A Mechanism for Intellectual Property Rights Discovery in a Digital Content Distribution System

By

Asif Iqbal

Submitted to the department of Electrical Engineering and Computer Science and the Faculty of the Graduate School of the University of Kansas in partial fulfillment of the requirements for the degree of Master of Science.

______________________________________

Dr. Susan Gauch (Chair)

______________________________________

Dr. Jeremiah James (Committee Member)

______________________________________

Dr. John Gauch (Committee Member)

______________________________________

Dr. Robert Buddemier (Committee Member)

Date thesis accepted
To my Parents and Sister
Acknowledgements

I would like to acknowledge the guidance and support of the people who helped me enormously during the course of this thesis.

My committee chair Dr. Susan Gauch for providing me with guidance and motivation for pursuing this thesis. My committee members Dr. Jerry James and Dr. John Gauch for providing me with valuable suggestions and insight and for serving on my committee. Special thanks is due to Dr. Robert Buddemier for funding me throughout my stay at KU and for imbibing a work ethic in me that has served me well.
Abstract

Digital distribution of content is becoming increasingly popular. The flexibility in business models that it provides and the ease of distribution of content are apparent benefits. These advantages of digital content distribution which give it the edge over conventional content distribution also make digital content an easy target for piracy. To prevent piracy Digital Rights Management (DRM) systems have been developed which enforce intellectual property rights. The current DRM systems for digital content distribution lack a mechanism for the consumer to discover intellectual property rights pertaining to content. This thesis addresses this need by designing and implementing a mechanism for discovering intellectual property rights in a digital content distribution system.
TABLE OF CONTENTS

1 INTRODUCTION 1

2 BACKGROUND 3

2.1 Business Models for Digital content distribution 4
  2.1.1 Pay-per-resource or download 4
  2.1.2 Pay-per-use 4
  2.1.3 Subscriptions 5

2.2 Rights Expression languages 5
  2.2.1 XrML (eXtensive rights Management Language) 6
  2.2.2 ODRL (Open digital rights language) 8

2.3 Digital Rights Management systems 10
  2.3.1 Architecture of a typical DRM system 10
  2.3.2 Security in a DRM system 11
    2.3.2.1 Symmetric and Asymmetric encryption 12
    2.3.2.2 Digital certificates 12
    2.3.2.3 Individualization 13
    2.3.2.4 Digital watermarks 13

2.4 Existing systems 13
  2.4.1 Ideas sought to be implemented in this thesis 14
  2.4.2 Comparison 14
    2.4.2.1 Google and IEEE 14
    2.4.2.2 OZAuthors 15
    2.4.2.3 TDB: A Database System for Digital Rights Management 16
    2.4.2.4 A Contract and Rights Management Framework Design for Interacting
    Brokers 16
    2.4.2.5 A Digital Object Approach to Interoperable Rights Management 17
    2.4.2.6 Windows media rights manager 18
    2.4.2.7 IBM Electronic media management system (EMMS) and Real
    networks Helix DRM 19
    2.4.2.8 InterTrust RIGHTS|SYSTEM 20

2.5 IPR discovery - The missing piece 20
3 DESIGN OF AN IPR SEARCH SOLUTION IN A DCD SCENARIO 22

3.1 Choice of parameters 22
   3.1.1 Business models 22
      3.1.1.1 Pay-per-resource 23
      3.1.1.2 Subscriptions 23
   3.1.2 Rights expression language 24
   3.1.3 DRM system used 24

3.2 Requirements for the system 24
   3.2.1 Flexibility of IPR 25
   3.2.2 Management and Validation of licenses 25
   3.2.3 Scalability 25
   3.2.4 Interoperability and reusability 26
   3.2.5 Flexibility of search 26
   3.2.6 Auditability without loss of privacy 26
   3.2.7 Revocation of licenses with immediate effect 27

3.3 System Design 27
   3.3.1 Interaction between consumer and publisher 27
   3.3.2 Interaction between consumer and the search system 28
   3.3.3 Interaction between publisher and search system 30

4 IMPLEMENTATION OF AN IPR SEARCH SOLUTION IN A DCD SCENARIO 33

4.1 DCD simulation programs 33

4.2 Indexing programs 34
   4.2.1 Content indexing program 34
   4.2.2 Offer indexing program 35
   4.2.3 User licenses indexing program 35
   4.2.4 Retrieval program 36
      4.2.4.1 Search for documents owned 36
      4.2.4.2 Search for documents which can be acquired with or without a price limit 38
   4.2.5 License Revocation program 39
   4.2.6 License Audit program 41

4.3 Database schema 42
   4.3.1 Table Name : Offers 42
      4.3.1.1 Fields 42
      4.3.1.2 primary key 43
      4.3.1.3 Non-primary key indexes 43
4.3.2 Table Name: user_licenses
   4.3.2.1 Fields
   4.3.2.2 primary key

4.4 Table Name: user_subscriptions
   4.4.1.1 Fields
   4.4.1.2 primary key

4.4.2 Table Name: revocations
   4.4.2.1 Fields
   4.4.2.2 primary key

5 TESTING

5.1 Test Setup
   5.1.1 General setup
   5.1.2 Hardware setup

5.2 Testing
   5.2.1 Flexibility of IPR
      5.2.1.1 Offers
      5.2.1.2 Licenses
   5.2.2 Validation of licenses
      5.2.2.1 Needle in a haystack test
      5.2.2.2 Load testing on license validation
      5.2.2.3 Types of invalidation
      5.2.2.4 Addition of rights to a license
   5.2.3 Scalability
      5.2.3.1 Indexing scalability
      5.2.3.2 Flexibility of search
      5.2.3.3 Content to which a user owns a license
      5.2.3.4 Content to which a user wants to acquire a license
      5.2.3.5 Rights queried less than the rights offered

