies loaned out.

The repositories perform the common closing transaction steps.

The server updates the usage rights data for the digital work. This may preclude use of the work until it is returned from the loan. The user on the requester platform can now use the transferred copies of the digital work. A user accessing the original repository cannot use the digital work, unless there are copies remaining. What happens next depends on the order of events in time.

Case 1. If the time of the loan period is not yet exhausted and the requester sends the repository a Return message.

The return message includes the requester identification, and the transaction ID.

The server decrements the copies-in-use field by the number of copies that were returned. (If the number of digital works returned is greater than the number actually borrowed, this is treated as an error.) This step may now make the work available at the server for other users.

The requester deactivates its copies and removes the contents from its memory.

Case 2. If the time of the loan period is exhausted and the requester has not yet sent a Return message.

The server decrements the copies-in-use field by the number of digital works that were borrowed.

US 8,393,007 B2

35

The requester automatically deactivates its copies of the digital work. It terminates all current uses and erases the digital work copies from memory. One question is why a requester would ever return a work earlier than the period of the loan, since it would be returned automatically anyway. One reason for early return is that there may be a metered fee which determines the cost of the loan. Returning early may reduce that fee.

The Play Transaction

A play transaction is a request to use the contents of a work. Typically, to “play” a work is to send the digital work through some kind of transducer, such as a speaker or a display device. The request implies the intention that the contents will not be communicated digitally to any other system. For example, they will not be sent to a printer, recorded on any digital medium, retained after the transaction or sent to another repository.

This term “play” is natural for examples like playing music, playing a movie, or playing a video game. The general form of play means that a “player” is used to use the digital work. However, the term play covers all media and kinds of recordings. Thus one would “play” a digital work, meaning, to render it for reading, or play a computer program, meaning to execute it. For a digital ticket the player would be a digital ticket agent.

The requester sends the server a message to initiate the play transaction. This message indicates the work to be played, the version of the play right to be used in the transaction, the identity of the player being used, and the file data for the work.

The server checks the validity of the player identification and the compatibility of the player identification with the player specification in the right. It ends with an error if these are not satisfactory.

The repositories perform the common opening transaction steps.

The server and requester read and write the blocks of data as requested by the player according to the transmission protocol. The requester plays the work contents, using the player.

When the player is finished, the player and the requester remove the contents from their memory.

The repositories perform the common closing transaction steps.

The Print Transaction

A Print transaction is a request to obtain the contents of a work for the purpose of rendering them on a “printer.” We use the term “printer” to include the common case of writing with ink on paper. However, the key aspect of “printing” in our use of the term is that it makes a copy of the digital work in a place outside of the protection of usage rights. As with all rights, this may require particular authorization certificates.

Once a digital work is printed, the publisher and user are bound by whatever copyright laws are in effect. However, printing moves the contents outside the control of repositories. For example, absent any other enforcement mechanisms, once a digital work is printed on paper, it can be copied on ordinary photocopying machines without intervention by a repository to collect usage fees. If the printer to a digital disk is permitted, then that digital copy is outside of the control of usage rights. Both the creator and the user know this, although the creator does not necessarily give tacit consent to such copying, which may violate copyright laws.

The requester sends the server a message to initiate a Print transaction. This message indicates the work to be played, the identity of the printer being used, the file data for the work, and the number of copies in the request.

36

The server checks the validity of the printer identification and the compatibility of the printer identification with the printer specification in the right. It ends with an error if these are not satisfactory.

The repositories perform the common opening transaction steps.

The server transmits blocks of data according to the transmission protocol.

The requester prints the work contents, using the printer.

When the printer is finished, the printer and the requester remove the contents from their memory.

The repositories perform the common closing transaction steps.

The Backup Transaction

A Backup transaction is a request to make a backup copy of a digital work, as a protection against media failure. In the context of repositories, secure backup copies differ from other copies in three ways: (1) they are made under the control of a Backup transaction rather than a Copy transaction, (2) they do not count as regular copies, and (3) they are not usable as regular copies. Generally, backup copies are encrypted.

Although backup copies may be transferred or copied, depending on their assigned rights, the only way to make them useful for playing, printing or embedding is to restore them.

The output of a Backup operation is both an encrypted data file that contains the contents and description of a work, and a restoration file with an encryption key for restoring the encrypted contents. In many cases, the encrypted data file would have rights for “printing” it to a disk outside of the protection system, relying just on its encryption for security. Such files could be stored anywhere that was physically safe and convenient. The restoration file would be held in the repository. This file is necessary for the restoration of a backup copy. It may have rights for transfer between repositories.

The requester sends the server a message to initiate a backup transaction. This message indicates the work to be backed up, the version of the backup right to be used in the transaction, the destination address information for placing the backup copy, the file data for the work.

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the work. Otherwise, a set of default rights for backup files of the original are transmitted by the server.

The requester records the work contents, data, and usage rights. It then creates a one-time key and encrypts the contents file. It saves the key information in a restoration file.

The repositories perform the common closing transaction steps.

In some cases, it is convenient to be able to archive the large, encrypted contents file to secure offline storage, such as a magneto-optical storage system or magnetic tape. This creation of a non-repository archive file is as secure as the encryption process. Such non-repository archive storage is considered a form of “printing” and is controlled by a print right with a specified “archive-printer.” An archive-printer device is programmed to save the encrypted contents file (but not the description file) offline in such a way that it can be retrieved.

The Restore Transaction

A Restore transaction is a request to convert an encrypted backup copy of a digital work into a usable copy. A restore operation is intended to be used to compensate for cata-

US 8,393,007 B2

37

strophic media failure. Like all usage rights, restoration rights can include fees and access tests including authorization checks.

The requester sends the server a message to initiate a Restore transaction. This message indicates the work to be restored, the version of the restore right for the transaction, the destination address information for placing the work, and the file data for the work.

The server verifies that the contents file is available (i.e. a digital work corresponding to the request has been backed-up.) If it is not, it ends the transaction with an error.

The repositories perform the common opening transaction steps.

The server retrieves the key from the restoration file. It decrypts the work contents, data, and usage rights.

The server transmits the requested contents and data to the requester according to the transmission protocol. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the work. Otherwise, a set of default rights for backup files of the original are transmitted by the server.

The requester stores the digital work.

The repositories perform the common closing transaction steps.

The Delete Transaction

A Delete transaction deletes a digital work or a number of copies of a digital work from a repository. Practically all digital works would have delete rights.

The requester sends the server a message to initiate a delete transaction. This message indicates the work to be deleted, the version of the delete right for the transaction.

The repositories perform the common opening transaction steps.

The server deletes the file, erasing it from the file system.

The repositories perform the common closing transaction steps.

The Directory Transaction

A Directory transaction is a request for information about folders, digital works, and their parts. This amounts to roughly the same idea as protection codes in a conventional file system like TENEX, except that it is generalized to the full power of the access specifications of the usage rights language.

The Directory transaction has the important role of passing along descriptions of the rights and fees associated with a digital work. When a user wants to exercise a right, the user interface of his repository implicitly makes a directory request to determine the versions of the right that are available. Typically these are presented to the user—such as with different choices of billing for exercising a right. Thus, many directory transactions are invisible to the user and are exercised as part of the normal process of exercising all rights.

The requester sends the server a message to initiate a Directory transaction. This message indicates the file or folder that is the root of the directory request and the version of the directory right used for the transaction.

The server verifies that the information is accessible to the requester.

In particular, it does not return the names of any files that have a HIDE-NAME status in their directory specifications, and it does not return the parts of any folders or files that have HIDE-PARTS in their specification. If the information is not accessible, the server ends the transaction with an error.

38

The repositories perform the common opening transaction steps.

The server sends the requested data to the requester according to the transmission protocol.

5 The requester records the data.

The repositories perform the common closing transaction steps.

The Folder Transaction

10 A Folder transaction is a request to create or rename a folder, or to move a work between folders. Together with Directory rights, Folder rights control the degree to which organization of a repository can be accessed or modified from another repository.

The requester sends the server a message to initiate a Folder transaction. This message indicates the folder that is the root of the folder request, the version of the folder right for the transaction, an operation, and data. The operation can be one of create, rename, and move file. The data are the specifications required for the operation, such as a specification of a folder or digital work and a name.

20 The repositories perform the common opening transaction steps.

The server performs the requested operation—creating a folder, renaming a folder, or moving a work between folders.

25 The repositories perform the common closing transaction steps.

The Extract Transaction

30 An extract transaction is a request to copy a part of a digital work and to create a new work containing it. The extraction operation differs from copying in that it can be used to separate a part of a digital work from d-blocks or shells that place additional restrictions or fees on it. The extraction operation differs from the edit operation in that it does not change the contents of a work, only its embedding in d-blocks. Extraction creates a new digital work.

35 The requester sends the server a message to initiate an Extract transaction. This message indicates the part of the work to be extracted, the version of the extract right to be used in the transaction, the destination address information for placing the part as a new work, the file data for the work, and the number of copies involved.

40 The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester according to the transmission protocol. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the new work. Otherwise, the rights of the original are transmitted. The Copy-Count field for this right is set to the number-of-copies requested.

45 The requester records the contents, data, and usage rights and stores the work. It records the date and time that new work was made in the properties of the work.

The repositories perform the common closing transaction steps.

55 The Embed Transaction

An embed transaction is a request to make a digital work become a part of another digital work or to add a shell d-block to enable the adding of fees by a distributor of the work.

60 The requester sends the server a message to initiate an Embed transaction. This message indicates the work to be embedded, the version of the embed right to be used in the transaction, the destination address information for placing the part as a work, the file data for the work, and the number of copies involved.

65 The server checks the control specifications for all of the rights in the part and the destination. If they are incompatible, the server ends the transaction with an error.

US 8,393,007 B2

39

The repositories perform the common opening transaction steps.

The server transmits the requested contents and data to the requester according to the transmission protocol. If a Next-Set-Of-Rights has been provided, those rights are transmitted as the rights for the new work. Otherwise, the rights of the original are transmitted. The Copy-Count field for this right is set to the number-of-copies requested.

The requester records the contents, data, and usage rights and embeds the work in the destination file.

The repositories perform the common closing transaction steps.

The Edit Transaction

An Edit transaction is a request to make a new digital work by copying, selecting and modifying portions of an existing digital work. This operation can actually change the contents of a digital work. The kinds of changes that are permitted depend on the process being used. Like the extraction operation, edit operates on portions of a digital work. In contrast with the extract operation, edit does not effect the rights or location of the work. It only changes the contents. The kinds of changes permitted are determined by the type specification of the processor specified in the rights. In the currently preferred embodiment, an edit transaction changes the work itself and does not make a new work. However, it would be a reasonable variation to cause a new copy of the work to be made.

The requester sends the server a message to initiate an Edit transaction. This message indicates the work to be edited, the version of the edit right to be used in the transaction, the file data for the work (including its size), the process-ID for the process, and the number of copies involved.

The server checks the compatibility of the process-ID to be used by the requester against any process-ID specification in the right. If they are incompatible, it ends the transaction with an error.

The repositories perform the common opening transaction steps.

The requester uses the process to change the contents of the digital work as desired. (For example, it can select and duplicate parts of it; combine it with other information; or compute functions based on the information. This can amount to editing text, music, or pictures or taking whatever other steps are useful in creating a derivative work.)

The repositories perform the common closing transaction steps.

The edit transaction is used to cover a wide range of kinds of works. The category describes a process that takes as its input any portion of a digital work and then modifies the input in some way. For example, for text, a process for editing the text would require edit rights. A process for "summarizing" or counting words in the text would also be considered editing. For a music file, processing could involve changing the pitch or tempo, or adding reverberations, or any other audio effect. For digital video works, anything which alters the image would require edit rights. Examples would be coloring, scaling, extracting still photos, selecting and combining frames into story boards, sharpening with signal processing, and so on.

Some creators may want to protect the authenticity of their works by limiting the kinds of processes that can be performed on them. If there are no edit rights, then no processing is allowed at all. A processor identifier can be included to specify what kind of process is allowed. If no process identifier is specified, then arbitrary processors can be used. For an example of a specific process, a photographer may want to allow use of his photograph but may not want it to be col-

40

orized. A musician may want to allow extraction of portions of his work but not changing of the tonality.

Authorization Transactions

There are many ways that authorization transactions can be defined. In the following, our preferred way is to simply define them in terms of other transactions that we already need for repositories. Thus, it is convenient sometimes to speak of "authorization transactions," but they are actually made up of other transactions that repositories already have.

A usage right can specify an authorization-ID, which identifies an authorization object (a digital work in a file of a standard format) that the repository must have and which it must process. The authorization is given to the generic authorization (or ticket) server of the repository which begins to interpret the authorization.

As described earlier, the authorization contains a server identifier, which may just be the generic authorization server or it may be another server. When a remote authorization server is required, it must contain a digital address. It may also contain a digital certificate.

If a remote authorization server is required, then the authorization process first performs the following steps:

The generic authorization server attempts to set up the communications channel. (If the channel cannot be set up, then authorization fails with an error.)

When the channel is set up, it performs a registration process with the remote repository. (If registration fails, then the authorization fails with an error.)

When registration is complete, the generic authorization server invokes a "Play" transaction with the remote repository, supplying the authorization document as the digital work to be played, and the remote authorization server (a program) as the "player." (If the player cannot be found or has some other error, then the authorization fails with an error.)

The authorization server then "plays" the authorization. This involves decrypting it using either the public key of the master repository that issued the certificate or the session key from the repository that transmitted it. The authorization server then performs various tests. These tests vary according to the authorization server. They include such steps as checking issue and validity dates of the authorization and checking any hot-lists of known invalid authorizations. The authorization server may require carrying out any other transactions on the repository as well, such as checking directories, getting some person to supply a password, or playing some other digital work. It may also invoke some special process for checking information about locations or recent events. The "script" for such steps is contained within the authorization server.

If all of the required steps are completed satisfactorily, the authorization server completes the transaction normally, signaling that authorization is granted.

The Install Transaction

An Install transaction is a request to install a digital work as runnable software on a repository. In a typical case, the requester repository is a rendering repository and the software would be a new kind or new version of a player. Also in a typical case, the software would be copied to file system of the requester repository before it is installed.

The requester sends the server an Install message. This message indicates the work to be installed, the version of the Install right being invoked, and the file data for the work (including its size).

The repositories perform the common opening transaction steps.

The requester extracts a copy of the digital certificate for the software. If the certificate cannot be found or the master

US 8,393,007 B2

41

repository for the certificate is not known to the requester, the transaction ends with an error.

The requester decrypts the digital certificate using the public key of the master repository, recording the identity of the supplier and creator, a key for decrypting the software, the compatibility information, and a tamper-checking code. (This step certifies the software.)

The requester decrypts the software using the key from the certificate and computes a check code on it using a 1-way hash function. If the check-code does not match the tamper-checking code from the certificate, the installation transaction ends with an error. (This step assures that the contents of the software, including the various scripts, have not been tampered with.)

The requester retrieves the instructions in the compatibility-checking script and follows them. If the software is not compatible with the repository, the installation transaction ends with an error. (This step checks platform compatibility.)

The requester retrieves the instructions in the installation script and follows them. If there is an error in this process (such as insufficient resources), then the transaction ends with an error. Note that the installation process puts the runnable software in a place in the repository where it is no longer accessible as a work for exercising any usage rights other than the execution of the software as part of repository operations in carrying out other transactions.

The repositories perform the common closing transaction steps.

The Uninstall Transaction

An Uninstall transaction is a request to remove software from a repository. Since uncontrolled or incorrect removal of software from a repository could compromise its behavioral integrity, this step is controlled.

The requester sends the server an Uninstall message. This message indicates the work to be uninstalled, the version of the Uninstall right being invoked, and the file data for the work (including its size).

The repositories perform the common opening transaction steps.

The requester extracts a copy of the digital certificate for the software. If the certificate cannot be found or the master repository for the certificate is not known to the requester, the transaction ends with an error.

The requester checks whether the software is installed. If the software is not installed, the transaction ends with an error.

The requester decrypts the digital certificate using the public key of the master repository, recording the identity of the supplier and creator, a key for decrypting the software, the compatibility information, and a tamper-checking code. (This step authenticates the certification of the software, including the script for uninstalling it.)

The requester decrypts the software using the key from the certificate and computes a check code on it using a 1-way hash function. If the check-code does not match the tamper-checking code from the certificate, the installation transaction ends with an error. (This step assures that the contents of the software, including the various scripts, have not been tampered with.)

The requester retrieves the instructions in the uninstallation script and follows them. If there is an error in this process (such as insufficient resources), then the transaction ends with an error.

The repositories perform the common closing transaction steps.

Distribution and Use Scenarios

To appreciate the robustness and flexibility of the present invention, various distribution and use scenarios for digital

42

works are illustrated below. These scenarios are meant to be exemplary rather than exhaustive.

Consumers as Unpaid Distributors

In this scenario, a creator distributes copies of his works to various consumers. Each consumer is a potential distributor of the work. If the consumer copies the digital work (usually for a third party), a fee is collected and automatically paid to the creator.

This scenario is a new twist for digital works. It depends on the idea that “manufacturing” is just copying and is essentially free. It also assumes that the consumers as distributors do not require a fee for their time and effort in distributing the work.

This scenario is performed as follows:

A creator creates a digital work. He grants a Copy right with fees paid back to himself. If he does not grant an Embed right, then consumers cannot use the mechanism to act as distributors to cause fees to be paid to themselves on future copies. Of course, they could negotiate side deals or trades to transfer money on their own, outside of the system.

Paid Distributors

In another scenario, every time a copy of a digital work is sold a fee is paid to the creator and also to the immediate distributor.

This scenario does not give special status to any particular distributor. Anyone who sells a document has the right to add a fee to the sale price. The fee for sale could be established by the consumer. It could also be a fixed nominal amount that is contributed to the account of some charity.

This scenario is performed as follows:

A creator creates a digital work. He grants a Copy right with fees to be paid back to himself. He grants an Embed right, so that anyone can add shells to have fees paid to themselves.

A distributor embeds the work in a shell, with fees specified to be paid back to himself. If the distributor is content to receive fees only for copies that he sells himself, he grants an Extract right on the shell.

When a consumer buys a copy from the distributor, fees are paid both to the distributor and to the creator. If he chooses, the consumer can extract the work from the distributor’s shell. He cannot extract it from the creator’s shell. He can add his own shell with fees to be paid to himself.

Licensed Distribution

In this scenario, a creator wants to protect the reputation and value of his work by making certain requirements on its distributors. He issues licenses to distributors that satisfy the requirements, and in turn, promises to reward their efforts by assuring that the work will not be distributed over competing channels. The distributors incur expenses for selecting the digital work, explaining it to buyers, promoting its sale, and possibly for the license itself. The distributor obtains the right to enclose the digital work in a shell, whose function is to permit the attachment of usage fees to be paid to the distributor in addition to the fees to be paid to the creator.

This differs from the previous scenario in that it precludes the typical copy owner from functioning as a distributor, since the consumer lacks a license to copy the document. Thus, a consumer cannot make copies, even for free. All copies must come initially from authorized distributors. This version makes it possible to hold distributors accountable in some way for the sales and support of the work, by controlling the distribution of certificates that enable distributors to legitimately charge fees and copy owners to make copies. Since licenses are themselves digital works, the same mechanisms give the creators control over distributors by charging for licenses and putting time limits on their validity.

US 8,393,007 B2

43

This scenario is performed as follows:

A creator purchases a digital distribution license that he will hand out to his distributors. He puts access requirements (such as a personal license) on the Copy and Transfer rights on the distribution license so that only he can copy or transfer it.

The creator also creates a digital work. He grants an Embed right and a Copy right, both of which require the distribution license to be exercised. He grants a Play right so that the work can be played by anyone. He may optionally add a Transfer or Loan right, so that end consumers can do some non-commercial exchange of the work among friends.

A distributor obtains the distribution license and a number of copies of the work. He makes copies for his customers, using his distribution license.

A customer buys and uses the work. He cannot make new copies because he lacks a distribution license.

Super Distributors

This is a variation on the previous scenarios. A distributor can sell to anyone and anyone can sell additional copies, resulting in fees being paid back to the creator. However, only licensed distributors can add fees to be paid to themselves.

This scenario gives distributors the right to add fees to cover their own advertising and promotional costs, without making them be the sole suppliers. Their customers can also make copies, thus broadening the channel without diminishing their revenues. This is because distributors collect fees from copies of any copies that they originally sold. Only distributors can add fees.

This scenario is performed similarly to the previous ones. There are two key differences. (1) The creator only grants Embed rights for people who have a Distribution license. This is done by putting a requirement for a distributor's license on the Embed right. Consequently, non-distributors cannot add their own fees. (2) The Distributor does not grant Extract rights, so that consumers cannot avoid paying fees to the Distributor if they make subsequent copies. Consequently, all subsequent copies result in fees paid to the Distributor and the Creator.

1-Level Distribution Fees

In this scenario, a distributor gets a fee for any copy he sells directly. However, if one of his customers sells further copies, he gets no further fee for those copies.

This scenario pays a distributor only for use of copies that he actually sold.

This scenario is performed similarly to the previous ones. The key feature is that the distributor creates a shell which specifies fees to be paid to him. He puts Extract rights on the shell. When a consumer buys the work, he can extract away the distributor's shell. Copies made after that will not require fees to be paid to the distributor.

Distribution Trees

In another scenario, distributors sell to other distributors and fees are collected at each level. Every copy sold by any distributor—even several d-blocks down in the chain—results in a fee being paid back to all of the previous distributors.

This scenario is like a chain letter or value chain. Every contributor or distributor along the way obtains fees, and is thereby encouraged to promote the sale of copies of the digital work.

This scenario is performed similarly to the previous ones. The key feature is that the distributor creates a shell which specifies fees to be paid to him. He does not grant Extract rights on the shell. Consequently, all future copies that are made will result in fees paid to him.

44

Weighted Distribution Trees

In this scenario, distributors make money according to a distribution tree. The fee that they make depends on various parameters, such as time since their sale or the number of subsequent distributors.

This is a generalized version of the Distribution Tree scenario, in that it tries to vary the fee to account for the significance of the role of the distributor.

This scenario is similar to the previous one. The difference is that the fee specification on the distributor's shell has provisions for changes in prices. For example, there could be a fee schedule so that copies made after the passage of time will require lower fees to be paid to the distributor. Alternatively, the distributor could employ a "best-price" billing option, using any algorithm he chooses to determine the fee up to the maximum specified in the shell.

Fees for Reuse

In this scenario, a first creator creates a work. It is distributed by a first distributor and purchased by a second creator. The second creator extracts a portion of the work and embeds in it a new work distributed by a second distributor. A consumer buys the new work from the second distributor. The first creator receives fees from every transaction; the first distributor receives fees only for his sale; the second creator and second distributor receive fees for the final sale.

This scenario shows how that flexible automatic arrangements can be set up to create automatic charging systems that mirror current practice. This scenario is analogous to when an author pays a fee to reuse a figure in some paper. In the most common case, a fee is paid to the creator or publisher, but not to the bookstore that sold the book.

The mechanisms for derived works are the same as those for distribution.

Limited Reuse

In this scenario, several first creators create works. A second creator makes a selection of these, publishing a collection made up of the parts together with some new interstitial material. (For example, the digital work could be a selection of music or a selection of readings.) The second creator wants to continue to allow some of the selected works to be extractable, but not the interstitial material.

This scenario deals with fine grained control of the rights and fees for reuse.

This scenario is performed as follows:

The first creators create their original works. If they grant extraction and embedding rights, then the second creator can include them in a larger collected work. The second creator creates the interstitial material. He does grant an Extract right on the interstitial material. He grants Extract rights on a subset of the reused material. A consumer of the collection can only extract portions that have that right. Fees are automatically collected for all parts of the collection.

Commercial Libraries

Commercial libraries buy works with the right to loan. They limit the loan period and charge their own fees for use. This scenario deals with fees for loaning rather than fees for making copies. The fees are collected by the same automatic mechanisms.

The mechanisms are the same as previous scenarios except that the fees are associated with the Loan usage right rather than the Copy usage right.

Demo Versions

A creator believes that if people try his work that they will want to buy it or use it. Consumers of his work can copy the work for free, and play (or execute) a limited version of the work for free, and can play or use the full featured version for

US 8,393,007 B2

45

a fee. This scenario deals with fees for loaning rather than fees for making copies. The fees are collected by the same automatic mechanisms.

This scenario is performed as follows:

The creator creates a digital work and grants various rights and fees. The creator grants Copy and Embed rights without a fee, in order to ensure widespread distribution of the work. Another of the rights is a limited play right with little or no fee attached. For example, this right may be for playing only a portion of the work. The play right can have various restrictions on its use. It could have a ticket that limits the number of times it is used. It could have internal restrictions that limit its functionality. It could have time restrictions that invalidate the right after a period of time or a period of use. Different fees could be associated with other versions of the Play right.

