Design of ODRL Extension for Rights Control of User-Generated Content

Jeongmin Kim and Hyunsook Chung, *Member, IAENG*

***Abstract*—Recently, many various kinds of authoring tools for creating digital content have been developed and distributed to personal users freely. Amateur users want to implement their ideas and take rewards as the result of sharing their created digital content to other users. To prevent uncontrolled distribution of the user-generated content, a DRM(Digital Rights Management) system is required. However, there is no DRM system that the amateur user can use to control their properties easily. In this paper, we propose an approach to support specifying rights expression and license control management based on the design of ODRL(Open Digital Rights Language) extension. Amateur authors can create various types of license for valid sharing of their UGC by using our license model and management system.**

***Index Terms*—license, ontology, rights, user-generated content**

I. Introduction

UNTIL now, user-created content(UGC) has been considered as the result of fun activity, sharing information, or collaboration in social network among amateur users[1] [2]. In these days, the creative user really enjoys creating digital items and sharing their works with other people on the Web. In addition, most UGC authors wish to make secured packages of their works and distribute them with the attachment of valid licenses [3] because they want to protect their original works and receive appropriate benefits as a result from the distribution of their works[4].

Current DRM systems have functionality for controlling and managing digital rights for use and distribution of digital assets. DRM systems must have a rights control module as a core component, which is implementing a particular rights expression specification[5][6]. The objective of DRM systems has been changed from the strict restriction of use and distribution of digital assets to the consumer-centric use, share, and distribute. According to the rapid evolving of authoring applications like apple iMovie, amateur users can make high quality multimedia contents easily. Further, they distribute and share their own contents by using social networks.

In this circumstance, we found the requirement of amateur users for protecting their digital assets. However, current DRM systems do not support packaging, licensing, usage tracing, and profit sharing of UGC of amateur users. To make the user-centric DRM functionality possible, we found that a license creation and control module, which is the core module of the DRM system, should be user interactive to enable amateur users to create and control the desired multiple types of licenses for the secured distribution of their digital assets [7].

In this paper, we define the semantic-based rights expression and management model for the user-generated content. Unlike the previous XML-based rights expression models, our license management model conceptualizes the internal and external knowledge structures of entities like user, content, role, license, right, constraint, device, and domain by adopting ontological engineering.

After creation of a digital asset, an amateur user can make an instance of the UGC asset class by defining property values about his asset, and then he creates a new license for the asset by specifying the required rights and constraints. Each asset can have one or more licenses with different types, reproduction, distribution, and usage. Based on our semantic license model, we can support a new business model in which users can sell and buy their created digital assets in a secure environment.

The structure of this paper is as follows. First, we introduce drawbacks of the standard XML-based RELs and previous ontological transformations in section 2. Section 3 introduces current ODRL abstract model and section 4 represents the definition of the ODRL extension model. Section 5 describes the system architecture to implement our proposed semantic license model. Finally, section 6 has our conclusions and future works.

II. Related Work

The Moving Pictures Expert Group(MPEG) proposes the specifications of a REL(Rights Expression Language) and RDD(Rights Data Dictionary) as the fifth part of the MPEG-21 standard that formalizes multimedia framework[8]. The MPEG-21 REL is an XML-based formal language that defines the syntax and semantics to specify rights and conditions for users to use digital assert under the controlled ways.

In the MPEG-21 REL data model, the license is the one important element, which contains one or more grants and an issuer. The license is a collection of grants, which allow principals to have rights over resources, issued by one or more issuers. The principal denotes the identification of an entity holding rights to execute digital works under some conditions. The rights granted to the principal specify acts to be performable on resources, such as print, play, move,

Manuscript received July 19, 2016; revised Aug 10, 2016.

Jeongmin Kim is a professor in the Department of Computer Engineering at Daejin University, Pocheon, 11159 Korea.(e-mail:jmkim@daejin.ac.kr)

Hyunsook Chung is a professor in the Department of Computer Engineering at Chosun University, Gwang-ju, 61452 Korea(corresponding author to provide phone: +82-62-230-7307; fax: +82-62-230-0978; e-mail: hsch@ chosun.ac.kr).

delete, and so on[8][9].

MPEG-21 REL can be extended for supporting new business models by defining extensions. However, the new business model of UGC has to support the more functions of consuming, share, distribute, sell, and adaptation performable by original authors as well as second authors. There are some approaches to define the ontology-based models for rights expression through transferring XML-based syntaxes to OWL syntaxes[10].