5.3 Auditability Test

5.4 Revocation of licenses with immediate effect

6 CONCLUSIONS AND FUTURE WORK

6.1 Scope for future work
   6.1.1 Search Web-service interoperability with DRM solutions
   6.1.2 Content recommendation service based on licenses owned

7 REFERENCES
# Table of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-1</td>
<td>XrML License Structure</td>
<td>7</td>
</tr>
<tr>
<td>2-2</td>
<td>ODRL Foundation model</td>
<td>9</td>
</tr>
<tr>
<td>2-3</td>
<td>Architecture of a typical DRM system</td>
<td>11</td>
</tr>
<tr>
<td>3-1</td>
<td>Interaction between consumer and publisher</td>
<td>28</td>
</tr>
<tr>
<td>3-2</td>
<td>User - Search System interaction</td>
<td>29</td>
</tr>
<tr>
<td>3-3</td>
<td>Publisher-Search system interaction for indexing offers</td>
<td>30</td>
</tr>
<tr>
<td>3-4</td>
<td>Publisher Search system interaction for revoking licenses</td>
<td>31</td>
</tr>
<tr>
<td>3-5</td>
<td>Publisher-Search system interaction for Auditing</td>
<td>32</td>
</tr>
<tr>
<td>4-1</td>
<td>Flowchart – Search for documents owned by a user</td>
<td>37</td>
</tr>
<tr>
<td>4-2</td>
<td>Flowchart – Search for documents sought to be acquired</td>
<td>38</td>
</tr>
<tr>
<td>4-3</td>
<td>Flowchart – License revocation program</td>
<td>40</td>
</tr>
<tr>
<td>4-4</td>
<td>Flowchart – Auditing program</td>
<td>41</td>
</tr>
</tbody>
</table>
1 Introduction

The use of computers has permeated every walk of life imaginable. The world wide web has become ubiquitous contributing to the increasing use of computers. The day is not far when the paper document will become a rare commodity. Electronic publishing is poised to take off in a big way. This is especially on the rise with the speedy development in the field of portable devices. These devices are small enough to be carried around and yet allow the non-geek user to perform any operation that a desktop computer performs. Indeed this invasion of the paper has already started. All major newspapers and magazines have an online version of their publication available. The newspapers, magazines and even books available online represent digital media with intellectual property associated with it. Digital subscriptions are often needed to access the digital media online. These digital subscriptions represent the intellectual property rights granted to a user.

As digital media keeps growing so does the need to express, discover, enforce and comply with IPR (Intellectual Property Rights) related to it. Rights expression languages such as XrML (eXtensive rights Markup Language), ODRL (Open Digital Rights Language) etc. are concerned with expressing IPR. DRM (Digital Rights Management) is concerned with enforcing and ensuring compliance of IPR expressed using rights expression languages. Discovery of IPR however seems to be the one area which is neglected. In one sense DRM systems have to discover the IPR to be
able to enforce them. But mechanisms by which users can discover IPR seem to be missing. IPR discovery by the user is key in any digital content distribution system because users will not buy content if they are not able to discover the IPR related to it in an easy to use manner. This thesis addresses the need for an IPR discovery mechanism to discover digital content and IPR related to it. Search engines are a very popular and effective means of discovering digital content on the web. The IPR discovery mechanism designed in this thesis interoperates with conventional search engines.

The organization of the thesis is as follows:

Chapter 2 explains the background on rights expression languages and Digital rights management systems. A comparison is also presented contrasting the objective of this thesis against some of the systems available today.

Chapter 3 presents the design of the system for IPR discovery. The requirements for the design and implementation are listed followed by the architecture of the components of the system.

Chapter 4 presents the implementation of the system for IPR discovery. The program flow for each of the implemented programs is presented here.

Chapter 5 presents the validation of the test results against the requirements presented in chapter 3.

Chapter 6 presents the conclusions of the thesis and the scope for future work.
2 Background

Digital content distribution is a relatively new method of distributing content. Despite being of relatively recent origin digital distribution of content is fast gaining popularity. Part of this rise in popularity has been fuelled by increasing advances in technology. Portable devices are fast becoming very popular with the consumer. The ease with which digital media can be acquired, stored, organized and preserved has made it an extremely attractive proposition. The ease with which digital media can be distributed has also proven to be a hit with the producers and distributors of digital media.

However the same features which make digital media easy to distribute also make it easier to be copied and shared. This copying while sometimes done well within the principles of fair use, is also abused by illegal file sharing notably over p2p networks. Preventing this illegal sharing of files while at the same time retaining the ease of distribution advantage that digital content offers is a challenge. This challenge has been the motivation behind DRM systems seeking to enforce and seek compliance of IPR. The various components which make digital content distribution an exciting prospect are examined in this chapter.
2.1 Business Models for Digital content distribution

This section examines the various common business models being employed for digital content distribution. Digital content distribution (DCD) also has the advantages of providing so much flexibility in developing business models that it is impossible to present all or even most of the possible models. The models presented in this section are the basic models used for DCD. One salient feature amongst the models presented is that they have been present in some form or the other in non-DCD scenarios as well.

2.1.1 Pay-per-resource or download

In this business model the user obtains some digital content and pays for rights to it while acquiring it. The payment is a one-time payment. There are no additional payments for the number of times the consumer consumes the product after it has been acquired. This model does allow that only limited rights are granted to the user for the one-time acquisition. The consumer may be allowed to read a book acquired as many times as desired but will not be allowed to copy its contents and share them or reproduce them. This model seems well suited for a scenario of book distribution.

2.1.2 Pay-per-use

The pay-per-use business model mandates a payment every time a consumer consumes a resource. This model is more likely to be used in a video or even music
distribution scenario. Variations of this model allow for an upper limit on the number of uses for a specific resource by a consumer beyond which the consumer is not charged a per-use fee.

