行政院國家科學委員會專題研究計畫 成果報告

給隱私權保護與著作權管理使用的語意式web規範互通語言

研究成果報告(精簡版)

計畫類別：個別型
計畫編號：NSC 98-2221-E-004-009-
執行期間：98年08月01日至99年10月31日
執行單位：國立政治大學資訊科學系

計畫主持人：胡毓忠

計畫參與人員：碩士班研究生-兼任助理人員：楊協達
碩士班研究生-兼任助理人員：楊竣展
碩士班研究生-兼任助理人員：吳建輝
碩士班研究生-兼任助理人員：郭弘毅

報告附件：國際合作計畫研究心得報告

處理方式：本計畫可公開查詢

中華民國 99年12月27日

附件一

行政院國家科學委員會補助專題研究計畫 成果報告

（計畫名稱）

給隱私權保護與著作權管理使用的語意式web規範互通語言

Semantics-enabled Web Policy Interchange Format（SewPIF）for Privacy Protection and Digital Rights Management

計畫類別：■個別型計畫　□整合型計畫
計畫編號：NSC　98－2221－E－004－009
執行期間：2009年8月1 日至2010年10月 31 日
執行機構及系所：國立政治大學資訊科學系

計畫主持人：胡毓忠
共同主持人：
計畫參與人員：郭弘毅、吳建輝、楊竣展、楊協達

成果報告類型(依經費核定清單規定繳交)：■精簡報告　□完整報告

本計畫除繳交成果報告外，另須繳交以下出國心得報告：
□赴國外出差或研習心得報告
□赴大陸地區出差或研習心得報告
■出席國際學術會議心得報告
□國際合作研究計畫國外研究報告

處理方式：除列管計畫及下列情形者外，得立即公開查詢
　　　　　□涉及專利或其他智慧財產權，□一年□二年後可公開查詢

中　華　民　國 99 年 12 月 6　日

附件二

國科會補助專題研究計畫成果報告自評表

請就研究內容與原計畫相符程度、達成預期目標情況、研究成果之學術或應用價值（簡要敘述成果所代表之意義、價值、影響或進一步發展之可能性）、是否適合在學術期刊發表或申請專利、主要發現或其他有關價值等，作一綜合評估。

1. 請就研究內容與原計畫相符程度、達成預期目標情況作一綜合評估

■達成目標

□　未達成目標（請說明，以 100 字為限）

□　實驗失敗

□　因故實驗中斷

□　其他原因

說明：

建構一個 SemPIF 語意式電腦規範互通的架構, 在這個 SemPIF 的規範架構平台，電腦的規範可以運用語意式規範語言來加以表示。我們可以更進一步利用 meta-PIF 的概念（也就是規範的規範）來提供規範之間整合時的管理與優先度的設定以便於規範之間衝突無法解決的窘境。我們落實個人資料分享與保護及著作權數位內容保護可以利用這個 SemPIF 的規範架構平台來實踐。

我們已經將研究成果發表於 2 項知名的國際研討會，RuleML09 與 Web Intelligence (WI) 10 與一個和語意網有關的專書 Introduction to the Semantic Web: Concepts, Technologies and Applications 篇章。另有一篇論文已完成投稿的工作，而另外一篇論文則在修定並且準備再投稿。

2. 研究成果在學術期刊發表或申請專利等情形：

論文：■已發表 ■未發表之文稿 □撰寫中 □無

專利：□已獲得 □申請中 □無

技轉：□已技轉 □洽談中 □無

其他：（以 100 字為限）

（已經發表的論文與已經投稿的論文請參考附錄）

Challenges for Rule Systems on the Web", The International RuleML Symposium on Rule Interchange and Applications (RuleML 2009), Las Vegas, Neveda, USA, Nov. 5-7, 2009, Springer-Verlag, LNCS 5858

Hu, Y. J. and H. Boley, SemPIF: A Semantic Meta-Policy Interchange Format for Multiple Web Policies, 2010 ACM/IEEE Web Intelligence (WI) Conference, Aug. 31-Sep. 3, 2010.

Hu, Y. J., Unifying Semantic Privacy Protection Web Policies for Digital Rights Management (DRM) System, (Book Chapter), Introduction to the Semantic Web: Concepts, Technologies and Applications, iConcept, 2010.

Hu, Y. J. and Jiun-Jan Yang , A Semantic Privacy-Preserving Model for Data Sharing and Integration, The International Conference on Web Intelligence, Mining and Semantics (WIMS'11) , 2011 (submitted).

Hu, Y. J. et al., Reasoning Personal Privacy Intentions on the Social Web", 2010, (In Revision).

學生碩士畢業論文：

郭弘毅、使用本體論與規則執行企業隱私保護規範

吳建輝、個人隱私揭露意願之推論

3. 請依學術成就、技術創新、社會影響等方面，評估研究成果之學術或應用價值（簡要敘述成果所代表之意義、價值、影響或進一步發展之可能性）（以 500 字為限）

本研究案的學術成果的意義與價值在於解決 Web(全球資訊網) 資料流通、分享、與保護的問題。並且能夠透過語意網的本體論和規則兩大知識系統表達與具體落實執行能力讓電腦軟體可以正確解讀出具有語意式的 Web 規範。利用標準的 Web 語言如 OWL, RIF 等來表示隱私權保護與著作權保護的規範可以讓規範所需要表達語意可以更明確以避免混淆不清的窘境。除了可以提升原有 XML 為基礎的權力表達語言如 ODRL, XrML, P3P, XACML 的缺失之外也可以讓現有的規範語言如 Rein, KaOS 可以透過這個 SemPIF 互通語言的架構來達成互通與規範整合的目的。

語意式電腦規範的價值在於找出人類法治規範可以落實到 Web 環境中自動化執行的概念與規則讓電腦系統可以有效且自動化的來解讀和執行以避免完全用人類手動式處理規範的困境。除此之外可以達成資訊有效分享與保護的目的，系統可以在資料分享與保護時在事先與進行中的程序來加以檢驗，避免事後的冗長訴訟程序與資料不當揭露與使用所產生的副作用。

本研究案的進一步發展可以將參照個人資料保護法與著作權保護法等法治規範將相對應的語意式電腦規範運用到雲端運算的環境中，來解決雲端環境資料分享與保護的目的，讓雲端中資料的個人資料保護與數位內容的著作權保護可以透過此電腦規範的表示與執行得到適當的解決。

附件四

國科會補助專題研究計畫項下出席國際學術會議心得報告

日期:99_年 12_月 07_日

計畫編號	NSC 98－2221－E－004－009		
計畫名稱	給隱私權保護與著作權管理使用的語意式 web 規範互通語言		
出國人員姓名	胡毓忠	服務機構及職稱	國立政治大學資訊科學系教授
會議時間	2010 年 8 月 31 日至 2010 年 9 月 3 日	會議地點	加拿大多倫多市
會議名稱	(中文)IEEE/WIC/ACM Web 智慧與智慧型代理者技術國際研討會 （英文）IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology		
發表論文題目	(中文)提供多重 Web 規範交換的語意式 meta 規範格式 (英文) SemPIF: A Semantic Meta-Policy Interchange Format for Multiple Web Policies		

一、參加會議經過

本研討會於 2010 年 8 月 31-9 月 3 日於加拿大多倫多市的 York 大學舉行。York 大學位於多倫多市近郊約 1 個多小時車程的郊區，附近的大學則有位於多倫多市世界知名的多倫多大學。

8 月 31 日的研討會首日進行的是 Workshop。9 月 1 日至 9 月 3 日則是正式的研討會。研討會分成兩部分來進行：Web Intelligence, Intelligent Agent。在這個基礎上論文的發表大致上分成幾個 track 來進行論文的發表。Web Information Retrieval and Filtering, Web Mining, Semantics and Ontology Engineering, Social Network Analysis, Web Services, Distributed Problem Solving, etc。從這些 tracks 的觀察我們可以知道這個研討會涵蓋面非常的廣，但是主要著重的還是 Web Mining 以及 Web Information Retrieval。計畫主持人發表論文 SemPIF: A Semantic Meta-Policy Interchange Format for Multiple Web Policies 所排定的時段是在 9 月 3 日早上的 Semantics and Ontology Engineering IV 的場次。我也順便主持了這個場次其他演講者的論文發表。本場次的演講共有 5 位的論文發表者，其中有一位大陸東北大學的研究學者並未出席，另外德州大學的論文發表者則請人代打。另外則有京都大學、多倫多大學、及本人（代表政治大學)。本研討會在 9 月 2 日晚上舉辦餐宴於多倫多港上的遊輪上，台灣也有不少的學者參與此盛會。

二、與會心得

對於本研討(IEEE/WIC/ACM Web 智慧與智慧型代理者技術國際研討會)會計畫主持人在過去幾年都是這個研討會 Web Intelligence (WI)的論文評審委員(Program Committee)。這個研討會雖然在語意網和智慧型代理者的知名度沒有 International Semantic Web Conference (ISWC), World Wide Web (WWW), 以及 Autonomous Agent and Multi-Agent System (AAMAS)來得大。但是根據往年和今年的經驗總論文的投稿數量也都可以達到 300-400 篇左右。主要的投稿來源還是來自於大陸的學者以及分佈在世界各地大學與研究單位的大陸學者。這個國際研討會的論文有被收錄到 IEEE Explore 的數位圖書資料庫，因此這也是吸引人投稿的因素之一。除此之外，本國際研討會也能夠每年分別在世界的五大洲選定國家來舉辦。整體來說研討會的論文主要議題還是在 Web Intelligence (WI)的部分，因此對於 Intelligent Agent 這一部份的學術成果發表顯得比較弱勢。實際上論文審稿作業也是 Web Intelligence, Intelligent Agent 分開審稿的方式來進行。因此其審稿委員的聘任和論文的挑選與評定也是分別去進行的。除此之外，在這個研討會發表的論文如果有獲得前幾名的殊榮還有機會在修稿並延伸之後在 WIAT 的期刊刊登。

三、考察參觀活動(無是項活動者略)

無。

四、建議

這幾年隨著大陸經濟起飛與學者國際化參與意願的提升，使得各項研討會都可以看到大陸學者的大量參與並且發表論文。除了他們在過去幾年在海外求學之後定居於當地形成一個有利的學術網絡之外，大陸的內陸學者只要有意願並且經費許可的情況之下都會主動的出國並且參加。除此之外，世界知名的電腦科學研討會也陸陸續續在組織運作上佔有很重要的一席之地，這是我們台灣學者必須要體認的一項事實。

五、攜回資料名稱及內容

研討會論文議事錄一本與全論文集光碟片一張。

六、其他

Challenges for Rule Systems on the Web

Yuh-Jong Hu[1], Ching-Long Yeh[2], and Wolfgang Laun[3]

[1] Emerging Network Technology (ENT) Lab.
Department of Computer Science
National Chengchi University, Taipei, Taiwan
hu@cs.nccu.edu.tw
[2] Department of Computer Science Engineering
Tatung University, Taipei, Taiwan
chingyeh@cse.ttu.edu.tw
[3] Thales Rail Signalling Solutions
GmbH, Austria
wolfgang.laun@gmail.com

Abstract. The RuleML Challenge started in 2007 with the objective of inspiring the issues of implementation for management, integration, interoperation and interchange of rules in an open distributed environment, such as the Web. Rules are usually classified as three types: deductive rules, normative rules, and reactive rules. The reactive rules are further classified as ECA rules and production rules. The study of combination rule and ontology is traced back to an earlier active rule system for relational and object-oriented (OO) databases. Recently, this issue has become one of the most important research problems in the Semantic Web. Once we consider a computer executable policy as a declarative set of rules and ontologies that guides the behavior of entities within a system, we have a flexible way to implement real world policies without rewriting the computer code, as we did before. Fortunately, we have de facto rule markup languages, such as RuleML or RIF to achieve the portability and interchange of rules for different rule systems. Otherwise, executing real-life rule-based applications on the Web is almost impossible. Several commercial or open source rule engines are available for the rule-based applications. However, we still need a standard rule language and benchmark for not only to compare the rule systems but also to measure the progress in the field. Finally, a number of real-life rule-based use cases will be investigated to demonstrate the applicability of current rule systems on the Web.

1 Introduction

The RuleML Challenge competitions started in 2007[1], so the RuleML-2009 Challenge will be the third year for the rule system competition. We offer participants the chance to demonstrate their commercial and open source tools, use cases, and applications for rule related technologies. For the past two RuleML Challenge

[1] RuleML-2007 Challenge, http://2007.ruleml.org/index-Dateien/Page787.htm

G. Governatori, J. Hall, and A. Paschke (Eds.): RuleML 2009, LNCS 5858, pp. 4–16, 2009.

competitions, only a minimum set of requirements was given for evaluating the submitted demo systems. The criteria were that declarative rules should have to play a central role in the application, and that the demo systems should preferably be embedded into a Web-based or distributed environment, etc. The Challenge winners were selected and 1st and 2nd places were awarded with prestigious prizes.