They create mapping rules between XML and OWL constructs and propose OWL documents corresponding to RELs of MPEG-21 REL, ODRL REL, and Creative Commons REL[11]. However, these ontology models are not appropriate for rights expression and management of UGC because they don't consider the semantic relationships identifiable in creating, editing, selling, sharing, distributing, and consuming UGC.

III. ODRL Core Model

ODRL(Open Digital Rights Language) is started as initial version ODRL1.1 in 2000. ODRL2.0 specification is announced in April 2012 and the latest version, ODRL2.1 specification, is published in March 2015. ODRL is an open international specification for Policy expressions, which means that only an explicitly permitted use may be executed and the other is prohibited by default[12].

Therefore, the central entity of the ODRL core model depicted in figure 1 is Policy which has to have at least one Permission and may have Prohibitions. In other words, a user can perform only actions specified in a Permission contained in an ODRL Policy on a related Asset. Implicitly the other actions are prohibited. The core model of ODRL consists of 8 entities, Policy, Permission, Prohibition, Asset, Party, Duty, Action, and Constraint. These entities define only the limited number of attributes in abstraction level to enable different types of applications adopt the ODRL specification to their implementation[13].

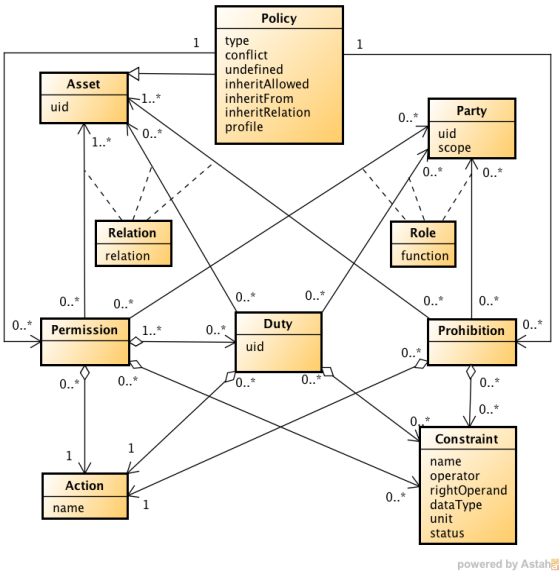

Fig. 1. The ODRL abstract class model. This core model defines 8 basic classes and directed relations between them. The abstract classes have no property or small number of core properties only.

IV. Design of ODRL Extension Model

Our modeling purpose is to support that an author creates multiple types of licenses and a consumer selects a specific license for a purchased digital asset. If there is no license, which a consumer wants to acquire for the purchased asset, he/she can request the author to create a new license. To realize the goal of our approach, we design license lifecycle, ownership transition, controlled sharing and distribution, and second market support. These functions are designed as the extension of the existing basic rights expression languages.

In this section, we describe the ODRL extension model based on the definition of data properties and object properties of ODRL classes. The following tables show property name, data type, and description defined in each class.

TABLE I
Data Properties of Action class

Property	Data type	Description
actionId	URI	Action identifier
actionType	URI	A type of action
actionName	string	A name of action
description	string	Explanation about the action

TABLE II
Data Properties of Asset class

Property	Data type	Description
assetId	URI	Asset identifier
assetType	URI	A type of digital asset
assetName	String	A name of digital asset
creator	URI	Author
resource	URI	An address of digital files of asset
createdDate	datetime	Date and time created an asset
valid	string	Status of validation

TABLE III
Data Properties of Constraint class

Property	Data type	Description
constraintId	URI	Constraint identifier
definition	string	Naming definition of a constraint
operator	token	An operator of a constraint
rightOperand	string	Right operand
datatype	token	Data type of the right operand
unit	token	Data unit of the right operand
status	string	Status of a constraint
description	string	Explanation about a constraint

TABLE IV
Data Properties of Duty class

Property	Data type	Description
dutyId	URI	Duty identifier
definition	string	Naming definition of duty
operator	token	An operator of duty
rightOperand	string	Right operand
datatype	token	Data type of the right operand
unit	token	Data unit of the right operand
status	string	Status of duty
description	string	Explanation about duty

ISBN: 978-988-14047-1-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

TABLE V
DATA PROPERTIES OF PARTY CLASS

Property	Data type	Description
partyId	URI	Party identifier
partyName	string	A name of party
partyType	token	A type of party
valid	string	Status of validation