2.1.3 Subscriptions

Subscriptions grant the consumer rights to a resource with a time period specified during which the granted rights are valid. The resource in this case can be at very levels. A single newspaper article may be considered a resource, as may all the articles which collectively form a daily edition of the newspaper. The flexibility in content distribution becomes obvious here as a consumer may be convinced to acquire an article from an edition of a newspaper much more easily than to obtain an entire newspaper.

Some of the many business models for DCD were presented in this section. While the many advantages including the flexibility of these models and their distribution efficiency are apparent, equally apparent is the need to express IPR related to digital content and to enforce such IPR. These are indeed key to ensure the success of the DCD business models.

2.2 Rights Expression languages

A DCD system requires that IPR be expressed. There are many objectives to be satisfied in designing a rights expression language. It must be flexible enough so that various types of business models can be accommodated. The language must be
extensible to allow for new business models which may spring up in the future and cannot be predicted now. Interoperability is another feature which needs to be supported. Apart from all these, perhaps the most challenging is the ability to express rights in a clear, concise, unambiguous manner. The challenges in fulfilling this last requirement become obvious because the rights expressed need to be machine interpretable as well as to be consistent with the law. This is so since for many principles such as fair use there is no clear definition in law. The interpretations of fair use are made specific to each case based on human judgment [1]. Authoring rights expression languages therefore is a difficult task. This section describes the relevant work with respect to rights expression languages. Two XML based languages seem to be leading the way in the field of rights expression languages. These are XrML (Extensible rights markup language) and ODRL (Open Digital Rights Language).

2.2.1 XrML (eXtensive rights Management Language)

XrML [2] developed by contentguard is an XML based rights expression language. XrML derives its roots from DRPL (Digital Rights protection language). XrML based rights expressions are referred to as licenses. A XrML license primarily expresses four things: The principal to whom the rights are granted, the resource for which the rights are granted, the rights that are granted and the terms and conditions under which the rights are granted. A license hence represents a principal being granted certain rights to a certain resource under certain conditions. The XrML schema consists of a core schema containing all the essential elements which form the
central concepts for XrML semantics. The core schema is supplemented by a standard extension schema which includes concepts that apply to a wide range of usage scenarios but are not core concepts. Domain specific extensions exist to cater to each domain. XML digital signature specification is used to ensure validity of the licenses. A format for a license reproduced from the XrML technical resources section [3].

Figure 2-1: XrML License Structure
2.2.2 ODRL (Open digital rights language)

ODRL provides for expression of digital rights in "open and trusted environments"[4]. ODRL is not concerned with the mechanisms to achieve secure architectures. ODRL is based on XML. It involves a number of core entities and their relationships. The three core entities in ODRL terminology are: assets, rights and parties. Assets refer to the content for which rights need to be expressed. The Rights include Permissions. The permissions can further contain Constraints, Requirements, and Conditions. Permissions are the actual usages or activities allowed over the Assets (e.g. Play a video Asset). Constraints are limits to these Permissions (e.g. Play the video for a maximum of 5 times). Requirements are the obligations needed to exercise the Permission (e.g. Pay $5 each time you Play the video). Conditions specify exceptions that, if become true, expire the Permissions and renegotiation may be required (e.g. If Credit Card expires then all Permissions are withdrawn to Play the video). The Parties include end users and Rights Holders. Parties can be humans, organizations, and defined roles. With these three core entities, the foundation model can then express Offers and Agreements. Offers are proposals from Rights Holders for specific Rights over their Assets. Agreements are when Parties enter into contracts or deals with specific Offers. The model can also then express revoking of any Offers or Agreements. A detailed explanation of ODRL can be obtained from [4].
The foundation model for ODRL is reproduced from [4] below.

![ODRL Foundation Model](image)

**Figure 2-2: ODRL Foundation model**

Both XrML and ODRL represent the state of the art in rights expression languages. Both offer similar functionality with different terminology. A digital magazine distribution scenario described in the XrML technical resources [3] which coincided with the business models sought to be implemented led the author to choose XrML. It is the authors belief that ODRL could prove to be an equally good choice for the implementation. Once rights have been expressed they need to be evaluated and enforced. This is the realm of DRM systems. There are many DRM
architectures using a variety of security schemes to enforce rights and ensure compliance. These are discussed in the next section.

2.3 Digital Rights Management systems

Digital content distribution requires a mechanism to ensure that IPR of a content be enforced. DRM systems control the digital content and also control the usage and distribution of content. The key concept in a DRM system is the license which the consumer has to acquire in order to gain access to content. The DRM system uses the license to validate the user and determine that the user is authorized to access the content being requested. The security mechanism used to secure the license is the prime function of the DRM mechanism. A brief overview of a typical DRM system is provided in this section. The different types of cryptographic mechanisms used to secure content are also discussed. Some of the commercially implemented DRM systems are mentioned. Detailed information can be obtained from [5] and [6].

2.3.1 Architecture of a typical DRM system

The typical DRM system comprises of four entities. The four entities are the content owner, the consumer, the distributor and the clearinghouse. The content owner owns the digital resource. The consumer represents the person interested in
acquiring rights to the content. The distributor provides access to the content after validating a digital license. The clearinghouse issues a digital license to the consumer after the user fulfills a condition (say payment of some amount of money). The distribution and clearinghouse entities may be present as one administrative body. Quite a few variations of the typical DRM systems discussed exist. The block diagram for a typical DRM system is reproduced from [5] below

![Diagram of a typical DRM system](image)

Figure 2-3: Architecture of a typical DRM system

### 2.3.2 Security in a DRM system

Security is the essence of the DRM system. Unlike a typical security system where two parties try to protect a shared secret, the situation is different in DRM systems. In DRM systems one of the entities, the user, cannot be trusted. Security is a process and not a product. Keeping this in mind there are two major factors to keep in mind when designing a DRM solution. The first of these is that a single breach of security should not compromise the whole system. The DVD region protection is a
example of a case where this principle was not followed. The second principle is that the cost of hacking into a specific resource representing digital content must be sufficiently high so as to dissuade hackers from doing so. Security models used in DRM systems are discussed below.