The RuleML-2009 Challenge follows similar processes and the evaluation criteria are the same as in the previous two events. But we consider inviting more participants to submit their rule related systems in this year. In the RuleML-2009 Challenge, we organize events as two tracks, one is by invitation, to demonstrate a commercial or open source environment for its rule systems, and the other is open to general public for a real system competition. In addition to the demo systems with reports submitted to the RuleML Challenge website[2], it is also possible to submit demo papers describing research and technical details, and the selected papers will be published in additional special Challenge proceedings, such as CEURS. A final selection of revised papers from the Challenge proceedings will be resubmitted to a special issue of a journal for publishing. In this RuleML Challenge survey paper, we point out the possible research and implementation challenges for rule systems on the Web that are related to the Challenge competition events in the forthcoming years.

1.1 Challenges for Rule Systems

Rules as human understandable policies are everywhere in our daily life to impose human behaviors. For example, before you take a flight, you need to read airline check-in and boarding time rules in the policy statement of your booking itinerary receipt. If you violate any rule you might miss your plane. Related situations in this scenario of using rules are early-bird conference registration, special discount hotel reservation, payment and refund policies, etc. These rules as policies are represented as human understandable natural language. However, we still need to transform these natural language policies into computer programming rules for computer system understanding and automatic execution. Sometimes, not all of the rules imposed on a human are necessarily and possibly represented as software programs to accomplish automatic execution in our computer systems. Usually, these rules restrict only human behavior, without direct connection with any software system. For example, we have law for privacy protection and digital rights management but not all of privacy rights and digital rights for human are required to be represented and evaluated in computer systems.

There are several challenges while implementing rule systems on the Web. Rules should be allowed to cope with the data model, such as RDB/OO-DB, or a knowledge base, such as an ontology, to permit query and modification services on the data models. Policies imposed on human behavior are declared in some policy language by the combination of rules and an ontology (or database), and these policies can be automatically interpreted and executed by a computer.

[2] http://ruleml-challenge.cs.nccu.edu.tw

There should be a standard language and framework for rule systems to enable rule interchange services on the Web. A certain number of use cases are easily represented and executed by rule and ontology reasoning engines with rule interchange and ontology merging standards to ensure rule interoperability and ontology compatibility.

In the early computer development stage, imperative programming languages such as C and Java were used to represent rules and execute them on a computer system. But these rules are inflexible and not easy to maintain when they are distributed on the Web and require interchange and integration between rule systems. Moreover, imperative programming languages are not appropriate to express concepts of human policies as computerized rules. Recently, people use declarative programming to specify the rules and execute them automatically, where XML is used as a standard syntax representation for interchange of declarative rules, such as RuleML [1], RIF [2], etc.

Even though an XML-based standard rule language and framework provides rule interchange service, pure XML cannot specify a well-defined semantics for rules. So people in the standard rule community constructed a logic foundation behind rule languages and their framework, to preserve the integrity of syntax and semantics of rules interchange for various rule systems. Similarly, OMG SBVR intends to define the vocabulary and rules for documenting semantics of business vocabulary, facts, and rules, as well as an XMI schema for interchange of business vocabularies and rules among organizations and between software tools[3].

In this paper, we first introduce the classification of rules, then, in section 2, we address the issue of rules, and databases and ontologies . In section 3, the current status of a declarative policy as the combination of ontology and rules will be introduced. In addition, Semantic Web Service (SWS) processes also require a declarative policy to express and execute Web Service rules to control information sharing and service execution. In section 4, we examine current different rule management systems and engines. In section 5, we investigate different rule interchange languages. In section 6, we look into the use cases that are possibly represented and executed by the rule systems. Finally, we conclude this study in section 7.

2 Rule and Data Model

2.1 The Classification of Rules

Rules are classified as three types: deductive rules (or derivation rules), normative rules (or integrity rules), and reactive rules (or active rules). One can use deductive rules and facts to trigger a forward or backward reasoning engine to derive implicit facts. Normative rules pose constraints on the data or on the business logic to ensure their consistency in the database or knowledge

[3] http://www.omg.org/spec/SBVR/1.0/

base. Without reactive rules, we cannot update a database or knowledge base by using deductive rules only.

Reactive rules are further subdivided into event-condition-action (ECA) rules and production rules. ECA rules are rules of the form *ON Event IF Condition DO Action*, where *Action* should be executed if the *Event* occurs, provided that the *Condition* holds. Production rules are rules of the form *IF Condition DO Action*, where *Condition* queries the working memory containing the data on which the rules operate. *Action* should be executed whenever a change to the underlying database makes *Condition* true [2].

In reactive rules, we verify the satisfaction of conditions and also execute the action whenever message arrival or timer event triggers the rule. Declarative rules extend their executive power by the combination of rule semantics and imperative programming in the action part.

2.2 Rules and Databases

As early as 1980, Ullman pointed out the principles of the integration of database and knowledge base systems [3] [4]. The foundation of database is relation algebra with SQL as a declarative database query language. However, first order logic (FOL) was also proposed as a way to represent "knowledge" and as a language for expressing operations on relations. The roots of relational theory is logic, and so we cannot deny that the foundation of relational DBMS is based on logic [5]. The simplest data model of FOL is "Datalog", which was coined to suggest a version of Prolog suitable for database systems where it does not allow function symbols in Datalog's predicate arguments. In the IDEA methodology, deductive rules and reactive rules were built on top of the object-oriented (OO) database as a way to express operations on the OO data model [6].

2.3 Rules and Ontologies

Concepts of the Semantic Web have been proposed by Tim Berners-Lee et al. since 2001 [7]. Graph-based RDF(S), including RDF and RDF-schema were the first standardized ontology languages to represent an ontology's schema and instances. Then, standardized ontology languages based on Description Logic (DL) [8], i.e., OWL-DL (later OWL 2), enhanced RDF(S) that plays the major role of knowledge representation for the Semantic Web. However, the logic program (LP) rule language was also introduced because of the limited expressive power of a DL-based ontology language in some situations, such as property chaining, and the manipulation of events, states, and actions.

Initially, the "rule" layer was laid on top of the "ontology" layer in the Semantic Web layered architecture but it has undergone several revisions reflecting the evolution of layers[4]. The most recent layered architecture of rule and ontology layers is one where they sit side by side to reflect their equal status but with some basic assumption differences between ontology and rule, such as the open world

[4] http://www.w3.org/2007/03/layerCake.svg

assumption (OWA) vs. the closed world assumption (CWA), or the non-unique name assumption (non-UNA) vs. the unique name assumption (UNA) [9].

It will be a challenge to resolve these basic assumption differences when we combine rule and ontology to execute rule systems on the Web.

Rules and RDF(S). Inspired from F-Logic, TRIPLE[5] was one of the earliest rule languages using Horn rules to access the RDF datasets. Another rule language called Notation3 (N3) uses a CWA forward reasoning engine to access the ontologies generated from RDF(S)[6]. SPARQL is another W3C standardized query language for querying RDF datasets. SPARQL queries are represented as Datalog rules so SPARQL's **CONSTRUCT** queries are viewed as deductive rules, which create new RDF triples from the RDF datasets.

Rules and OWL. In addition to the Semantic Web Rule Language (SWRL) [10], Rule Interchange Format (RIF) is an emerging rule interchange language from W3C RIF WG [2]. It intends to provide core and extend languages with a common exchange syntax for all of the classification rule languages, i.e., deductive, normative, and reactive rules. The requirements of integrating different types of rules with possible data (and meta data) accessing representation, i.e., RDB, XML, RDF, and OWL, drive the development of a RIF core interchange format, the*RIF Core,* and its extensions, *RIF dialects.* Another recent development is to combine RIF and OWL 2 in RIF, RDF, and OWL that specifies the interactions between RIF, RDF and OWL for their compatibilities[7].

2.4 Combination of Rule and Ontology

A one-way knowledge flow exists from an ontology module to a rule module for knowledge acquisition, where an ontology module's instances are imported as basic facts and filtered with conditions in the rules. This passive knowledge query only uses deductive rules. If a rule engine derives implicit new facts not in an ontology module and furthermore updates new facts back to an ontology module, then it provides another reverse knowledge flow from a rule module to an ontology module. In this two-way knowledge flow process, normative and reactive rules are also required to check the knowledge consistency and trigger the message passing for updating the ontology's knowledge base.

The idea of combining rules and ontologies is to fulfill a goal of two-way knowledge flow. The combination is classified as two types: tightly coupled integration and loosely coupled integration [11]. In the tightly coupled integration model, all of the terms in the rule's body and head are specified in the ontology schema, but in the loosely coupled integration model we do not have this requirement. So, some rules have their own defined terms in the rules' body or head. This loosely coupled integration model enhances the expressive power of ontology and rule as compared to the tightly coupled one.

[5] http://triple.semanticweb.org/

[6] http://www.w3.org/2000/10/swap/doc/cwm

[7] http://www.w3.org/2005/rules/wiki/OWLRL

Description Logic Program (DLP) [12] and SWRL are two well-known tightly coupled integration models. In general, both DL and LP are subsets of FOL in knowledge representation but each has its own part that cannot be expressed in the other part. DLP only takes intersection of DL and LP so knowledge representation in this model is limited. In SWRL, the major knowledge representation is OWL-DL with additional Datalog rules from LP to enhance the lack of property chaining in OWL-DL. In SWRL, DL-safe is the condition where variables occurring in each rule's head are also required to occur in its body to ensure the decidable reasoning of the rule engine. The availability of SWRL rule and ontology integration development in the popular Protégé environment[8] makes the SWRL model the most attractive one for people to use.

In the loosely coupled integration, DL-log [13], AL-log [14], and DL+Log [15] are three well-known models. In these models, rules are extended to Horn rules. Besides, not all of the terms in rules are required from ontology so rule module in these models provides more powerful knowledge representation and rule reasoning than the ones in SWRL. However, none of loosely coupled integration models provide standardized XML markup languages and a development environment, as SWRL does in Protégé. Obviously, this will be a challenge to represent and execute rule systems for loosely coupled integration on the Web. Moreover, the reactive rules [16] have not been seriously considered in all of the ontology and rule integration models. This will be the biggest impediment to implement rule and ontology systems for distributed applications on the Web.

3 Policy as Ontology and/or Rule

Since computers understand the data semantics in the Semantic Web, people are much more satisfied with the search results when a semantic search engine is fully developed. Policy-aware Web extends Semantic Web that provides computerized policies, such as privacy protection or digital rights management policies for computers to understand and execute automatically [17]. However, pure rule and/or ontology languages are not explicit enough to represent policies that regulate human behavior in the real world. We need a well-defined policy language that describes the concepts of rights, obligation, conditions, resources, etc. between resource owner and user to represent and execute access control policies of resources on the Web.

Following [18], policies are considered as knowledge bases, allowing deontic classes, properties, and access control rules. This has the advantage that many operations are automated, thereby reducing ad hoc program coding to a minimum and enabling automated documentation. Regulations imposed on human behavior and activity are simulated by computerized policies that are specified by using policy languages, such as Rei or KAoS [19]. The semantics of these policy languages is only DL-based, and needs to be further extended by using LP-based semantics of rule languages, such as RuleML, RIF or Protune [20]. Recently, AIR (AMORD In RDF) is a policy language that considers using both

[8] http://protege.stanford.edu/

RDF ontology language and N3 rule language for the privacy protection policy execution[9].

3.1 Policy for Semantic Web Services

The idea of Web Services in the SOA of distributed software systems has become a tremendous success. Semantic Web Services (SWS) employ Semantic Web technology in the Web Services area: service functionality, service inputs and outputs, preconditions and effects, etc.; all are expressed and executed in knowledge representation languages, i.e., ontology and rule languages [21]. A policy can be considered in the SWS because of using similar ontology and rule languages' semantics on the Policy-aware Web. Thus policies are represented and executed as Web Service rules for the compliance of human regulations to control information sharing and service execution.

One of the challenges to implement rule systems on the Policy-aware Web is how to design and implement rules as computerized policy by the integration of rule and ontology. This computerized policy imitates human regulation for controlling information sharing and service execution for a composite web service on the Web. And the ultimate goal is the satisfaction of legal regulation compliance from the execution of a computerized policy. This idea is similar to the Legal Knowledge Interchange Format (LKIF) proposed in the past EU FP6 project [22].

4 Rule Management Systems and Engines

Before looking into the details of rule management systems, we need to decide about a rule management systems implementation platform. If we choose a rule system that is also embedded in the Semantic Web development environment, then we have several advantages. First, it provides sufficient facilities to implement subsystems for rules and the data model. Second, both the ontology and rule languages used in the Semantic Web are complementary to each other so we can leverage on the declarative knowledge representation. Third, we have a standard query language or a rule language to support the access of underlying knowledge bases for ontology or rule bases. Finally, if applications are embedded in Java or some other popular imperative programming language, we have language typing, control flow, and interaction mechanism available for the implementation of application system on the Web.

