12-2009

Digital Rights Management, Fair Use, and Privacy: Problems for Copyright Enforcement through Technology

Eric A. Robinson
University of St. Augustine for Health Sciences, erobinson@usa.edu

Author(s) ORCID Identifier:
https://orcid.org/0000-0001-9554-8754

Digital Rights Management, Fair Use, and Privacy: Problems for Copyright Enforcement through Technology

Eric A. Robinson
San Jose State University
https://orcid.org/0000-0001-9554-8754

Abstract

This article discusses the nature of Digital Rights Management (DRM) systems with regard to the problems they pose to traditional exceptions to copyright restrictions. Problems of fair use and the copying of material for preservation are examined in the context of the architecture of digital rights management systems, and the limitations of current DRM systems in accommodating these policies are examined. The monitoring of usage by the licensing modules of these systems is also criticized for its lack of protection of user privacy and the potential chilling of intellectual freedom. Various potential solutions to these are briefly surveyed with a view of improving DRM and preserving traditional library values.

Introduction

The advent of the Internet has enabled the sharing of information on a level unprecedented in human history. Simple and speedy transferral of digital content has created widely available educational opportunities and the possibility for broader dissemination of vast libraries of cultural content like music, art, and film in electronic forms. This incredible ease of dissemination has enabled file sharing and use on vast scales that have strained traditional interpretations of copyright law and spurred larger media firms to invest in digital technologies for controlling use of electronic files. These technologies, referred to as Digital Rights Management (DRM) systems rely upon computer systems to impose restrictions on the use of digital content that adhere to the wishes of the copyright holders, even in situations where individuals have paid for and own the content in question.

Prior to the Internet era, ownership of content artifacts like books or CDs allowed the user the opportunity for relatively free use of the content. Although copyright law would ostensibly restrict unlawful use of material, in reality, economic factors worked more strongly to inhibit copying and distribution of protected content. It was simply time-consuming to photocopy an entire work, only to be left with a copy of vastly inferior quality, e.g., an unbound ream of loose-leaf pages of poor readability (Coyle, 2004a). The time required for duplication and the unsatisfactory product, thus, previously made duplication less desirable than the digital environment does today. Digital content has drastically changed this precarious economic balance, enabling instantaneous duplication and broad dissemination with no loss of quality. Such capability creates incredible potential to adversely affect the marketability and profit-value of created works.

DRM systems were created in an effort to justifiably to constrain illegitimate duplication and

uses that would violate the legal protections granted to copyright holders, giving the publishers greater control over the use and distribution of their work (Stefik, 1997; Stefik & Silverman, 1997). The nature of these systems (as they currently stand) remains problematic, however, as they are unable to accommodate the subtleties of copyright law. DRM has regularly been criticized for overreaching the bounds of copyright legislation, enforcing a degree of control in excess of the protections granted under copyright law and hindering the exercise of user rights as granted under the law (Mulligan, 2003; Samuelson, 2003). Logical implementation of the rights of use in DRM software typically falls far short of the subtle consideration needed to evaluate legitimate reproduction or dissemination which traditionally would fall under the exceptions to copyright law known as fair use. Further, many of the schemas implemented by DRM controls rely upon user tracking and retention of information regarding content use that could easily be viewed as violation of the privacy of content users.

This paper will examine briefly the nature of Digital Rights Management Systems (DRMS) and the problems they pose to the conventional exceptions to copyright law. It is argued that the current state of DRMS do not make allowance for the exceptional uses guaranteed under copyright law, and that DRMS need to be further developed to allow for anonymous tracking of user information to ensure intellectual freedom.

DRM Architecture

DRM systems are intended to control and restrict illegitimate uses of digital media. In defining and controlling access, the DRMS must coordinate a multiplicity of terms and access rights. These access rights may vary with the class of user or the conditions of use. Typically such conditions will be tied to a financial transaction, such as a purchase, a subscription, or some other licensing agreement. All of these parameters must be coordinated to determine the level of access authorization. If access is granted, limited allowances may be made by the system for utilization of the file.

A variety of types of access rights exist which determine how digital content may be used. The most often encountered rights needed in a digital library setting are *rendering rights.* Rendering refers here to the production of accessible content from an encrypted or controlled file. Rendering can include playback of music, viewing of a video file, and printing or screen-viewing of a text or image file. However, other rights exist that allow transfer of ownership, distribution, or excerption for critical or parodic uses. Each of these possibilities must be explicitly defined and implemented in the DRM.

Erickson (2003) defines a taxonomy of at least four functions which must be served in DRM rendering rights. First and foremost, user actions must be tied to policy-level terms, and any external rendering application must be forced to receive authorization from an evaluation system before allowing content to be rendered. Second, policies must be evaluated by an intermediary system that examines requests and evaluates applicable rules in order to make an authorization determination. Third, governing policies must be in place, defining the rights and conditions of use. Finally these policies must be built into the system and either embedded or linked to the content resource in a machine-readable language. Such systems have been termed “trusted systems” since publishers can assign rights and access conditions and then rely upon the system to enforce those terms (Stefik & Silverman, 1997).

While a wide range of schemata can be utilized to implement DRM, most are composed of variations

on the following generic components which implement the taxonomy described by Erickson (2003): a content server with some sort of DRM metadata packaging; a license server which utilizes rights description metadata to to generate access licenses; and a client server with a DRM controller used to determine access from the license and decrypt content for use in various rendering applications as expressed in the rights agreement. The relationship between these elements and their components, as discussed below, is outlined in Figure 1 (Rosenblatt, Trippe, & Mooney, 2001).