TABLE VI
DATA PROPERTIES OF GROUP CLASS

Property	Data type	Description
groupId	URI	Group identifier
groupName	string	A name of group
creator	URI	A creator of group

TABLE VII
DATA PROPERTIES OF PERMISSION CLASS

Property	Data type	Description
permissionId	URI	Permission identifier
permType	URI	A type of permission
definition	string	Explanation

TABLE VIII
DATA PROPERTIES OF POLICY CLASS

Property	Data type	Description
policyId	URI	Policy identifier
policyType	URI	A type of policy
conflict	string	Conflict existence
profile	string	Profile description
creator	URI	Creator of policy
numOfAgree	number	The number of agreement

TABLE VIIII
DATA PROPERTIES OF PROHIBITION CLASS

Property	Data type	Description
prohibitionId	URI	Prohibition identifier
creator	URI	Creator of prohibition
prohibitName	string	A name of prohibition
description	string	Explanation

TABLE X
DATA PROPERTIES OF ROLE CLASS

Property	Data type	Description
roleId	URI	Role identifier
role	string	Role description
roleValue	string	Role value

A license can be classified into three categories, which are Consume, Distribution, and Modification, in our semantic license model. Consume license category is composed of usage actions, such as play, display, print, execute, and so on. Distribution license category is composed of sharing actions, such as copy, move, lend, lease, and so on. These actions can be performed with a valid rights expression and control mechanism. Modification license category is composed of reproduction actions, such as modify, convert, extract, combine, and so on. These actions can raise the issue of copyright violation of original works. However, a creator can propose a valid modification license to other consumers with appropriate rewards in our proposed semantic license. If consumers buy the proposed modification licenses, they can reproduce the original contents with valid grants. Table 11 shows action types and actions to be attributed to Permission class.

TABLE XI
KINDS OF USAGE ACTIONS DEFINED IN ACTION TYPES AND DUTY

Action Type	Action	Description
Consume	Play	Play multimedia files.
	Execute	Execute applications or computer programs.
	Print	Print documents or text files.
	Read	Read documents, web pages, or magazines.
	Copy	Copy files.
	Download	Download as files in the storage.
	View	Display the contents of files on the screen.
	Preview	Display previews.
	Domain	Allow device domains.
	CocurrentUse	Use content on the devices simultaneously.
Distribution	Move	Move files to another device. Original file should be removed.
	Sell	Allow sales of files.
	Lend	Borrow files without fee. After a specific period, the files should be returned.
	Give	Give ownership of files to others.
	Present	Send usage licenses to anyone as a gift. The original should be removed and the license should be invalidated.
	Copy	Copy usage licenses to anyone. Original can be remained and used.
	Broadcast	Play or execute files publicly.
	Share	Share files with the authorized consumer group.
	Lease	Borrow files to others with payments. After a specific period, the files should be returned.
Modification	Modify	Modify the partial contents of files.
	Format	Change formats of files.
	Combine	Combine multiple fragments into an integrated content.
	Extract	Extract a part of content.
	Aggregate	Create a content group.
	Annotation	Add explanation to the content.
	Append	Addition of new part into the existing content.
	AttachResource	Connect auxiliary resources to the content.
	Include	Include other content into the content as a part.
	Write	Newly rewrite the content.
Duty	Pay	Make a payment for consuming of the content.
	Inform	Inform the consuming of the content.
	Delete	Delete the content or license.
	Return	Return the content or license.

To define different kinds of relationship(link) types

ISBN: 978-988-14047-1-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

between concepts of the proposed semantic license model we explore two broad categories of link types, structural relations and associated relations. Structural relations form graph structures of ontologies by connecting the related concepts with structural links, such as is-a, subclass-of, instance-of, have/has, container/containee, same-as, part-of, composition, is-sibling, and so on. These links establish direct connections between concepts and do not require any further explanation about the relations. On the other hand, associated relations create semantic links between concepts. These links require additional concepts for describing and explaining the relationships of concepts semantically. Table 12 shows object properties to establish the relationships between classes.