4.1 Rule Systems in the Semantic Web Framework

The SemWebCentral[10] is one of the well-known websites providing Open Source development tools for the Semantic Web. The Semantic Web system development framework can be subdivided into three subsystem modules: an application module, a controller module and a view module. The application module

[9] http://dig.csail.mit.edu/TAMI/2007/amord/air-specs.html

[10] http://www.semwebcentral.org/

contains reasoning functions, including task and inference, domain schema and knowledge base. The controller module handles interactions with the user and functions in the application model. The view module provides output for the user. The Semantic Web system development framework usually includes two development parts, one is for ontology and the other is for rule. For example, Protégé has been successfully developed for ontology and rule, such as Jena[11] and Jess[12]. The Jena rule engine was integrated in the Semantic Web system development framework Protégé for having rule-based inference with the access to knowledge base in the ontologies of RDF and OWL[13]. In addition, the system for development of ontology and rule combination, such as SWRL is also available in the Protégé with SWRLTab[14].

4.2 Standalone Rule Systems

A number of standalone rule systems have been investigated by the RIF Working Group[15]. A rule system is defined as a piece of software that implements or supports a rule language in some way (e.g., a rule engine or a rule editor). Among the RIF list, some rule systems are developed for commercial usage but others are for open source purposes. Based on the classification of rule types, some rule systems are developed for a deductive rule engine but others are implemented for a reactive rule engine.

Commercial Rule Systems. IBM ILOG Business Rule Management Systems (BRMS)[16] provides a complete BRMS for analysts, architects and developers, featuring tools of rule authoring and rule management besides its rule engine. In fact, ILOG JRules is one of the best-known production rule systems. JBoss Drools[17] Enterprise BRMS is a well-known open source rule system which provides perfect integration with the service-oriented architecture (SOA) Web service solutions. On the other hand, existing rule systems, such as Prova[18] and ruleCore[19] are also available for ECA rules inference. For more details about reactive rules on the Web please refer to [16].

Some commercial rule systems are developed from a matured prototype of the Semantic Web middleware, such as OntoBroker[20]. Oracle Business Rules integrates with the Business Process Execution Language (BPEL) and tries to enrich decision making for processes in the SOA[21]. In general, commercial rule

[11] http://jena.sourceforge.net/

[12] http://www.jessrules.com/

[13] http://protege.stanford.edu/plugins/owl/jena-integration.html

[14] http://protege.cim3.net/cgi-bin/wiki.pl?SWRLTab

[15] http://www.w3.org/2005/rules/wg/wiki/List_of_Rule_Systems

[16] http://ilog.com/products/businessrules/index.cfm

[17] http://www.jboss.com

[18] http://www.prova.ws

[19] http://www.rulecore.com

[20] http://www.ontoprise.de

[21] http://www.oracle.com/technology/products/ias/business_rules/index.html

systems use proprietary rule languages for the development of the rule bases so we need a standard rule interchange language, such as RIF to obtain rule interoperability among these rule systems.

Academic Rule Systems. The academic ECA rule system XChange, with its integration of the Web query language Xcerpt, provides the access of data sources to obtain information on the dynamic Web. Other academic rule systems are deductive reasoning rule engines, such as jDREW and its object-oriented extension OO jDREW[22]. An Object-Oriented Knowledge Base Language FLORA-2 provides frame-based logic reasoning engine with the knowledge base development environment[23].

Logic programming systems are also used to develop rule-based applications. For example, Logic Programming Associates Prolog provides a complete rule development environment with a graphical interface for rule editing[24]. Thea is a Prolog library for generating and manipulating OWL content on the Semantic Web. The Thea OWL parser uses SWI-Prolog's Semantic Web library for parsing RDF/XML serialization of OWL documents into RDF triples, and then it builds a representation of the OWL ontology[25].

One of the challenges for implementing rule systems on the Web is to be aware of the current rule management systems, including commercial and academic ones, and, furthermore, an understanding of their system features and which rule type reasoning they can support. Moreover, we need to investigate the possible application domains they intent to accomplish through the underlying rule interchange standard.

4.3 Performance Benchmark for Rule Systems

It is not easy to propose an acceptable measurement benchmark to evaluate the performance of current rule systems because rule systems vary considerably with respect to rule syntax and features. In [23], a set of benchmarks were proposed for analyzing and comparing the performance of numerous rule systems. In this OpenRuleBench, they include five rule technologies to compare with: Prolog-based, deductive databases, production rules, triple engines, and general knowledge bases. Jena and OntoBroker we mentioned before were also two of the selective rule systems in their comparison list. We envision that the benchmark performance evaluation output will be just one of the criteria for people to decide for which rule system they are going to adopt in their application development.

5 Rule Interchange Languages

In early expert systems, a specific language, such as Prolog or LISP was used to encode expert domain knowledge into rules and facts, for execution in a standalone

[22] http://www.jdrew.org/oojdrew/

[23] http://flora.sourceforge.net/

[24] http://www.lpa.co.uk/

[25] http://www.semanticweb.gr/TheaOWLLib/

system. However, when rules and facts are created in different rule systems and distributed on the Web, we need a rule standard exchange language for the interchange of heterogeneous rule formats. Otherwise, we cannot implement an application, such as composite (semantic) web services that might require rules created and distributed in the different rule systems [21]. Therefore, a common rule format facilitates decision making on the network environment with multiple rule formats. For example, the therapeutic guideline recommendation rules for diabetes type 2 are constructed with the combination of clinical and therapeutic criteria as the condition part and therapeutic options as the actions. When users or organizations switch rules from one rule product to another, they can employ the rule interchange technologies without re-developing their rules.

Proposed rule interchange languages include RuleML [1], REWERSE Rule Markup Language (R2ML) [24], and W3C RIF[26], where R2ML attempts at integrating aspects of RuleML, SWRL, and Object Constraint Language (OCL). The most recent W3C RIF[27] was proposed to achieve the objective of rules interchange and interoperability for major rule systems. These rule interchange languages provide XML schemata to guarantee the comparability of rule syntax and semantics from source to target rule systems and vice versa. The other important rule language is Semantics of Business Vocabulary and Business Rules (SBVR), submitted by Business Rule Group (BRG) to OMG on the standardization of semantics for business vocabulary and rules[28].

One of the challenges to apply rule systems on the Web is to finalize a rule interchange language to provide a rule interchange framework and format of rules for current major rule systems. When agents proceed towards a two way rule interchange, a rule interchange language with the framework ensures the compatibility of rules' syntax and semantics between rule systems. The related challenge is the requirement to have a software development system and a runtime environment for people to build, design, and implement standardized rule interchange formats to automatically extract and transform rules from different rule systems on the Web.

6 Use Cases with Rules

Rules are used to express computational or business logic in the information systems which do not have explicit control flow, so rules are more suitable for execution in the dynamic situations for business collaborations. Along with the rapid development of the Web, multiparty collaborations for carrying out business services in this environment are more significant than ever before. For example, when a credit card transaction is requested from a merchant, a customer needs a payment authorization from the merchant and the card issuer (the bank) to accomplish a successful transaction service. In this case, both merchant and bank have their own policies as rules to conduct their authorization processes.

[26] http://www.w3.org/TR/rif-bld/

[27] http://www.w3.org/2005/rules/wiki/RIF_Working_Group

[28] http://www.businessrulesgroup.org/sbvr.shtml

If both parties are required to combine their policies, we hope they can transform the rules into a formal common rule format, such as RIF. For example, rules from the bank are directly imported by the merchant and processed with his local rule engine to derive an authorization decision. In addition, this situation can be extended to other relevant web services for conducting composite web services. Another use case is a seller, posting his price discount and refund policies for execution as rules on his website, to attract potential customers for making a purchase decision from his selling goods. Moreover, a vendor advertises his lead time policies in formal rules to attract customers and also as a part of contract negotiation in the supply chain management.

Use cases such as the ones we have shown above are categorized by the W3C RIF Working Group as a type of policy-based transaction authorization policy for access control with the interchange of human-oriented business rules. Several other interesting use cases focusing on different application domains are also available on this website[29]. Another interesting use case study was proposed by the Business Rule Group (BRG) to use SBVR for illustrating business rule concepts of EU-Rent, EU-Fly, and EU-Stay. They are available on the BRG website[30]. The challenge here is whether we have enough use cases that can be accomplished by current rule systems on the Web to convince people to adopt and use this technology.

7 Conclusion

In this study, we outlined the objectives of RuleML-Challenge competitions started in 2007. Alos, we have elaborated the possible research and implementation challenges for rule systems on the Web that are closely related to the Challenge competition events in the forthcoming years.

The first challenge is to perfectly implement rule systems with the data model, either from a relational or object-oriented database or from a DL-based knowledge base. The second challenge is to enable computerized policies, created in a policy language that is compliant with human legal regulations. In addition to the legalized policy implementation with policies created from the policy language, computerized policy can be shown as a combination of ontology and/or rule languages for the purpose of information sharing and web service execution. The third challenge is full awareness of current available commercial and open source rule management systems and, moreover, finding out the pros and cons of each rule system by a standard evaluation benchmark to verify its scalability and performance. The fourth challenge is to achieve rule interoperability using available rule interchange languages for rules created and distributed on the Web. The fifth challenge is to demonstrate sufficient use cases implemented from rule systems, while interchanging their rules through one of rule interchange language.

[29] http://www.w3.org/2005/rules/wg/wiki/Use_Cases

[30] http://www.businessrulesgroup.org/egsbrg.shtml

Acknowledgements

This research was partially supported by the NSC Taiwan under Grant No. NSC 95-2221-E004-001-MY3 and NSC 98-2918-E-004-003.

References

1. Boley, H.: The ruleML family of web rule languages. In: Alferes, J.J., Bailey, J., May, W., Schwertel, U. (eds.) PPSWR 2006. LNCS, vol. 4187, pp. 1–17. Springer, Heidelberg (2006)
2. Boley, H., Kifer, M., Pătrânjan, P.-L., Polleres, A.: Rule interchange on the web. In: Antoniou, G., Aßmann, U., Baroglio, C., Decker, S., Henze, N., Patranjan, P.-L., Tolksdorf, R. (eds.) Reasoning Web. LNCS, vol. 4636, pp. 269–309. Springer, Heidelberg (2007)
3. Ullman, D.J.: Principles of Database and Knowledge-Base Systems Volume I. Computer Science Press, Rockville (1988)
4. Ullman, D.J.: Principles of Database and Knowledge-Base Systems Volume II. Computer Science Press, Rockville (1989)
5. Date, C.J.: Logic and Databases: The Roots of Relational Theory. Trafford Publishing (2007)
6. Ceri, S., Fraternali, P.: Designing Database Applications with Objects and Rules: The IDEA Methodology. Addison-Wesley, Reading (1997)
7. Berners-Lee, T., et al.: The semantic web. Scientific American (2001)
8. Brachman, J.R., Levesque, H.J.: Knowledge Representation and Reasoning. Morgan Kaufmann, San Francisco (2004)
9. Patel-Schneider, F.P., Horrocks, I.: A comparison of two modelling paradigms in the semantic web. Journal of Web Semantics, 240–250 (2007)
10. Horrocks, I., et al.: SWRL: A semantic web rule language combining OWL and RuleML (2004)
11. Maluszynski, J.: Hybrid integration of rules and DL-based ontologies. In: Maluszynski, J. (ed.) Combining Rules and Ontologies. A survey. EU FP6 Network of Excellence (NoE), pp. 55–72. REWERSE (2005)
12. Grosof, N.B., et al.: Description logic programs: Combining logic programs with description logic. In: World Wide Web 2003, Budapest, Hungary, pp. 48–65 (2003)
13. Motik, B., Sattler, U., Studer, R.: Query answering for OWL-DL with rules. In: McIlraith, S.A., Plexousakis, D., van Harmelen, F. (eds.) ISWC 2004. LNCS, vol. 3298, pp. 549–563. Springer, Heidelberg (2004)
14. Donini, M.F., et al.: AL-log: Integrating datalog and description logics. Journal of Intelligent Information Systems 10, 227–252 (1998)
15. Rosati, R.: DL+log: Tight integration of description logics and disjunctive datalog. In: Proc. of the 10th International Conference on Principles of Knowledge Representation and Reasoning, KR (2006)
16. Berstel, B., Bonnard, P., Bry, F., Eckert, M., Pătrânjan, P.-L.: Reactive rules on the web. In: Antoniou, G., Aßmann, U., Baroglio, C., Decker, S., Henze, N., Patranjan, P.-L., Tolksdorf, R. (eds.) Reasoning Web. LNCS, vol. 4636, pp. 183–239. Springer, Heidelberg (2007)
17. Weitzner, D.J., et al.: Creating a policy-aware web: Discretionary, rule-based access for the world wide web. In: Ferrari, E., Thuraisingham, B. (eds.) Web and Information Security, pp. 1–31. Idea Group Inc., USA (2006)