The content server typically houses both the information *content repository*, containing the actual content files, such as music .mp3 files or text .pdf files, and the *DRM packager*. The DRM packager relies upon a database of product content metadata to prepare information for digital distribution (Rosenblatt et al., 2001). This component associates metadata for the identification of a content item, as well information for its discovery, but will also contain a complex description of the rights associated with the item. Depending on the level of sophistication of the system, it may also include statistical tracking for usage monitoring. Metadata preparation may be performed in advance and stored with the content or generated as material is downloaded. These rights descriptions are accessed by other components of the DRMS to evaluate users' rendering rights for viewing, printing, transferring or copying content to the conditions of the agreement, and may even restrict the full digital transferral of the data files, instead requiring the user to view the content in an online or streaming format (Rosenblatt et al., 2001).

The *license server* utilizes the above-mentioned rights descriptions to generate encryption codes or controlled-use licenses for transmission to the client with the content. The *DRM license generator* houses rights information and the codes

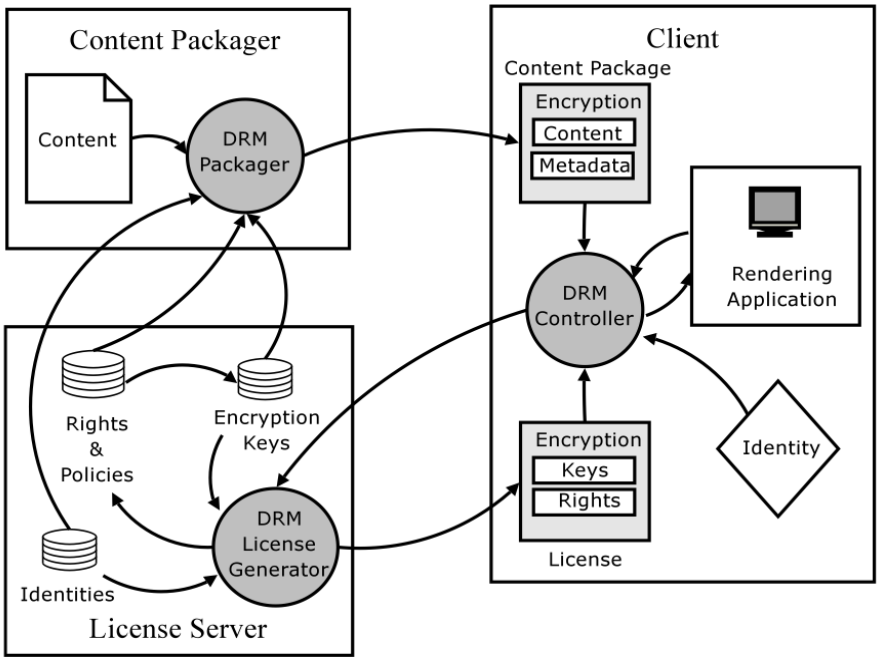

Figure 1: Generic DRM architecture. Source: Rosenblatt et al., 2002.

for the generation of encryption keys used by the DRM to restrict access beyond the legitimate user. In order to ensure that usage is restricted to licensed limitations, DRMs also require an identity store, which houses information on individual use, e.g. the number of pages of an e-book printed by a user, or the time frame allowed for viewing a document.

The rights defined in these two systems are typically expressed in the form of a Rights Expression Language (REL). RELs provide a machine-readable vocabulary for expressing the relationships among data elements and the policy provisions outlining the conditions of use (D. Mulligan & A. Burstein, 2002). These are intended to meet three broad goals: the statement of legal copyright, the expression of contractual language in computer-interpretable form, and the implementation of usage controls (Coyle, 2004b). They thus express licensing controls as a digital formatting of permissions. Outlined in the REL will be the parties of a license or contract, statements of classes of access and usage, and necessary financial transaction information (Coyle, 2004b). These basic relationships are outlined in Figure 2.

These relationships attempt to capture the unique conditions for each potential usage permission that might be encountered. They use explicit conditional statements combined with the rights metadata in the content package to comprise directions for action on digital content packages (Erickson, 2003). Thus, as an example, the DRM system might use an REL to express that user A has paid a $10.00 fee to access an audio edition of Moby Dick, the rights to which are held by Penguin Press. They will also capture time frame or subscription information expressing that for example, the above user will be able to listen to the audio book as many times as he likes within a three-week period. These systems require very precise language to specify the rights and conditions that is *completely unambiguous* in order to be expressed in programming code. As we will see in the discussion of copyright exceptions below, this unambiguous expression creates problems for vaguer notions, such as fair use, which are difficult to model in precise language.

The last element of the DRM architecture, the client, is the system employed on the user- side to render the content. It includes several components for controlling access and decrypting content for

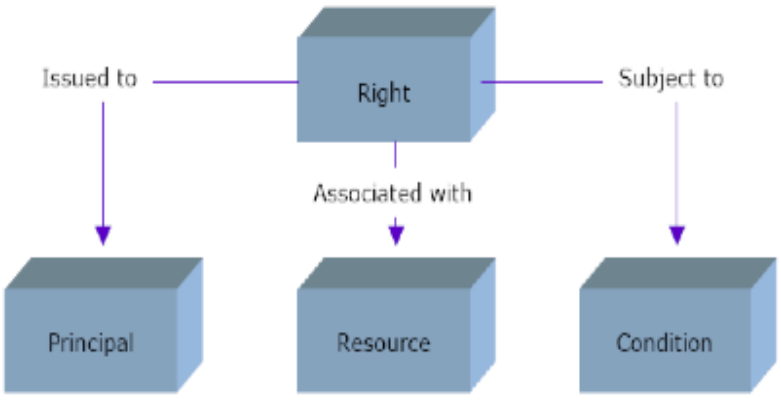

Figure 2: Rights relations in RELs. Source: Coyle (2004b)

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.