2.3.2.1 Symmetric and Asymmetric encryption

Both symmetric and asymmetric encryption are used to prevent digital content from being copied and to prevent digital licenses from being tampered. Using asymmetric encryption a mathematically related key pair is generated. Content signed with the private key generated will only verify with the corresponding public key. Any malicious attempt to tamper with the digital license will result in the license being rendered invalid. As a result when a tampered digital license is sought to be validated with the content owners public key it will not verify.

2.3.2.2 Digital certificates

Digital certificates are used in DRM systems to verify the identity of the parties involved. Fake certificates can be revoked using a certificate revocation list. This limits the damage that can be caused by a malicious hacker.
2.3.2.3 Individualization

Some DRM systems rely on an unique identification of the device on which the protected content is allowed to be consumed. This scheme though quite effective prevents the user from consuming content that he has acquired on the various devices that he owns. In order to allow for this some systems allow users to backup their licenses and carry them across devices. The number of times that this process of backing up is allowed for a user is limited to prevent malicious users from abusing the system. The Microsoft windows media player makes use of such a process.

2.3.2.4 Digital watermarks

Digital watermarks are imperceptible signals embedded in digital content. The digital watermarks can be used to store license information for digital content within it. Watermarks can also be used to trace piracy of digital content.

2.4 Existing systems

A comparison with existing systems is essential especially in order to stress the uniqueness of the ideas implemented. In this section the ideas sought to be implemented are compared with some of the existing systems. This section tries to establish the concepts that make the treatment in thesis different from existing work in this area.
2.4.1 Ideas sought to be implemented in this thesis

The main objective of this thesis is to provide a framework and implementation which provides a mechanism for IPR discovery in a digital content distribution scenario. The chosen mechanism for IPR discovery is a system which will interoperate with a conventional search engine system. The salient characteristics of such a system can be identified as:

- A search system which indexes IPR along with the content it represents
- A search system which allows users to manage their licenses
- A search system which allows users to search for content based on IPR
- A system which allows publishers to revoke user licenses issued by them
- A system which allows publishers to audit the system

2.4.2 Comparison

This section will attempt a comparison between the system sought to be implemented and other systems implemented based on the features summarized above,

2.4.2.1 Google and IEEE

Google is one of the most popular search engines today. A google search allows users to search by query terms they submit. The results include protected content websites like the IEEE. When a user attempts to click on such a result it leads to the IEEE website page where the user is required to submit his credentials obtained when
payment for access to the website was made. IPR is not being expressed using any rights expression language. None of the other features are present either.

2.4.2.2 OZAuthors

OzAuthors is an online ebook store. It is a publishing venture between IPR Systems and the Australian Society of Authors (ASA) based on the digitization, digital distribution and digital rights management of textual works. All of this is made possible through the underlying infrastructure - Digital Book eXchange (DBX). By utilizing DBX authors can publish, promote, sell and distribute their works online. Content owners are allowed to specify the cost for their content and the rights that they offer for the cost. A number of preview pages can also be specified by the publishers. The information is stored using ODRL. This system also allows multiple contributors to a content to specify the percentage of royalty they are entitled to. A number of security options are also offered including watermarking, preventing printing of content etc. This system addresses the need for content owners to be able to publish their content digitally. The author could not access the website. However from [7] it appears as though users provided a preview of content. Whether the user can only browse for content or also search for content and if so can a search be based on offers or user owned licenses is unclear. If such capabilities exist they are not mentioned in [7]. OZAuthors however is a digital publishing service and not a search service. It may be possible for the system to offer search capabilities but even then such capabilities will only be for content published through the OZAuthors system.
The search system sought to be implemented in this thesis is a generic system which can be used by multiple publishers.

2.4.2.3 TDB: A Database System for Digital Rights Management

In [8] a database system is designed which allows a persistent storage of the users contracts representing their rights in the users machine. It is a database designed for DRM applications and is tightly integrated with C++ programming language. The main aim of this system seems to be to store the data related to the contracts on the users machine in the database designed and prevent the user from tampering with the database. The design focuses on tamper-proofing the database and tamper detection. Such a database is more of a contract management and IPR enforcement solution. It does not focus on IPR discovery and license management (from the users perspective). This differentiates it from the system sought to be implemented in this thesis.

2.4.2.4 A Contract and Rights Management Framework Design for Interacting Brokers

[9] identifies open issues in contract and rights management especially applied to the educational domain. The issues mainly targeted are contract management (supports the phrasing, storage, processing, management, editing, revocation and export of contracts) and mapping the data model of the domain to that of the rights
expression language. The solution is designed for interfacing brokers, for example those in P2P networks. The data model for the domain is mapped by the system in [9] to the ODRL space. Contract enforcement and access control are addressed by a contract and rights management engine. IPR discovery is identified as one of the typical value added services that can be provided. An offer/contract database repository is created which stores the contracts and offers and a contract interpreter is used to interpret the contracts. The system that is implemented is identified as being extensible to provide the additional value added services including IPR discovery that are identified. The focus of this thesis is IPR discovery identified in [9] as one of the value added services that can be supported.

2.4.2.5 A Digital Object Approach to Interoperable Rights Management

The author in [10] presents a need for a higher level of abstraction for securely packaged content in order to supply the metadata required by indexing and search services. The discussion in [10] uses the interoperable repository model developed by CNRI and the Digital Library research group at Cornell, codified as the Repository Access Protocol (RAP) on top of a webservices layer. This thesis addresses the need for a IPR discovery framework as a value added service supplementing a search engine. The author of [10] has consistently presented the need for an IPR discovery mechanism as an essential part of a Digital rights management system.
2.4.2.6 Windows media rights manager

Windows media rights manager encrypts and locks content to be distributed using a key. According to [11] the key is stored in an encrypted license which is distributed separately. The packaged file can be placed on a Web site for download, placed on a media server for streaming, distributed on a CD, or e-mailed to consumers. The content provider chooses a license clearing house that stores the specific rights or rules of the license and implements the Windows Media Rights Manager license services. The role of the clearing house is to authenticate the consumer's request for a license.