18. Bonatti, P., Olmedilla, D.: Policy language specification, enforcement, and integration. project deliverable D2, working group I2. Technical report, REWERSE (2005)
19. Tonti, G., Bradshaw, J.M., Jeffers, R., Montanari, R., Suri, N., Uszok, A.: Semantic web languages for policy representation and reasoning: A comparison of KAoS, Rei, and Ponder. In: Fensel, D., Sycara, K., Mylopoulos, J. (eds.) ISWC 2003. LNCS, vol. 2870, pp. 419–437. Springer, Heidelberg (2003)
20. Antonious, G., et al.: Rule-based policy specification. In: Security in Decentralized Data Management, Springer, Heidelberg (2007)
21. Studer, R., Grimm, S., Abecker, A.: Semantic Web Services: Concepts, Technologies and Applications. Springer, Heidelberg (1990)
22. Gordon, F.T.: The legal knowledge interchange format (LKIF). Estrella deliverable d4.1, Fraunhofer FOKUS Germany (2008)
23. Liang, S., Fodor, P., Wan, H., Kifer, M.: Openrulebench: an analysis of the performance of rule engines. In: Word Wide Web 2009, pp. 601–610 (2009)
24. Wagner, G., Damásio, C.V., Antoniou, G.: Towards a general web rule language. International Journal Web Engineering and Technology 2 (2005)

2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology

SemPIF: A Semantic Meta-Policy Interchange Format for Multiple Web Policies

Yuh-Jong Hu
Dept. of Computer Science
NCCU, Taipei, Taiwan
hu AT cs.nccu.edu.tw

Harold Boley
IIT – e-Business,
NRC-CNRC, Canada,
Harold.Boley AT nrc.gc.ca

***Abstract*—We propose a semantics-enabled layered policy architecture for the exchange and management of multiple policies created by different policy languages on the Web. This architecture consists of four layers: Unifying Logic (UNL), Policy Interchange Format (PIF), Privacy Protection/DRM (PPD), and Domain Specific Applications (DSA). A meta-Policy Interchange Format (meta-PIF) layer is also introduced, side by side with the corresponding PIF layer, allowing agents in the facilitator to provide uniform services of interchange, reconciliation, and combination of policies. This SemPIF architecture extends W3C's Semantic Web architecture to permit the reuse of earlier work. A scenario of agents in the facilitator employing SemPIF for Digital Rights Management (DRM) and privacy protection policies on digital library subscription services will be demonstrated.**

***Keywords*-semantic web; ontology and rule; computer policy; privacy protection; digital rights management**

I. Introduction

In the Semantic Web, information is given well-defined meaning to better enable computers and people to work in cooperation. The well-known Semantic Web layered architecture[1] has undergone revisions reflecting the evolution of layers such as the Description Logic (DL)-based ontology language OWL [1], the Horn Logic (HL)-based rule language RIF [2], and their relationship. On the other hand, policy languages, such as Rei [3], KAoS [4], Protune [5], have also been proposed – on the basis of DL and LP – to allow agents understand policies and to enforce these policies as intended by their semantics. However, the semantic bases of policy languages vary considerably, ranging from DL to HL to Logic Programming (LP), e.g. leading to different stances w.r.t. the unique name assumption (UNA) and the closed world assumption (CWA) [6]. This makes policies created in these policy languages hard to interchange and combine with each other.

Policies are formulated and treated as knowledge bases, i.e. ontologies and rules ($\mathcal{O}+\mathcal{R}$). Many operations can be automated, thereby reducing ad-hoc program coding to a minimum and enabling automated documentation [5]. Policy frameworks also need to support interoperability. Moreover, the context of a policy is itself described in a machine-understandable way. Therefore, we propose a semantics-enabled policy architecture consisting of four layers: Unifying Logic (UNL), Policy/meta-Policy Interchange Format (PIF/meta-PIF), Privacy Protection/DRM (PPD), and Domain Specific Applications (DSA). Here UNL directly corresponds to the layer "Unifying Logic" of the most recent version of the Semantic Web architecture. We also introduce a meta-PIF layer, side by side with the corresponding PIF layer, allowing software agents in the facilitator, to provide the management functions of interchange, reconciliation, and combination of policies. The Policy Web architecture can be viewed as an extension of the Semantic Web architecture shown as Fig. 1.

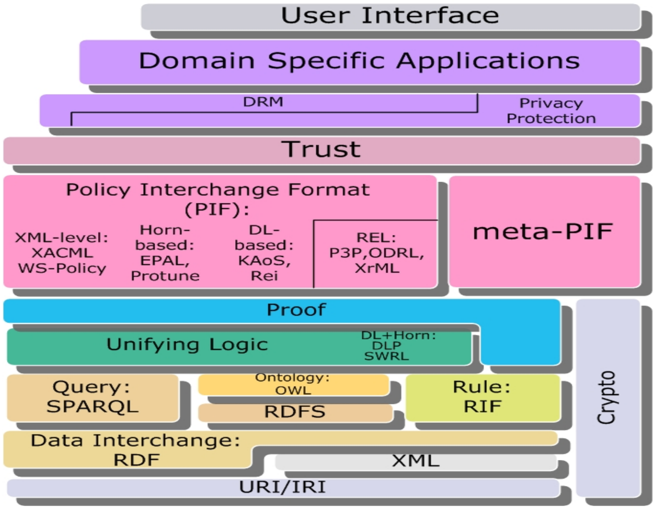

Fig. 1. SemPIF: A semantics-enabled layered policy model centered on semantic policy interchange format (PIF) and meta-PIF.

PIF is built on DL-based ontologies and LP-based rules, i.e. $\mathcal{O}+\mathcal{R}$, that allow agents in the facilitator to support interchange services of policies created from different policy languages. In addition, we may use meta-PIF to specify meta-policies for managing policies created from different policy languages. A meta-policy is a policy about policies that provides a set of rules for realizing services needed for the management of policies. Moreover, a meta-policy consists of a set of rules for setting up the priority of polices to coordinate,

[1] http://www.w3.org/2007/03/layerCake.svg

978-0-7695-4191-4/10 $26.00 © 2010 IEEE
DOI 10.1109/WI-IAT.2010.238

enforce, and even negotiate policies [7].

In a particular policy language framework, policy management services could be implemented as meta-policies as shown in the Rei framework [3] or it could be implemented as policy administration tools as shown in KAoS [4]. In the Protune framework, the role of meta-policies is in governing the behavior to reduce ad-hoc programming efforts and to improve policy readability and maintainability. However, policy management services in these frameworks were only allowed to operate within their own environments. For added flexibility, SemPIF allows agents in the facilitator to use meta-policies providing the management services of policy interchange, combination, and negotiation across multiple heterogeneous domains.

In contrast to other policy languages, such as KAoS, Rei, and Protune, PIF follows W3C $\mathcal{O}+\mathcal{R}$ standards [8] and strives to provide a mechanism for agents to preserve different policy syntaxes and semantics throughout its policy integration and interchange. In addition, agents can use meta-PIF, providing further management and reconciliation services of PIF-enabled multiple policies across various domains (see Fig. 2).

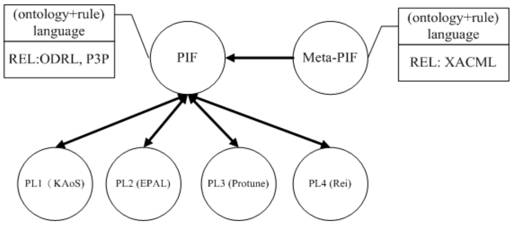

Fig. 2. Policy integration and interchange for various policy languages (PL) through PIF and meta-PIF in the SemPIF

An XML-based Rights Expression Language (REL) lacks semantic expressive power so it is a restricted form of policy language in the PIF layer. Currently, there are three RELs available, i.e. P3P for privacy protection as well as ODRL and XrML for DRM. Unfortunately, policies created from these XML-level RELs lack formal semantics, which prevents agents from automatically and accurately interpreting and processing these policies.

A formal semantic model for policies could be expressed and enforced as a combination of $\mathcal{O}+\mathcal{R}$. Obviously, if we do not know what are the available expressive features of each $\mathcal{O}+\mathcal{R}$ combination, then we cannot decide which combination will be the best one to represent the formal semantics of RELs. We have shown the semantics of DRM policies in PPD as a homogeneous combination of $\mathcal{O}+\mathcal{R}$, i.e. SWRL, where both $\mathcal{O}+\mathcal{R}$ are embedded in a logical language $\mathcal{L}$, to structure the semantics of ODRL [9]. We also have shown the semantics of privacy protection policies in PPD as a hybrid combination of $\mathcal{O}+\mathcal{R}$, i.e. DL+log, where a strict separation between the rule predicates and ontology predicates, to formalize the semantics of P3P in the PIF layer [10].

Another issue addressed by our investigation of the DRM vs. privacy usage control problem is the following. While DRM systems are collecting personal information for usage control, it is quite possible that they might also invade privacy rights. To reconcile this conflict, the $(\mathcal{O}+\mathcal{R})$ language in meta-PIF permits agents to enforce the fine-grained mapping and merging of ontologies with interchangeable rules from policies on privacy protection and DRM. This paves the way for accomplishing the objective of unifying multiple Web policies through SemPIF.

Finally, we will show a scenario for digital library subscription services in a client-server model and demonstrate how to use it by agents in the facilitator to eliminate possible conflicts between a server's DRM policies and a client's privacy protection policies.

II. Semantic Web Layered Architecture

We have both Web markup languages and Semantic Web languages in the Semantic Web Layered Architecture (SWLA) (see Fig. 1). XML / XML Schema and URI/IRI references constitute the foundation, which provides interoperable syntax for RELs at the PIF layer. The semantics of RELs need to be formalized with one of the $\mathcal{O}+\mathcal{R}$ combinations from the UNL, to provide meanings for policies. For Semantic Web languages, we have ontology languages, rule languages, and a combination of ontology and rule languages, i.e. $\mathcal{O}+\mathcal{R}$ languages. The ontology languages include the graph-based RDF(S) and the DL-based OWL. The Horn-based rule languages and their extensions to LP-based rule languages include RIF and RuleML. SWRL is a Semantic Web language using a combination of OWL-DL ontologies and Datalog RuleML rules so it is an $\mathcal{O}+\mathcal{R}$ language [11]. OWL 2 RL and its combination with RIF is another emerging $\mathcal{O}+\mathcal{R}$ language that can be compared with DLP[2].

A. Unifying Logic

PIF and Sem-PIF are built on the unifying logic of DL-based ontologies and LP-based rules [6]. In the UNL layer, DL is a subset of the First Order Logic (FOL). DL provides a basic logic foundation for an ontology language, such as OWL 2. Similarly, Horn logic and LP provides a basic logic foundation for rule languages, such as RIF or RuleML. One of LP's characteristics, procedural attachment, is not included in DL (or FOL) but this feature is very important for the execution of policy's actions.

DLP was introduced as the intersection of DL and LP [12], which has quite limited expressive power when compared to other $\mathcal{O}+\mathcal{R}$ combinations, such as $\mathcal{AL}$-log, $\mathcal{DL}$-log, etc. [13] [14]. The homogeneous $\mathcal{O}+\mathcal{R}$ combination of DL and Datalog provides the logic foundation of SWRL. These combinations of $\mathcal{O}+\mathcal{R}$ have much more expressive power than DLP regarding $\mathcal{O}+\mathcal{R}$, which is needed for Sem-PIF. However, given the ongoing $\mathcal{O}+\mathcal{R}$ research[3], we have not fixed yet any one combination for SemPIF's representation and enforcement requirements.

[2] The combination of OWL 2 and RIF has been shown in http://www.w3.org/2005/rules/wiki/OWLRL.

[3] See, e.g., ONTORULE http://ontorule-project.eu/.

B. Policy Interchange Format

The PIF layer consists of regular DL-based policy languages, such as Rei, KAoS; or Horn-based policy languages, such as EPAL [15], and XML-syntax policy languages, such as XACML [16]. P3P and EPAL were proposed as policy languages for privacy protection in the corresponding client-server and server-server models [17] [18]. As REL sublayer, ODRL and XrML were proposed for designing DRM policies [19] [20]. DL-based or Horn-based logic can be used as foundations to underpin these RELs with explicitly defined semantics for policy languages.

A policy's explicit representation in terms of ontologies or rules depends on what the underlying logic foundation of your policy language is. If policies are created from a DL-based policy language, such as Rei or KAoS, then ordinary policies are shown as $\mathcal{TB}$ox ontology schemas and $\mathcal{AB}$ox instances. Otherwise, with policies created from LP-based policy languages, such as EPAL, ordinary policies are sets of rules and facts using unary or binary predicates.

These policy languages in the PIF layer do not fully utilize the syntax and semantics expressive power of OWL or RDF(S) shown in the SWLA. Therefore, we do not expect these policy languages to be able to leverage the power of existing ontology or rule languages. Another restriction is policies created from different policy languages might not be able to interchange or negotiate with each other. This calls for the use of SemPIF to achieve policy interchange, combination, reconciliation, and negotiation.