Upgrading a Digital Work with a Vendor

A consumer buys a digital work together with an agreement that he can upgrade to a new version at a later date for a modest fee, much less than the usual purchase price. When the new version becomes available, he goes to a qualified vendor to make the trans action.

This scenario deals with a common situation in computer software. It shows how a purchase may include future “rights.” Two important features of the scenario are that the transaction must take place at a qualified vendor, and that the transaction can be done only once per copy of the digital work purchased.

This scenario is performed as follows:

The creator creates a digital work, an upgrade ticket, and a distribution license. The upgrade ticket uses the a generic ticket agent that comes with repositories. As usual, the distribution license does not have Copy or Transfer rights. He distributes a bundled copies of the work and the ticket to his distributors as well as distribution licenses.

The distributor sells the old bundled work and ticket to customers.

The customer extracts the work and the ticket. He uses the work according to the agreements until the new version becomes available.

When the new work is ready, the creator gives it to distributors. The new work has a free right to copy from a distributor if a ticket is available.

The consumer goes to distributors and arranges to copy the work. The transaction offers the ticket. The distributor’s repository punches the ticket and copies the new version to the consumer’s repository.

The consumer can now use the new version of the work.
Distributed Upgrading of Digital Works

A consumer buys a digital work together with an agreement that he can upgrade to a new version at a later date for a modest fee, much less than the usual purchase price. When the new version becomes available, he goes to anyone who has the upgraded version and makes the transaction.

This scenario is like the previous one in that the transaction can only be done once per copy of the digital work purchased, but the transaction can be accomplished without the need to connect to a licensed vendor.

This scenario is similar to the previous one except that the Copy right on the new work does not require a distribution license. The consumer can upgrade from any repository having the new version. He cannot upgrade more than once because the ticket cannot work after it has been punched. If desired, the repository can record the upgrade transaction by posting a zero cost bill to alert the creator that the upgrade has taken place.

46

Limited Printing

A consumer buys a digital work and wants to make a few ephemeral copies. For example, he may want to print out a paper copy of part of a digital newspaper, or he may want to make a (first generation) analog cassette tape for playing in his car. He buys the digital work together with a ticket required for printing rights.

This scenario is like the common practice of people making cassette tapes to play in their car. If a publisher permits the making of cassette tapes, there is nothing to prevent a consumer from further copying the tapes. However, since the tapes are “analog copies,” there is a noticeable quality loss with subsequent generations. The new contribution of the present invention is the use of tickets in the access controls for the making of the analog copies.

This scenario is performed as follows:

The creator sells a work together with limited printing rights. The printing rights specify the kind of printer (e.g., a kind of cassette recorder or a kind of desktop paper printer) and also the kind of ticket required. The creator either bundles a limited number of tickets or sells them separately. If the tickets use the generic ticket agent, the consumer with the tickets can exercise the right at his convenience.

Demand Publishing

Professors in a business school want to put together course books of readings selected from scenario studies from various sources. The bookstore wants to be able to print the books from digital masters, without negotiating for and waiting for approval of printing of each of the scenarios. The copyright holders of the scenarios want to be sure that they are paid for every copy of their work that is printed.

On many college campuses, the hassle of obtaining copy clearances in a timely way has greatly reduced the viability of preparing course books. Print shops have become much more cautious about copying works in the absence of documented permission.

Demand Publishing is performed as follows: the creator sells a work together with printing rights for a fee. There can be rights to copy (distribute) the work between bookstore repositories, with or without fee. The printing rights specify the kind of printer. Whenever a bookstore prints one of the works (either standalone or embedded in a collection), the fee is credited to the creator automatically. To discourage unauthorized copying of the print outs, it would be possible for the printer to print tracer messages discretely on the pages identifying the printing transaction, the copy number, and any other identifying information. The tracer information could be secretly embedded in the text itself (encoded in the grey scale) or hidden in some other way.

Metered Use and Multiple Price Packages

A consumer does not know what music to purchase until he decides whether he likes it. He would like to be able to take it home and listen to it, and then decide whether to purchase. Furthermore, he would like the flexibility of paying less if he listens to it very infrequently.

This scenario just uses the capability of the approach to have multiple versions of a right on a digital work. Each version of the right has its own billing scheme. In this scenario, the creator of the work can offer the Copy right without fee, and defer billing to the exercise of the Play right. One version of the play right would allow a limited performance without fee—a right to “demo”. Another version of the right could have a metered rate, of say \$0.25 per hour of play. Another version could have a fee of \$15.00 for the first play, but no fee for further playing. When the consumer exercises a play right, he specifies which version of the right is being selected and is billed accordingly.

US 8,393,007 B2

47

Fees for Font Usage

A designer of type fonts invests several months in the design of special fonts. The most common way of obtaining revenue for this work is to sell copies of the fonts to publishers for unlimited use over unlimited periods of time. A font designer would like to charge a rate that reflects the amount that the font is used.

This scenario is performed as follows: the font designer creates a font as a digital work. He creates versions of the Play right that bill either for metered use or "per-use". Each version of the play right would require that the player (a print layout program) be of an approved category. The font designer assigns appropriate fees to exercise the Copy right. When a publisher client wants to use a font, he includes it as input to a layout program, and is billed automatically for its use. In this way, a publisher who makes little use of a font pays less than one who uses it a lot.

Rational Database Usage Charges

Online information retrieval services typically charge for access in a way that most clients find unpredictable and uncorrelated to value or information use. The fee depends on which databases are open, dial-up connect time, how long the searches require, and which articles are printed out. There are no provisions for extracting articles or photographs, no method for paying to reuse information in new works, no distinction between having the terminal sit idly versus actively searching for data, no distinction between reading articles on the screen and doing nothing, and higher rates per search when the centralized facility is busy and slow servicing other clients. Articles can not be offloaded to the client's machine for off-site search and printing. To offer such billing or the expanded services, the service company would need a secure way to account for and bill for how information is used.

This scenario is performed as follows:

The information service bundles its database as files in a repository. The information services company assigns different fees for different rights on the information files. For example, there could be a fee for copying a search database or a source file and a different fee for printing. These fees would be in addition to fees assigned by the original creator for the services. The fees for using information would be different for using them on the information service company's computers or the client's computers. This billing distinction would be controlled by having different versions of the rights, where the version for use on the service company's computer requires a digital certificate held locally. Fees for copying or printing files would be handled in the usual way, by assigning fees to exercising those rights. The distinction between searching and viewing information would be made by having different "players" for the different functions. This distinction would be maintained on the client's computers as well as the service computers. Articles could be extracted for reuse under the control of Extract and Embed rights. Thus, if a client extracts part of an article or photograph, and then sells copies of a new digital work incorporating it, fees could automatically be collected both by the information service and earlier creators and distributors of the digital work. In this way, the information retrieval service could both offer a wider selection of services and billing that more accurately reflects the client's use of the information.

Print Spooling with Rights

In the simplest scenario, when a user wants to print a digital document he issues a print command to the user interface. If the document has the appropriate rights and the conditions are satisfied, the user agrees to the fee and the document is

48

printed. In other cases, the printer may be on a remote repository and it is convenient to spool the printing to a later time. This leads to several issues. The user requesting the printing wants to be sure that he is not billed for the printing until the document is actually printed. Restated, if he is billed at the time the print job is spooled but the job is canceled before printing is done, he does not want to pay. Another issue is that when spooling is permitted, there are now two times at which rights, conditions and fees could be checked: the time at which a print job is spooled and the time at which a print is made. As with all usage rights, it is possible to have rights that expire and to have rights whose fee depends on various conditions. What is needed is a means to check rights and conditions at the time that printing is actually done.

This scenario is performed as follows: A printing repository is a repository with the usual repository characteristics plus the hardware and software to enable printing. Suppose that a user logs into a home repository and wants to spool print jobs for a digital work at a remote printing repository. The user interface for this could treat this as a request to "spool" prints. Underneath this "spooling" request, however, are standard rights and requests. To support such requests, the creator of the work provides a Copy right, which can be used to copy the work to a printing repository. In the default case, this Copy right would have no fees associated for making the copy. However, the Next-Set-Of-Rights for the copy would include the Print rights, with the usual fees for each variation of printing. This version of the Copy right could be called the "print spooling" version of the Copy right. The user's "spool request" is implemented as a Copy transaction to put a copy of the work on the printing repository, followed by Print transactions to create the prints of the work. In this way, the user is only billed for printing that is actually done. Furthermore, the rights, conditions and fees for printing the work are determined when the work is about to be printed.

Thus, a system for enforcing the usage rights of digital works is disclosed. While the embodiments disclosed herein are preferred, it will be appreciated from this teaching that various alternative, modifications, variations or improvements therein may be made by those skilled in the art, which are intended to be encompassed by the following claims.

APPENDIX A**Glossary****Authorization Repository:**

A special type of repository which provides an authorization service. An authorization may be specified by a usage right. The authorization must be obtained before the right may be exercised.

Billing Clearinghouse:

A financial institution or the like whose purpose is to reconcile billing information received from credit servers. The billing clearinghouse may generate bills to users or alternatively, credit and debit accounts involved in the commercial transactions.

Billing Transactions:

The protocol used by which a repository reports billing information to a credit server.

Clearinghouse Transactions:

The protocol used between a credit server and a clearinghouse.

Composite Digital Work:

A digital work comprised of distinguishable parts. Each of the distinguishable parts is itself a digital work which has usage rights attached.

US 8,393,007 B2

49

Content:

The digital information (i.e. raw bits) representing a digital work.

Copy Owner:

A term which refers to the party who owns a digital work stored in a repository. In the typical case, this party has purchased various rights to the document for printing, viewing, transferring, or other specific uses.

Creator:

A term which refers to a party who produces a digital work.

Credit Server:

A device which collects and reports billing information for a repository. In many implementations, this could be built as part of a repository. It requires a means for periodically communicating with a billing clearinghouse.

Description Tree:

A structure which describes the location of content and the usage rights and usage fees for a digital work. A description tree is comprised of description blocks. Each description block corresponds to a digital work or to an interest (typically a revenue bearing interest) in a digital work.

Digital Work (Work):

Any encapsulated digital information. Such digital information may represent music, a magazine or book, or a multimedia composition. Usage rights and fees are attached to the digital work.

Distributor:

A term which refers to a party who legitimately obtains a copy of a digital work and offers it for sale.

Identification (Digital) Certificate:

A signed digital message that attests to the identity of the possessor. Typically, digital certificates are encrypted in the private key of a well-known master repository.

Master Repository:

A special type of repository which issues identification certificates and distributes lists of repositories whose integrity have been compromised and which should be denied access to digital works (referred to as repository "hotlists").

Public Key Encryption:

An encryption technique used for secure transmission of messages on a communication channel. Key pairs are used for the encryption and decryption of messages. Typically one key is referred to as the public key and the other is the private key. The keys are inverses of each other from the perspective of encryption. Restated, a digital work that is encrypted by one key in the pair can be decrypted only by the other.

Registration Transactions:

The protocol used between repositories to establish a trusted session.

Rendering Repository:

A special type of repository which is typically coupled to a rendering system. The rendering repository will typically be embodied within the secure boundaries of a rendering system.

Rendering System:

The combination of a rendering repository and a rendering device. Examples of a rendering systems include printing systems, display systems, general purpose computer systems, video systems or audio systems.

Repository:

Conceptually a set of functional specifications defining core functionality in the support of usage rights. A repository is a trusted system in that it maintains physical, communications and behavioral integrity.

50

Requester Mode:

A mode of a repository where it is requesting access to a digital work.

Revenue Owners:

5 A term which refers to the parties that maintain an interest in collecting fees for document use or who stand to lose revenue if illegitimate copies of the digital work are made.

Server Mode:

10 A mode of a repository where it is processing an incoming request to access a digital work.

Shell Description Block:

A special type of description block designating an interest in a digital work, but which does not add content. This will typically be added by a distributor of a digital work to add their fees.

Transactions:

A term used to refer to the protocols by which repositories communicate.

Usage Fees:

20 A fee charged to a requester for access to a digital work. Usage fees are specified within the usage rights language.

Usage Rights:

25 A language for defining the manner in which a digital work may be used or distributed, as well as any conditions on which use or distribution is premised.

Usage Transactions:

A set of protocols by which repositories communicate in the exercise of a usage rights. Each usage right has its own transaction steps.

30 What is claimed:

1. A computer-implemented method of distributing digital content to at least one recipient computing device to be rendered by the at least one recipient computing device in accordance with usage rights information, the method comprising:

35 determining, by at least one sending computing device, if the at least one recipient computing device is trusted to receive the digital content from the at least one sending computing device;

40 sending the digital content, by the at least one sending computing device, to the at least one recipient computing device only if the at least one recipient computing device has been determined to be trusted to receive the digital content from the at least one sending computing device; and

45 sending usage rights information indicating how the digital content may be rendered by the at least one recipient computing device, the usage rights information being enforceable by the at least on recipient computing device.

50 2. The method of claim 1, wherein the usage rights information further includes a condition under which the content can be rendered.

3. The method of claim 1, wherein the determination of trust comprises:

55 receiving a request from at least one recipient computing device for an authorization object required to render the digital content; and

60 transmitting the authorization object to the at least one recipient computing device when it is determined that the request should be granted.

4. The method of claim 1, wherein the determination of trust comprises:

65 receiving a registration message from the at least one recipient device, the registration message including an identification certificate of the recipient computing device and a random registration identifier, the identification certificate being certified by a master device;

US 8,393,007 B2

51

validating the authenticity of the at least one recipient device;

exchanging messages including at least one session key with the at least one recipient device, the session key to be used in communications; and

conducting a secure transaction using the session key, wherein the secure transaction includes sending the digital content to the at least one recipient device.

5. The method of claim 1, wherein the validating comprises:

verifying the identification certificate of the at least one recipient device;

generating a message to test the authenticity of the at least one recipient device, the generated message including a nonce;

sending the generated message to the at least one recipient device; and

verifying if the at least one recipient device correctly processed the generated message.

6. A sending apparatus for distributing digital content to at least one recipient computing device to be rendered by the at least one recipient computing device in accordance with usage rights information, the sending apparatus comprising:

one or more processors; and

one or more memories operatively coupled to at least one of the one or more processors and having instructions stored thereon that, when executed by at least one of the one or more processors, cause at least one of the one or more processors to:

determine if the at least one recipient computing device is trusted to receive the digital content from the sending apparatus;

send the digital content, by the sending apparatus, to the at least one recipient computing device only if the at least one recipient computing device has been determined to be trusted to receive the digital content from the sending apparatus; and

send usage rights information indicating how the digital content may be rendered by the at least one recipient computing device, the usage rights information being enforceable by the at least on recipient computing device.

7. The apparatus of claim 6, wherein the usage rights information further includes a condition under which the content can be rendered.

8. The apparatus of claim 6, wherein the determination of trust comprises:

receiving a request from at least one recipient computing device for an authorization object required to render the digital content; and

transmitting the authorization object to the at least one recipient computing device when it is determined that the request should be granted.

9. The apparatus of claim 6, wherein the determination of trust comprises:

receiving a registration message from the at least one recipient device, the registration message including an identification certificate of the recipient computing device and a random registration identifier, the identification certificate being certified by a master device;

validating the authenticity of the at least one recipient device;

exchanging messages including at least one session key with the at least one recipient device, the session key to be used in communications; and

52

conducting a secure transaction using the session key, wherein the secure transaction includes sending the digital content to the at least one recipient device.

10. The apparatus of claim 9, wherein the validating comprises:

verifying the identification certificate of the at least one recipient device;

generating a message to test the authenticity of the at least one recipient device, the generated message including a nonce;

sending the generated message to the at least one recipient device; and

verifying if the at least one recipient device correctly processed the generated message.

11. At least one non-transitory computer-readable medium storing computer-readable instructions that, when executed by at least one sending computing device, cause the at least one sending computing device to:

determine if the at least one recipient computing device is trusted to receive the digital content from the at least one sending computing device;

send the digital content, by the at least one sending computing device, to the at least one recipient computing device only if the at least one recipient computing device has been determined to be trusted to receive the digital content from the at least one sending computing device; and

send usage rights information indicating how the digital content may be rendered by the at least one recipient computing device, the usage rights information being enforceable by the at least on recipient computing device.

12. The at least one non-transitory computer-readable medium of claim 11, wherein the usage rights information further includes a condition under which the content can be rendered.

13. The at least one non-transitory computer-readable medium of claim 11, wherein the determination of trust comprises:

receiving a request from at least one recipient computing device for an authorization object required to render the digital content; and

transmitting the authorization object to the at least one recipient computing device when it is determined that the request should be granted.

14. The at least one non-transitory computer-readable medium of claim 11, wherein the determination of trust comprises:

receiving a registration message from the at least one recipient device, the registration message including an identification certificate of the recipient computing device and a random registration identifier, the identification certificate being certified by a master device;

validating the authenticity of the at least one recipient device;

exchanging messages including at least one session key with the at least one recipient device, the session key to be used in communications; and

conducting a secure transaction using the session key, wherein the secure transaction includes sending the digital content to the at least one recipient device.

15. The at least one non-transitory computer-readable medium of claim 14, wherein the validating comprises:

verifying the identification certificate of the at least one recipient device;

US 8,393,007 B2

53

generating a message to test the authenticity of the at least one recipient device, the generated message including a nonce;
sending the generated message to the at least one recipient device; and

54

verifying if the at least one recipient device correctly processed the generated message.

* * * * *



(12) **United States Patent**
Nguyen et al.

(10) **Patent No.:** **US 8,001,053 B2**
 (45) **Date of Patent:** ***Aug. 16, 2011**

(54) **SYSTEM AND METHOD FOR RIGHTS OFFERING AND GRANTING USING SHARED STATE VARIABLES**

(75) Inventors: **Mai Nguyen**, Buena Park, CA (US); **Xin Wang**, Torrance, CA (US); **Eddie J. Chen**, Rancho Palos Verdes, CA (US); **Bijan Tadayon**, Germantown, MD (US)

(73) Assignee: **ContentGuard Holdings, Inc.**, Wilmington, DE (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 278 days.
 This patent is subject to a terminal disclaimer.

(21) Appl. No.: **10/956,070**

(22) Filed: **Oct. 4, 2004**

(65) **Prior Publication Data**
 US 2005/0137984 A1 Jun. 23, 2005

Related U.S. Application Data

(63) Continuation-in-part of application No. 10/162,212, filed on Jun. 5, 2002, which is a continuation-in-part of application No. 09/867,745, filed on May 31, 2001, now Pat. No. 6,754,642.

(60) Provisional application No. 60/296,113, filed on Jun. 7, 2001, provisional application No. 60/331,625, filed on Nov. 20, 2001, provisional application No. 60/331,624, filed on Nov. 20, 2001.

(51) **Int. Cl.**
H04L 9/00 (2006.01)

(52) **U.S. Cl.** **705/57; 705/51; 705/53; 705/59**

(58) **Field of Classification Search** 705/1, 54, 705/57
 See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS
 3,263,158 A 7/1966 Janis
 (Continued)

FOREIGN PATENT DOCUMENTS
 BR 9810967 A 10/2001
 (Continued)

OTHER PUBLICATIONS
 Delaigle, "Digital Watermarking," Spie Conference in Optical Security and Counterfeit Deterrence Techniques, San Jose, CA (Feb. 1996).

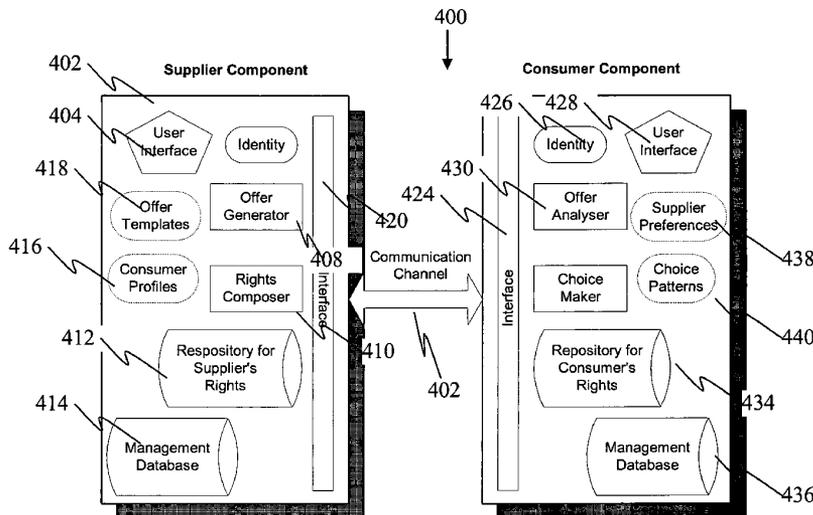
(Continued)

Primary Examiner — Evens J Augustin
 (74) *Attorney, Agent, or Firm* — Reed Smith LLP; Marc S. Kaufman; Stephen M. Hertzler

(57) **ABSTRACT**

A method, system and device for sharing rights adapted to be associated with items, the method and system including generating at least one of usage rights and meta-rights for the items; defining, via the usage rights, a manner of use for the items; and defining, via the meta-rights, a manner of rights transfer for the items. The device including receiving at least one of usage rights and meta-rights for the items; interpreting, via the usage rights, a manner of use for the items; and interpreting, via the meta-rights, a manner of rights transfer for the items. The usage rights or the meta-rights include at least one state variable that is shared by one or more rights.