Digital media files and licenses are distributed and stored separately, making it easier to manage the entire system. To play a packaged digital media file, the consumer must first acquire a license key to unlock the file. The process of acquiring a license begins automatically when the consumer attempts to acquire the protected content, acquires a predelivered license, or plays the file for the first time. Windows Media Rights Manager either sends the consumer to a registration page where information is requested or payment is required, or "silently" retrieves a license from a clearing house. To play the digital media file, the consumer needs a media player that supports Windows Media Rights Manager. The consumer can then play the digital media file according to the rules or rights that are included in the license. From the information provided by [11] some kind of license management system resides on
the client system. However no search system seems to exist for searching for offers or even to search for content amongst the many licenses owned by a user.

2.4.2.7 IBM Electronic media management system (EMMS) and Real networks Helix DRM

According to [12] "The IBM Electronic Media Management System (EMMS) provides an industry-independent foundation for delivery of digital assets that creates new business models, enables flexible digital rights management (DRM) and helps protect the assets through their entire life cycle". The security in the EEMS system is based on a private key-public key encryption system. A key is needed to unlock the digital content. Keys can be obtained by an authenticated transaction with the EEMS system. The focus on this system as in the Windows media rights manager is on DRM enforcement and discovery of IPR as an integral part of discovery of content seems not to have been addressed directly. The use of a secure container in these systems while providing a reliable security model is also the primary reason for the author in [12] to express the need for a "higher level of abstraction ". The Real networks helix DRM technology seems to be using a similar architecture using secure containers and a license server to unlock these containers. Based on [13] no IPR discovery mechanism or service seems to have been provided.
2.4.2.8 InterTrust RIGHTS|SYSTEM

The InterTrust RIGHTS|SYSTEM system [14] uses an architecture similar to the Windows and IBM EMMS systems. Content is packaged using secure encryption and rights need to be acquired by the user for access. The emphasis is on IPR discovery and the system lacks a mechanism for IPR discovery.

Many of the DRM systems being offered seem to prefer some kind of secure container technology for content distribution rendering it tamper proof. This mechanism for DRM enforcement and content distribution is very good as indicated by the large number of solutions using it. The need for an IPR discovery system has been expressed but no solutions seem to have been implemented to address the issue. Its the authors belief that many of the existing digital content distribution systems would have an enhanced user base if a search service such as the one proposed in this thesis were provided.

2.5 IPR discovery - The missing piece

A DCD system involves various aspects. It needs to support many different business models and be able to adapt to future models which spring up. IPR needs to be expressed for digital content for DCD to be successful. Rights expression languages are key to express IPR for digital content. ORDL and XrML are two of the promising rights expression languages available. Once IPR have been expressed they
need to be evaluated and enforced. DRM systems with secure encryption methods provide for enforcement of IPR. While all the above mentioned components are necessary for the success of a DCD system they are not sufficient. From a consumers point of view its important that discovery of content be seamless and easy. For protected content in a DRM system its important that the consumer be able to discover not only content but also the IPR that go with it. There is no system available currently which allows this. There is also a lack of a system where a user can manage all his licenses by authenticating them once and having a single point authenticated access to all content that he has rights to. Indeed if the consumers are unable to discover content and IPR related to it in an easy way the whole system might fail for lack of enough users.

A search engine is one of the most popular method of discovering content on the internet today. Google in itself has become a phenomenon and it provides answers to a lot of people seeking content on the web. This thesis attempts to extend the search engine by making it a mechanism for IPR discovery in a DCD scenario. Also included are license management features for users. The design and implementation of the system are described in the next two chapters respectively.
3 Design of an IPR search solution in a DCD scenario

Digital content distribution requires an IPR discovery and license management mechanism for its consumers. DRM mechanisms used in DCD are required to enforce the rights expressed in licenses but they do not offer a means of discovering IPR or for managing licenses to the users. Search engines are by far the most effective means of searching for content on the web today. An extension of the search engine to provide IPR discovery and license management to the consumers is an obvious solution that fills the gap that exists in the DCD system. A design of the search system which is used for IPR discovery and license management is presented here.

3.1 Choice of parameters

An important task to be addressed before the system is designed are the parameters of the DCD system that exist and will be utilized by the system. These include a definition of the business models sought to be supported, rights expression language used and the DRM system to be used. These parameters are described in this section.

3.1.1 Business models

The number of different types of content for which DCD can be used is huge. It was decided to limit the focus of this thesis to a certain type of content. The choice
was a digital magazine publication scenario. Even for a particular content for which DCD is used there exist many different business models for DCD. It was decided to narrow down on the number of business models that were sought to be implemented in this thesis. Two of these the pay-per-resource and subscriptions are considered in this thesis.

### 3.1.1.1 Pay-per-resource

In the pay-per-resource model the user obtains a license with specific rights to a resource. For example a consumer obtains the read and print rights to an ebook. There are no time based limitations on the rights obtained through the license in this model.

### 3.1.1.2 Subscriptions

In the subscriptions business model the user obtains a license with specific rights to a resource but these rights are valid only for a specific time period. For example a consumer might obtain the right to read an article between 01-january-2003 and 31-january-2004.
3.1.2 Rights expression language

A rights expression language needs to be chosen in order to express the IPR. XrML and ODRL were considered by the author. XrML was chosen over ODRL because a use case scenario for a digital magazine presented in [3] were coincident with the business model scenario chosen.

3.1.3 DRM system used

The XML digital signature specification is supported in XrML. It was decided that the licenses created would be signed by the content owners private key. Such licenses cold be verified with the public key of the content owner.