C. Privacy Protection and DRM

Privacy protection and DRM are introduced as independent but intertwined layers on top of PIF / meta-PIF and the Trust layer (see Fig. 1). This relationship reflects that access rights enforcements for these two domains are closely related with each other. In [21], the authors proposed that a DRM system should consider user-desirable privacy rights indicated in the Fair Information Principles (FIP), such as data collection, retention, use, disclosure, and destruction, etc., when it enforces privacy protection policies. Otherwise, user privacy rights might be violated. A scenario will be demonstrated to show how agents in the facilitator employ SemPIF to integrate DRM and privacy protection policies (see section III).

III. A SCENARIO OF DIGITAL LIBRARY SUBSCRIPTION

Protection policies are created from various policy languages, such as ODRL, P3P, XACML, and EPAL, for enforcing DRM and privacy protection. This access-control scenario is extended from policy-aware access control for the open Web environment [22]. Agents in the facilitator use PIF-based policies to provide services of integrating semantics-enabled protection policies between a client and a server. Moreover, Agents use meta-PIF-based policies to manage policies, which permits clients and a server to compromise on their respective rights and obligations(see Fig. 3).

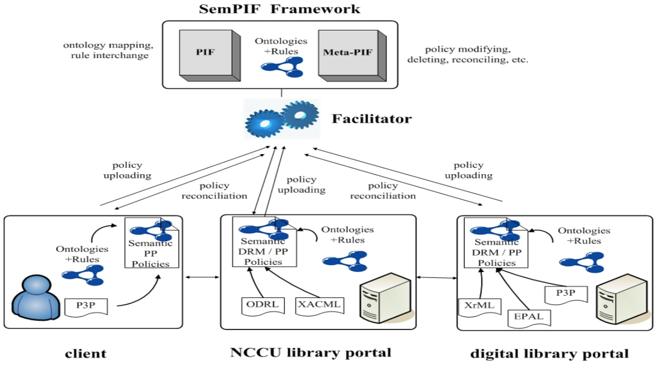

Fig. 3. Agents in the facilitator provide policy interchange services with PIF-based policies and policy management services with meta-PIF-based policies

A. Web server's policies

The $NCCU$ university library has subscribed to $IEEE$, ACM, and $Springer$ digital library services, which provide a set of `eJournal` article access rights for authorized students and staff. There are two types of policy for an IEEE Web server: one for DRM and the other for the declaration of privacy statements.

1) Policies in the IEEE server: The IEEE publisher has two PIF-based policies: $\mathtt{policy(drm1-IEEE)}$ for DRM and $\mathtt{policy(pp1-IEEE)}$ for privacy declaration. $\mathtt{policy(drm1-IEEE)}$ indicates that the policy's name is drm1-IEEE, which corresponds to a URI as a policy indicator for agents to apply a meta-PIF policy to manage it. The predicates of each RIF rule are specified in PIF-based ontologies (see Fig. 4 and Fig. 5).

- $\mathtt{policy(drm1-IEEE)}$:
  If a student owns a valid student ID issued by a department of a university, e.g. a registrar, and its library is one of the subscribers on the IEEE publisher's list, then the student is endowed with DRM usage rights $\{\mathtt{download, view, print}\}$ for `eJournals` from a Web server of the IEEE publisher's delegation.

$$\begin{aligned}
&\mathtt{?st\#Student}\wedge\mathtt{?id\#StudentID}\wedge\mathtt{?st[own}\rightarrow\mathtt{?id]}\\
&\wedge\mathtt{?uni[nccuHasPartR}\rightarrow\mathtt{?rg]}\wedge\mathtt{?st[enrolledAt}\rightarrow\mathtt{?uni]}\\
&\wedge\mathtt{?rg[issue}\rightarrow\mathtt{?id]}\wedge\mathtt{?uni[nccuhasPartN}\rightarrow\mathtt{?lib]}\\
&\wedge\mathtt{?lib[subscribedTo}\rightarrow\mathtt{IEEE]}\\
&\wedge\mathtt{IEEE[hasPublished}\rightarrow\mathtt{?ejr]}\\
&\wedge\mathtt{IEEE[endowedWith}\rightarrow\mathtt{?rgt]}\wedge\mathtt{?rgt[appliedTo}\rightarrow\mathtt{?ejr]}\\
&\wedge\mathtt{IEEE[delegate}\rightarrow\mathtt{?st]}\Longrightarrow\mathtt{?st[endowedWith}\rightarrow\mathtt{?d]}\\
&\wedge\mathtt{?st[endowedWith}\rightarrow\mathtt{?v]}\ \wedge\mathtt{?st[endowedWith}\rightarrow\mathtt{?p]}\\
&\wedge\mathtt{?d\#Download}\wedge\mathtt{?d[appliedTo}\rightarrow\mathtt{?ejr]}\\
&\wedge\mathtt{?v\#View}\wedge\mathtt{?v[appliedTo}\rightarrow\mathtt{?ejr]}\\
&\wedge\mathtt{?p\#Print}\wedge\mathtt{?p[appliedTo}\rightarrow\mathtt{?ejr]}.
\end{aligned}$$

- $\mathtt{policy(pp1-IEEE)}$:
  If a person is endowed with DRM usage rights from a Web server of the IEEE's publisher and this publisher has the purpose of enforcing DRM control for collecting,

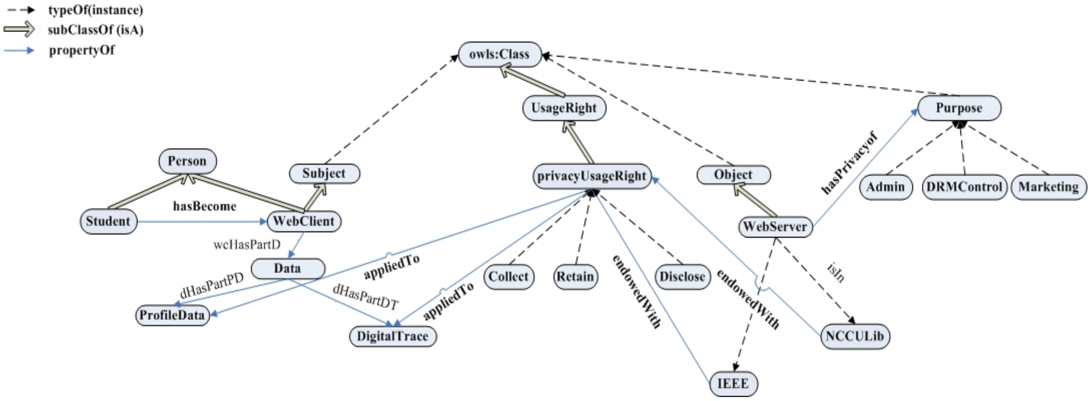

Fig. 4. A PIF-based ontology for privacy protection policies

retaining, and disclosing a person's data, then the IEEE publisher is endowed with privacy usage rights $\{\texttt{collect}, \texttt{retain}, \texttt{disclose}\}$ on this data from a person's delegation, including profiles and digital traces in the Web server under condition of a retention period of two months after the data are first collected.

$$
\begin{aligned}
&?\texttt{per}[\texttt{endowedWith} \rightarrow ?\texttt{drmr}] \wedge ?\texttt{drmr}[\texttt{appliedTo} \rightarrow ?\texttt{ejr}] \\
&\wedge \texttt{IEEE}[\texttt{hasPublished} \rightarrow ?\texttt{ejr}] \\
&\wedge \texttt{IEEE}[\texttt{hasPrivacyOf} \rightarrow \texttt{DRMControl}] \\
&\wedge ?\texttt{per}[\texttt{dHasPartPD} \rightarrow ?\texttt{prf}] \wedge ?\texttt{per}[\texttt{dHasPartDT} \rightarrow ?\texttt{dif}] \\
&\wedge ?\texttt{per}[\texttt{endowedWith} \rightarrow ?\texttt{ppr}] \wedge ?\texttt{per}[\texttt{delegate} \rightarrow \texttt{IEEE}] \\
&\wedge \texttt{Retain}[\texttt{hasDuration} \rightarrow = 2\text{Month}] \\
&\wedge ?\texttt{sdtime}[\texttt{dHasPartD} \rightarrow ?\texttt{dtime}] \\
&\wedge ?\texttt{edtime}[\texttt{dHasPartD} \rightarrow ?\texttt{dtime}] \\
&\wedge \text{subtract-dateTimes}(?\texttt{edtime}, ?\texttt{sdtime}) \leq \texttt{Retain} \\
&\Longrightarrow \texttt{IEEE}[\texttt{endowedWith} \rightarrow ?\texttt{ppr}] \\
&\wedge ?\texttt{ppr}[\texttt{appliedTo} \rightarrow ?\texttt{prf}] \ \wedge ?\texttt{ppr}[\texttt{appliedTo} \rightarrow ?\texttt{dit}].
\end{aligned}
$$

In $\texttt{policy}(\texttt{drm1} - \texttt{IEEE})$, we use ODRL basic primitive vocabularies $\texttt{principal}, \texttt{asset}, \texttt{right}$, or $\texttt{obligation}$ to define a license agreement between principals, i.e., library, registrar, university, and publisher. Similarly, in $\texttt{policy}(\texttt{pp1} - \texttt{IEEE})$, we use P3P basic primitive vocabularies, such as $\texttt{user}, \texttt{owner}, \texttt{purpose}, \texttt{rights}, \texttt{obligation}$ to define a privacy protection agreement between data user and data owner. All of the basic vocabularies are defined in the DRM or privacy protection ontology's schema (see Fig. 4 and Fig. 5), so the semantics of ODRL and P3P RELs are formalized. Furthermore, policies specified in other policy languages for DRM and privacy protection can be mapped to the PIF-based policies for the purposes of policy interchange and integration.

2) Policies in a Web client: A student, $John$, as a Web client has privacy protection policies, i.e., $\texttt{policy}(\texttt{pp3} - \texttt{John})$, $\texttt{policy}(\texttt{pp4} - \texttt{John})$ to address how and what of his personal data can (or cannot) be collected, retained, or disclosed from a Web server. Here we show the policies in natural language only.

- $\texttt{policy}(\texttt{pp3} - \texttt{John})$:
  If an $\texttt{eJournal}$ distributor from {ACM, IEEE, Springer} has the purpose of enforcing DRM control by collecting, retaining, and disclosing data on John as the Web client, then it is endowed with privacy usage rights $\{\texttt{collect}, \texttt{retain}\}$ on the profiles of John as the Web client under the condition of a retention period of less than thirty days after the profiles are first collected.
- $\texttt{policy}(\texttt{pp4} - \texttt{John})$:
  If the distributor IEEE $\texttt{Journal}$ has the purpose of enforcing DRM control by collecting, retaining, and disclosing data on John as the Web client, then IEEE is endowed with the privacy usage rights $\{\texttt{collect}, \texttt{retain}\}$ from the digital traces of John as the Web client, where the data retention period is less than fourteen days after the trace data are first collected.

IV. SemPIF for Multiple Web Policies

Agents in the facilitator provide policy interchange to avoid possibly inconsistent or ambiguous syntax and semantics between source and target policies.

A. Meta-PIF

We envision several important issues in the design of agents while using SemPIF as a mediation architecture to enforce policy management services, such as policy sequencing, adding, deleting, merging, etc.

- In the SemPIF architecture, agents use PIF to provide basic interchange services of various policy languages.
- The basic vocabularies of PIF for interchange of policies are specified in the various REL policy languages, such as P3P or ODRL. They are $\texttt{principal}, \texttt{subject}(\texttt{owner}), \texttt{object}(\texttt{user})$, $\texttt{resource}(\texttt{asset})$, $\texttt{right}, \texttt{obligation}, \texttt{purpose}$, and $\texttt{condition}$. In addition, access right vocabularies for privacy protection and DRM are different. We have $\{\texttt{collect}, \texttt{retain}, \texttt{disclose}\}$, etc. for privacy protection and we have $\{\texttt{download}, \texttt{view}, \texttt{print}\}$, etc. for DRM.
- Most of the basic vocabularies for meta-PIF are the same as PIF's except some of them are directly related to

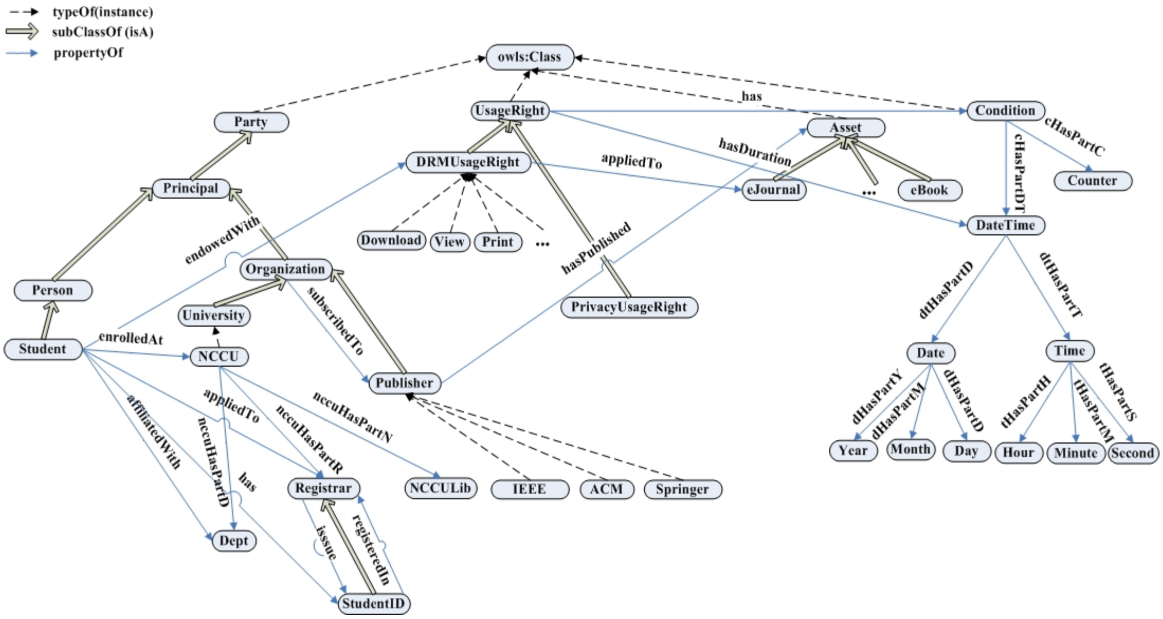

Fig. 5. A PIF-based ontology for DRM policies

PIF-based policies. The policy itself is introduced as a resource with respective users, rights, and conditions, etc. for agents to enforce its policy management services.