35 Claims, 17 Drawing Sheets



US 8,001,053 B2

Page 2

U.S. PATENT DOCUMENTS							
3,609,697	A	9/1971	Blevins et al.	5,386,369	A	1/1995	Christiano
3,790,700	A	2/1974	Callais et al.	5,390,297	A	2/1995	Barber et al.
3,798,605	A	3/1974	Feistel	5,394,469	A	2/1995	Nagel et al.
4,159,468	A	6/1979	Barnes et al.	5,410,598	A	4/1995	Shear
4,200,700	A	4/1980	Mäder	5,412,717	A	5/1995	Fischer
4,220,991	A	9/1980	Hamano et al.	5,414,852	A	5/1995	Kramer et al.
4,278,837	A	7/1981	Best	5,428,606	A	6/1995	Moskowitz
4,323,921	A	4/1982	Guillou	5,432,849	A	7/1995	Johnson et al.
4,361,851	A	11/1982	Asip et al.	5,438,508	A	8/1995	Wyman
4,423,287	A	12/1983	Zeidler	5,444,779	A	8/1995	Daniele
4,429,385	A	1/1984	Cichelli et al.	5,453,601	A	9/1995	Rosen
4,442,486	A	4/1984	Mayer	5,455,953	A	10/1995	Russell
4,529,870	A	7/1985	Chaum	5,457,746	A	10/1995	Dolphin
4,558,176	A	12/1985	Arnold et al.	5,473,687	A	12/1995	Lipscomb et al.
4,593,376	A	6/1986	Volk	5,473,692	A	12/1995	Davis
4,614,861	A	9/1986	Pavlov et al.	5,485,577	A	1/1996	Eyer et al.
4,621,321	A	11/1986	Boebert et al.	5,499,298	A	3/1996	Narasimhalu et al.
4,644,493	A	2/1987	Chandra et al.	5,502,766	A	3/1996	Boebert et al.
4,658,093	A	4/1987	Hellman	5,504,814	A	4/1996	Miyahara
4,713,753	A	12/1987	Boebert et al.	5,504,816	A	4/1996	Hamilton et al.
4,736,422	A	4/1988	Mason	5,504,818	A	4/1996	Okano
4,740,890	A	4/1988	William	5,504,837	A	4/1996	Griffeth et al.
4,796,220	A	1/1989	Wolfe	5,509,070	A	4/1996	Schull
4,816,655	A	3/1989	Musyck et al.	5,530,235	A	6/1996	Stefik et al.
4,817,140	A	3/1989	Chandra et al.	5,532,920	A	7/1996	Hartrick et al.
4,827,508	A	5/1989	Shear	5,534,975	A	7/1996	Stefik et al.
4,868,376	A	9/1989	Lessin et al.	5,535,276	A	7/1996	Ganesan
4,888,638	A	12/1989	Bohn	5,539,735	A	7/1996	Moskowitz
4,891,838	A	1/1990	Faber	5,553,143	A	9/1996	Ross et al.
4,924,378	A	5/1990	Hershhey et al.	5,557,678	A	9/1996	Ganesan
4,932,054	A	6/1990	Chou et al.	5,563,946	A	10/1996	Cooper et al.
4,937,863	A	6/1990	Robert et al.	5,564,038	A	10/1996	Grantz et al.
4,949,187	A	8/1990	Cohen	5,568,552	A	10/1996	Davis
4,953,209	A	8/1990	Ryder et al.	5,619,570	A	4/1997	Tsutsui
4,961,142	A	10/1990	Elliott et al.	5,621,797	A	4/1997	Rosen
4,975,647	A	12/1990	Downer et al.	5,625,690	A	4/1997	Michel et al.
4,977,594	A	12/1990	Shear	5,629,980	A	5/1997	Stefik et al. 705/54
4,999,806	A	3/1991	Chernow et al.	5,633,932	A	5/1997	Davis et al.
5,010,571	A	4/1991	Katznelson	5,634,012	A	5/1997	Stefik et al.
5,014,234	A	5/1991	Edwards	5,636,346	A	6/1997	Saxe
5,023,907	A	6/1991	Johnson et al.	5,638,443	A	6/1997	Stefik et al.
5,047,928	A	9/1991	Wiedemer	5,638,513	A	6/1997	Ananda
5,050,213	A	9/1991	Shear	5,649,013	A	7/1997	Stuckey et al.
5,052,040	A	9/1991	Preston et al.	5,655,077	A	8/1997	Jones et al.
5,058,164	A	10/1991	Elmer et al.	5,671,412	A	9/1997	Christiano
5,103,476	A	4/1992	Waite et al.	5,708,709	A	1/1998	Rose
5,113,519	A	5/1992	Johnson et al.	5,708,717	A	1/1998	Alasia
5,129,083	A	7/1992	Cutler et al.	5,715,403	A	2/1998	Stefik
5,136,643	A	8/1992	Fischer	5,734,823	A	3/1998	Saigh et al.
5,138,712	A	8/1992	Corbin	5,734,891	A	3/1998	Saigh
5,146,499	A	9/1992	Geffrotin	5,737,413	A	4/1998	Akiyama et al.
5,148,481	A	9/1992	Abraham et al.	5,737,416	A	4/1998	Cooper et al.
5,159,182	A	10/1992	Eisele	5,745,569	A	4/1998	Moskowitz et al.
5,174,641	A	12/1992	Lim	5,745,879	A	4/1998	Wyman
5,183,404	A	2/1993	Aldous et al.	5,748,783	A	5/1998	Rhoads
5,191,193	A	3/1993	Le Roux	5,757,907	A	5/1998	Cooper et al.
5,204,897	A	4/1993	Wyman	5,758,069	A	5/1998	Olsen
5,222,134	A	6/1993	Waite et al.	5,761,686	A	6/1998	Bloomberg
5,235,642	A	8/1993	Wobber et al.	5,764,807	A	6/1998	Pearlman et al.
5,247,575	A	9/1993	Sprague et al.	5,765,152	A	6/1998	Erickson
5,255,106	A	10/1993	Castro	5,768,426	A	6/1998	Rhoads
5,260,999	A	11/1993	Wyman	5,787,172	A	7/1998	Arnold
5,263,157	A	11/1993	Janis	5,790,664	A	8/1998	Coley et al.
5,263,158	A	11/1993	Janis	5,790,677	A	8/1998	Fox et al.
5,276,444	A	1/1994	McNair	5,794,207	A	8/1998	Walker et al.
5,276,735	A	1/1994	Boebert et al.	5,812,664	A	9/1998	Bernobich et al.
5,287,408	A	2/1994	Samson	5,825,876	A	10/1998	Peterson
5,291,596	A	3/1994	Mita	5,825,879	A	10/1998	Davis
5,293,422	A	3/1994	Loiacono	5,825,892	A	10/1998	Braudaway et al.
5,301,231	A	4/1994	Abraham et al.	5,838,792	A	11/1998	Ganesan
5,311,591	A	5/1994	Fischer	5,848,154	A	12/1998	Nishio et al.
5,319,705	A	6/1994	Halter et al.	5,848,378	A	12/1998	Shelton et al.
5,335,275	A	8/1994	Millar et al.	5,850,443	A	12/1998	Van Oorschot et al.
5,337,357	A	8/1994	Chou et al.	5,892,900	A	4/1999	Ginter et al.
5,339,091	A	8/1994	Yamazaki et al.	5,910,987	A	6/1999	Ginter et al.
5,341,429	A	8/1994	Stringer et al.	5,915,019	A	6/1999	Ginter et al.
5,347,579	A	9/1994	Blandford	5,917,912	A	6/1999	Ginter et al.
5,381,526	A	1/1995	Ellson	5,920,861	A	7/1999	Hall et al.
				5,925,127	A	7/1999	Ahmad

US 8,001,053 B2

Page 3

5,933,498 A	8/1999	Schneck et al.	7,136,838 B1 *	11/2006	Peinado et al.	705/59
5,940,504 A	8/1999	Griswold	7,149,722 B1 *	12/2006	Abburi	705/59
5,943,422 A	8/1999	Van Wie et al.	7,181,438 B1 *	2/2007	Szabo	1/1
5,949,876 A	9/1999	Ginter et al.	7,233,948 B1 *	6/2007	Shamoon et al.	1/1
5,982,891 A	11/1999	Ginter et al.	7,319,759 B1 *	1/2008	Peinado et al.	380/277
5,987,134 A	11/1999	Shin et al.	2001/0009026 A1	7/2001	Terao et al.	
5,991,306 A	11/1999	Burns et al.	2001/0011276 A1	8/2001	Durst, Jr. et al.	
5,999,624 A	12/1999	Hopkins	2001/0014206 A1	8/2001	Artigalás et al.	
5,999,949 A	12/1999	Crandall	2001/0032312 A1 *	10/2001	Runje et al.	713/172
6,006,332 A	12/1999	Rabne et al.	2001/0037467 A1	11/2001	O'Toole, Jr. et al.	
6,009,401 A	12/1999	Horstmann	2001/0039659 A1	11/2001	Simmons et al.	
6,020,882 A	2/2000	Kinghorn et al.	2001/0051996 A1 *	12/2001	Cooper et al.	709/217
6,047,067 A	4/2000	Rosen	2002/0001387 A1	1/2002	Dillon	
6,056,786 A	5/2000	Rivera et al.	2002/0019814 A1 *	2/2002	Ganesan	705/59
6,073,234 A	6/2000	Kigo et al.	2002/0035618 A1	3/2002	Mendez et al.	
6,091,777 A	7/2000	Guetz et al.	2002/0044658 A1	4/2002	Wasilewski et al.	
6,112,181 A	8/2000	Shear et al.	2002/0051540 A1 *	5/2002	Glick et al.	380/258
6,112,239 A	8/2000	Kenner et al.	2002/0056118 A1	5/2002	Hunter et al.	
6,115,471 A	9/2000	Oki et al.	2002/0069282 A1	6/2002	Reisman	
6,135,646 A	10/2000	Kahn et al.	2002/0099948 A1	7/2002	Kocher et al.	
6,138,119 A	10/2000	Hall et al.	2002/0127423 A1	9/2002	Kayanakis	
6,141,754 A	10/2000	Choy	2002/0141584 A1 *	10/2002	Razdan et al.	380/203
6,157,719 A	12/2000	Wasilewski et al.	2002/0169974 A1 *	11/2002	McKune	713/200
6,157,721 A	12/2000	Shear et al.	2003/0028488 A1 *	2/2003	Mohammed et al.	705/59
6,169,976 B1	1/2001	Colosso	2003/0066884 A1 *	4/2003	Reddy et al.	235/382.5
6,185,683 B1	2/2001	Ginter et al.	2003/0097567 A1	5/2003	Terao et al.	
6,189,037 B1	2/2001	Adams et al.	2004/0052370 A1	3/2004	Katznelson	
6,189,146 B1	2/2001	Misra et al.	2004/0172552 A1	9/2004	Boyles et al.	
6,209,092 B1	3/2001	Linnartz				
6,216,112 B1	4/2001	Fuller et al.				
6,219,652 B1	4/2001	Carter et al.				
6,226,618 B1 *	5/2001	Downs et al.	EP 0 067 556 B1	12/1982		
6,233,684 B1 *	5/2001	Stefik et al.	EP 0 084 441	7/1983		
6,236,971 B1	5/2001	Stefik et al.	EP 0 180 460	5/1986		
6,237,786 B1	5/2001	Ginter et al.	EP 0 257 585 A2	3/1988		
6,240,185 B1	5/2001	Van Wie et al.	EP 0 262 025 A2	3/1988		
6,253,193 B1	6/2001	Ginter et al.	EP 0 332 304 A2	9/1989		
6,292,569 B1	9/2001	Shear et al.	EP 0 332 304 A3	9/1989		
6,301,660 B1	10/2001	Benson	EP 0 332 707	9/1989		
6,307,939 B1	10/2001	Vigarie	EP 0 393 806 A2	10/1990		
6,327,652 B1	12/2001	England et al.	EP 0 450 841 A2	10/1991		
6,330,670 B1	12/2001	England et al.	EP 0 529 261 A2	3/1993		
6,345,256 B1	2/2002	Milsted et al.	EP 0 613 073 A1	8/1994		
6,353,888 B1	3/2002	Kakehi et al.	EP 0 651 554	5/1995		
6,363,488 B1	3/2002	Ginter et al.	EP 0 668 695	8/1995		
6,389,402 B1	5/2002	Ginter et al.	EP 0 678 836 A1	10/1995		
6,397,333 B1	5/2002	Söhne et al.	EP 0 679 977 A1	11/1995		
6,401,211 B1	6/2002	Brezak, Jr. et al.	EP 0 715 243 A	6/1996		
6,405,369 B1	6/2002	Tsuria	EP 0 715 243 A1	6/1996		
6,424,717 B1	7/2002	Pinder et al.	EP 0 715 244 A	6/1996		
6,424,947 B1	7/2002	Tsuria et al.	EP 0 715 244 A1	6/1996		
6,442,517 B1 *	8/2002	Miller et al.	EP 0 715 245 A1	6/1996		
6,487,659 B1	11/2002	Kigo et al.	EP 0 715 246 A	6/1996		
6,516,052 B2	2/2003	Voudouris	EP 0 725 376	8/1996		
6,516,413 B1	2/2003	Aratani et al.	EP 0 731 404 A1	9/1996		
6,523,745 B1	2/2003	Tamori	EP 0 763 936 A2	3/1997		
6,636,966 B1 *	10/2003	Lee et al.	EP 0 818 748 A2	1/1998		
6,697,944 B1 *	2/2004	Jones et al.	EP 0 840 194 A2	5/1998		
6,772,340 B1 *	8/2004	Peinado et al.	EP 0 892 521 A2	1/1999		
6,775,655 B1 *	8/2004	Peinado et al.	EP 0 934 765 A1	8/1999		
6,796,555 B1	9/2004	Blahut	EP 0 946 022 A2	9/1999		
6,816,596 B1 *	11/2004	Peinado et al.	EP 0 964 572 A1	12/1999		
6,829,708 B1 *	12/2004	Peinado et al.	EP 1 041 823 A2	10/2000		
6,850,252 B1 *	2/2005	Hoffberg	EP 1 103 922 A2	5/2001		
6,885,748 B1 *	4/2005	Wang	GB 1483282	8/1977		
6,947,571 B1 *	9/2005	Rhoads et al.	GB 2022969 A	12/1979		
6,947,910 B2 *	9/2005	Hsu et al.	GB 2 136 175	9/1984		
6,973,444 B1 *	12/2005	Blinn et al.	GB 2 236 604	4/1991		
6,985,588 B1 *	1/2006	Glick et al.	GB 2236604 A	4/1991		
6,993,131 B1 *	1/2006	Meyers	GB 2309364 A	7/1997		
7,010,808 B1 *	3/2006	Leung et al.	GB 2316503 A	2/1998		
7,024,393 B1 *	4/2006	Peinado et al.	GB 2354102 A	3/2001		
7,039,615 B1 *	5/2006	Gajjala et al.	JP 62-241061	10/1987		
7,051,005 B1 *	5/2006	Peinado et al.	JP 64-068835	3/1989		
7,065,507 B2 *	6/2006	Mohammed et al.	JP 3-063717 A	3/1991		
7,068,787 B1 *	6/2006	Ta et al.	JP 04-369068	12/1992		
7,103,574 B1 *	9/2006	Peinado et al.	JP 5-100939	4/1993		
7,120,254 B2 *	10/2006	Glick et al.	JP 5168039 A2	7/1993		
7,134,144 B2 *	11/2006	McKune	JP 05-268415	10/1993		
			JP 6-131371 A	5/1994		

FOREIGN PATENT DOCUMENTS

US 8,001,053 B2

Page 4

JP	06-175794	6/1994	Cox, "Superdistribution" Wired Magazine (Sep. 1994) XP002233405 URL: http://www.wired.com/wired/archive/2.09/superdis_pr.html&gt .
JP	06-215010	8/1994	Dunlop et al, Telecommunications Engineering, pp. 346-352 (1984).
JP	7-36768	2/1995	Elgamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," IEEE Transactions on Information Theory IT-31(4):469-472 (Jul. 1985).
JP	07-084852	3/1995	Gheorghiu et al., "Authorization for Metacomputing Applications" (no date).
JP	07-200317	8/1995	Iannella, ed., Open Digital Rights Language (ODRL), pp. 1-31 (Nov. 21, 2000).
JP	07-244639	9/1995	Kahle, wais.concepts.txt, Wide Area Information Server Concepts, Thinking Machines Version 4, Draft, pp. 1-18 (Nov. 3, 1989).
JP	0 715 241	6/1996	Kahn, "Deposit, Registration and Recordation in an Electronic Copyright Management System," Technical Report, Corporation for National Research Initiatives, Reston, Virginia (Aug. 1992) URL: http://www.cni.org/docs/ima.ip-workshop/kahn.html .
JP	11031130 A2	2/1999	Kahn et al, "The Digital Library Project, vol. 1: The World of Knowbots (DRAFT), An Open Architecture for a Digital Library System and a Plan for its Development," Corporation for National Research Initiatives, pp. 1-48 (Mar. 1988).
JP	11032037 A2	2/1999	Kohl et al, Network Working Group Request for Comments: 1510, pp. 1-112 (Sep. 1993).
JP	11205306 A2	7/1999	Lee et al, CDMA Systems Engineering Handbook (1998) [excerpts but not all pages numbered].
JP	11215121 A2	8/1999	Mambo et al, "Protection of Data and Delegated Keys in Digital Distribution," Information Security and Privacy. Second Australian Conference, ACISP '97 Proceedings, pp. 271-282 (Sydney, NSW, Australia, Jul. 7-9, 1997, 1997 Berlin, Germany, Springer-Verlag, Germany), XP008016393 ISBN: 3-540-63232-8.
JP	2000215165 A2	8/2000	Mambo et al, "Proxy Cryptosystems: Delegation of the Power to Decrypt Ciphertexts," IEICE Trans. Fundamentals vol. E80-A, No. 1:54-63 (Jan. 1997) XP00742245 ISSN: 0916-8508.
JP	2005218143 A2	8/2005	Microsoft Word, Users Guide, Version 6.0, pp. 487-489, 549-555, 560-564, 572-575, 599-613, 616-631 (1993).
JP	2005253109 A2	9/2005	Ojanperä and Prasad, eds., Wideband CDMA for Third Generation Mobile Communications (1998) [excerpts but not all pages numbered].
JP	2006180562 A2	7/2006	Perritt, "Knowbots, Permissions Headers and Contract Law," Paper for the Conference on Technological Strategies for Protecting Intellectual Property in the Networked Multimedia Environment, pp. 1-22 (Apr. 2-3, 1993 with revisions of Apr. 30, 1993).
WO	WO 83/04461 A1	12/1983	Raggett, (Hewlett Packard), "HTML+(Hypertext markup language)," pp. 1-31 (Jul. 12, 1993) URL: http://citeseer.ist.psu.edu/cor-rect/340709 .
WO	WO 92/20022	11/1992	Samuelson et al, "Intellectual Property Rights for Digital Library and Hypertext Publishing Systems: An Analysis of Xanadu," Hypertext '91 Proceedings, pp. 39-50 (Dec. 1991).
WO	WO 92/20022 A1	11/1992	No Author, "Softlock Services Introduces . . . Softlock Services" Press Release (Jan. 28, 1994).
WO	WO 93/01550	1/1993	No Author, "Appendix III—Compatibility with HTML," No Title, pp. 30-31 (no date).
WO	WO 93/01550 A1	1/1993	No Editor, No Title, Dictionary pages, pp. 469-472, 593-594 (no date).
WO	WO 93/11480 A1	6/1993	Benoit, Digital Television MPEG-1, MPEG-2 and Principles of the DVB System, pp. 75-80, 116-121 (no date).
WO	WO 94/01821	1/1994	Benoit, Digital Television MPEG-1, MPEG-2 and Principles of the DVB System, 2 nd edition, pp. 74-80 (no date).
WO	WO 94/03003 A1	2/1994	AH Digital Audio and Video Series, "DTV Receivers and Measurements," Understanding Digital Terrestrial Broadcasting, pp. 159-164 (no date).
WO	WO 96/13814 A1	5/1996	O'Driscoll, The Essential Guide to Digital Set—Top Boxes and Interactive TV, pp. 6-24 (no date).
WO	WO 96/24092	8/1996	Ius Mentis, "The ElGamal Public Key System," pp. 1-2 (Oct. 1, 2005) online at http://www.iusmentis.com/technology/encryption/elgamal/ .
WO	WO 96/24092 A2	8/1996	Schneier, "Crypto Bibliography," Index of Crypto Papers Available Online, pp. 1-2 (online) (no date).
WO	WO 96/27155 A2	9/1996	No Author, No Title, pp. 344-355 (no date).
WO	WO 97/25800 A1	7/1997	No Author, "Part Four Networks," No Title, pp. 639-714 (no date).
WO	WO 97/37492 A1	10/1997	Microsoft Word User's Guide, pp. 773-774, 315-316, 487-489, 561-564, 744, 624-633 (1993).
WO	WO 97/41661 A2	11/1997	
WO	WO 97/43761 A2	11/1997	
WO	WO 97/48203	12/1997	
WO	WO 98/09209 A1	3/1998	
WO	WO 98/10561 A1	3/1998	
WO	WO 98/11690	3/1998	
WO	WO 98/11690 A1	3/1998	
WO	WO 98/19431 A1	5/1998	
WO	WO 98/42098	9/1998	
WO	WO 98/43426 A1	10/1998	
WO	WO 98/45768 A1	10/1998	
WO	WO 99/24928 A2	5/1999	
WO	WO 99/34553 A1	7/1999	
WO	WO 99/35782 A1	7/1999	
WO	WO 99/48296 A1	9/1999	
WO	WO 99/49615	9/1999	
WO	WO 99/60461 A1	11/1999	
WO	WO 99/60750 A2	11/1999	
WO	WO 00/04727 A2	1/2000	
WO	WO 00/05898 A2	2/2000	
WO	WO 00/08909 A	2/2000	
WO	WO 00/46994 A1	8/2000	
WO	WO 00/59152	10/2000	
WO	WO 00/59152 A2	10/2000	
WO	WO 00/62260 A1	10/2000	
WO	WO 00/72118 A1	11/2000	
WO	WO 00/73922 A2	12/2000	
WO	WO 01/03044 A1	1/2001	
WO	WO 01 13198 A	1/2001	
WO	WO 01/24530 A2	4/2001	
WO	WO 01/37209 A1	5/2001	
WO	WO 01/63528	8/2001	
WO	WO 2004/034223 A2	4/2004	
WO	WO 2004/103843	12/2004	

OTHER PUBLICATIONS

Perritt, "Technologies Strategies for Protecting Intellectual Property in the Networked Multimedia Environment," Knowbots, Permissions Headers and Contract Law (Apr. 2-3, 1993).

Blaze et al, "Divertible Protocols and Atomic Proxy Cryptography" 1998 Advances in Cryptography—Euro Crypt International Conference on the Theory and Application of Crypto Techniques, Springer Verlag, DE.

Blaze et al, "Atomic Proxy Cryptography" DRAFT (Online) (Nov. 2, 1997) XP002239619 Retrieved from the Internet.

No Author, "Capability- and Object-Based Systems Concepts," Capability-Based Computer Systems, pp. 1-19 (no date).

US 8,001,053 B2

Page 5

- No Author, "What is the ElGarnal Cryptosystem," p. 1 (Nov. 27, 2006) online at <http://www.x5.net/faqs/crypto/q29.html>.
- Johnson et al., "A Secure Distributed Capability Based System," ACM, pp. 392-402 (1985).
- Wikipedia, "El Gamal Encryption," pp. 1-3 (last modified Nov. 2, 2006) online at http://en.wikipedia.org/wiki/ElGamal_encryption.
- Blaze, "Atomic Proxy Cryptography," p. 1 Abstract (Oct. 20, 1998).
- Blaze, "Matt Blaze's Technical Papers," pp. 1-6 (last updated Aug. 6, 2006)].
- Online Search Results for "inverted file", "inverted index" from www.techweb.com, www.cryer.co.uk, computing-dictionary.thefreedictionary.com, www.nist.gov, en.wikipedia.org, www.cni.org, www.tiscali.co.uk (Jul. 15-16, 2006).
- Corporation for National Research Initiatives, "Digital Object Architecture Project", <http://www.nnri.reston.va.us/dao.html> (updated Nov. 28, 2006).
- Stefik, Summary and Analysis of A13 (Kahn, Robert E and Vinton G Cerf, "The Digital Library Project, vol. 1: The World of Knowbots (DRAFT), An Open Architecture for a Digital Library System and a Plan for its Development," Corporation for National Research Initiatives (Mar. 1988)), pp. 1-25 (May 30, 2007).
- Johnson et al., "A Secure Distributed Capability Based System," Proceedings of the 1985 ACM Annual Conference on the Range of Computing: MID-80's Perspective: MID-80's Perspective *Association for Computing Machinery* pp. 392-402 (1985).
- "National Semiconductor and EPR Partner for Information Metering/Data Security Cards" Mar. 4, 1994, Press Release from Electronic Publishing Resources, Inc.
- Weber, R., "Digital Rights Management Technology" Oct. 1995.
- Flasche, U. et al., "Decentralized Processing of Documents", pp. 119-131, 1986, *Comput. & Graphics*, vol. 10, No. 2.
- Mori, R. et al., "Superdistribution: The Concept and the Architecture", pp. 1133-1146, 1990. *The Transactions of the IEICE*, Vo. E 73, No. 7, Tokyo, JP.
- Weber, R., "Metering Technologies for Digital Intellectual Property", pp. 1-29, Oct. 1994, A Report to the International Federation of Reproduction Rights Organizations.
- Clark, P.C. et al., "Bits: A Smartcard protected Operating System", pp. 66-70 and 94, Nov. 1994, *Communications of the ACM*, vol. 37, No. 11.
- Ross, P.E., "Data Guard", pp. 101, Jun. 6, 1994, *Forbes*.
- Saigh, W.K., "Knowledge is Sacred", 1992, Video Pocket/Page Reader Systems, Ltd.
- Kahn, R.E., "Deposit, Registration and Recordation in an Electronic Copyright Management System", pp. 1-19, Aug. 1992, Corporation for National Research Initiatives, Virginia.
- Hilts, P. et al., "Books While U Wait", pp. 48-50, Jan. 3, 1994, *Publishers Weekly*.
- Strattner, A., "Cash Register on a Chip may Revolutionize Software Pricing and Distribution; Wave Systems Corp.", pp. 1-3, Apr. 1994, *Computer Shopper*, vol. 14, No. 4, ISSN 0886-0556.
- O'Conner, M., "New Distribution Option for Electronic Publishers; iOpener Data Encryption and Metering System for CD-ROM use; Column", pp. 1-6, Mar. 1994, *CD-ROM Professional*, vol. 7, No. 2, ISSN: 1409-0833.
- Willett, S., "Metered PCs: Is Your System Watching You? Wave System beta tests new technology", pp. 84, May 2, 1994, *InfoWorld*.
- Linn, R., "Copyright and Information Services in the Context of the National Research and Education Network", pp. 9-20, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1, Issue 1.
- Perritt, Jr., H., "Permission Headers and Contract Law", pp. 27-48, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1, Issue 1.
- Upthegrove, L., "Intellectual Property Header Descriptors: A Dynamic Approach", pp. 63-66, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1, Issue 1.
- Sirbu, M., "Internet Billing Service Design and prototype Implementation", pp. 67-80, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1, Issue 1.
- Simmell, S. et al., "Metering and Licensing of Resources: Kala's General Purpose Approach", pp. 81-110, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1, Issue 1.
- Kahn, R., "Deposit, Registration and Recordation in an Electronic Copyright Management System", pp. 111-120, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1, Issue 1.
- Tygar, J. et al., "Dyad: A System for Using Physically Secure Coprocessors", pp. 121-152, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1, Issue 1.
- Griswold, G., "A Method for Protecting Copyright on Networks", pp. 169-178, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1, Issue 1.
- Nelson, T., "A Publishing and Royalty Model for Networked Documents", pp. 257-259, Jan. 1994, *IMA Intellectual Property Project Proceedings*, vol. 1, Issue 1.
- Robinson, E., "Redefining Mobile Computing", pp. 238-240, 247-248 and 252, Jul. 1993, *PC Computing*.
- Abadi, M. et al., "Authentication and Delegation with Smart-cards", pp. 1-24, 1990, Research Report DEC Systems Research Center.
- Mark Stefik, "Letting Loose the Light: Igniting Commerce in Electronic Publication", pp. 219-253, 1996, *Internet Dreams: Archetypes, Myths, and Metaphors*, IDSN 0-262-19373-6.
- Mark Stefik, "Letting Loose the Light: Igniting Commerce in Electronic Publication", pp. 2-35, Feb. 8, 1995, *Internet Dreams: Archetypes, Myths and Metaphors*.
- Henry H. Perritt, Jr., "Technological Strategies for Protecting Intellectual Property in the Networked Multimedia Environment", Apr. 2-3, 1993, *Knowbots, Permissions Headers & Contract Law*.
- Perritt, "Technologies Strategies for Protecting IP in the Networked Multimedia Environment", Apr. 2-3, 1993, *Knowbot Permissions*.
- Delaille, "Digital Watermarking", *Spie Conference in Optical Security and Counterfeit Deterrence Techniques*, San Jose, CA Feb. 1996, vol. 2659 pp. 99-110.
- "The C++ Programming Language—Second Edition", Bjarne Stroustrup, Addison-Wesley, ISBN 0-201-53992-6, 1991.