TABLE XII
THE OBJECT PROPERTY FOR DEFINING RELATIONSHIPS BETWEEN CLASSES

Object Property	Domain	Range
hasMember	Group	Party
	Package	Asset
hasRole	Party	Role
hasAsset	Permission	Asset
	Prohibition	Asset
hasAction	Permission	Action
	Prohibition	Action
hasParty	Permission	Party
	Prohibition	Party
hasGroup	Permission	Group
hasConstraint	Permission	Constraint
	Prohibition	Constraint
hasDuty	Permission	Duty
inheritAllowed	Policy	Policy
inheritFrom	Policy	Policy
inheritRelation	Policy	Policy
hasPermission	Policy	Permission
hasProhibition	Policy	Prohibition
isPartOf	Permission	Policy
isPartOf	Prohibition	Policy

V. SYSTEM AND PROCESS DESIGN

In this section, we describe the system architecture to enable our proposed ODRL extension model. As depicted in figure 2, our system is a client-server architecture and is composed of user interface modules of front-end, the engine components of back-end, and data repositories.

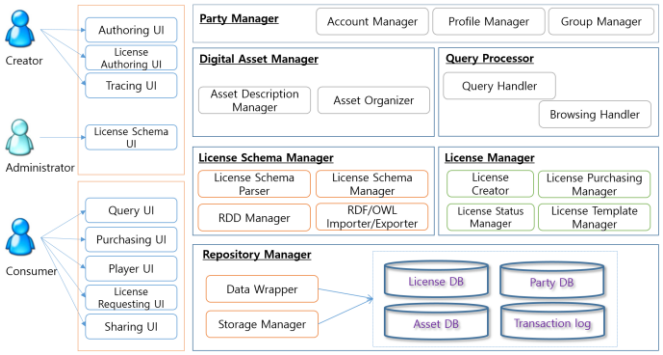

Fig. 2. The system architecture for supporting UGC license management based on the ODRL extension model. The user like digital asset creators, system managers, and asset consumers accesses to the system through user interface modules of the front-end. The back-end of the system is composed of multiple engine components and repositories to provide UGC license services.

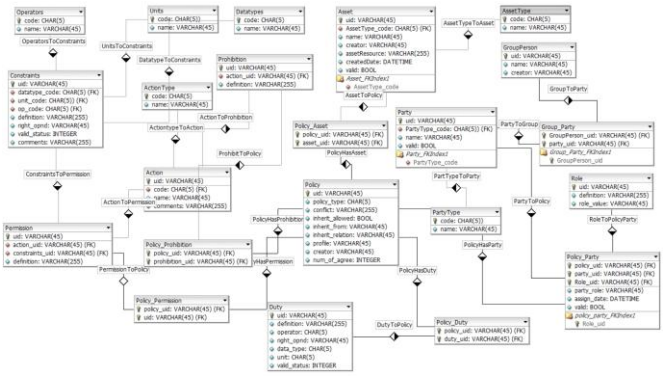

Fig. 3. Physical table diagram for storing asset and license data.

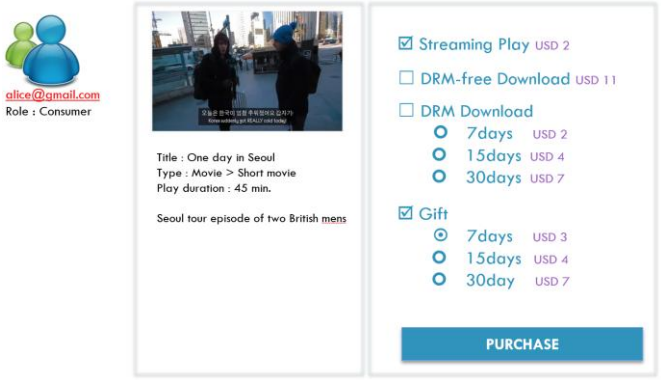

Fig. 4. A consumer, Alice, chooses a wanted asset and reviews multiple kinds of licenses. She takes two licenses of which one is streaming play and the other is gift.

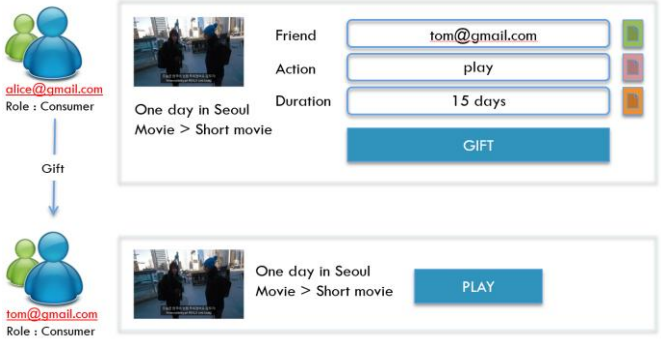

Fig. 5. A user interface for transferring a license of the asset to a friend as gift. Alice sends the purchased license to her friend, Tom, as gift. After transferring, Alice's license should be removed.