- Meta-PIF is a meta-policy language for PIF and only provides management services for PIF-based policies. If meta-PIF attempts to provide an interchange format for different meta-policies in PIF, then we also have to provide policy management interoperability services for different policy languages. This requires further study.

B. Agents in the facilitator enable meta-PIF policies

We use P3P basic vocabularies to specify data owner (or subject), user (or object), type, right, obligation as ontology classes with associated properties to formalize the semantics of privacy protection [10]. Furthermore, we also use Datalog-based rules to decide whether a Web server is allowed to collect, retain, or disclose a particular client's profiles and digital traces. Policies are defined as a combination of $\mathcal{O} + \mathcal{R}$. In order to unify policies from a client and a server, we allow agents in the facilitator to collect client and server's ontologies, and to enable SemPIF policy transformation and management services shown as follows:

1) **Ontologies mapping and aligning**
We map and align vocabularies from domain dependent ontologies of DRM and privacy protection policies. In Section III scenario, the vocabularies of class "Student, Publisher" in policy(drm1 − IEEE) and policy(pp1 − IEEE) correspond to "WebClient, WebServer" vocabularies of class in policy(pp3 − John) to policy(pp4 − John). Furthermore, we align the ontology schemas constructed with the vocabularies of class and property.

2) **Semantics mediation and unification**
We mediate and unify the semantic differences of vocabularies and schemas in the ontologies belonging to different protection policies. For example, a condition of Retain[hasDuration → =2Month] in the policy(pp1 − IEEE) corresponds to a condition of "retention period less than fourteen days" in the policy(pp4 − John).

3) **Conflicts resolving**
Agents initiate the reconciliation processes between conflicting policies using the meta-PIF framework. In this example, IEEE declares its intention to collect, retain, and disclose a Web user's data in the policy(pp1 − IEEE) for two months. The data include a Web user's profiles and digital traces. Web user John does not allow an IEEE Web server to disclose his personal profile to the other partners. Thus, policies between policy(pp1 − IEEE) and policy(pp3 − John), policy(pp4 − John) are inconsistent. Agents enable a policy priority-setting with meta-PIF-based policies to avoid the policy conflicts. In this example, an agent gives a higher priority to a client's policy(pp3 − John) and policy(pp5 − John) than to a server's policy(pp1 − IEEE). The defeasible logic of a meta-PIF's expression, Overrides(policy(?pid1), policy(?pid2)), for resolving conflicts of policies is a possible solution, where policy(?pid1) is bound to policy(pp3 − John) and policy(pp4 − John); policy(?pid2) is bound to policy(pp1 − IEEE). This negotiation protocol requires further study.

V. Related Work

REL is a subset of the PIF layers. FOL-semantics-enabled policy models for RELs have been proposed to specify the semantics of ODRL, XrML, and P3P [23] [24] [25]. However, it is still unclear how to design semantics-enabled policy languages from the FOL-enabled RELs to allow policies to be machine readable and understandable on the Web.

Tonti et. al. compared the three policy languages KAoS, Rei, and Ponder w.r.t. the representation and reasoning of specific policies [4]. The policy semantics of KAoS and Rei came from DL-based ontology. Rei has a policy management services framework for agents to manage policies but agents still cannot interoperate and cooperate with other agents across different frameworks. Moreover, policies created from LP-based policy languages, such as EPAL [15], were not able to interoperate and cooperate with the DL-based policies. We need a *de facto* standard policy interchange language as attempted by the W3C PLING[4] and with OMG's SBVR[5] to achieve policy interoperability.

The idea of meta-policies was proposed almost two decades ago [7]. It was used for policy management services in the Rei and Protune frameworks [5] [3]. In the Rei framework, the authors tried to propose a policy interchange mechanism instead of using a single policy language for describing all policies. Thus, SemPIF can be seen as bringing the objective of the Rei framework is close to the Semantic Web. In the Protune framework, meta-policies provide a simple means to specify which parts of a policy are sensitive, and how application-specific atomic conditions are to be verbalized in the documentation. However, predating SemPIF, the Rei and Protune frameworks did not show yet how a semantics-enabled policy layered architecture can be compatible with the current Semantic Web architecture.

VI. Conclusion and Future Work

We propose a semantics-enabled policy architecture, SemPIF, which extends W3C's Semantic Web layered architecture. We have introduced the SemPIF architecture as a 4-layer framework, i.e., UNL, PIF, PPD, and DSA. A meta-PIF layer is also introduced, side by side with the corresponding PIF, allowing agents in the facilitator to provide uniform services of interchange, reconciliation, and combination of policies from various domains on the Web. A scenario of employing SemPIF for DRM and privacy protection policies on digital library subscription services is described to demonstrate the feasibility of the SemPIF architecture. Future work include refining the PIF and meta-PIF languages to enable a multiple Web policies system on the Web.

Acknowledgements

This research was partially supported by the NSC Taiwan under Grant No. NSC 95-2221-E004-001-MY3, NSC 98-2918-E-004-003, and NSC 98-2221-E-004-009.

References

[1] S. Bechhofer et al., "OWL web ontology language reference", Tech. Rep., W3C, Feb. 2004.

[2] H. Boley et al., "Rule interchange on the web", in *Reasoning Web 2007, Third International Summer School*, Dresden, Germany, Sep. 2007, LNCS 4636, Springer.

[3] L. Kagal et al., "Using semantic web technologies for policy management on the web", in *21st National Conference on Artificial Intelligence (AAAI)*. July 2006, AAAI.

[4] G. Tonti et al., "Semantic web languages for policy representation and reasoning: A comparison of KAoS, Rei, and Ponder", in *2nd International Semantic Web Conference (ISWC) 2003*, 2003, LNCS 2870, pp. 419–437.

[5] P. Bonatti and D. Olmedilla, "Policy language specification, enforcement, and integration. project deliverable D2, working group I2", Tech. Rep., REWERSE, 2005.

[6] F. P. Patel-Schneider and I. Horrocks, "A comparison of two modelling paradigms in the semantic web", *Journal of Web Semantics*, pp. 240–250, 2007.

[7] H. Hilary Hosmer, "Metapolicies I", *ACM SIGSAC Review*, vol. 10, no. 2-3, pp. 18–43, 1992.

[8] Jos de Bruijn, "RIF RDF and OWL compatibility", Tech. Rep., W3C, Oct. 2009, http://www.w3.org/TR/rif-rdf-owl/.

[9] Y. J. Hu, "Semantic-driven enforcement of rights delegation policies via the combination of rules and ontologies", in *Workshop on Privacy Enforcement and Accountability with Semantics in conjunction with ISWC+ASWC'07*, 2007.

[10] Y. J. Hu, H. Y. Guo, and G. D. Lin, "Semantic enforcement of privacy protection policies via the combination of ontologies and rules", in *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC 2008)*, Taichung, Taiwan, June 2008.

[11] I. Horrocks et al., "SWRL: A semantic web rule language combining OWL and RuleML", 2004.

[12] N. B. Grosof et al., "Description logic programs: Combining logic programs with description logic", in *World Wide Web 2003*, Budapest, Hungary, 2003, pp. 48–65.

[13] M. F. Donini et al., "$\mathcal{AL}$-log: Integrating datalog and description logics", *Journal of Intelligent Information Systems*, vol. 10, no. 3, pp. 227–252, 1998.

[14] B. Motik, U. Sattler, and R. Studer, "Query answering for OWL-DL with rules", in *3rd International Semantic Web Conference (ISWC) 2004*. 2004, LNCS 3298, pp. 549–563, Springer.

[15] G. Karjoth and M. Schunter, "A privacy policy model for enterprises", in *15th IEEE Computer Security Foundations Workshop (CSFW)*. June 2002, IEEE.

[16] E. Rissanen, "eXtensible Access Control Markup Language (XACML) ver. 3.0", May 2007.

[17] A. H. Anderson, "A comparison of two privacy policy languages: EPAL and XACML", in *Proceedings of the 3rd ACM Workshop on Secure Web Services (SWS'06)*. 2006, pp. 53–60, ACM.

[18] L. Cranor et al., "The platform for privacy preferences (P3P) 1.0 (P3P 1.0) specification", 2002, http://www.w3.org/P3P/.

[19] Inc. ContentGuard, "XrML: The digital rights language for trusted content and services", Tech. Rep., ContentGuard Inc., 2002.

[20] S. Guth and R. Iannella, "Open Digital Rights Language (ODRL) version 2", ODRL initiative working draft, The ODRL Initiative, February 2005.

[21] E. J. Cohen, "DRM and privacy", *Commun. ACM*, vol. 46, no. 4, pp. 47–49, 2003.

[22] D. J. Weitzner et al., "Creating a policy-aware web: Discretionary, rule-based access for the world wide web", in *Web and Information Security*, E. Ferrari and B. Thuraisingham, Eds., pp. 1–31. Idea Group Inc., 2006.

[23] Y. J.and Vicky Weissman Halpern, "A formal foundation for XrML", *Journal of the ACM*, vol. 55, no. 1, pp. 1–42, 2008.

[24] N. Li, T. Yu, and A. I. Antón, "A semantics-approach to privacy languages", *Computer Systems and Engineering (CSSE)*, vol. 21, no. 5, Sep. 2006.

[25] R. Pucella and V. Weissman, "A formal foundation for ODRL", arXiv:cs/0601085v1, Cornell University, January 2006, http://arxiv.org/abs/cs/0601085.

[4] Policy Interchanges Interest Group - PLING, http://www.w3.org/Policy/pling.

[5] Semantics of Business Vocabulary and Business Rules (SBVR), http://www.omg.org/spec/SBVR/1.0/.

Unifying Semantic Privacy Protection Web Policies for the Digital Rights Management (DRM) System

Yuh-Jong Hu
Dept. of Computer Science,
National Chengchi University, Taipei, Taiwan
hu@cs.nccu.edu.tw

Abstract

There are several right expression languages (RELs) for the information model of data or content protection representation on the Web. For example, P3P/EPAL exists for privacy data protection and ODRL/XrML is used for digital rights control in the digital rights management (DRM) system. However, these XML-based RELs lack the ability to express the semantics of the web protection policies unambiguously enough for as a software program to process them automatically. We need declarative semantics-enabled policies formulated as knowledge bases, i.e., ontologies and/or rules to solve this problem. Here, we consider employing a combination of ontologies and rules as a semantic model of the REL so that the power of Description Logic (DL) for ontologies and Logic Program (LP) for rules can be leveraged simultaneously. In fact, using different ontology and rule $\mathcal{O}+\mathcal{R}$ combinations, such as DLP, SWRL, and OWL 2 RL for the RELs implies variant semantic expressions that can be derived from DL and LP. Certainly, a semantic model of the REL based on $\mathcal{O}+\mathcal{R}$ will affect the robustness of semantics fulfillment for a protection policy. We propose a semantics-enabled policy framework, where privacy protection policies are unified with the digital rights policies in the DRM system. The customizable privacy protection of the DRM system satisfies a server's privacy protection promise to its users.