* cited by examiner

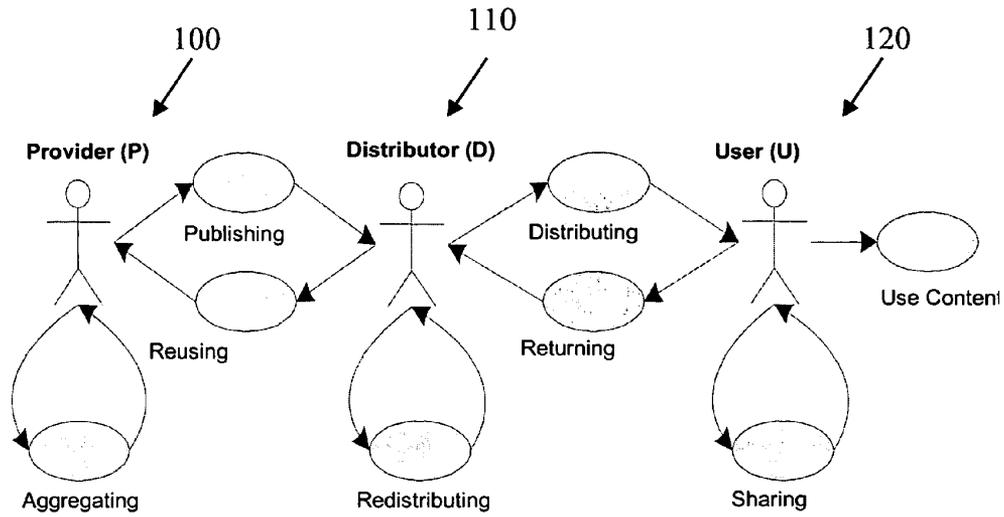


Fig. 1

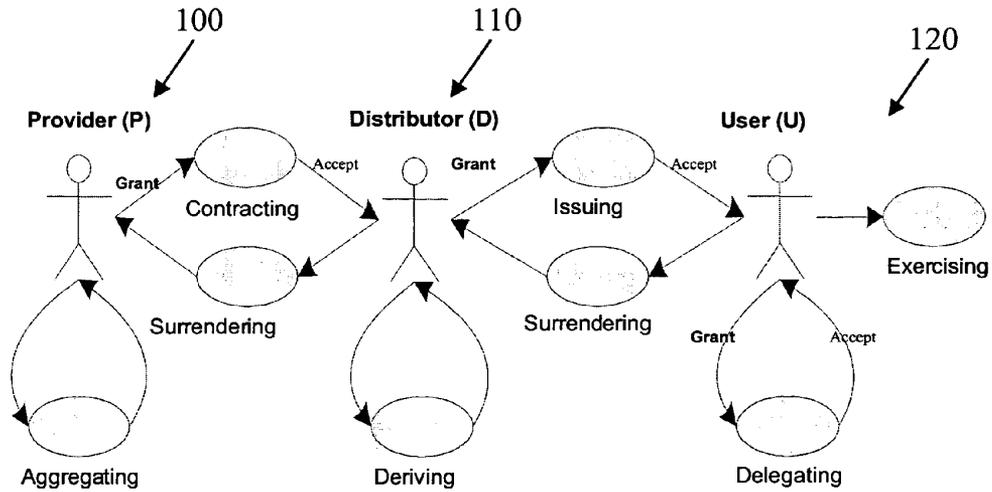


Fig. 2

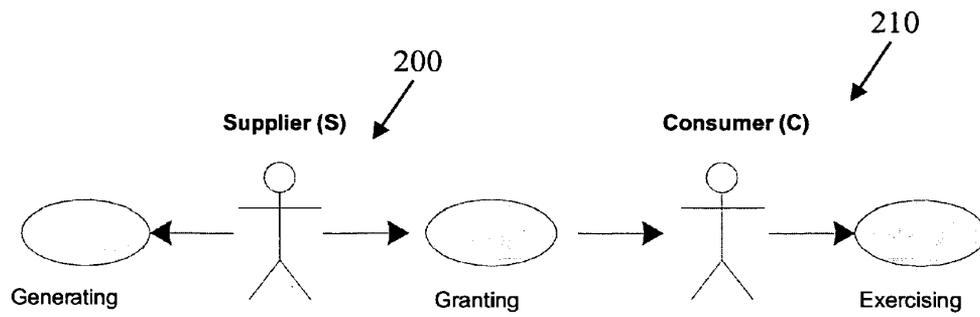


Fig. 3(a)

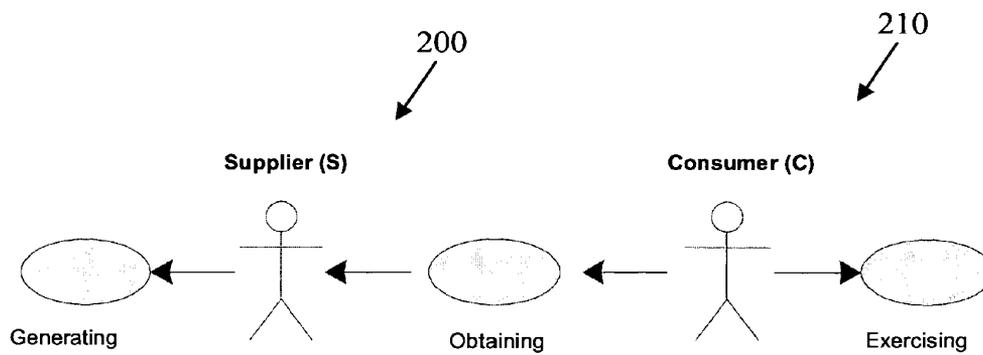


Fig. 3(b)

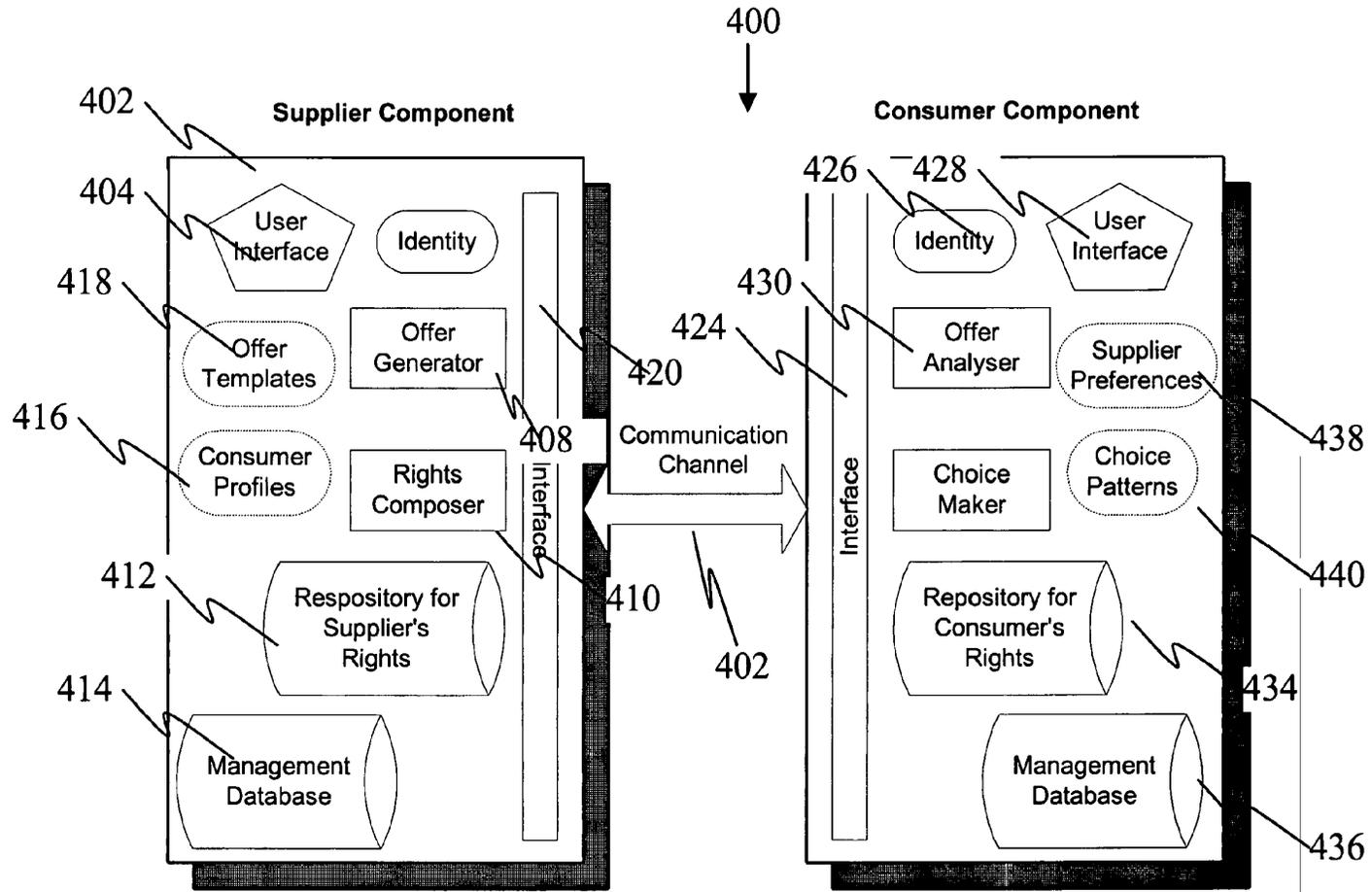


Fig. 4

Appx000356

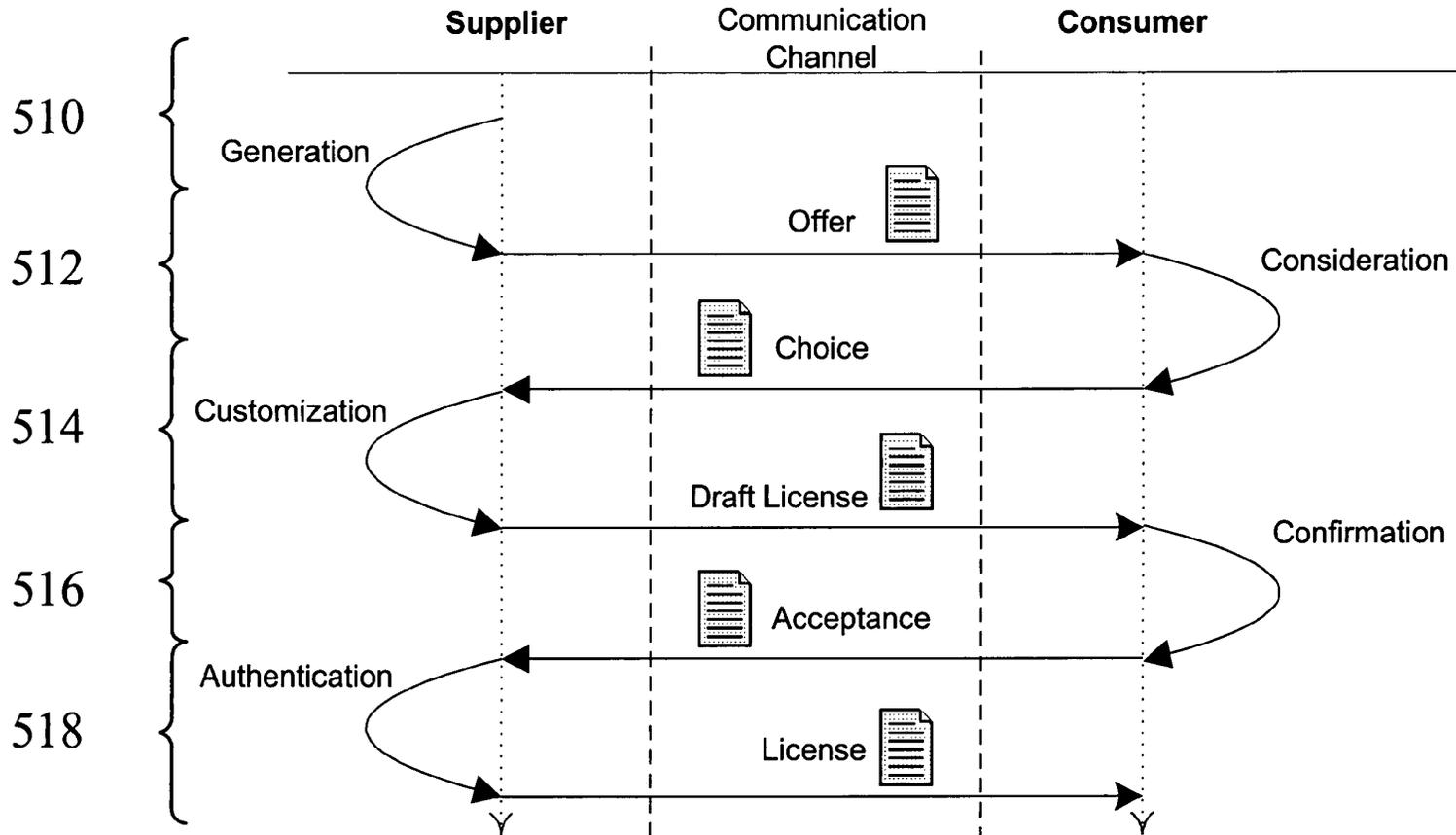


Fig. 5(a)

Appx000357

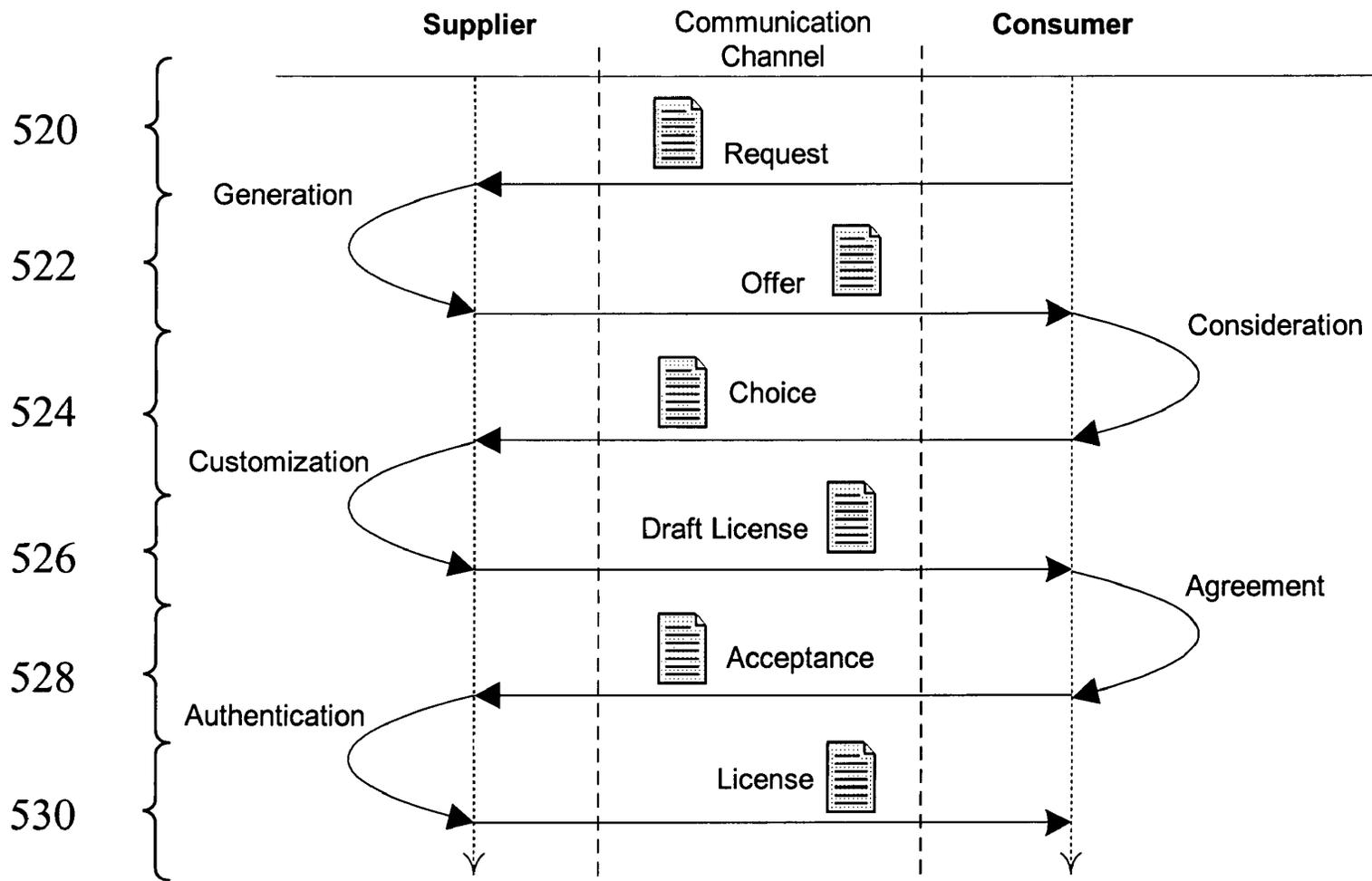


Fig. 5(b)