With the above parameters chosen a block diagram of the system within which the search system sought to be implemented would have to operate is provided below.

3.2 Requirements for the system

It is essential that a set of requirements be defined for any system before it is designed. The requirements guide the design of the system. They also allow for both the design and implementation to be checked to determine that the requirements have been met. The testing can also be done to this end.
3.2.1 Flexibility of IPR

The IPR related to a document must be allowed to be flexible. All combinations of the two business models chosen to be supported must be allowed. A combination of various rights say read, print, copy and extract may need to be expressed. Subscriptions need to be expressed as licenses too.

3.2.2 Management and Validation of licenses

The consumer must be allowed to upload licenses owned to the system. Updation of licenses by periodic upload of new licenses must be supported. The user must be able to view all managed licenses at any point of time. The system must ensure that the validity of licenses is done perfectly. No invalid licenses must be allowed access.

3.2.3 Scalability

The solution must scale to a huge number of resources, offers, licenses and consumers while having a manageable tradeoff with respect to system resources and performance.
3.2.4 Interoperability and reusability

The design must be such that any conventional search engine can be plugged into this system with minimum changes to accommodate it.

3.2.5 Flexibility of search

The system must allow the consumer the following kinds of searches:

- Content to which a user owns a license
- Content to which a user acquire a license
  - based on certain rights
  - based on a certain price limit

3.2.6 Auditability without loss of privacy

The system must have a mechanism to allow the IP owners to audit it. Auditing refers to being able to compare the license details for a license as stored by the search system with that stored by the IP owner. While doing so no more information than is needed must be revealed and the privacy of the user must be protected.
3.2.7 Revocation of licenses with immediate effect

IP owners must be able to revoke any license they issued at any time with the revocation taking effect across the entire system immediately.

3.3 System Design

With the set of requirements defined, the system was designed to carry out all the requirements in an efficient manner. The system design has been explained in terms of communication interfaces in the system. There are three different communication interfaces which exist in the system. These are: between the consumer and the publisher (henceforth the content owner is referred to as publisher which is the case for the scenario being considered), between the consumer and the search system developed and between the publisher and the search system. These are defined in detail in this section.

3.3.1 Interaction between consumer and publisher

The consumer discovers an offer of choice and contacts the publisher to obtain a license. A payment is made to the publisher for the required amount. Each publisher has a unique private key-public key pair. The publisher creates a license for the offer sought and signs it with the private key. The consumer by this process has a number
of licenses. A block diagram for the interaction between a consumer and a publisher is shown below.

3.3.2 Interaction between consumer and the search system

The consumer registers with the search system and uploads all licenses owned to the search system. The search system stores the license details for each user in a database. The consumer searches for

- Content to which license is owned
- Content to which a license is sought to be acquired
- based on certain rights
- based on a certain price limit
- on a combination of certain rights and a price limit

The consumer periodically uploads new licenses acquired. The consumer also reviews all licenses owned on a periodic basis.

The search system does a search on the query term entered and based on the offer details for the documents retrieved and the license details for the user trying to search, filters the search results. A block diagram for the interaction between a user and the search system is shown below:

![Figure 3-2: User - Search System interaction](image-url)

29
3.3.3 Interaction between publisher and search system

Publishers allow search systems crawler access to the content and the offer files related to content. The search system crawls the offer files in addition to the content and stores the offer details in a database. A block diagram for the interaction between a publisher and the search system is shown below.
The publisher can revoke licenses from the search systems database as shown below:

![Diagram showing the interaction between the Search System, Publisher, Database, Index of User licenses, and Revoked licenses]

Licenseid to revoke

Result of revocation

Search System

Publisher

Licenseid to revoke

Result of revocation

Database

Delete from

Add to

Index of User licenses

Revoked licenses

Figure 3-4: Publisher Search system interaction for revoking licenses

The publisher also looks up license details of licenses issued from the search system database and compares them to the ones in its own database to ensure the search system has not been breached by an invalid license.
This process called auditing is shown in the below

![Diagram of Publisher-Search system interaction for Auditing](image)

Figure 3-5: Publisher-Search system interaction for Auditing

The design for an IPR discovery mechanism was presented in this chapter in terms of the interactions between various interfaces. The design was based on the requirements specified. An implementation of the design is presented in the next chapter.
4 Implementation of an IPR search solution in a DCD scenario

The design of a IPR discovery mechanism as part of a DCD mechanism was described in the previous chapter. The implementation of such a system is described in detail in this section. The programs implemented include the programs needed for the functionality of the search solution and also programs which simulated some conditions of a DCD scenario.

4.1 DCD simulation programs

A number of programs had to be written in order to simulate conditions in a DCD scenario based on the design. The content was represented by a bunch of 100,000 files obtained from the ODP (open directory project). These files were in folders having a unique name with each folder representing a publisher name. The filenames contained the publishers name as prefix. A program had to be written to create one or more offer files at random for each of the 100,000 content files. This program resulted in 130,000 offer files being created in total. Users are needed to test the various programs as they are being developed. A program was developed to create users. This program took as input the name of the user to be created, the number of licenses to be created and created a zip file containing a number of both subscription and non-time bound licenses. These licenses and subscriptions were signed with the private key which was part of a unique key pair. While generating licenses it was
checked to see if a key-pair had been created for the publisher before. If a key-pair had been created for the publisher before then the private key of existing key-pair was used to sign the license generated. If a key-pair did not exist for the publisher a new key-pair was generated and stored in a separate folder.