To construct repositories of assets and licenses we have designed database schema as depicted in figure 3. The entity-relation model shown in figure 3 implementsclassess and property model described in section 4.

Figure 4 shows an example of user interface for choosing licenses of the retrieved asset. A consumer can retrieve and select the desired asset from the list of assets by using query interface and processing engine component. The system displays all licenses attached to the selected asset. The consumer purchases one or more licenses consume the selected asset.

Figure 5 represents a user interface for sending a license for one user to his/her friend as a gift. In the scenario, Alice sends a license, which can play a movie, with her friend, Tom. After completing of license transfer, Tom can play the movie. But, Alice cannot enjoy the movie because her license was already removed from her has purchased license list.

For every transaction, instances of classes are generated,

ISBN: 978-988-14047-1-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

managed, and stored into repositories. Figure 6 shows an example of instance graph, which represents license transfer between consumers.

VI. Conclusion

Current DRM-based media service platforms are closed systems. They do not accept UGC created by amateur users as their assets. Therefore, interoperability of DRM systems as well as support of UGC-based business model is an important issue that should be handled in the near future. Our approach represents the meaningful early stage of rights expression and management for creating the semantic map of UGC. In this paper, we described the semantic-based rights expression and management model for the user-generated content. The contribution of our proposed model is the invention of a new semantic model, which supports selling and buying user created contents under control of different types of licenses. Further, we concentrate the development of the complete ontology model to support a marketplace for user created content and the related intelligent services.

References

[1] P. O'Connor, "User-Generated Content and Travel: A Case Study on TripAdvisor.com", In Proceedings of Information and Communication Technologies in Tourism, pp. 47-58, Innsbruck, Austria, 2008.
[2] V. Dhar and E. A. Chang, "Does Chatter Matter? The Impact of User-Generated Content on Music Sales", Journal of Interactive Marketing, Vol23, No.4, pp. 300-307, 2009.
[3] L. Nixon, "MediaMixer: Innovative Multimedia Solutions ready for transfer to industry", The Journal of Innovation Impact, Vol.7, No.1, pp. 266-274, 2014.
[4] James Won-Ki Hong,Young-Mi Shin, Myoung-Sup Kim, Jae-Young Kim, Young-Ho Suh, "Design and implementation of a distributed multimedia collaborative environment", Cluster Computing, Vol.2, Issue.1, pp. 45-59, 1999.
[5] W. Rosenblatt, W. Trippe and S. Mooney, "Digital Rights Management: Business and Technology", M&T Books, New York, 2002.
[6] S. R. Subramanya, K. Y. Byung, "Digital Rights Management", IEEE Potentials Magazine, Vol.25, No.2, pp. 31-34, 2006.
[7] Giuseppe Di Modica, Orazio Tomarchio, "Matching the business perspectives of providers and customers in future cloud markets", Cluster Computing, Vol.18, Issue.1, pp. 457-475, 2015.
[8] G. Rust and C. Barlas, "The MPEG-21 Rights Data Dictionary", IEEE Transactions on Multimedia, Vol.7, No.2, 2005.
[9] H. Ueno, M. Sumita, and N. Ishikawa, "Standardization of OMA DRM-Technologies for Handling Digital Content Distribution and Copyright Management", NTT DoCoMo Technocal Journal, Vol.6, No.4, pp. 43-50, 2005.
[10] R. García, R. Gil, and J. Delgado, "A Web Ontologies Framework for Digital Rights Management", Journal of Artificial Intelligence and Law, Vol.5, No.2, pp. 137-154, 2007.
[11] E. Rodriguez and J. Delgado, "Towards the interoperability between MPEG-21 REL and Creative Commons licenses", in Proceedings of AXMEDIS'06, 2006.
[12] C. Canali, M. Colajanni, and R. Lancellotti, "A Two-level distributed architecture for the support of content adaptation and delivery services", Cluster Computing, Vol.13, Issue.1, pp. 1-17, 2013.
[13] R. Iannella, S. Guth, D. Paehler, and A. Kasten, "ODRL Version 2.1 Core Model", Final Specification, https://www.w3.org/community/odrl/model/2-1, 2015.

ISBN: 978-988-14047-1-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)