Appx000358

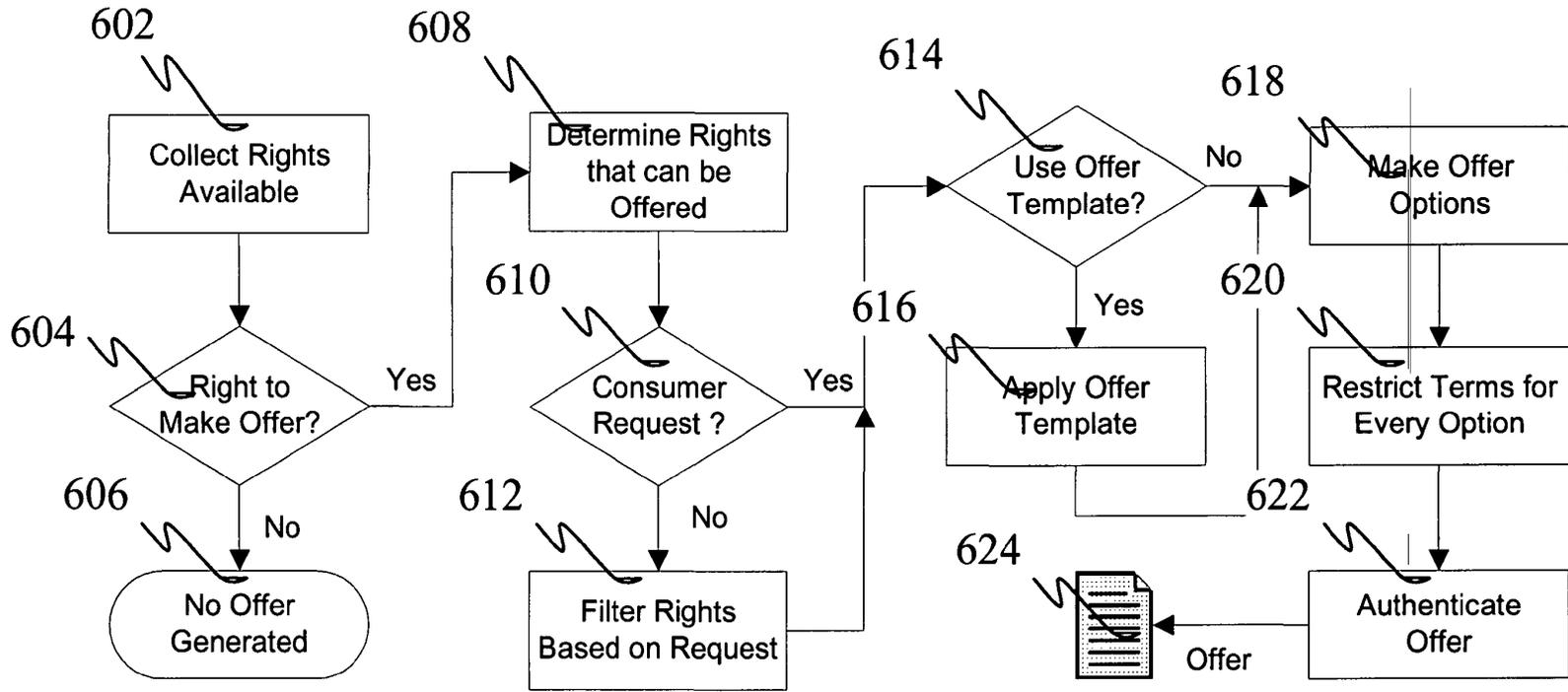


Fig. 6

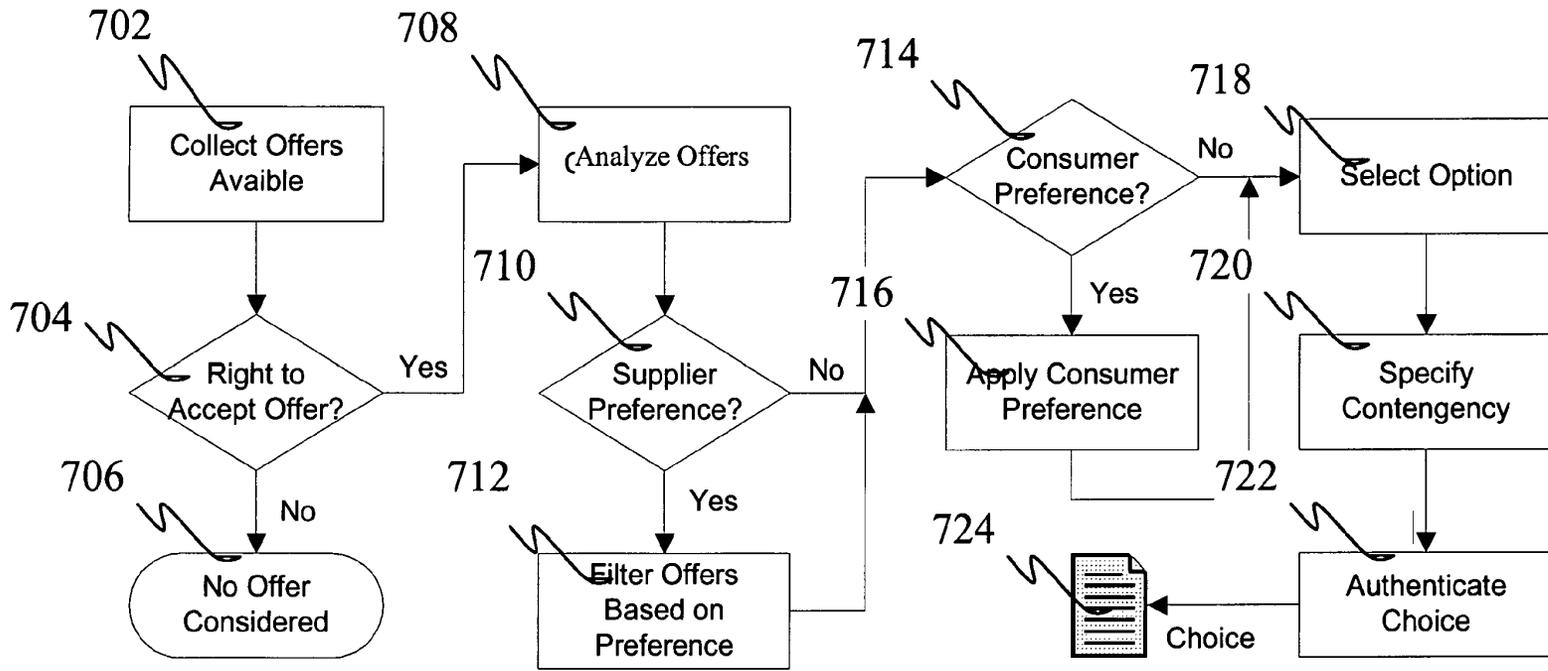


Fig. 7

Appx000360

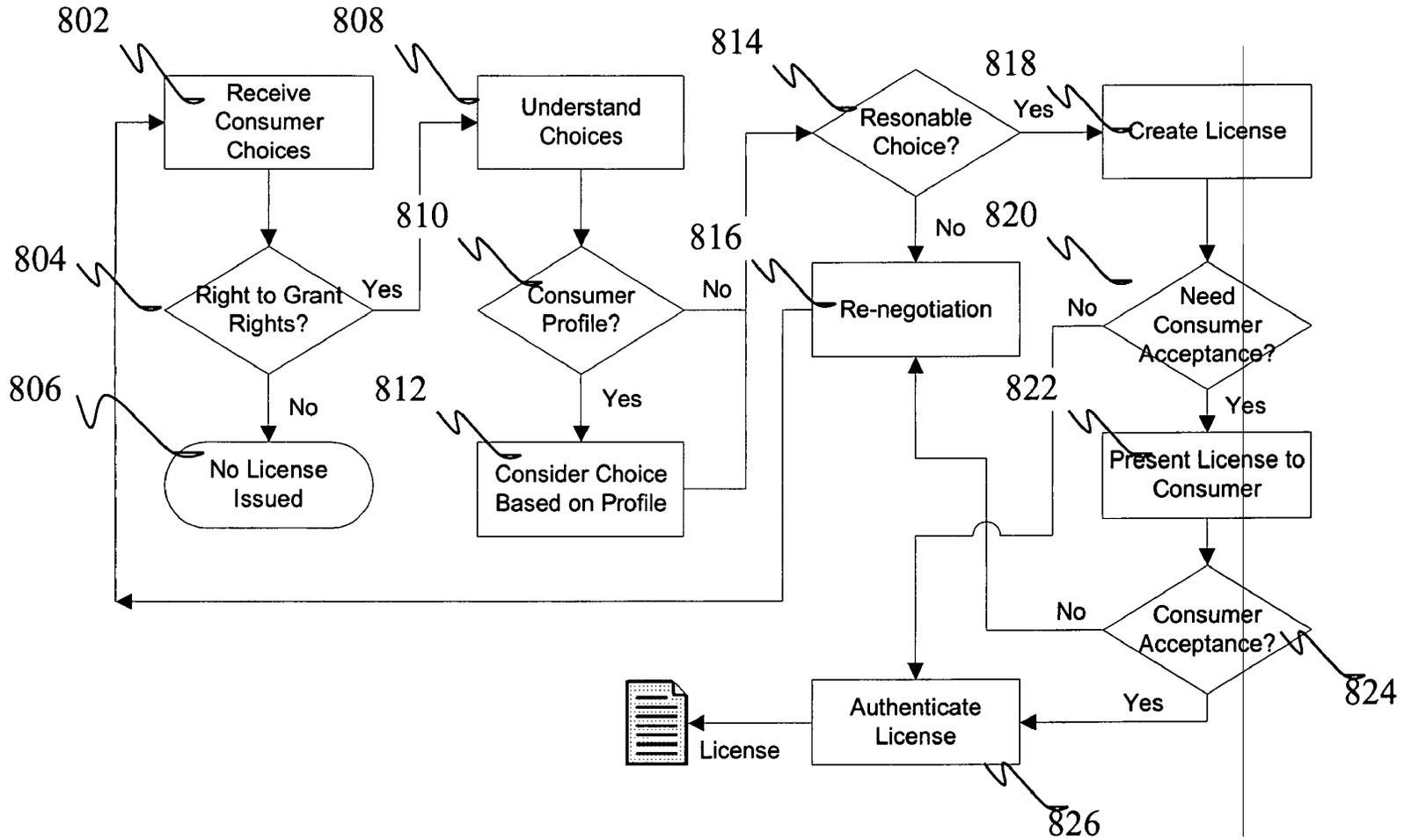


Fig. 8

Appx000361

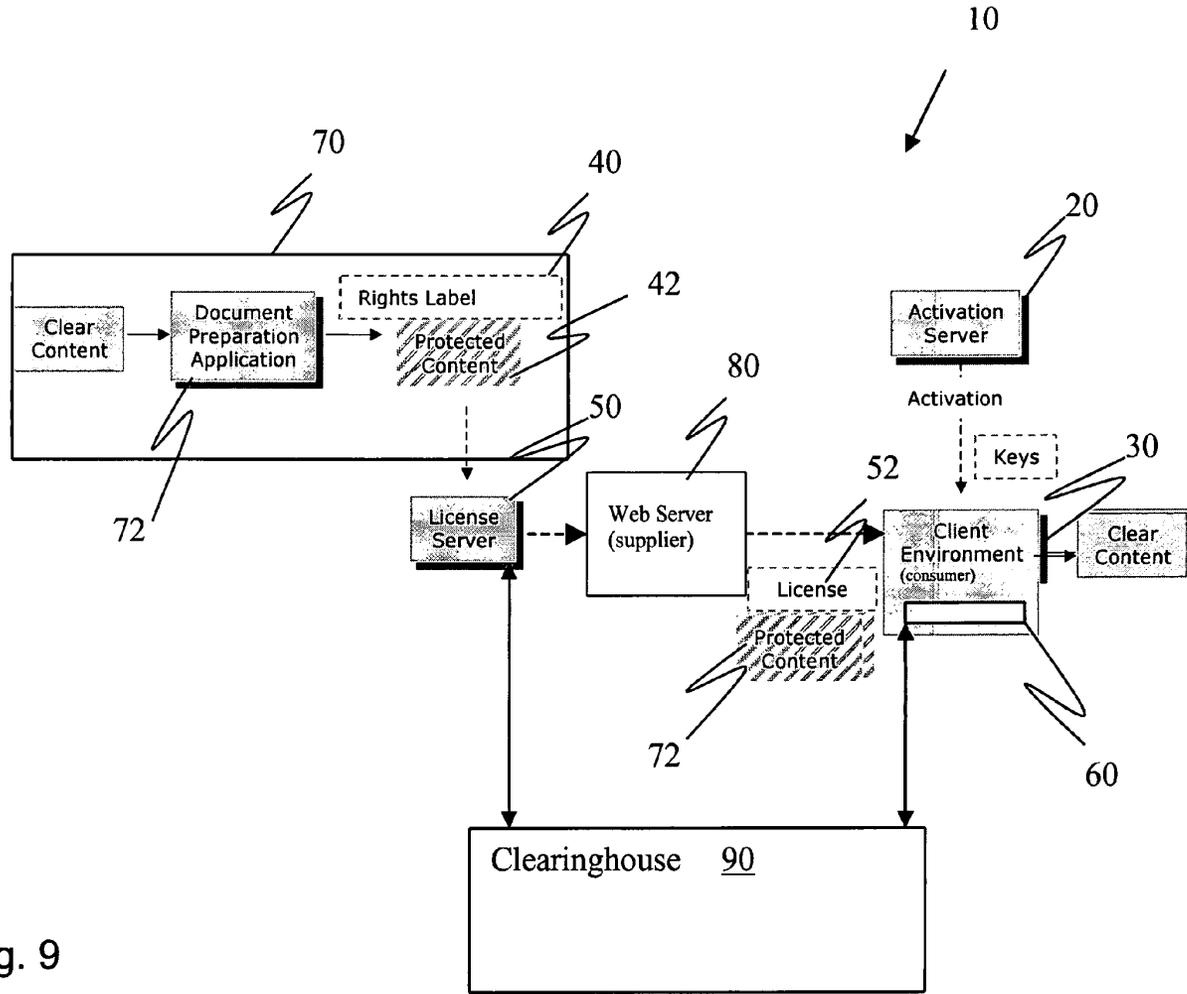


Fig. 9

Appx000362

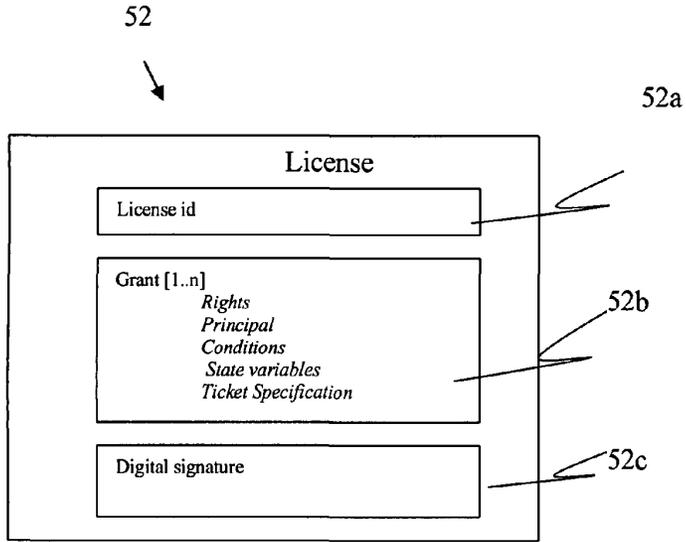


Fig. 10

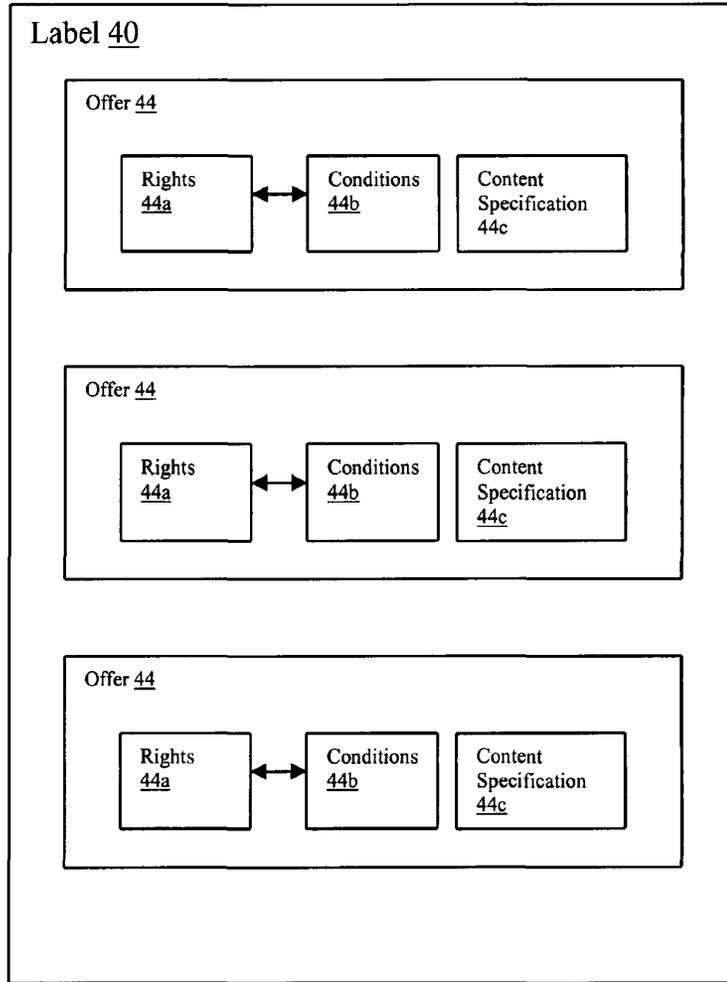


Fig. 11

Appx000363

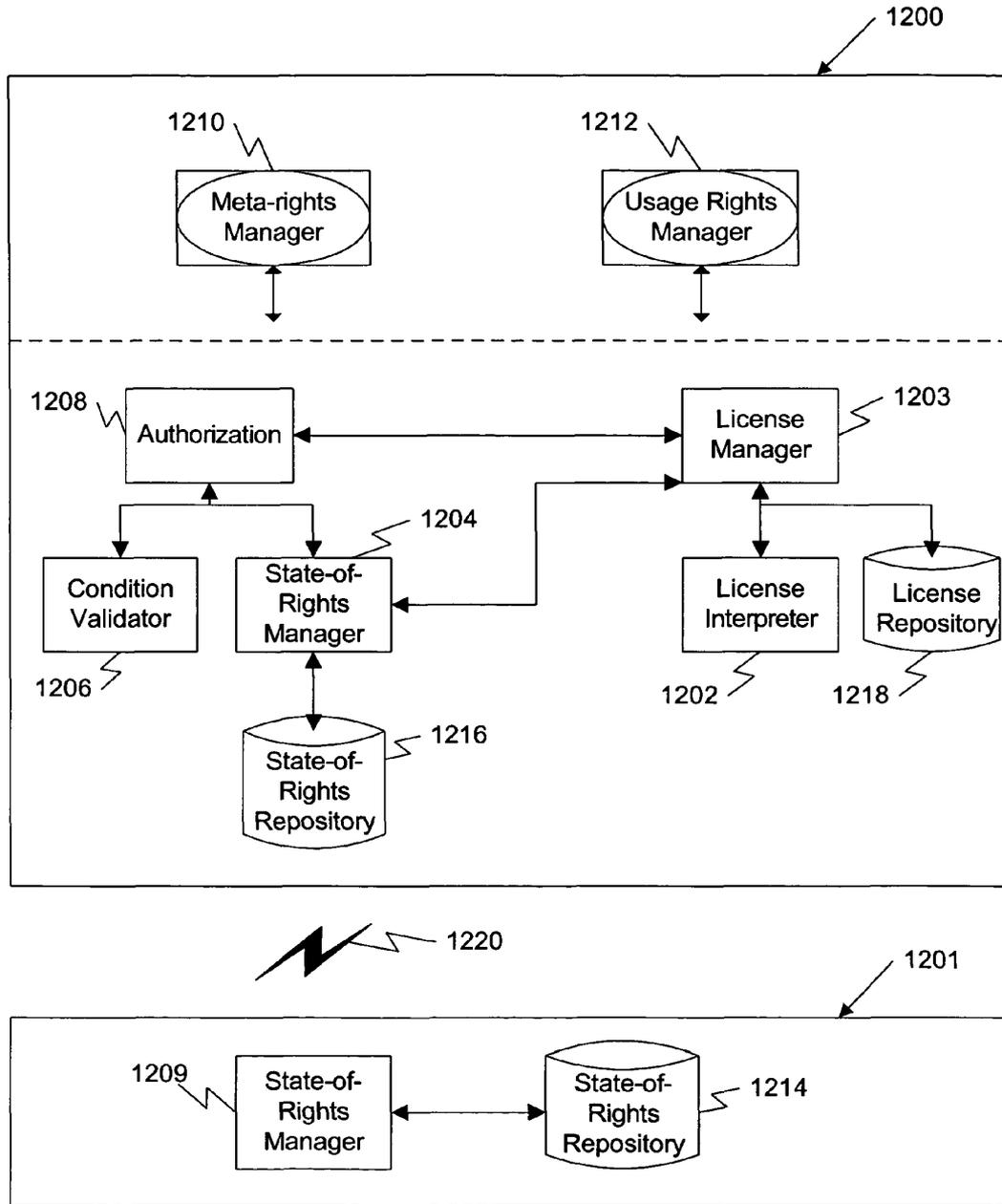
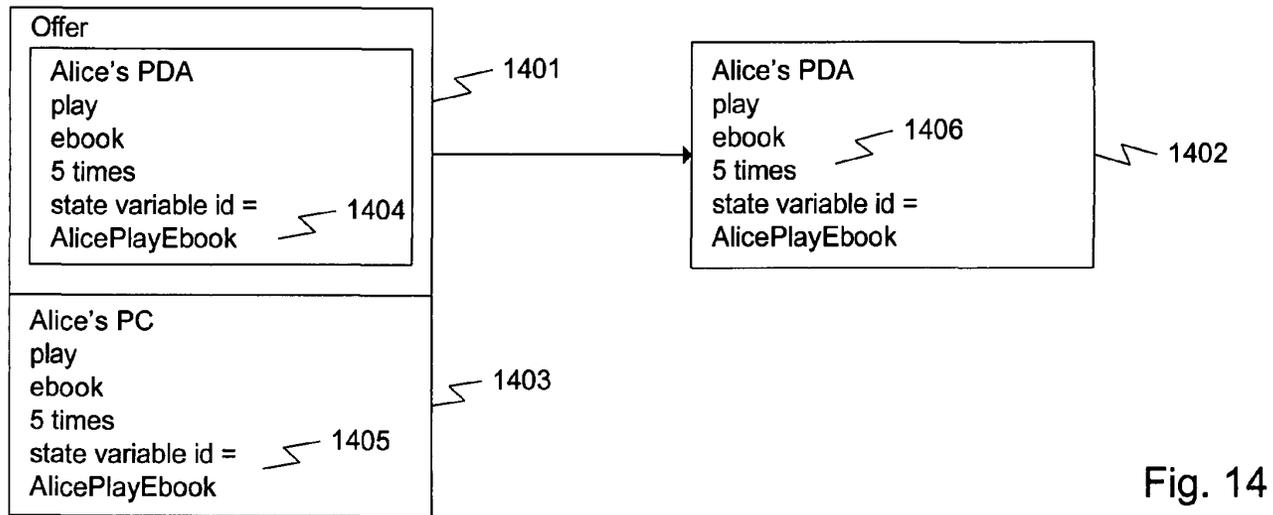
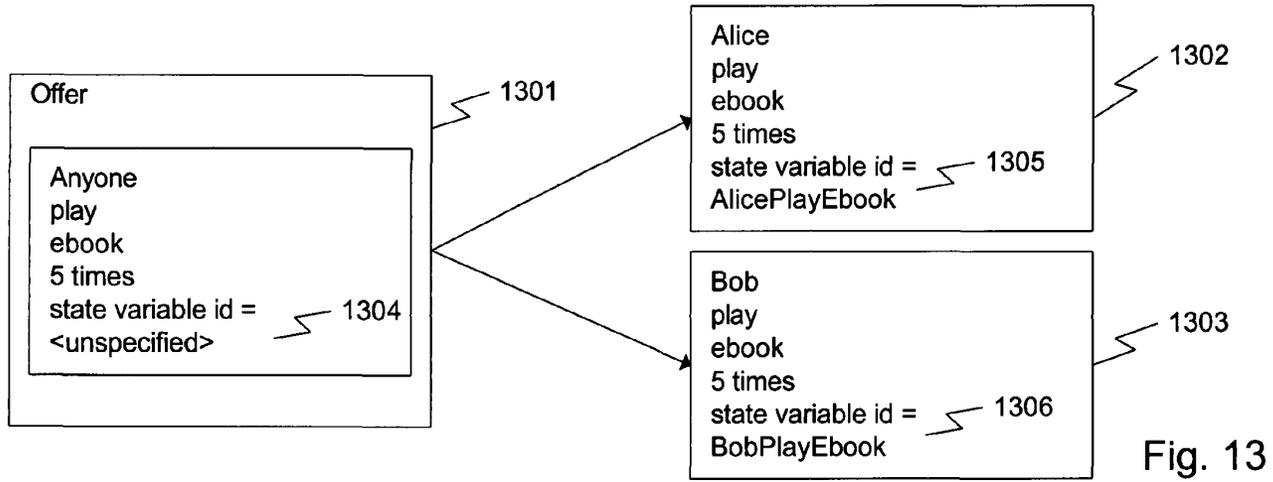