4.2 Indexing programs

4.2.1 Content indexing program

A conventional search engine solution called lucene was employed to index and retrieve the content files. Jakarta Lucene is a high-performance, full-featured text search engine written entirely in Java. The main features of lucene are reproduced from [15] below:

- Scalable, High-Performance Indexing
- over 200MB/hour on Pentium II/266
- incremental indexing as fast as batch indexing
- small RAM requirements, only 1MB heap
- index size roughly 30% the size of text indexed
- Powerful, Accurate and Efficient Search Algorithms
- ranked searching -- best results returned first
- Boolean and phrase queries

It was decided to use lucene because of all its features as well as to ensure that the interoperability requirement of the design was met. This means that the system
implemented can be plugged into any other conventional search engine system with little or no modifications. It is the authors belief that it is key to keep the content search separate from the IPR search. This will also make it more feasible for the existing search engine solutions to adopt this system since their system will only need to be extended and no major modifications of the existing programs will be required.

4.2.2 Offer indexing program

Once the content has been indexed the offers need to be indexed. XPath queries are used to determine the docid, the rights offered and the price at which they are being offered. One document can have multiple offers with varying rights and prices for each offer. Hence the database table used to store the index of offers has a primary key which includes the columns docid, price, and the rights columns (play, print, copy and extract).

4.2.3 User licenses indexing program

The user is allowed to upload a zip file containing his licenses. The user license indexing program extracts each license file in the zip file. For each of the licenses the validity of the license is verified with the public key of the corresponding publisher. If the license verifies correctly then the revocations table is checked to see if the licenseid has been revoked. If the licenseid passes that test based on the type of the license (subscription license or a non-subscription license) the details of the license
are extracted using XPath. The details common to both types of licenses are the license id, the docid and the username to whom the license is granted. The subscription license in addition contains the begin date and the end date for the subscription. The non-subscription license contains the rights (any combination of play, print, copy and extract). The details from the subscriptions are stored in the user_subscriptions table. The non-subscription licenses are stored in the user_licenses table.

4.2.4 Retrieval program

There are two different scenarios when a user initiates search. The two search scenarios lead to two different paths in the program. The two scenarios and the respective paths are described in this section.

4.2.4.1 Search for documents owned

In this scenario the user searches for content for which the user owns licenses. The program sends the query term to the lucene retrieval program. The lucene retrieval program has been modified to return a comma delimited list of docids which match the query terms specified by the user. The list of docids is then used in a query to the user_subscriptions and the user_licenses table to determine if any of the relevant documents is owned by the user. The list of documents owned by the user is then displayed to the user sorted by relevance of the query term.
A flowchart representing the search for documents owned by a user is shown below:

Figure 4-1: Flowchart – Search for documents owned by a user
4.2.4.2 Search for documents which can be acquired with or without a price limit

In this scenario the user searches for content which can be acquired with specific minimum rights. A flow chart representing the flow of this search program is shown below:

![Flowchart]

Figure 4-2: Flowchart – Search for documents sought to be acquired
A price limit can also be set by the user for content to be searched. The program sends the query term to the lucene retrieval program. The lucene retrieval program has been modified to return a comma delimited list of docids which match the query terms specified by the user. The docids are then used in a query to the offers table to find if any offers exist for any of the docids which offer at least the rights that the user queried. If the user has specified a price limit, the price of the offers is also matched to be less than that input by the user.

4.2.5 License Revocation program

The publisher sends a licenseid to be revoked. The user_licenses and the user_subscriptions tables are searched for the licenseid. If the licenseid exists its details are entered in revocation table. If the licenseid is a subscription the licenseid is entered in the revocations table before it is deleted from the user_subscriptions table. If the licenseid is a non-subscription license the licenseid is entered in the revocations table before it is deleted from the user_licenses table.

The revoked license details are stored in the revocations table in order to prevent malicious users from attempting to upload revoked licenses. Theses licenses will be validated correctly by the public key of the publisher but when the license indexing program checks the revocations table it will detect the revocation. This program is implemented with web interface and also as a web-service. The program used by a publisher to revoke licenses issued in the publishing system can easily call the revocation web-service and the revocation propagates throughout the search.
system immediately. A flowchart representing the program flow for revoking a license is shown below:

Figure 4-3: Flowchart – License revocation program
4.2.6 License Audit program

A flowchart for the license audit program is shown below:

Figure 4-4: Flowchart – Auditing program
The publisher sends a license id to be audited. The program retrieves the details of the license and returns them. This program has a web interface and is also implemented as a web service. The publisher can implement the program which periodically pick a random licenseid from the license issuing system and call the webservices to get the license details from the search system. A comparison of the two which yields no difference will serve as a successful audit of the system.

4.3 Database schema

MySQL has been used as the database server. The database consists of the following tables.

4.3.1 Table Name: Offers

4.3.1.1 Fields

docid varchar
price float
play int
print int
copy int
extract int
4.3.1.2 primary key
   (docid,price,play,copy,extract)

4.3.1.3 Non-primary key indexes
   idx_docid on docid

4.3.2 Table Name: user_licenses

4.3.2.1 Fields

   licenseid varchar
   username varchar
   docid varchar
   play int
   print int
   copy int
   extract int

4.3.2.2 primary key

   licenseid

4.4 Table Name: user_subscriptions

4.4.1.1 Fields

   licenseid varchar
   username varchar
docid varchar
begintime datetime
endtime datetime

4.4.1.2 primary key
 licenseid

4.4.2 Table Name : revocations

4.4.2.1 Fields

 licenseid varchar

4.4.2.2 primary key
 licenseid
5 Testing

The system needs to be tested to ensure that all programs function correctly and efficiently. This also serves to ensure that the requirements defined for the system have been met. The testing done on the system is explained in this chapter.

5.1 Test Setup

5.1.1 General setup

A collection of 100,000 documents from the ODP were indexed using an indexing solution lucene offered by the apache project. Offer files were created with random offers for each file with at least 1 offer per file. The total number of offers generated were about 130,000. 80 users were created with each user having a random set of licenses. The number of licenses per user varied between 10 and 400.

5.1.2 Hardware setup

All tests were run on a desktop PC system with an AMD athlon 1700+ processor (1.4 GhZ) with 256 MB RAM running Redhat 9, jdk 1.4.2, apache 2.4.3 and perl.
5.2 Testing

The testing done was to validate the requirements that were identified in chapter 3. Various aspects of the testing done are presented in this section.