1 Introduction

We envision a promising future for the use of standardized rights expression languages (RELs) for privacy protection policy's representations in the digital rights management (DRM) system. The objective of a DRM system is to allow the legal sharing of digital content with a content owner's consent. In addition, we expect that this DRM system should also protect each content user's privacy rights. It will be a grand challenge to use RELs, such as XACML, for access control policies with an additional protection of a user's privacy in the DRM system . Moreover, we can enforce the web protection policies in a machine for digital content and personal information access control without too much human intervention. Several RELs, such as ODRL and P3P already have the capacity to represent digital content control policies in a digital rights license agreement for a server and privacy protection policies for a client. More specifically, ODRL and XrML have been proposed as RELs to express the digital content usage contracts in the DRM system [ContentGuard, 2002] [Guth & Iannella, 2005b]. P3P and EPAL are other RELs used to express privacy statements in protection policies, in order for a server to fulfil its privacy protection promises to its users regarding information disclosure [Antón et al., 2007].

However, the progress of modeling semantics for RELs has either gone unnoticed or exists at a very primitive stage [ContentGuard, 2002] [Guth & Iannella, 2007]. In general, the semantics of REL was previously shown as English descriptions or as computer algorithms that grant the access right permissions based on a set of license agreements. These approaches are quite inflexible and inextensible as are the semantics-enabled web policies. Therefore, the building of a formal semantic model from a variety of RELs for web protection policies has recently become a highly significant research area, such as research into XrML and ODRL for digital rights [Halpern, 2008] [Pucella & Weissman, 2006], and APPEL/P3P for privacy rights [Yu et al., 2004] [Li et al., 2006].

Theoretically, any XML-based RELs, such as ODRL and XrML do not have unambiguous expressive power to declare and enforce the semantics of the web policies in their information models unless we have a corresponding ontology language, such as the model theory found in RDF(S), OWL [Patel-Schneider & Siméon, 2002]. Another option for REL's formal semantics is a logic program (LP)-based datalog rule. The formal semantics of the ODRL and XrML RELs were recently modeled as a First-Order Logic (FOL) [Pucella & Weissman, 2006] [Halpern, 2008]. In these studies, authors showed how to express a decidable fragment of the FOL as the semantics of REL for a license agreement. XACML and EPAL are other XML-based RELs proposed to address the issue of information disclosure control in privacy protection systems [Anderson, 2006]. In fact, XACML is also a general access control language for usage control of digital content in the DRM system.

EPAL [Karjoth & Schunter, 2002], which was derived from the FAF model [Jajodia et al., 2001] was proposed for use in the enterprise's privacy protection policies on the web. When the information usage control for the privacy protection is similar to the content usage access control for the DRM, sensitive personal information may possibly be disseminated over the entire web without a user's consent. The semantic representation model of EPAL is far from satisfactory, because EPAL is only based on a logic program, so it lacks a well-defined semantics for its structure data model to specify the data access control policies.

In this chapter, we introduce the current status of numerous RELs for privacy protection and digital rights management. Then, we survey a taxonomy of existing semantics or not-semantics-enabled policy languages. Third, we demonstrate how semantic web policy language can be constructed from the REL by using a combination of ontologies and rules. After that, we propose a semantics-enabled policy framework for unifying the semantic privacy protection policies and the digital rights management policies in the DRM system. Finally, we provide a scenario of digital content subscription services to demonstrate how to apply our semantics-enable policies in the framework to unify privacy protection policies with digital rights policies in the DRM system .

2 Goal

The major criteria we need to consider when designing a formal semantic model for a privacy protection system are not the same as when designing a DRM system. Therefore, we propose a unifying semantic model to resolve the criteria discrepancy between privacy protection and content usage control in the DRM system. For instance, when a data user submits an information disclosure request, EPAL-based policy uses a purpose criterion of data usage from the data user. But in a DRM content usage control policy, when we apply ODRL and XrML RELs we do not consider a data user's purpose criterion. The truth is most DRM systems, do not consider using a purpose criterion unless a fair use or personal privacy protection rights are requested by the DRM's users. But these are statutory rights granted by the Copyright Act and privacy protection law. Therefore, the DRM system eventually has to enforce the fair use and privacy protection rights for their content users [Arnab & Hutchison, 2005] [Erickson, 2003].

The goal of this chapter is to resolve the lack of the formal semantics problem in RELs in web policies' representation and enforcement as we design privacy protection policies in the DRM system. Current *de facto* RELs, such as P3P, ODRL, XACML, etc usually exist semantic ambiguity in the representation of license agreements and access control policies. Instead of using hard-coded computer algorithms or natural language descriptions, we propose a unifying formal semantic model that can be overlaid on the existing RELs to express and enforce the access control web policies. Under this unifying formal semantic model, the semantics-enabled REL can be used to express license agreements based on associated web policies to achieve rights protection when enforced by software programs without causing ambiguity.

3 Taxonomy of policy languages

The term "policy" encompasses different notions and usages, including security policies, trust management policies, business rules, etc. Computer policy is an executable and declarative rule that is created from a specific policy language to regulate the behavior of entities on the web [Vimercati et al., 2007]. Requirements for designing policies from a policy language are declarative, and have well-defined semantics, ontology support, rule support, and after-disclosure control, etc [Bonatti et al., 2006]. We select several well-known declarative RELs and policy languages and categorize them as follows:

- XML-based RELs (not semantics-enabled): P3P [Cranor et al., 2002], ODRL, XrML
- XML-based policy language (not semantics-enabled): XACML [Anderson, 2006], WS-Policy
- Logic Program (LP)-based policy language (semantics-enabled): EPAL [Antoniou et al., 2007]
- Description Logic (DL)-based policy language (semantics-enabled): KAoS, Rei [Tonti et al., 2003]
- $\mathcal{O}+\mathcal{R}$-based policy language (semantics-enabled): Protune [Bonatti et al., 2006], AIR[1]

We will argue why we need a formal semantic model based on a combination of ontologies and rules $\mathcal{O}+\mathcal{R}$ but not on an ontology $\mathcal{O}$ or a rule $\mathcal{R}$ alone. Moreover, the important criteria that people use to select a $\mathcal{O}+\mathcal{R}$ for web policy's semantics representation will also also be shown. We propose a unifying formal semantic model based on a particular $\mathcal{O}+\mathcal{R}$ with DL-based ontologies and LP-based datalog rules. This semantics-enabled web policy provides clear and decidable semantic enforcement of access control policies to ensure a user's privacy rights or a server's content digital rights.

4 REL, license agreement, policy, and rights protection

The $UCON_{ABC}$ model provides a solution for the content usage right control in the DRM and other similar problems, such as privacy protection, etc [Park & Sandhu, 2004]. This $UCON_{ABC}$ access control model uses *Authorization(A), oBligations(B), and Conditions(C)* as specifications for its access control policy. In fact, usage control is a generalization of a regular access control that covers authorization, obligations, conditions, continuity (ongoing controls), and mutability attributes. The $UCON_{ABC}$ model improves the original server-based access control approach, which cannot be used for usage and rights protection on the web. The $UCON_{ABC}$ model still cannot provide after-disclosure control because the original data owner is hard-pressed to enforce its protection policies once the digital content or personal information is disseminated on the web.

[1] **A**MORD **I**n **R**DF (**AIR**), `http://dig.csail.mit.edu/TAMI/2007/AIR/`

4.1 REL for a license agreement

A rights expression language (REL) provides an information modelling format for representing the digital rights usage and delegation on the web. The RELs are used for digital license agreements, access control policies, and rights protection systems. A license agreement is an instance of a digital contract for participants under which a principal, $\texttt{Prin}_o$ allows another principal, $\texttt{Prin}_u$ to use an asset, `r`, presumably owned (or controlled) by $\texttt{Prin}_o$. $\texttt{Prin}_o$ is an asset owner and $\texttt{Prin}_u$ is an asset user. We might also allow a license agreement specifically for a single asset owner to have multiple asset users, but to formalize the expressions and to enforce a license agreement without any ambiguous semantics is more challenging in this case.

A license agreement supports the expression of rights that are formal digital contracts, but there is no guarantee of clear semantics to stipulate the content, terms and conditions of rights usage for all the parties involved in this agreement. Formally speaking, a license agreement must contain at least one `asset` entity, at least one `permission` and `prohibition` entity, at least one `party` entity with an `assigner` role, and at least one `party` with `assignee(s)`(or `consumer(s)`) role(s) [Guth & Iannella, 2007].

4.2 REL for an access control policy

Most existing XML-based standard RELs, such as XACML and EPAL, only have vocabularies consisting of terms and conditions for protected resources. As a standard REL for privacy protection, XACML was compared with EPAL on the privacy policy enforcement and decisions [Anderson, 2006]. On the other hand, ODRL and XrML are intended to provide models for an access control policy on content publishing, distribution and consuming in the DRM system [ContentGuard, 2002] [Guth & Iannella, 2005b].

Although these RELs have a formal information model to express assets usage rights, obligations of conditions and requirements, the semantics used by these RELs to represent and enforce access control policies are sometimes ambiguous. ODRL specifications simply contain expression language, data dictionary elements, and XML syntax to encode the DRM's expressions and elements. However, ODRL does not enforce or mandate any rights protection policies for the DRM system. It only provides expressions of the policies for digital rights [Guth & Iannella, 2005b].

4.3 A rights protection system

Even REL contains enough necessary vocabularies to specify a license agreement with a formal information model to express the access control web policies. But we still need a run time engine to derive a permission (or not permission) in the rights protection system when we use these web policies for each service request. The information usage and delegation rights are shown as web policies to achieve after-disclosure control when enforced in a rights protection system. It processes a license agreement by execution the respective protection policies, which semantics are described as an algorithm or logic-based declarative ontologies and rules.

5 A formal semantic model

We propose a unifying formal semantic model of REL based on ODRL for DRM and P3P/EPAL for privacy protection. A semantics-enabled license agreement is a digital contract created from this model to describe the concepts of protection for content usage rights and user privacy rights. Furthermore, the semantics-enabled access control policies are also represented in the same model to enforce the privacy protection in the DRM system (see Figure 1).

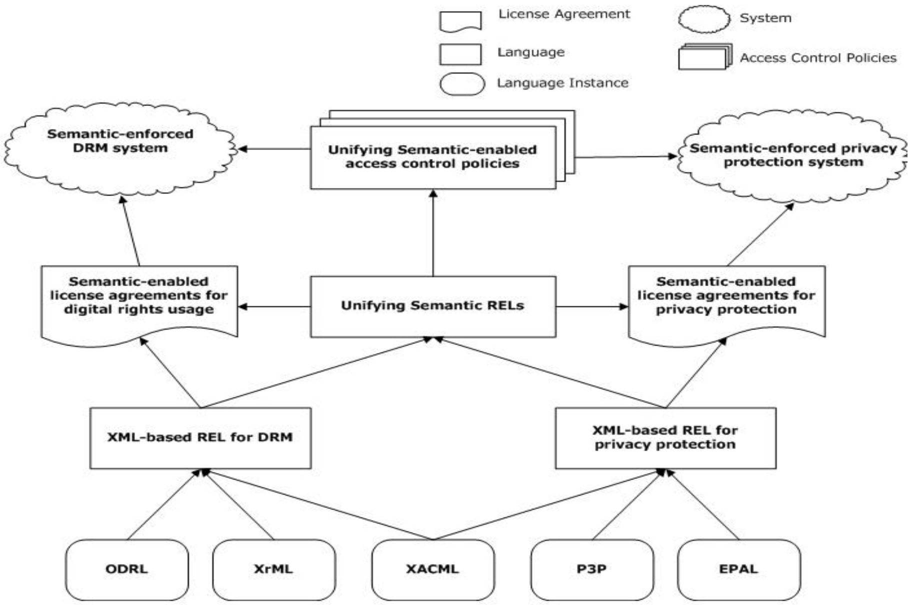
License Agreement
Language
Language Instance
System
Access Control Policies
Semantic-enforced DRM system
Unifying Semantic-enabled access control policies
Semantic-enforced privacy protection system
Semantic-enabled license agreements for digital rights usage
Unifying Semantic RELs
Semantic-enabled license agreements for privacy protection
XML-based REL for DRM
XML-based REL for privacy protection
ODRL
XrML
XACML
P3P
EPAL

Figure 1: A unifying semantic model of REL for describing license agreements and associated access control policies for privacy protection in the DRM system

5.1 FOL as a formal semantic model

In [Halpern, 2008] [Pucella & Weissman, 2006], a formal FOL foundation was elaborated to find out what are the tractable fragments for ODRL and XrML that provide the verification of access control permission implied by a set of licenses and policies. This result has a strong impact on the revised new version of ODRL and XrML information models [Guth & Iannella, 2005a] [Guth & Iannella, 2007].

However, the FOL-based formal semantic model is far from perfect because the tractable fragments of FOL for REL do not allow the updating of a license agreement conditions with its policies. Moreover, communication mechanisms between protection systems cannot be defined as a simple FOL-based REL. Therefore, it is impossible to deploy a generic FOL-based formal semantic model on the web unless we have web-enabled supporting markup languages, similar to RDF(S) and OWL ontology languages to express its semantics.