Fig. 12



Appx000365

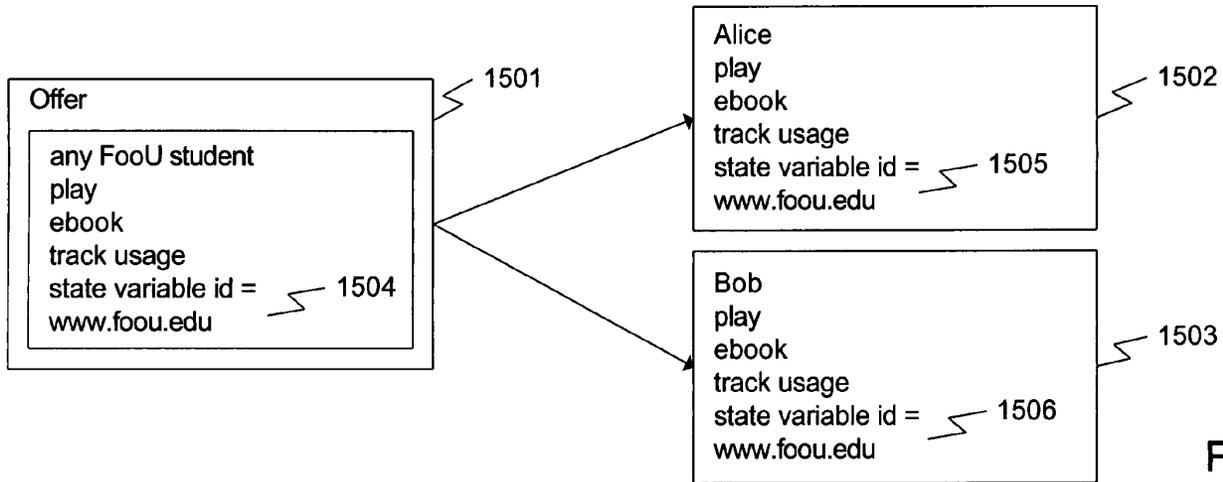


Fig. 15

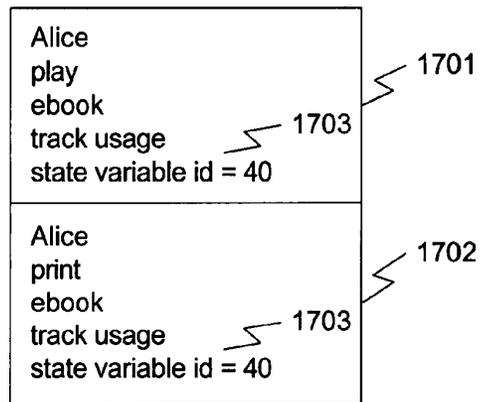


Fig. 17

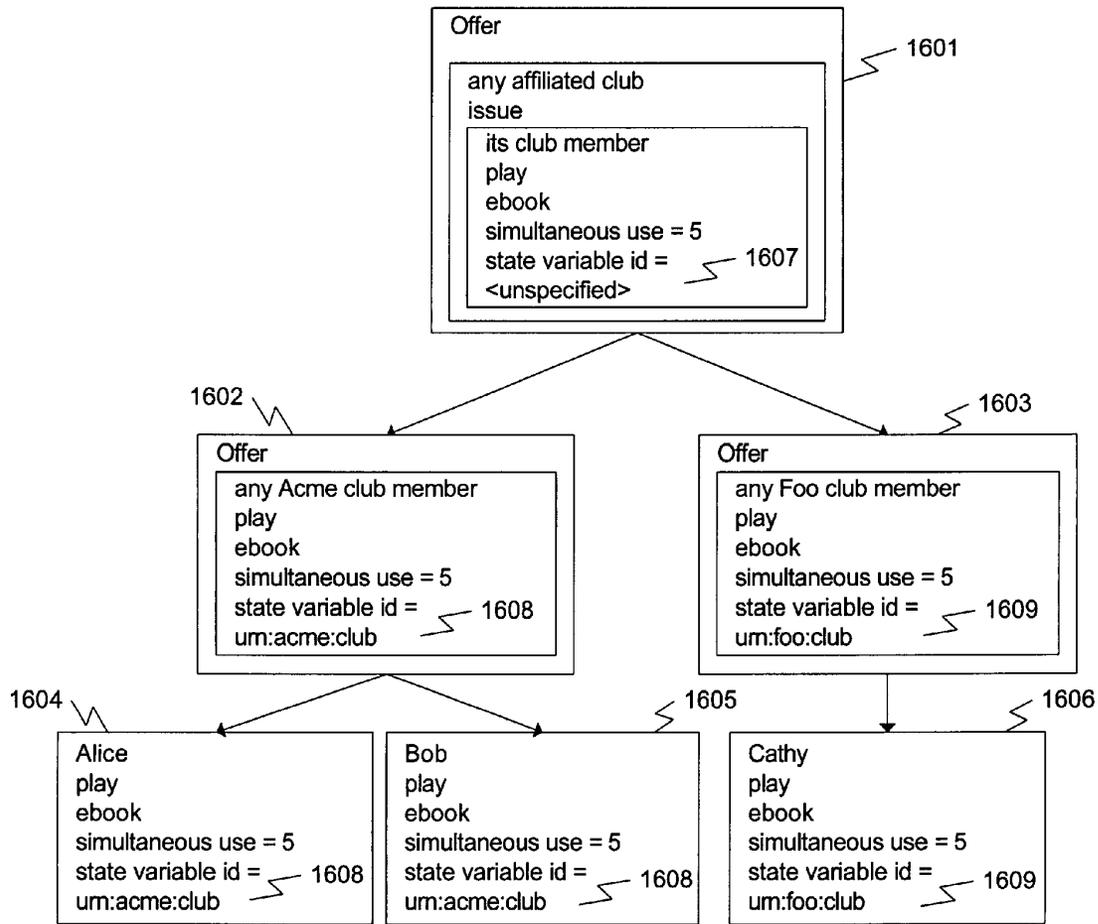


Fig. 16

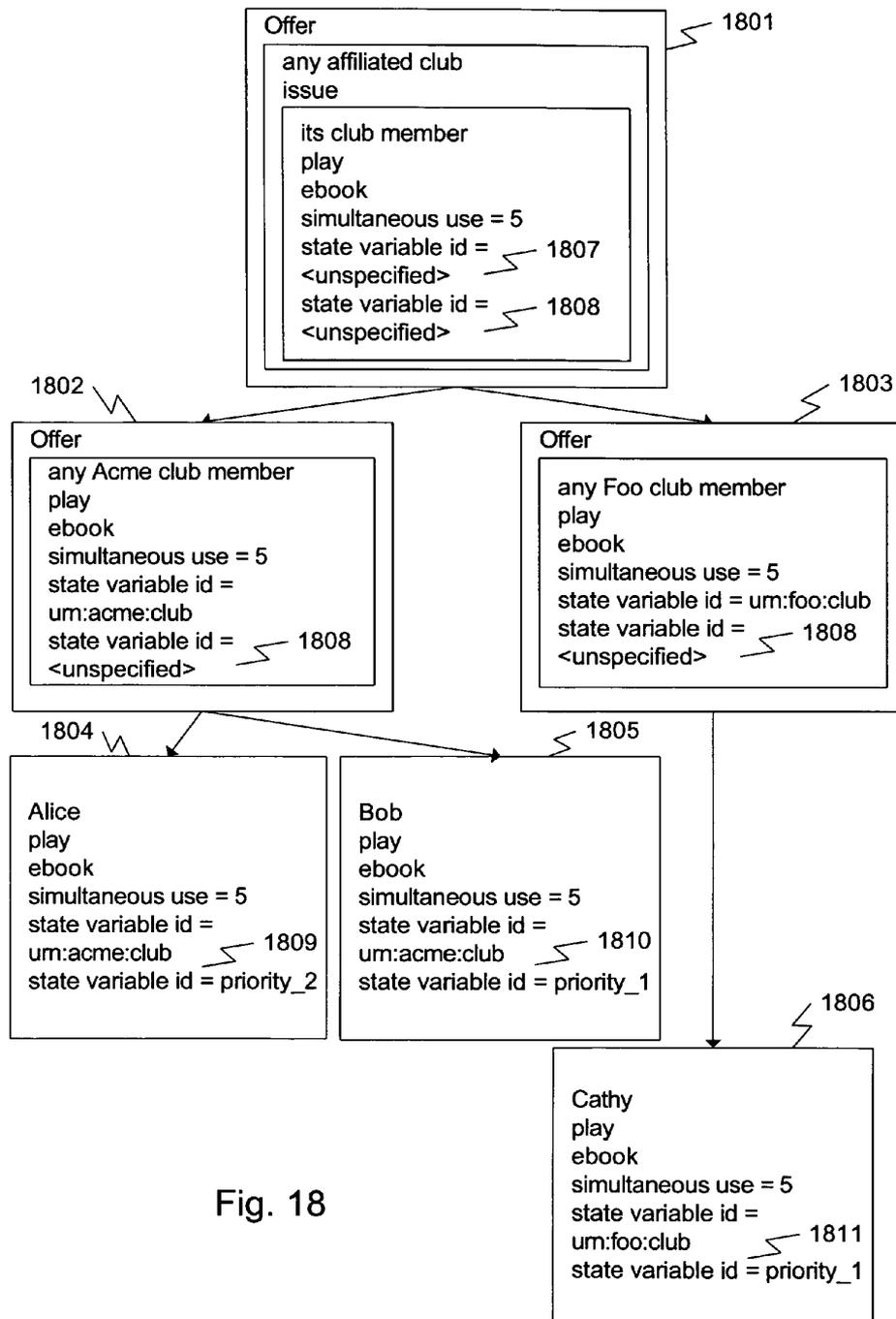


Fig. 18

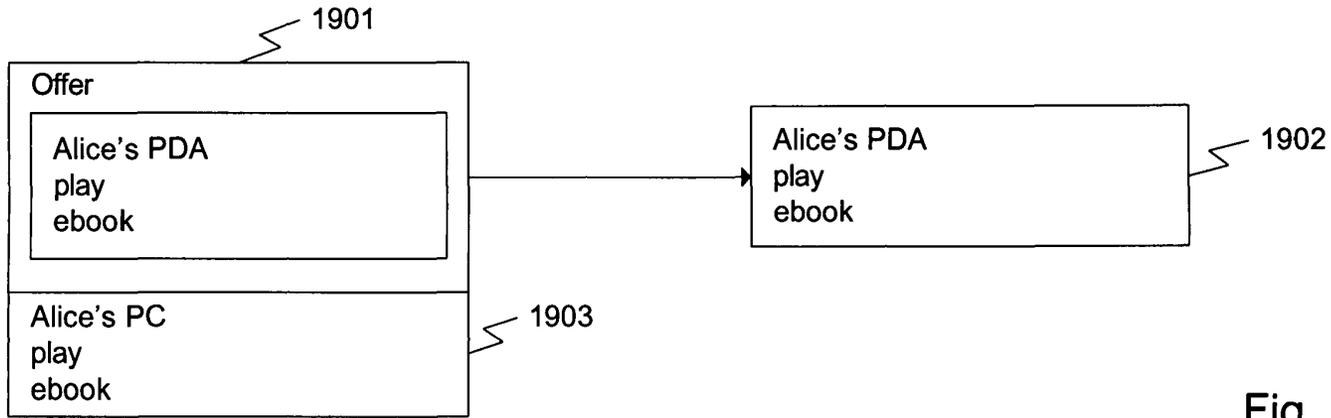


Fig. 19

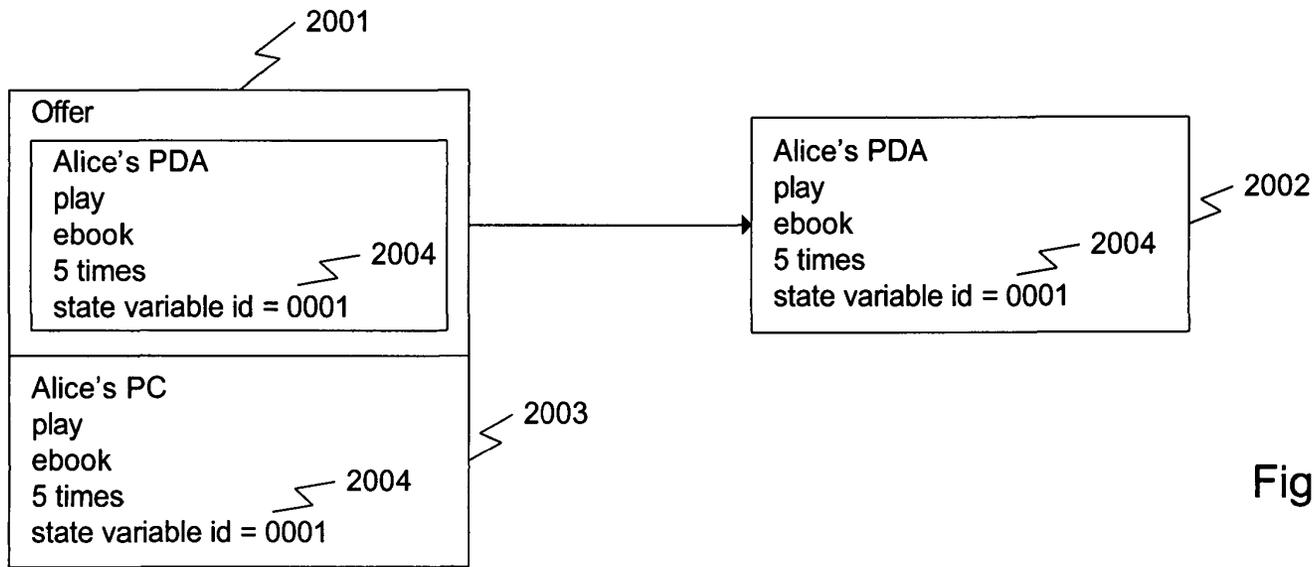


Fig. 20

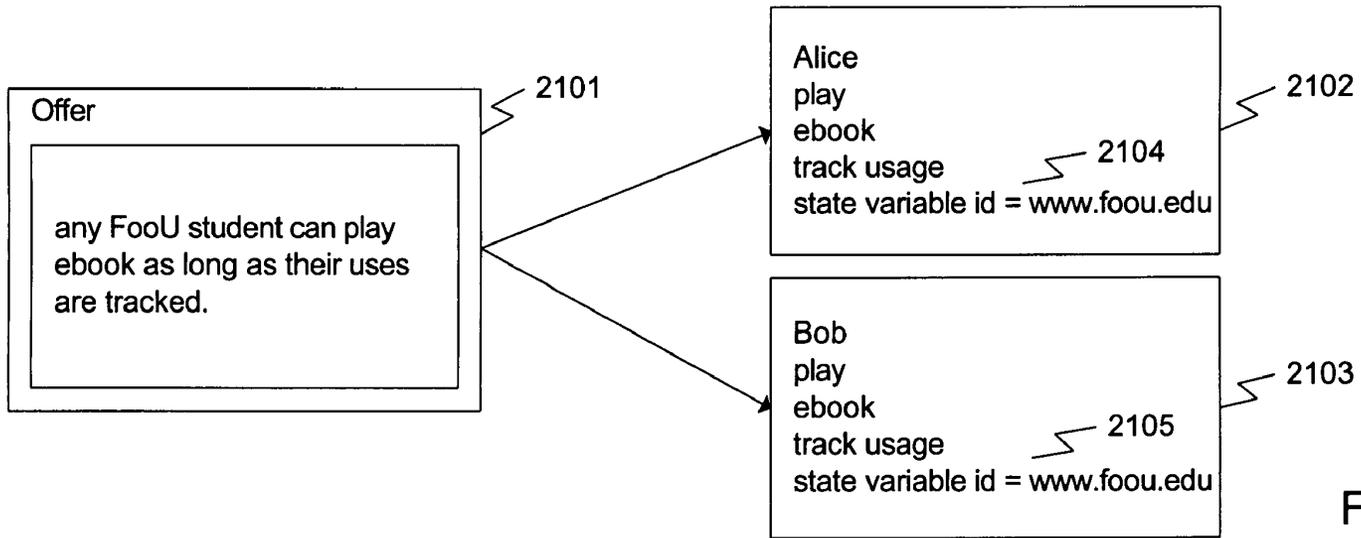


Fig. 21

US 8,001,053 B2

1

**SYSTEM AND METHOD FOR RIGHTS
OFFERING AND GRANTING USING SHARED
STATE VARIABLES**

RELATED APPLICATION DATA

This application is a continuation-in-part application of co-pending application Ser. No. 10/162,212 filed on Jun. 5, 2002, which is a continuation-in-part application of application Ser. No. 09/867,745 filed on May 31, 2001, and which claims benefit from U.S. provisional application Ser. No. 60/296,113, filed in Jun. 7, 2001, U.S. provisional application, Ser. No. 60/331,625 filed in Nov. 20, 2001, and U.S. provisional application Ser. No. 60/331,624 filed on Nov. 20, 2001, the entire disclosures of all of which are hereby incorporated by reference herein.

FIELD OF THE INVENTION

The present invention generally relates to offering and granting of rights and more particularly to a method, system and device for offering and granting of rights using shared state variables.

BACKGROUND OF THE INVENTION

The digital age has greatly increased concerns about ownership, access, and control of copyrighted information, restricted services and valuable resources. Rapid evolution and wide deployment has occurred for computers, and other electronic devices such as cellular phones, pagers, PDAs, and e-book readers, and these devices are interconnected through communication links including the Internet, intranets and other networks. These interconnected devices are especially conducive to publication of content, offering of services and availability of resources electronically.

One of the most important issues impeding the widespread distribution of digital works (i.e. documents or other content in forms readable by computers), via electronic means, and the Internet in particular, is the current lack of ability to enforce the intellectual property rights of content owners during the distribution and use of digital works. Efforts to resolve this problem have been termed "Intellectual Property Rights Management" ("IPRM"), "Digital Property Rights Management" ("DPRM"), "Intellectual Property Management" ("IPM"), "Rights Management" ("RM"), and "Electronic Copyright Management" ("ECM"), collectively referred to as "Digital Rights Management (DRM)" herein. There are a number of issues to be considered in effecting a DRM System. For example, authentication, authorization, accounting, payment and financial clearing, rights specification, rights verification, rights enforcement, and document protection issues should be addressed. U.S. Pat. Nos. 5,530, 235, 5,634,012, 5,715,403, 5,638,443, and 5,629,980, the disclosures of which are incorporated herein by reference, disclose DRM systems addressing these issues.

Two basic DRM schemes have been employed, secure containers and trusted systems. A "secure container" (or simply an encrypted document) offers a way to keep document contents encrypted until a set of authorization conditions are met and some copyright terms are honored (e.g., payment for use). After the various conditions and terms are verified with the document provider, the document is released to the user in clear form. Commercial products such as Cryptolopes and Digiboxes fall into this category. Clearly, the secure container approach provides a solution to protecting the document during delivery over insecure channels, but does not provide any

2

mechanism to prevent legitimate users from obtaining the clear document and then using and redistributing it in violation of content owners' intellectual property.

In the "trusted system" approach, the entire system is responsible for preventing unauthorized use and distribution of the document. Building a trusted system usually entails introducing new hardware such as a secure processor, secure storage and secure rendering devices. This also requires that all software applications that run on trusted systems be certified to be trusted. While building tamper-proof trusted systems is a real challenge to existing technologies, current market trends suggest that open and untrusted systems, such as PC's and workstations using browsers to access the Web, will be the dominant systems used to access digital works. In this sense, existing computing environments such as PC's and workstations equipped with popular operating systems (e.g., Windows, Linux, and UNIX) and rendering applications, such as browsers, are not trusted systems and cannot be made trusted without significantly altering their architectures. Of course, alteration of the architecture defeats a primary purpose of the Web, i.e. flexibility and compatibility.

Some DRM systems allow content owners to specify usage rights and conditions, and associate them with content. These usage rights control how the recipient thereof can use the content. Usually after a content distributor or consumer has completed selecting and ordering specific content, the content is delivered either electronically from some content repository or via a conventional distribution channel to the recipient, such as tangible media sent via a common carrier. Corresponding DRM systems used by the recipient, for example the distributor or consumer, will then interpret the rights and conditions associated with the content, and use them to control how the content is distributed and/or used. Examples of usage rights include view, print and extract the content, and distribute, repackage and loan content. Associated conditions may include any term upon which the rights may be contingent such as payment, identification, time period, or the like.

U.S. Pat. No. 5,634,012, discloses a system for controlling the distribution of digital documents. Each rendering device has a repository associated therewith. A predetermined set of usage transaction steps define a protocol used by the repositories for enforcing usage rights associated with a document. Usage rights persist with the document content. The usage rights can permit various manners of use such as, viewing only, use once, distribution, and the like. Usage rights can be contingent on payment or other conditions.

However, there are limitations associated with the above-mentioned paradigms wherein only usage rights and conditions associated with content are specified by content owners or other grantors of rights. Once purchased by an end user, a consumer, or a distributor, of content along with its associated usage rights and conditions has no means to be legally passed on to a next recipient in a distribution chain. Further the associated usage rights have no provision for specifying rights to derive other rights, i.e. Rights to modify, transfer, offer, grant, obtain, transfer, delegate, track, surrender, exchange, transport, exercise, revoke, or the like. Common content distribution models often include a multi-tier distribution and usage chain. Known DRM systems do not facilitate the ability to prescribe rights and conditions for all participants along a content distribution and usage chain. Therefore, it is difficult for a content owner to commercially exploit content unless the owner has a relationship with each party in the distribution chain.

SUMMARY OF THE INVENTION

Exemplary aspects of the present invention include a method, system and device for sharing rights adapted to be

US 8,001,053 B2

3

associated with items, the method and system including generating at least one of usage rights and meta-rights for the items; defining, via the usage rights, a manner of use for the items; and defining, via the meta-rights, a manner of rights transfer for the items. The device including receiving at least one of usage rights and meta-rights for the items; interpreting, via the usage rights, a manner of use for the items; and interpreting, via the meta-rights, a manner of rights transfer for the items. The usage rights or the meta-rights include at least one state variable that is shared by one or more rights.

Still other aspects, features, and advantages of the present invention are readily apparent from the following detailed description, simply by illustrating a number of exemplary embodiments and implementations, including the best mode contemplated for carrying out the present invention. The present invention is also capable of other and different embodiments, and its several details can be modified in various respects, all without departing from the spirit and scope of the present invention. Accordingly, the drawings and descriptions are to be regarded as illustrative in nature, and not as restrictive.

BRIEF DESCRIPTION OF THE DRAWINGS

Exemplary embodiments of this invention will be described in detail, with reference to the attached drawings in which:

FIG. 1 is a schematic diagram of a three-tier model for content distribution;

FIG. 2 is a schematic diagram illustrating rights offering and granting processes in the model of FIG. 1;

FIG. 3(a) is a schematic diagram of a simple supplier-consumer push model for rights generating, issuing and exercising;

FIG. 3(b) is a schematic diagram of a simple supplier-consumer pull model for rights generating, issuing and exercising;

FIG. 4 is a block diagram of a rights offering-granting architecture in accordance with the preferred embodiment;

FIGS. 5a and 5b are workflow diagrams for examples of offering and granting rights between a rights supplier and a rights consumer with a push and pull model respectively;

FIG. 6 is a flow chart of a rights offer generation process in accordance with the preferred embodiment;

FIG. 7 is a flow chart of a rights offer consideration process in accordance with the preferred embodiment;

FIG. 8 is a flow chart of a rights offer customization process in accordance with the preferred embodiment;

FIG. 9 is block diagram of a DRM system that may be utilized in connection with the preferred embodiment;

FIG. 10 is a block diagram of an exemplary structure of a license containing usage rights and meta-rights of the preferred embodiment;

FIG. 11 is a schematic illustration of a rights label of the preferred embodiment;

FIG. 12 illustrates an exemplary system including a state-of-rights server;

FIG. 13 illustrates employing of a state variable in deriving exclusive usage rights;

FIG. 14 illustrates employing of a state variable in deriving inherited usage rights;

FIG. 15 illustrates employing of a state variable in deriving rights that are shared among a known set of rights recipients;

FIG. 16 illustrates employing of a state variable in deriving rights that are shared among a dynamic set of rights recipients;

4

FIG. 17 illustrates employing of a state variable in maintaining a state shared by multiple rights;

FIG. 18 illustrates employing of multiple state variables to represent one state of rights;

FIG. 19 illustrates a case where not all rights are associated with states;

FIG. 20 illustrates a case where not all rights which are associated with states are shared or inherited; and

FIG. 21 illustrates a case of rights sharing based on an offer which does not explicitly include meta-rights.

DETAILED DESCRIPTION

Prior to providing detailed description of the apparatus and method for offering and granting rights, a description of a DRM system that can be utilized to specify and enforce usage rights and meta-rights for specific content, services, or other items is first described below.

FIG. 9 illustrates DRM System 10 that includes a user activation component, in the form of activation server 20, that issues public and private key pairs, or other identification mechanisms, to content users in a protected fashion, as is well known. Typically, when a user uses DRM system 10 for the first time, the user installs software that works with, or includes, a rendering application for a particular content format. The software is installed in client environment 30, a computer associated with the content recipient, for example. The software is part of DRM 10 system and is used to enforce usage rights for protected content. During the activation process, some information is exchanged between activation server 20 and client environment 30. Client component 60 preferably is tamper resistant and contains the set of public and private keys issued by activation server 20 as well as other components, such as rendering components for example.

Rights label 40 is associated with content 42 and specifies usage rights and meta-rights that are available to a recipient, i.e. a consumer of rights, when corresponding conditions are satisfied. License Server 50 manages the encryption keys and issues licenses 52 for protected content 42. Licenses 52 embody the actual granting of rights, including usage rights and meta-rights, to an end user. For example, rights offer 40 may permit a user to view content for a fee of five dollars and print content for a fee of ten dollars, or it may permit a user to offer rights to another user, for example, by utilizing the concept of meta-rights described below. License 52 can be issued for the view right when the five dollar fee has been paid. Client component 60 interprets and enforces the rights, including usage rights and meta-rights, that have been specified in the license. Rights label 40 and license 52 are described in detail below.

FIG. 11 illustrates rights label 40 in accordance with the preferred embodiment. Rights label 40 includes plural rights options 44. Each rights option 44 includes usage rights 44a, conditions 44b, and content specification 44c. Content specification 44c can include any mechanism for referencing, calling, locating, or otherwise specifying content 42 associated with rights offer 44.

As shown in FIG. 10, license 52 includes license 52a, grant 52b, and digital signature 52c. Grant 52b includes granted usage rights and/or meta-rights selected from label. The structure of the grant also includes one or more principals, to whom the specified usage rights and/or meta-rights are granted, a list of conditions, and state variables required to enforce the license. Like usage rights, access and exercise of the granted meta-rights are controlled by the condition list and state variables as described below.

US 8,001,053 B2

5

Clear (unprotected) content can be prepared with document preparation application 72 installed on computer 70 associated with a content publisher, a content distributor, a content service provider, or any other party. Preparation of content consists of specifying the usage rights, meta-rights, and conditions under which content 42 can be used and distributed, associating rights label 40 with content 42 and protecting content 42 with some crypto algorithm. A rights language such as XrML can be used to specify the rights and conditions. However, the usage rights and meta-rights can be specified in any manner. Also, the rights can be in the form of a pre-defined specification or template that is merely associated with the content. Accordingly, the process of specifying rights refers to any process for associating rights with content. Rights label 40 associated with content 42 and the encryption key used to encrypt the content can be transmitted to license server 50.

Rights can specify transfer rights, such as distribution rights, and can permit granting of rights to others or the derivation of rights. Such rights are referred to as "meta-rights". Meta-rights are the rights that one has to manipulate, modify, or otherwise derive other meta-rights or usage rights. Meta-rights can be thought of as usage rights to usage rights. Meta-rights can include rights to offer, grant, obtain, transfer, delegate, track, surrender, exchange, and revoke usage rights to/from others. Meta-rights can include the rights to modify any of the conditions associated with other rights. For example, a meta-right may be the right to extend or reduce the scope of a particular right. A meta-right may also be the right to extend or reduce the validation period of a right.

Often, conditions must be satisfied in order to exercise the manner of use in a specified right. For, example a condition may be the payment of a fee, submission of personal data, or any other requirement desired before permitting exercise of a manner of use. Conditions can also be "access conditions" for example, access conditions can apply to a particular group of users, say students in a university, or members of a book club. In other words, the condition is that the user is a particular person or member of a particular group. Rights and conditions can exist as separate entities or can be combined.

State variables track potentially dynamic states conditions. State variables are variables having values that represent status of an item, usage rights, license or other dynamic conditions. State variables can be tracked, by clearinghouse 90 license or server 30 another device, based on identification mechanisms in license 52. Further, the value of state variables can be used in a condition. For example, a usage right can be the right to print content 42 three times. Each time the usage right is exercised, the value of the state variable "number of prints" is incremented. In this example, when the value of the state variable is three, the condition is not longer satisfied and content 42 cannot be printed. Another example of a state variable is time. A condition of license 52 may require that content 42 is printed within thirty days. A state variable can be used to track the expiration of thirty days. Further, the state of a right can be tracked as a collection of state variables. The collection of the change is the state of a usage right represents the usage history of that right.

A typical workflow for DRM system 10 is described below. A recipient, such as a user, operating within client environment 30 is activated for receiving content by activation server 20. This results in a public-private key pair (and some user/machine specific information) being downloaded to client environment 30 in the form of client software component 60 in a known manner. This activation process can be accomplished at any time prior to the issuing of a license.

6

When a user wishes to use protected content 42, the user makes a request for the content 42. For example, a user might browse a Web site running on Web server 80 associated with a grantor of rights such as a content distributor, using a browser installed in client environment 30, and attempt to download protected content 42. During this process, the user may go through a series of steps possibly including a fee transaction (as in the sale of content) or other transactions (such as collection of information). When the appropriate conditions and other prerequisites, such as the collection of a fee and verification that the user has been activated, are satisfied, Web server 80 contacts license server 50 through a secure communications channel, such as a channel using a Secure Sockets Layer (SSL). License server 50 then generates license 52 for the content and Web server 80 causes both protected content 42 and license 52 to be downloaded. License 52 can be downloaded from license server 50 or an associated device. Content 42 can be downloaded from computer 70 associated with a publisher, distributor, or other party.