5.2.1 Flexibility of IPR

Rights expression languages are used to express two different types of expressions. The first of these are offer files that a publisher creates in order to express the price and rights that are being offered for content. The second of these are license files which express the rights that have been granted to a particular user for some content. The business scenarios to which the attention in this thesis was focused were of two types. The non-subscription license grants the user rights to content with no time limitation on the rights being granted. The subscription license grants all rights to a user with a time frame specified for which the rights are valid. It was necessary to test out the creation of each of these types of files in the system.

5.2.1.1 Offers

Offer files with varying cost and varying combination of rights were created. Multiple offer files were created for the same document. Specific offer files were chosen to be manually cross-checked with the indexed details created by the system in the offers table. Random search results containing offer details obtained from the retrieval system were also manually verified,
5.2.1.2 Licenses

Subscription and non-subscription licenses with various combinations of rights were created in the test setup. The index created for various users by the user license indexing system and stored in the user_license and user_subscriptions table were manually compared with the license files.

5.2.2 Validation of licenses

License files represent rights granted to the user by the publisher. Invalid licenses were deliberately created and sought to be validated.

5.2.2.1 Needle in a haystack test

A bunch of 400 licenses with just 1 invalid license was tested to verify if the system would identify the 'needle in a haystack'. The system identified the invalid license in this scenario.

5.2.2.2 Load testing on license validation

A large number of licenses (up to 400) were sought to be validated at a time to check that validation worked without error on high load.

5.2.2.3 Types of invalidation

Licenses were invalidated in various different ways to ensure that the system did not miss any particular type of invalidation. The types of invalidation attempted were the following.
5.2.2.4 Addition of rights to a license

- Changing the docid for which the license was issued
- Changing the username on the license
- Changing the time-period in subscription licenses.
- Tampering the modulus and exponent of the digital signature in the license
- Signing a custom created license with an invalid private key-public key pair.

The system detected the invalidation in all the different type of invalidation scenarios presented above.

5.2.3 Scalability

An important characteristic of any system is its scalability. With the huge amount of digital content available and the rate at which it is increasing it is essential that the system proposed be tested for scalability.

5.2.3.1 Indexing scalability

Around 130,000 offer files were created and sought to be indexed. The system indexed the files in approximately 38 minutes. A large number of user licenses (around 400) were sought to be indexed at the same time. The system was able to index the files in real time. It is advisable to limit the number of licenses being
uploaded for indexing to the system at a time to around 200 licenses for instantaneous response. The indexing was performed on the hardware setup mentioned above which indicates that the hardware requirements for the system are unlikely to be monstrously large even for a much larger number of files.

5.2.3.2 Flexibility of search

The system was designed to allow various different kinds of searches by a user. Each of the 80 different users created in the system was tested for the following kinds of searches.

5.2.3.3 Content to which a user owns a license

A license id owned by a user was identified and a search term contained in the document pertaining to the license was manually identified. A search was then performed for the search term identified for the user identified with the option to show only owned documents in the search results. The results were validated if they contained the document identified. The system validated all such searches performed.

5.2.3.4 Content to which a user wants to acquire a license

An offer was identified for a document with specific rights and price. A user was identified who did not own the document. For this user a search was conducted
for a query term present in the document identified. This search was repeated with the various combinations indicated below.

5.2.3.5 Rights queried less than the rights offered

The search was done with the rights queried (say play and print rights) less than the rights offered (say play, print, and copy). The results were validated when the search returned the document identified.

5.2.3.5.1 Rights queried more than the rights offered

The search was done with the rights queried (say play and print rights) more than the rights offered (say play and copy). The results were validated when the search did not return the document identified.

5.2.3.5.2 Price queried less than the price offered

The search was done with the price queried (say $5) less than the price offered (say $10). The results were validated when the search did not return the document identified.

5.2.3.5.3 Price queried more than the price offered

The search was done with the price queried (say $10) more than the price offered (say $5). The results were validated when the search returned the document identified.
5.2.3.5.4 Search before and after the revocation of a license

A licenseid owned by a user was identified. A search term was identified from the corresponding document. A search was made for the user identified with the search term identified and the option for returning only owned documents enabled. It was ensured that the results returned contained the document identified. The licenseid identified was revoked. The search described was repeated. The returned search results did not contain the document identified validating the effect of revocation.

5.3 Auditability Test

Content owners would want to audit the system to check if the details for licenses present in the search systems database are the same as that in the license issuing database. A random licenseid was manually identified and used as input to the auditing program. The results of the auditing program were manually compared with both the search system database and the license itself. All auditability tests validated successfully.

5.4 Revocation of licenses with immediate effect

Content owners must be able to revoke any license they issued at any time with the revocation taking effect across the entire system immediately. A licenseid and the
user to which it belongs was chosen at random. The audit web-service was called with the licenseid. The audit web-service returns the details of the licenseid. The revocation web-service was called with the licenseid. After the revocation web-service returns a successful result the audit web-service was again called with the licenseid identified. A return code of licenseid not found by the audit web-service validates the revocation.
6 Conclusions and future work

A mechanism for IPR discovery in a digital content distribution system was presented in this thesis. The implementation was designed to interoperate with conventional search engines. It is hoped that DRM solutions currently available will adopt such a system for IPR discovery.

6.1 Scope for future work

6.1.1 Search Web-service interoperability with DRM solutions

Web services were developed in the system implemented for revocation and auditing. Web-services could also be developed for the search itself so that commercially available DRM solutions could interoperate with and provide the search service to the consumers.

6.1.2 Content recommendation service based on licenses owned

Based on licenses owned by a user similar content which can be acquired can be identified. A content recommendation service can thus be developed.
7 References


[15] Jakarta Lucene Overview, [online],