Client component 60 in client environment 30 will then proceed to interpret license 52 and allow use of content 42 based on the rights and conditions specified in license 52. The interpretation and enforcement of usage rights are well known generally. The steps above may take place sequentially or approximately simultaneously in various order.

DRM system 10 addresses security aspects of protecting content 42. In particular, DRM system 10 may authenticate license 52 that has been issued by license server 50. One way to accomplish such authentication is for application 60 to determine if the licenses can be trusted. In other words, application 60 has the capability to verify and validate the cryptographic signature of digital signature 52c, or other identifying characteristic of the license. During the activation step described above, both client environment 30 and license server 50 receive a set of keys in a tamper-resistant software "package" that also includes other components, such as the necessary components for activated client environment 30 to verify signature 52 of license 52 in a known manner. Of course, the example above is merely one way to effect a DRM system. For example, the license and content can be distributed from different entities. Also, rights offer 40 can be associated with content by a party other than the party preparing the content. Also, clearinghouse 90 can be used to process payment transactions and verify payment prior to issuing a license.

For any set of rights, there are two kinds of entities involved, the "supplier" and the "consumer". The function of the supplier is to offer, and possibly grant, the rights, and the function of the consumer is to select, and possibly exercise the rights. Both the supplier and consumer may actually represent two or more entities. In general, multiple entities may collectively make an offer and grant rights to multiple entities. The supplier and consumer represent any two entities in the content value chain that have a direct relationship with each other regarding the granting of rights. At the beginning of the value chain, the supplier and consumer may be author and publisher. Going down along the value chain, the supplier and consumer may be a publisher and another publisher (for content aggregation), a publisher and distributor (for content distribution), a distributor and another distributor (for multi-tier content distribution), a distributor and a retailer (for content retailing), a retailer and a consumer (for content consumption), and a consumer and another consumer (for content supper-distribution or personal lending).

An "offer of rights" or "rights offer" expresses how a consumer (e.g. a content distributor or user) can acquire a

US 8,001,053 B2

7

particular instance of content together with its associated usage rights and/or meta-rights. An offer may or may not contain financial terms. An offer is an expression of mere willingness to commerce negotiation and also an expression of willingness to grant on terms stated. An offer may be expressed in the form of a rights label. A “consideration of rights” is a process as part of the rights granting in which the rights consumer has examined the rights being offered and possibly bargained them and associated terms and conditions. A “choice of rights” is a selection of rights and their associated terms and conditions from a rights offer. It indicates the intent of the consumer to accept these rights and the corresponding terms and conditions. For example, selection can comprise selecting one option **44** from label **40**. “Customization of rights” is a process as part of the rights granting in which the rights supplier assembles rights and terms and conditions based on a choice of the rights consumer. The output of this process can be a draft license to be accepted by the rights consumer. A “license of rights” is an expression of rights and possibly conditions accepted and agreed upon by the rights supplier and consumer. It is the output of the rights offering and granting process. A license is a grant to exercise the rights that govern the usage (possibly including further distribution) of content or other items.

As described above, a rights label, such as rights label **40**, may contain a number of options **44** allowing the consumer to make a selection and conduct negotiation (if permitted), while license **52** contains rights the consumer has selected and accepted. Note that the accepted rights may include a right to present offers to others or make selections of offers.

An example of a distribution chain model is illustrated in FIG. 1. The distribution chain includes a content provider **100**, distributor **110**, and end user **120**. Of course content may be prepared in the manner described above. It is assumed that the content has already been prepared in the model of FIG. 1. FIG. 1 is directed to the transfer of content and shows that, in this example, provider **100** may publish content to distributor **110** or receive content for reuse from distributor **110**. Distributor **110** may in turn distribute content to user **120** or receive returned content form user **120**. User **100** can use content. To further illustrate the potential complexities of multi-tier distribution chains provider **100** can aggregate content from others, distributor **110**, can receive content from other distributors for redistribution, and user **120** can share content with the other users. It is clear that there are plural stages in the content life cycle and plural relationships between the various parties. A precise and consistent specification of rights at the different stages of the life cycle and relationships is important and crucial to persistent protection of content in multi-tier distribution and usage.

FIG. 2 illustrates the flow of rights in the same model, including rights generating, aggregating, issuing, relinquishing, driving, granting, surrendering, delegating and exercising. The model of FIG. 2 includes the same entities, provider **100**, distributor **110**, and user **120**. It can be seen that, with respect to the flow of rights, each party can grant and accept rights. User **120** can grant and accept rights from other users, a process called “delegation”, in this example.

The model of FIG. 2 covers many specific content publishing, distribution and use relationships. Other models can be derived from on this model by a different consolidation or segregation of the parties. For example, every provider can be a distributor. This is “direct publishing”, which allows individual authors to distribute/sell their content without any intermediate publisher. Further, every consumer can be a potential distributor. This allows consumers to pass content to each other. This includes supper-distribution, gifting, and

8

personal lending. In a “Web community” and everyone is able to publish, distribute and consume content. “Content aggregation” allows publishers to compose content from other publishers into composite works. Site license and enterprise use allows sharing content among consumers.

In general, all the rights relationships shown in FIG. 2 can be captured by two generic supplier-consumer models, as shown in FIGS. 3(a) and 3(b). FIG. 3(a) shows a “push” model and FIG. 3(b) shows a “pull” model. In the push model shown in FIG. 3(a), rights supplier **200** initiates the rights offering and granting process by generating an offer and granting the rights to the rights consumer **210**. In the pull model shown in FIG. 3(b), rights consumer **210** initiates the process by requesting an offer and accepting the rights from the rights supplier **200**.

An architecture of the preferred embodiment for rights offering and granting is shown in FIG. 4. Architecture **400** can be implemented as a combination of computer hardware and software and includes rights supplier component **402**, rights consumer component **438** and communication channel **422** linking these two components. For example, communication channel **42** can be Internet, a direct computer to computer connection, a LAN, a wireless connection or the like. Supplier component **402** is associated with the supplier, i.e. the entity making rights available to a consumer who is the entity going to exercise, i.e., consume the rights. The supplier could be the content owner or provider, or could be a distributor or any “middle-man,” such as a retailer or operator of a web site. Consumer component **438** is associated with the consumer who could be the ultimate user (i.e., content consumer) or a “middle-man,” such as a retailer, whole-seller, or reseller. Keep in mind that the consumer consumes rights and does not necessarily use (i.e. consume) the content. Both supplier component **402** and consumer component **438** can embody any type of hardware devices, and/or software modules, such as a personal computer, a handheld computer, a mobile phone a server, a network, or any combination of the same. Supplier component **402** generates rights label **40** as offers, presents draft licenses and grants license **52** to the consumer. Consumer component **438** issues requests, select choices of options **44** from rights labels **40**, generates counter offers, and accepts licenses **52**. Supplier component **402** and consumer component **438** can be embodied in the same device(s) and communication channel **422** can be an internal channel.

Supplier component **402** contains user interface module **404**, communication interface module **420** identity module **406** repository **412** for supplier’s rights (e.g., in the form of issued licenses) and database **414** for management related information. User interface **404** accomplishes presentation to the user of the component functions and acceptance of user interactions in a known manner. Communication interface **422** provides the proper formatting and protocols for messages between supplier component **402** and consumer component **438**. Identity module **406** ensures that the identity of supplier component **402** can be authenticated by consumer component **438** and may contain authentication information like a password, cryptographic keys or biometric information of the user of supplier component **402**. Rights repository **412** stores rights granted to the user of supplier component **402** and may include functions for indexing, searching and updating the rights stored within. Management database **414** is used to archive information generated during the rights offering and granting processes. Such information includes information related to initial offers, consumer choices, possible counter-offers, agreements and final licenses.

Consumer component **438** includes user interface module **428**, communication interface module **424**, identity module

US 8,001,053 B2

9

426, repository 434 for consumer's rights (e.g., in the form of issued licenses), and database 436 for management related information. User interface 424 deals handles presentation to the user of the component and acceptance of user interactions. Communication interface 422 provides the proper formatting and protocols for rights offering and granting messages between supplier component 402 and consumer component 438. Identity module 426 ensures that the identity of the consumer component 438 can be authenticated by supplier component 402 and may contain authentication information like a password, cryptographic keys or biometric information of the user. Rights repository 434 stores rights granted to the user of consumer component 438 and may include functions for indexing, searching and updating the rights stored within. Management database 436 is used to archive information generated during the rights offering and granting process. The information includes that related to offers 44, consumer choices, possible counter-offers, agreements and licenses 52. Note that database 436 can store information that is the same as or different from database 414 because the parties may interact with other parties and thus have different archived information.

Supplier component 402 also includes offer generator module 408 for generating offers, rights composer module 410 for composing licenses, offer templates module 418 for providing templates for generating offers based on previous transactions and common formality of offers, and consumer profiles module 416 for customizing and granting rights based on past consumer characteristics and relationships.

Consumer component 438 also includes offer analyzer module 430 for understanding rights and their terms and conditions presented within offers, a choice maker module 432 for selecting favorable options specified in offers, a supplier preference module 438 for describing any preferred suppliers based on past and existing supplier characteristics and relationships, and choice patterns module 440 for providing patterns and interests in selection options in offers. For example, the choice pattern module 440 may include a list of preferred suppliers or a list of lowest prices for the item of interest to the consumer. Offer analyzer module 430 and choice maker module 432, respectively, may be combined into one module.

The process of offering and granting rights within architecture 400 is based on protocols followed by supplier component 402 and consumer component 438. These protocols generally consist of an offer and acceptance of that offer. Specifically, the protocols include an offering of rights by one party to another and acceptance of that offer by the person to whom it is made. An offer, once made, may be styled so that it may be revoked before acceptance or the offeror could style it so that it cannot be revoked at all or only under certain circumstances definable by the offeror. An offer can also expire in various way, for example if a deadline for acceptance passes. If there is no specified deadline, then the offer could expire in a predetermined reasonable time, depending on the subject matter of the offer. For periodically available content such as magazines, journals, and even newspapers, a reasonable time could be accord to the period of the content publication, for example. For dynamically generated or provided content such as streaming content, a reasonable time could be any time before the availability of the content. The rights supplier can dictate other terms of the acceptance, to which the rights consumer is bound. For example, the offer may require acceptance in sending back in a certain form via an email or through a certain web page interface.

FIG. 5(a) illustrates the workflow of protocol 500 of a push model for rights granting. Supplier component 402 generates

10

an offer of rights in the form of rights label 40 for example, with possibly many options 44, and sends it to consumer component 438 (510). Consumer component 438 considers the offer and its possible options, and responds to supplier component 402 with a choice of any of the optional rights offer 44 (512). Supplier component 402 customizes rights according to the consumer's response, and issues the rights the user of consumer component 432 (514) in the form of a draft license.

Consumer component 438 then accepts the draft license if it corresponds to the choice made and is otherwise acceptable (516). Upon acceptance, supplier component 402 generates license 52 and transmits license 52 to consumer component (518). Keep in mind that grant 52b of license 52 can include usage rights and/or meta-rights. Therefore license 52 can permit the user of consumer component 438 to grant rights to others in a similar fashion. However, the derivable rights are controlled by upstream parties through the use of meta-rights. Additionally, the protocol can include steps where supplier component 402 requests to make payment through a credit card of the user of consumer component 438, and the user component 402 provides the information and authorizes the charge. Both supplier component 402 and consumer component 438 can generate status reports on success or failure of the process. Further, parties can authenticate each other during the process and maintain authentication through the process.

FIG. 5(b) shows a protocol of pull model for rights granting. First, consumer component 438 sends a request to supplier component 402 to indicate an interest in obtaining certain rights in content (520). Supplier component 402 then responds with an offer, in the form of label 40 having plural offer options 44, covering the rights requested by consumer component 438, and sends the offer to consumer component 438 (522).

Consumer component 438 then considers the offer and its options, and responds to supplier component 402 with a choice of one of the offer options (524). Supplier component 402 customizes rights according to the response, and grant the rights to the consumer in the form of a draft license (526). Consumer component 438 then accepts the draft license (528) and supplier component 402 issues license 52 granting rights to consumer component 438 (530). Once again the rights can include meta-rights.

FIG. 6 illustrates the offer generation process 600 performed by offer generator module 408 in supplier component 402. In offer generation process 600, available rights are first collected in block 602. Rights may be available from a previous supplier by being derived from meta-rights granted to the supplier or may be originally created rights. In step 604 it is determined whether supplier has a right to make an offer to the consumer. For example, if the consumer is known to be a minor and the content is restricted to an adult consumer or if the consumer is on a list of those prohibited from receiving content, the supplier may not make an offer. In such case, the offer generation process terminates in step 606. If the supplier has the right to make an offer, the process then determines all the rights that can be offered to the consumer in step 608 by parsing the rights collected in step 602. Next, in step 610, the process determines whether the consumer has requested any specific rights. If a request has been received, the process further filters the determined rights that can be offered, taking the received consumer requested rights into consideration and comparing them to the available rights. Then, the process determines whether an offer template needs to be applied in steps 614.

US 8,001,053 B2

11

For example, the consumer might be offered standard rights included in the template, such as printing right, archiving right, etc. of the content. If an offer template is available and needed, the offer template is then applied in steps 616. In steps 618, human intervention may be provided to further make adjustments to the offer template or to any of the rights that are available for offering thus far in the process. Next, restrictions can be applied, through conditions and/or state variables. For example, a time restriction may be placed on certain rights in step 620. Finally, a digital signature or other authentication is provided with the collection of rights to be offered in step 622 and an authenticated offer, in the form of rights label 40 is made in step 624 and presented to consumer component 438 in step 624.

FIG. 8 illustrates rights customization process 800 which is performed by rights composer module 410 in supplier component 402. Initially, consumers choices are received in step 802. Choices are rights and conditions of an option 44 selected label 40 of step 624 (FIG. 6). The process then determines if supplier component 402 has the right to grant rights to consumer component 438 in step 804. For example, if the consumer fails to meet a certain requirement, such as minimum age or proof of residence in a locale where content may be licensed, for example, granting a license may not be proper, and the rights customization process 800 terminates in step 806. Otherwise, consumer selected choices are analyzed in step 808 to ascertain if they are discernible by supplier component 402. For example, the choices can be parsed to see if they are understandable.

Next, the process determines if consumer information is available in step 810. For example, consumer profiles may be stored in database 414 (FIG. 4). If available, the consumer information is taken into consideration in step 812 for further analysis of consumer choices. In step 812, dynamic information can also be considered as described below. For example, the profile may include a trust rating or address of the consumer that renders it desirable or undesirable to provide certain rights. The process then determines if the choices are reasonable in step 814. This determination may be carried out, for example, computationally or with human intervention. If the customer's choices are deemed unreasonable, re-negotiation of the customer's choices is then performed in block 816. In this re-negotiation process, the customer is presented with a new proposed offer based on the previously analyzed choices, the customer is given an opportunity to submit new choices offered, and the right customization process 800 begins again in step 802. Otherwise, a license including the selected rights is created in step 818.

After a license is created, if consumer acceptance is necessary (step 820), it is presented to the consumer for review in step 822. If the consumer does not agree with the terms in the license in step 824, re-negotiation is then initiated in step 816, which re-starts the rights customization process 800 again in step 802. In step 820, if a review by the consumer is not required, then the license is authenticated in step 826 to create a completed license 52 in step 828 which is to be issued and associated with content 42.

FIG. 7 illustrates offer consideration process 700 which is performed by offer analyzer module 430 and choice maker module 432 of consumer component 438. Available offers are first collected in step 702. In step 704, process 700 determines whether it has a right to accept offers from the supplier. For example, if the consumer certain restrictions on the purchase of content, such as an age restriction or a restriction against accepting content from outside an enterprise, the consumer may not accept an offer. In such a case, the offer consideration process terminates in step 706. If the consumer has the right

12

to accept offers from the supplier, the offers are then analyzed in step 708 to ascertain if they are discernible. If it is determined that supplier preferences are available in step 710, the offers are filtered in step 712 based on the preferences. For example, the consumer may trust a specific supplier, or otherwise prefer transactions with that supplier, more than other suppliers. Next, step 714 determines if consumer preferences are available and, if so, they are applied in step 716 to the offers. Once all the offers are analyzed, by applying the logic of steps 708-714 and any other desired logic, the consumer then selects options in block 718 and specifies contingencies in block 720. The selection of options can be done automatically. If human intervention is desired, the customer can intervene and further specify additional choices or conditions desired. Any preferences, rules, or other logic can be used to analyze offers.

Overall, as can be seen in the description of FIGS. 6, 7, and 8 above, the consumer sends a request, and then a license is constructed. Either the supplier or the consumer could draft the content of the license, but in the example above the supplier does so. The request is a subset of an offer and the offer has one or more options. The supplier makes the offer available to the consumer sending the request (and to other consumers if that is the desire), and the consumer (including other consumers, if applicable) makes choices. Then, the supplier analyzes the choices, and constructs the license (i.e. a grant of rights). Note that the request can also be rejected, or a counter proposal could be made and the same process could then repeat for the counter proposal.

Also, when the supplier analyzes the request, the analysis may be done automatically, or with human intervention. When the consumer considers the offer, the choice or acceptance may be done automatically, or with human intervention. Either the offer or a license, or both, may be generated based on the dynamic information, the consumer's information, and the consumer's request, such as described above.

The dynamic information may include many kinds of information including information related to pricing, status of the network, the traffic of a web site at each moment of time, discounts given, coupons given, the habits of the consumer, how many times the content has been used, for how long the content was used, where it was used, or the like. The dynamic information can be tracked as state variables and the values of the state variables can be checked and updated as necessary.

Dynamic information is information capable of being (although, it need not actually be) changed or created by or by reference to a non-static element. For example, the dynamic information can be obtained based on a formula, database, curve, predetermined table, percentage of a value, a function, reference to other data, such as the prime rate of interest or the change in a stock market index, and/or by a human intervention of the user or distributor, and/or consumer's input.

The consumer's information may include information such as the age of the consumer, the credit history of the consumer, the credit limit of the consumer, income of the consumer, what kind of rights or licenses obtained, the password of the consumer, the key assigned to the consumer, club membership for access or discount, the class of the consumer based on a predetermined criteria, or any other data, identification characteristics and information. The supplier's information may include some or all of the subjects of information as the consumer's information, and may also include, for example, available options or variations, suppliers, shipping information, and other information.

The system and processes disclosed in this invention support multi-tier and super distributions of content. The following is a use case that shows how this can be modeled and

US 8,001,053 B2

13

supported. It illustrates the process of offering and granting rights by showing the process of transforming offered rights to a rights supplier (the content distributor in this case) to granted rights to a rights consumer (the end user in this case). It specifically shows how an offer is generated from an existing license, how this offer is considered with a choice, and how a final license is issued. Meta-rights provide a mechanism for permitting the transfer of rights from one party to the next party in a content distribution chain.

Suppose that a content provider P of some content C wants to specify that a distributor D may sell, to any end user within the region of the United States (US), the “play” right at a flat rate of \$1 and the “print” right at a cost of \$4 per copy (both are paid by D to P). The provider also allows the content distributor to add its own conditions to the “play” and “print” rights it issues to end users.

A license from the content provider to the distributor may resemble the following using the XrML rights language.

```

<license>
  <grant>
    <forAll varName="user"/>
    <forAll varName="distributorConditionForPlay"/>
    <principal id="distributor"/>
    <issue/>
    <grant>
      <principal varRef="user"/>
      <play/>
      <digitalResource licensePartId="book"/>
      <allCondition>
        <region regionCode="US"/>
        <condition varRef="distributorConditionForPlay"/>
      </allCondition>
    </grant>
    <fee>
      <flat currencyCode="USD">1</flat>
      <to licensePartId="provider"/>
    </fee>
  </grant>
  <grant>
    <forAll varName="user"/>
    <forAll varName="distributorConditionForPrint"/>
    <principal id="distributor"/>
    <issue/>
    <grant>
      <principal varRef="user"/>
      <play/>
      <digitalResource licensePartId="book"/>
      <allCondition>
        <region regionCode="US"/>
        <condition varRef="distributorConditionForPrint"/>
      </allCondition>
    </grant>
    <fee>
      <perUse regionCode="USD">5</perUse>
      <to licensePartId="provider"/>
    </fee>
  </grant>
  <issuer id="provider"/>
</license>

```

The distributor may make an offer to the end user based on the rights it has as expressed in the license above. Note that usage rights and conditions of each option are set forth as XML elements between <grant> tags. In the following offer, note that the distributor adds a fee condition for getting the “play” right, charging the end user \$2 (\$1 more than it pays to the provider), and another fee condition for the “print” right, charging the end user \$6 per print copy (\$1 more than it pays to the provider). The distributor also limits the offer to an acceptance time period (up to Dec. 31, 2002). Meta rights granted to the distributor permit the distributor to modify the grant in the license, as described above, and make the offer.

14

```

<offer>
  <grant>
    <forAll varName="user"/>
    <principal varRef="user"/>
    <obtain/>
    <grant>
      <principal varRef="user"/>
      <play/>
      <digitalResource licensePartId="book"/>
      <region regionCode="US"/>
    </grant>
    <fee>
      <flat currencyCode="USD">2</flat>
      <to licensePartId="distributor"/>
    </fee>
  </grant>
  <grant>
    <forAll varName="user"/>
    <principal varRef="user"/>
    <obtain/>
    <grant>
      <principal varRef="user"/>
      <print/>
      <digitalResource licensePartId="book"/>
      <allCondition>
        <region regionCode="US"/>
        <fee>
          <perUse currencyCode="USD">6</perUse>
          <to licensePartId="distributor"/>
        </fee>
      </allCondition>
    </grant>
    <issuer id="distributor"/>
    <validityInterval>
      <until>2002:12:31</until>
    </validityInterval>
  </grant>
</offer>

```

When the offer is presented to an end user, the end user may choose to get only the right to “play” for the flat fee of \$2 and responds to the distributor with a choice set forth as an XML element between <choice> tags as follows.

```

<choice>
  <grant>
    <principal id="anEndUser"/>
    <obtain/>
    <grant>
      <principal id="anEndUser"/>
      <play/>
      <digitalResource licensePartId="book"/>
      <region regionCode="US"/>
    </grant>
    <fee>
      <flat currencyCode="USD">2</flat>
      <to licensePartId="distributor"/>
    </fee>
  </grant>
  <issuer id="anEndUser"/>
  <validityInterval>
    <until>2002:12:31</until>
  </validityInterval>
</choice>

```

Note that the request can also be rejected. Note also that a response can also be constructed as a counter offer for rights not originally offered by the distributor. When the distributor receives the choice from the end user, it then issues a license to the user as shown below.

US 8,001,053 B2

15

```

</license>
  <grant>
    <principal id="anEndUser"/>
    <obtain/>
    <grant>
      <principal id="anEndUser"/>
      <play/>
      <digitalResource licensePartId="book"/>
      <region regionCode="US"/>
    </grant>
    <fee>
      <flat currencyCode="USD">2</flat>
      <to licensePartId="distributor"/>
    </fee>
  </grant>
  <issuer id="distributor">
    <issuedTime>
      2002:05:06
    </issuedTime>
  </issuer>
</license>

```

Note that in all the XML documents above, the issuers may choose to digitally sign the documents using some digital signature algorithms. The recipients of these documents have options to verify the validity of these documents by checking the validity of the attached digital signatures. Access to the various documents, and elements thereof, can be controlled using known techniques.

In some situations offering and granting result in a license with a fresh state for content usage. As one starts to exercise the rights, derived rights, obtained as a result of meta-rights, may inherit and/or share the state variable values associated with the rights. For example, when one is granted with the right to print 5 times and make 4 copies of some document, all new copies may have the same set of rights but share the state (or remaining rights) with the original. After the original has been printed 2 times and a new copy was then made, the copy and original can all together print 3 times and make 2 more new copies.

Thus, the exemplary embodiments include a method for transferring usage rights adapted to be associated with items. The method includes generating, by a supplier, at least one first offer containing usage rights and meta-rights for the item, the usage rights defining a manner of use for the items, the meta-rights specifying rights to derive usage rights or other meta-rights, presenting the offer to a first consumer, receiving a selection from the first consumer indicating desired usage rights and meta-rights, and generating a first license granting the desired usage rights and meta-rights to the first consumer. The exemplary embodiments further include a system for transferring usage rights adapted to be associated with an item to be licensed in multi-tier channels of distribution with downstream rights and conditions assigned at least one level. The system includes a supplier component, comprising a supplier user interface module, an offer generator module for generating an offer containing at least usage rights and of meta-rights, a rights composer module for composing a draft license, and a repository for supplier's rights, a supplier management database. The system further includes a consumer component comprising a consumer user interface module, an offer-consideration module configured to analyze the offers generated by the supplier component and select offers based on the analysis, and a repository for consumer's rights, a consumer management database. The exemplary embodiments still further include a method for generating a license to digital content to be used within a system for at least one of managing use and distribution of the digital content.

16

The method includes presenting a consumer with an offer including meta-rights, receiving a selection by the consumer of at least one meta-right in the offer, generating a license based on the selection, wherein the license permits the consumer to exercise the at least one meta-right and permits the consumer to offer at least one derived right derived from the at least one meta-right and generate a license including the at least one derived right.

FIG. 12 illustrates an exemplary system including a common state-of-rights server, according to the present invention. In FIG. 12, the exemplary system can include a common state-of-rights server of the system 1201, including a state-of-rights manager 1209, and one or more state-of-rights repositories 1214, and one or more license servers 1200, including a meta-rights manager 1210, a usage rights manager 1212, an authorization component 1208, a condition validator 1206, a state-of-rights manager 1204, one or more state-of-rights repositories 1216, a license manager 1203, a license interpreter 1202, and one or more license repositories 1218.

The common state-of-rights server 1201 can be configured as a remote server connected with one or more of the license servers 1200. The common state-of-rights server 1201 provides comparable services as the state-of-rights manager 1204 in the license servers 1200 via the state-of-rights manager 1209. The services provided by the state-of-rights server 1201 are accessible and states that the server 1201 manages can be shared by one or more rights suppliers and rights consumers (not shown).

The state-of-rights server 1201 can be configured as a remote server connected with one or more of the license servers 1200 via one or more communication links 1220, and the like. The services provided by the state-of-rights server 1201 also can be integrated within one or more of the license server 1200 and such services can be accessible by other rights suppliers, rights consumers, and the like.

The license manager 1203 derives new rights based on an offer, which can include any suitable machine-readable expression, and optionally including meta-rights. While deriving rights, the license manager 1203 can create new state variables to be associated with derived rights. The creation of state variables and their scopes can be prescribed in the offer or by some other function in the system. The state variables can be created in one or more instances, for example, prior to rights derivation, during rights derivation, upon fulfillment of conditions, during a first exercise of rights associated with the state variables, and the like. The state variables can be designated exclusively for a specific rights consumer, can be shared among rights consumers, and can be shared among rights consumers and other entities, such as rights suppliers, and the like. The license manager 1203 can interact with the state-of-rights manager 1204 to associate new state variables with physical addresses in one or more of the state-of-rights repositories 1216. The state-of-rights manager 1204 can access the one or more state-of-rights repositories 1216 and can interact with the state-of-rights server 1201 to access shared state variables from one or more of the state-of-rights repositories 1214.

Designated state variables can be used to support a license that grants a recipient of the license a right to print content 5 times, shared state variables can be used to support a site license that grants a group of authorized users a right to print content an aggregated total of 100 times, and the like. A designated state variable can be updated when the corresponding right is exercised, whereas a shared state variable can be updated when an authorized user exercises the corresponding right. In other words, a shared state variable can

US 8,001,053 B2

17

include a data variable that is updated in response to actions by a plurality of users and which is globally applied to each of the users.

There are multiple ways to specify the scope of state variables, each of which can affect whether the derivative state variables can be shared, how the derivative state variables can be shared, and the like. For example, a state variable can be local, and solely confined to a recipient or can be global, and shared by a predetermined group of recipients. A global state variable can be shared by a group of recipients not determined when derived rights are issued, but to be specified later, perhaps based on certain rules defined in the license or based on other means. A global state variable can be shared between one or more rights suppliers, predetermined recipients, unspecified recipients, and the like. Advantageously, depending on the sharing employed with a given a business model and the rights granted in the meta-rights, state variables can be created at different stages of the value chain.

A set of non-exhaustive exemplary usages of state variables will now be described. For example, a state variable can be unspecified in meta-rights, which means the identifier and value of the state variable are yet to be determined by the meta-rights manager module 1210 and included in the derived right. If a distinct state variable is assigned to each derived right, the scope of the state variable in the derived right is typically exclusive to the recipient.

FIG. 13 is used to illustrate employing of a state variable in deriving exclusive usage rights, according to the present invention. In FIG. 13, rights 1302 and 1303 derived from an offer 1301 are exclusive to each respective consumer. The offer 1301 is a type of meta-right of which the recipients have the rights to obtain specific derivative rights when the conditions for obtaining such rights are satisfied. Accordingly, the exemplary offer 1301 has an unspecified state variable 1304. However, specific state variable 1305 and 1306, each with uniquely assigned identifications (IDs) are included in the derived rights 1302 and 1303. The derived state variables 1305 and 1306 are bound to their associated derived rights, e.g., "AlicePlayEbook" (i.e., Alice has the right to play Ebook) is bound to derived right 1302, and "BobPlayEbook" (i.e., Bob has the right to play Ebook) is bound to derived right 1303. The "AlicePlayEbook" variable can be updated when Alice exercises her play right, whereas the "BobPlayEbook" variable can be updated when Bob exercises his play right.

Other than deriving rights from an offer, a right can transfer from an entity to a recipient. When a right is transferred, the governing of the associated state variable is also transferred to the recipient. After a right is transferred, the source principal typically can no longer exercise the right, whereas the recipient can exercise the right. The license server governing the exercising of a right of a recipient assumes the responsibility for state management. If, however, the state variables are managed by the common state of right server 1201, the state of right server 1201 needs to be informed of the transfer of right. Specifically, the state variable can be managed in the context of the recipient after the transfer of right.

When a right is to be shared between the source principal and the recipient, the associated state variable is referenced in the derived right. If the same right is shared with multiple recipients, then typically all of the recipients share the same state variables with the source principal. In this case, a shared state can be managed by an entity that is accessible by all sharing principals.

FIG. 14 is used to illustrate employing of a state variable in deriving inherited usage rights, according to the present invention. In FIG. 14, a derived right can inherit a state variable from meta-rights. For example, a personal computer

18

(PC) of a user, Alice, can be configured to play an e-book according to a license 1403. A personal data assistant (PDA) of Alice also can obtain a right to play the e-book according to offer 1401, if the PC and PDA share the same state variables 1404 and 1405, e.g., "AlicePlayEbook." A derived right 1402 allows Alice also to play the e-book on her PDA as long as the PDA and the PC share a same count limit 1406 of 5 times.

When a usage right is to be shared among a predetermined set of recipients, a state variable for tracking a corresponding usage right can be specified in a meta-right using a same state variable identification for all recipients. During a process of exercising the meta-right, the same state variable identification is included in every derived right.

FIG. 15 illustrates the use of state variable in deriving rights that are shared among a known set of rights recipients, according to the present invention. In FIG. 15, a site license 1501 is issued to FooU university. For example, via the site license 1501, a librarian is granted a right to issue rights that allow FooU students to play, view, and the like, corresponding content, such as e-books and the like, as long as such usage is tracked by a state variable 1504, e.g., "www.fooou.edu." Accordingly, rights 1502 and 1503 derived from the site license 1501 include state variables 1505 and 1506, "www.fooou.edu," which can be updated when corresponding students, Alice and Bob, play the e-book.

When a usage right is to be shared among a dynamic set of recipients, the state variable can stay unspecified in the usage right. When exercising a meta-right and a set of recipients is known, a state variable can be specified using some identification unique to the known recipients and can be included within a derived right.

FIG. 16 is used to illustrate employing of a state variable in deriving rights that are shared among a dynamic set of rights recipients, according to the present invention. In FIG. 16, an offer 1601 specifies that a distributor can issue site licenses to affiliated clubs, allowing 5 members of each club to concurrently view, play, and the like, content, such as an e-book. A corresponding state variable 1607 associated with such a right can be unspecified in the offer 1601. When corresponding rights 1602 and 1603 are issued to affiliated clubs, the corresponding club identities are used to specify state variables 1608 and 1609 in the issued rights. The offers 1602 and 1603 are meta-rights derived from the offer 1601, with offer being assigned the distinct state variables 1608 and 1609. Further rights 1604-1606 can be derived from the offers 1602 and 1603 to be shared among members of each respective club. The licenses 1604 and 1605 are examples of rights derived from the offer 1602, and which inherit the state variable 1608, e.g., "urn:acme:club," whereas the license 1606 inherits the state variable 1609, e.g., "urn:foo:club."

Not only can state variables be shared among principals, such as rights suppliers, consumers, and the like, a state variable can be shared among multiple exercisable rights. FIG. 17 is used to illustrate employing of a state variable for maintaining a state shared by multiple rights, according to the present invention. In FIG. 17, a same state variable 1703 is associated to both a right to print 1702 and the right to play 1701, so that the total number of playing, printing, and the like, can be tracked together.

The state of rights can depend on more than one state variable. FIG. 18 is used to illustrate employing of multiple state variables to represent one state of rights, according to the present invention. The example described with respect to FIG. 18 builds upon the example described with respect to FIG. 16. In FIG. 18, a usage right can be tracked by employing multiple state variables 1807 and 1808 in an offer 1801. The state variable 1808, for example, representing a priority level,

US 8,001,053 B2

19

can stay unspecified in the corresponding offers **1802** and **1803** (e.g., site licenses). The corresponding state variables **1809-1811**, for example, used for setting a priority, can be assigned to each member in the corresponding licenses **1804**, **1805** and **1806**. The corresponding right to view, play, and the like, can now be dependent on two state variables, effectively restricting 5 simultaneous views, plays, and the like, per priority level.

One state variable can represent a collection of states. For example, a unique identification can be used to represent a state variable, and an appropriate mechanism can be employed to map such unique id to a database of multiple variables, where each variable represents a distinct state.

The scope of state variables can be used to determine entities by which the state variables can be managed. For example, for a local state variable, usage tracking of associated rights thereof can be managed solely by a trusted agent embedded within a rights consumption environment, such as a media player, and the like. In addition, such usage tracking can be conducted by a trusted remote service, such as the common state-of-rights server **1201**. Further, shared global state variables can be made accessible by multiple trusted agents. To avoid privacy issues, security issues, trust issues, rights issues, and the like, associated with accessing content, such as data, and the like, included within a peer rights consumption environment, managing of such shared global state variables can be performed by a remote service, such as the state-of-rights server **1201**.

A counter is a common form of state variable usage. For example, such state sharing can include counter sharing where a state represents a number of times a right has been exercised, an event has occurred, and the like. Such counter sharing can be manifested in various forms and occur in many contexts, such as: tracking a number of simultaneous uses, tracking a number of sequential uses, sequencing (e.g., a commercial must be viewed before free content can be accessed), a one-time use constraint, a transaction count, a delegation control level, a super-distribution level, dependency on at least one or more services or devices, and the like.

In addition, state variables can be incarnated in a wide variety of forms. For example, a state variable can be used to track specific time slots within a period of time, such as used by a movie studio to transfer syndication rights to a specific TV station, to transfer syndication rights shared by a group of stations, to transfer syndication rights assigned through a bidding process, and the like.

State variables also can be employed, for example, with regional selling or distribution rights, in a statement from a financial clearing house to acknowledge that an appropriate fee has been paid, as a status of whether a commercial has been watched before free content can be accessed, and the like.

Not all rights need be associated with states. FIG. **19** is used to illustrate a case where not all rights are associated with states, according to the present invention. In FIG. **19**, an offer **1901** allows a user, Alice, to grant an unlimited play right, view right, and the like, to her PDA. Such a play right need not be associated with any state. Accordingly, derived right **1902** also has an unlimited play right to the content, as well as the right **1903** for her PC.

Not all rights which are associated with states are shared or inherited. For example, some rights are meant for off-line usage, can be transferred in whole to another device, and hence are not shared with other devices. FIG. **20** is used to illustrate a case where not all rights which are associated with states are shared or inherited, according to the present invention. In FIG. **20**, even though a play right **2003** of a user, Alice,

20

a play right **2002** of a PDA of Alice, and a play right **2003** of a PC of Alice specify a same state variable identification **2004**, a same state need not be shared since each device can track a state thereof locally. Advantageously, such an implementation would allow the PC and the PDA to each play the corresponding content up to 5 times.

FIG. **21** illustrates a form of an offer which does not explicitly include meta-rights. In FIG. **21**, an offer **2101** is configured as a site license written in English. Licenses **2102** and **2103** are instances derived from the offer **2101**. In an exemplary embodiment, variables **2104** and **2105** can be created based on interpretation of the offer **2101**, for example, by the system of FIG. **12**.

The preferred embodiment can utilize various devices, such as a personal computers, servers, workstations, PDA's, thin clients, and the like. For example, the client environment can be a handheld device such as a mobile phone or a PDA. Various channels for communication can be used. Further, the various functions can be integrated in one device. For example, the license server function can be accomplished by software within the client environment. Further, the function of the license server or other modules for making offers, selecting rights and granting licenses can be accomplished in the same device. The disclosed functional modules are segregated by function for clarity. However, the various functions can be combined or segregated as hardware and/or software modules in any manner. The various functions can be useful separately or in combination.

The various elements and portions thereof can be stored on the same device or on different devices. For example, a license can be stored together with, or separate from, content. Further, the various elements of a license can be stored on separate devices. For example the values of state variables can be stored in a state variable repository of a system that tracks the current value of state variables. Various links, references, specifications, and the like can be used to associate the elements.

The invention has been described through exemplary embodiments and examples. However, various modifications can be made without departing from the scope of the invention as defined by the appended claims and legal equivalents.

What is claimed is:

1. A method for sharing rights adapted to be associated with an item, the method comprising:
 - specifying, in a first license, using a processor, at least one usage right and at least one meta-right for the item, wherein the usage right and the meta-right include at least one right that is shared among one or more users or devices;
 - defining, via the at least one usage right, using a processor, a manner of use selected from a plurality of permitted manners of use for the item;
 - defining, via the at least one meta-right, using a processor, a manner of rights creation for the item, wherein said at least one meta-right is enforceable by a repository and allows said one or more users or devices to create new rights;
 - associating, using a processor, at least one state variable with the at least one right in the first license, wherein the at least one state variable identifies a location where a state of rights is tracked;
 - generating, in a second license, using a processor, one or more rights based on the meta-right in the first license, wherein the one or more rights in the second license includes at least one right that is shared among one or more users or devices; and

US 8,001,053 B2

21

associating at least one state variable with the at least one right that is shared in the second license, wherein the at least one state variable that is associated with the second license is based on the at least one state variable that is associated with the first license.

2. The method of claim 1, wherein the state variable in the first or second license inherits a state thereof for content usage or rights derivation from other generated usage rights and meta-rights.

3. The method of claim 1, wherein the state variable in the first or second license shares a state thereof for content usage or rights derivation with other generated usage rights and meta-rights.

4. The method of claim 1, wherein the state variable in the first or second license inherits a remaining state for content usage or rights derivation from other generated usage rights and meta-rights.

5. The method of claim 1, wherein the state variable in the first or second license is updated upon exercise of a right associated with the state variable.

6. The method of claim 1, wherein the state variable in the first or second license represents a collection of states.

7. The method of claim 1, further comprising:

generating in a third license, using a processor, one or more rights from at least one of the usage right and the meta-right in the second license,

wherein the one or more rights in the third license includes at least one right that is shared among one or more users or devices;

associating, using a processor, at least one state variable with the at least one right that is shared in the third license,

wherein the at least one state variable that is associated with the third license is based on the at least one state variable that is associated with the second license.

8. The method of claim 1, further comprising a plurality of state variables that determine the state of the at least one right that is shared in the first or the second license.

9. The method of claim 1, wherein the state variable in the second license is transferred from the at least one right in the first license and is associated with the right that is shared in the second license.

10. The method of claim 1, wherein the plurality of permitted manners of use for the item include copy, transfer, loan, play, print, delete, extract, embed, edit, authorize, install, and un-install the item.

11. The method of claim 1, further comprising:

generating in a further license, using a processor, one or more rights based on the meta-right in the second license, wherein the one or more rights in the further license includes at least one right that is shared among one or more users or devices; and

associating, using a processor, at least one state variable with the at least one right that is shared in the further license, wherein the at least one state variable that is associated with the further license is based on the at least one state variable that is associated with the second license.

12. The method of claim 1, wherein the at least one state variable that is associated with the second license is the same as the at least one state variable that is associated with the first license, if the at least one state variable that is associated with the first license does not identify an unspecified location.

13. The method of claim 1, wherein the at least one state variable that is associated with the second license is assigned

22

a new location identification, if the at least one state variable that is associated with the first license identifies an unspecified location.

14. The method of claim 1, wherein two or more of the specifying, defining, associating, and generating steps may be carried out using a single processor.

15. A system for sharing rights adapted to be associated with an item, the system comprising:

a processor for specifying in a first license at least one usage right and at least one meta-right for the item, wherein the usage right and the meta-right include at least one right that is shared among one or more users or devices;

a processor for defining, via the at least one usage right, a manner of use selected from a plurality of permitted manners of use for the item;

a processor for defining, via the at least one meta-right, a manner of rights creation for the item, wherein said at least one meta-right is enforceable by a repository and allows said one or more users or devices to create new rights;

a processor for associating at least one state variable with the at least one right in the first license, wherein the at least one state variable identifies a location where a state of rights is tracked;

a processor for generating in a second license one or more rights based on the meta-right in the first license, wherein the one or more rights in the second license includes at least one right that is shared among one or more users or devices; and

a processor for associating at least one state variable with the at least one right that is shared in the second license, wherein the at least one state variable that is associated with the second license is based on the at least one state variable that is associated with the first license.

16. The system of claim 15, wherein the state variable in the first or second license inherits a state thereof for content usage or rights derivation from other generated usage rights and meta-rights.

17. The system of claim 15, wherein the state variable in the first or second license shares a state thereof for content usage or rights derivation with other generated usage rights and meta-rights.

18. The system of claim 15, wherein the state variable in the first or second license inherits a remaining state for content usage or rights derivation from other generated usage rights and meta-rights.

19. The system of claim 15, wherein the state variable in the first or second license is updated upon exercise of a right associated with the state variable.

20. The system of claim 15, wherein the state variable in the first or second license represents a collection of states.

21. The system of claim 15, further comprising:

a processor for generating in a third license one or more rights from at least one of the usage right and the meta-right in the second license, wherein the one or more rights in the third license includes at least one right that is shared among one or more users or devices;

a processor for associating at least one state variable with the at least one right that is shared in the third license, wherein the at least one state variable that is associated with the third license is based on the at least one state variable that is associated with the second license.

22. The system of claim 15, including a plurality of state variables that determine the state of the at least one right that is shared in the first or the second license.

US 8,001,053 B2

23

23. The system of claim 15, wherein the state variable in the second license is transferred from the at least one right in the first license and is associated with the right that is shared in the second license.

24. The system of claim 15, wherein the plurality of permitted manners of use for the item include copy, transfer, loan, play, print, delete, extract, embed, edit, authorize, install, and un-install the item.

25. The system of claim 15, wherein a single processor may be used to carry out two or more of the specifying, defining, associating, and generating steps.

26. A device for sharing rights adapted to be associated with an item, the device comprising:

a repository for receiving a first license specifying at least one usage right and at least one meta-right for the item, wherein the usage right and the meta-right include at least one right that is shared among one or more users or devices, the least one usage right defines a manner of use selected from a plurality of permitted manners of use for the item, the at least one meta-right defines a manner of rights creation for the item, said at least one meta-right is enforceable by a repository and allows said one or more users or devices to create new rights, at least one state variable is associated with the at least one right in the first license and identifies a location where a state of rights is tracked; and

a processor for generating in a second license one or more rights based on the meta-right in the first license, wherein the one or more rights in the second license includes at least one right that is shared among one or more users or devices, at least one state variable is associated with the at least one right that is shared in the second license, and the at least one state variable that is associated with the second license is based on the at least one state variable that is associated with the first license.

27. The device of claim 26, wherein the state variable in the first or second license inherits a state thereof for content usage or rights derivation from other generated usage rights and meta-rights.

24

28. The device of claim 26, wherein the state variable in the first or second license shares a state thereof for content usage or rights derivation with other generated usage rights and meta-rights.

29. The device of claim 26, wherein the state variable in the first or second license inherits a remaining state for content usage or rights derivation from other generated usage rights and meta-rights.

30. The device of claim 26, wherein the state variable in the first or second license is updated upon exercise of a right associated with the state variable.

31. The device of claim 26, wherein the state variable in the first or second license represents a collection of states.

32. The device of claim 26, wherein a third license includes one or more rights from at least one of the usage right and the meta-right in the second license,

the one or more rights in the third license includes at least one right that is shared among one or more users or devices,

at least one state variable is associated with the at least one right that is shared in the third license, and

the at least one state variable that is associated with the third license is based on the at least one state variable that is associated with the second license.

33. The device of claim 26, including a plurality of state variables that determine the state of the at least one right that is shared in the first or the second license.

34. The device of claim 26, wherein the state variable in the second license is transferred from the at least one right in the first license and is associated with the right that is shared in the second license.

35. The device of claim 26, wherein the plurality of permitted manners of use for the item include copy, transfer, loan, play, print, delete, extract, embed, edit, authorize, install, and un-install the item.

* * * * *

CERTIFICATE OF SERVICE

I, Daniela Lattes, hereby certify that on October 13, 2016, I caused one copy of the documents listed below:

**CORRECTED BRIEF FOR PLAINTIFF-APPELLANT
CONTENTGUARD HOLDINGS, INC.**

to be filed by CM/ECF with:

Clerk of Court

United States Court of Appeals for the Federal Circuit

717 Madison Place, N.W.

Washington, D.C. 20439

Tel: (202) 275-8000

Fax: (202) 275-9678

and to be served via electronic mail pursuant to Fed. Cir. R. ECF-9(b) on the following:

Robert Unikel

robert.unikel@kayescholer.com

Deanna Keysor

deanna.keysor@kayescholer.com

Kaye Scholer LLP

70 West Madison Street, Suite 4200

Chicago, IL 60602

Telephone: (312) 583-2340

Fax: (312) 583-2543

Michael J. Malecek
michael.malecek@kayescholer.com
Peter E. Root
peter.root@kayescholer.com
Kaye Scholer LLP
3000 El Camino Real
2 Palo Alto Square
Palo Alto, CA 94306
Telephone: (650) 319-4500
Fax: (650) 319-4700

Dan L. Bagatell
Perkins Coie LLP
3 Weatherby Road
Hanover, NH 03755
dbagatell@perkinscoie.com
Telephone: (602) 351-8250
Fax: (602) 648-7150

Neil P. Sirota
neil.sirota@bakerbotts.com
Robert Lawrence Maier
robert.maier@bakerbotts.com
Jennifer Cozeolino Tempesta
jennifer.tempesta@bakerbotts.com
Baker Botts, LLP
30 Rockefeller Plaza, 45th Floor
New York, NY 10112
Telephone: (212) 408-2548
Fax: (212) 259-2548

Terry Duane Garnett
tgarnett@loeb.com
Donald A. Miller
dmiller@loeb.com
Loeb & Loeb LLP
10100 Santa Monica Boulevard, Suite 2200
Los Angeles, CA 90067
Telephone: (310) 282-2199
Fax: (310) 919-3935

Scott F. Partridge
scott.partridge@bakerbotts.com
Michelle Jacobson Eber
michelle.eber@bakerbotts.com
Bradley Bowling
brad.bowling@bakerbotts.com
Baker Botts, LLP
One Shell Plaza, 910 Louisiana
Houston, TX 77002
Telephone: (713) 229-1569
Fax: (713) 229-7769

I declare that I am employed by McKool Smith P.C. at whose direction the service was made.

Executed on October 13, 2016, at New York.

/s/ Daniela Lattes
Daniela Lattes
Senior Paralegal
MCKOOL SMITH P.C.
One Bryant Park, 47th floor
New York, New York 10036
212.402.9499 (t)
212.402.9444 (f)
dlattes@mckoolsmith.com

CERTIFICATE OF COMPLIANCE

I certify that the foregoing Brief for Plaintiff-Appellant ContentGuard Holdings, Inc.:

1. complies with the type-volume limitation of Fed. R. App. P. 32(a)(7)(B).

This brief contains 9,557 words, excluding the parts of the brief exempted by Fed. R. App. P. 32(a)(7)(B)(iii) and Fed. Cir. R. 32(b). Microsoft Word 2003 was used to calculate the word count; and

2. complies with the typeface requirements of Fed. R. App. P. 32(a)(5) and the type style requirements of Fed. R. App. P. 32(a)(6). This brief has been prepared in proportionally-spaced typeface using Microsoft Word 2003 in 14-point Times New Roman type style.

Dated: October 13, 2016

Respectfully submitted,

/s/ Dirk Thomas

Dirk D. Thomas

McKool Smith P.C.

1999 K Street, Suite 600

Washington, DC 20006

(202) 370-8302

*Attorneys for Plaintiff-Appellant
ContentGuard Holdings, Inc.*