5.2 DL as a formal semantic model

A DL-based formal semantic model of REL provides the taxonomy of digital asset and usage rights as an ontology `TBox` for a DL-based reasoning engine, such as Racer or Pellet to decide the hierarchy implicit relationships [Garcia et al., 2005]. We can further use the standardized ontology languages, such as RDF(S) and OWL, to enforce the rights protection policies deployed on the web. Since DL is a subset of FOL, the limitations of expressive power in DL is similar to that in FOL for the updating of a license agreement and on the enabling of a message passing between the rights protection systems. This prevents us from building a full scale rights protection system on the web [Hu, 2007].

5.3 LP as a formal semantic model

The representations of LP-based FAF semantics and relational semantics were shown previously in the formal semantic models of EPAL and P3P [Antoniou et al., 2007] [Yu et al., 2004]. In order to have a decidable computational complexity for a rule language, the fragment of LP-based expressions are usually restricted to datalog rules with no function parameter within the predicates of each rule. However, a LP-based formal semantic model for REL has limitations on the representation of the concept and relationship hierarchy that prevent us from easily modeling the rights protection policies and their underlying REL's terms for rights, obligations, and usage conditions of protected resources.

5.4 $\mathcal{O} + \mathcal{R}$ as a formal semantic model

We exploited one of the homogeneous $\mathcal{O} + \mathcal{R}$ combinations, i.e., SWRL for the semantic representation and reasoning of a license agreement with its policies in the semantic DRM [Hu, 2007]. SWRL combines OWL-DL's ontology language with an additional datalog rule language, where a datalog rule language is shown as an axiom of ontology, a little extension of the OWL-DL language that overcomes the limitations of property chaining in the OWL-DL language [Horrocks et al., 2005]. The computation complexity of answering SWRL-based policies might be undecidable regarding the verification of rights access permission unless these policies satisfy the $\mathtt{DL} - \mathtt{Safe}$ conditions [Motik et al., 2004]. Moreover, SWRL did not include reactive rules, such as an event-condition-action (ECA) rule or production rule in its language design. Therefore, it is also impossible to provide the function of message passing by using SWRL-based policies alone in the protection systems.

6 Semantic REL for license agreements and protection policies

Ontology specifies the unambiguous concepts in a well-defined format for agents to process and understand. We have ontology languages, such as RDF(S) and OWL, to provide a well-defined standardized vocabulary to specify the concept and relationship of an ontology. Rule languages based on a (declarative) datalog rule, such as RuleML and RIF, can further enhance the expressive power of the ontology language to enforce information querying, updating, and communication related actions [Boley et al., 2007].

6.1 Semantic REL as an $\mathcal{O} + \mathcal{R}$ combination

Ontologies and rules based on logic foundations, i.e. description logic (DL) and logic program (LP) are primary knowledge representations. Ontologies based on DL are a subset of FOL that represents the shared knowledge domain of concepts and roles. On the other hand, rules based on datalog of LP are also a subset of FOL with the expressive power to overcome the limitations of ontology. Furthermore, reactive rules provide additional power, such as event acceptance and action trigger to enable data communication and updating operations [Berstel et al., 2007].

We propose a unifying semantic model of REL in license agreement and web protection policies for the verification of access rights permission in the DRM system (see Figure 2)

The benefits of using $\mathcal{O} + \mathcal{R}$ combination as the semantic model of REL are shown as follows:

- More expressive power for the semantics representations from ontologies and rules.
- Standardized web-enabled ontology language and rule language are possibly available for agents to automatically process license agreements and policies without facing semantics ambiguity.
- A flexible and scalable reasoning engine are available from DL and LP for executing the privacy protection policies in the DRM system.

The important criteria for using an $\mathcal{O} + \mathcal{R}$ combination as a semantic model of REL are shown as follows:

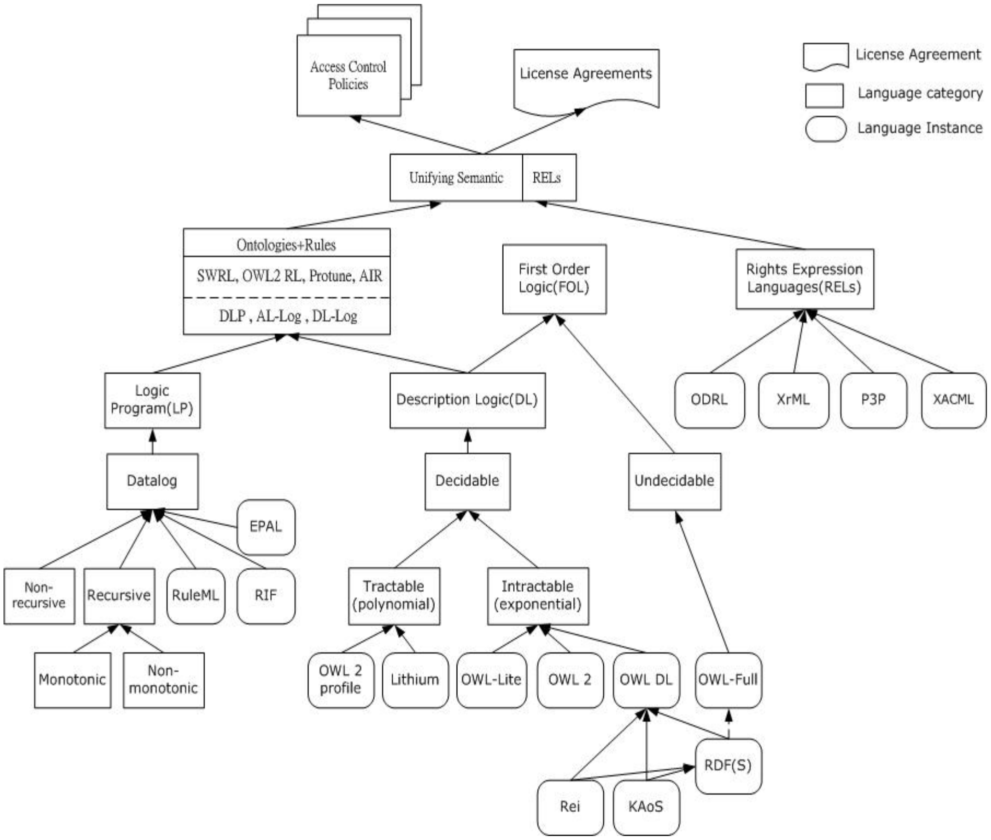

Figure 2: A unifying semantic model of REL from FOL, LP, RELs for license agreements and access control policies

- To use an $\mathcal{O}+\mathcal{R}$ combination which has a decidable computation fragment. Otherwise, we might not be able to obtain an answer for every rights permission query.
- To decide what semantic expressions of REL are from ontologies and rules.
- To resolve the semantic assumption of difference between ontologies and rules. DL-based ontology is an open world assumption (OWA) but an LP-based rule is a closed world assumption (CWA). This difference has side effects on the decision of protection policy when enforced.
- To have a bi-directional or a uni-directional information flow from ontologies to rules. In a uni-directional case, concepts and properties in the ontology are used to specify unary and binary predicates in the rules. In a bi-directional case, reaction rules provide the updating of the facts, e.g. $\mathcal{AB}\wr\S$ for the ontology.

There are two types of $\mathcal{O}+\mathcal{R}$ combinations [Eiter & Ianni, 2008]: homogeneous (tight) integration and heterogeneous (loose) integration. *DLP* [Grosof et al., 2003], *SWRL* [Horrocks et al., 2005], and *OWL 2 RL* [Grau et al., 2008] are a type of tight $\mathcal{O}+\mathcal{R}$ integration. *DLP* is too restricted to use for ontology and in the license agreements and web policies. On the other hand, *SWRL* might have a fragment of undecidable computation on the decision of rights permission unless we request that all of the rules satisfy the $\mathtt{DL}-\mathtt{Safe}$ condition[2]. Another *OWL 2 RL* [Hitzler et al., 2010] is an emerging $\mathcal{O}+\mathcal{R}$ combination, and thus needs further study.

AL-log [Donini et al., 1998] and *DL-log* are $\mathcal{O}+\mathcal{R}$ heterogeneous combinations. *AL-log* has decidable computation to answer a request of license or privacy rights. *AL-log* is based on *A*ttribute *L*anguage with *C*omplements (*ALC*) of the DL-based ontology for monotonic (positive) recursive $\mathtt{DL}-\mathtt{Safe}$ datalog rules. The other is *DL-log*, which has a decidable computation from the decidable DL-based ontologies and non-monotonic, recursive $\mathtt{DL}-\mathtt{Safe}$ datalog rules [Rosati, 2006]. In this study, we use a SWRL-based $\mathcal{O}+\mathcal{R}$ combination with the satisfaction of $\mathtt{DL}-\mathtt{Safe}$ conditions to avoid the undecidable computation of decisions for each request for license or privacy rights.

6.2 A unifying semantic model of REL

The future of online privacy will be closely linked to copyright enforcement in the DRM system [Feigenbaum et al., 2002]. In the past, personal information and digital traces could be easily collected and a dossier of the user's preferences could be built by the DRM system. In order to avoid information disclosure without a user's awareness and agreement, we propose a unifying semantic model of REL based on a SWRL-based $\mathcal{O}+\mathcal{R}$ combination, where the features of privacy protection can be incorporated into the DRM system (see [Cohen, 2003]). This unifying semantic model of REL is used to resolve the rights protection dilemma between privacy protection and content usage control in the DRM system (Figure 3). In this unifying semantic model, the abstract concepts for describing the enforcement of content usage rights under certain conditions with corresponding obligations are modeled as ontologies. Furthermore, the criteria for privacy rights protection, such as `purpose`, `action`, `data`, `datauser`, and `obligation` are also considered and incorporated into the rule module of the DRM system. The access control web policies to enforce the digital usage rights and the protection web policies for a DRM user's privacy rights are integrated together to avoid the possible right protection conflicts between these two systems.

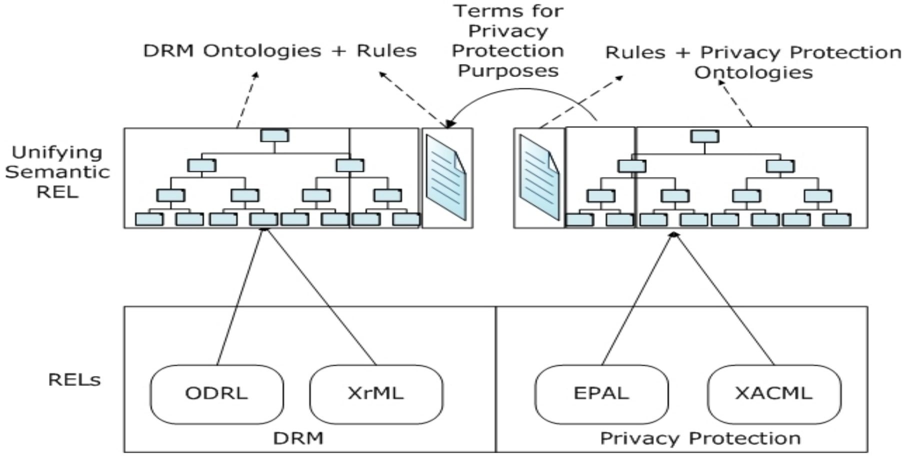

Figure 3: A unifying semantic model of REL shown as a $\mathcal{O}+\mathcal{R}$ combination in a license agreement and access control policies to avoid the possible conflicts of content usage and privacy protection rights in the DRM system

[2] All variables in each rule must appear in at least one of the datalog predicates, i.e., not predicates directly adopted from ontology for the rule's pre-conditions (or body).

6.3 The compromise of rights protection

In the EPAL data model a 5-tuple, i.e., (`user`, `data`, `purpose`, `right`, `obligation`) is used as the set of attributes for privacy protection policies [Karjoth & Schunter, 2002]. The use of a 5-tuple indicates that a particular type of `users` ask for `data` achieving a `purpose` with a certain `right(s)` and `obligation(s)`. In a standalone semantics-enabled privacy protection system, web protection policies for privacy are enforced as verifying the satisfaction of constraints for data user, data type, purpose, rights, and obligation in the ontologies and rules. When we consider enforcing privacy protection rights and content usage rights in the DRM system, we do not directly combine the privacy protection system and the DRM system because this would entail more research on the merging and aligning of ontologies and rules in the heterogeneous systems.

The easiest way to resolve a rights protection dilemma for privacy and content usage is to incorporate a user's privacy rights in the DRM system (see Figure 3). In our unifying semantic model of REL, we allow a content user to specify privacy protection rights unambiguously in the semantics-enabled privacy protection web policies. A user's privacy rights are ensured when the user asks for usage rights to digital content because a user's profile and digital trace usage permission are modeled as a pre-condition of the rules, acting as an extra constraint of execution access control policies in the DRM system.

We extend our previous SWRL-based semantics-enabled DRM system to grant a content user privacy and fair use rights in the DRM system [Hu, 2007] [Hu et al., 2008]. We first request that the content distributor fulfill fair use statutory rights of the copyright law, i.e., allowing a content user to reuse his or her copyrighted digital contents in certain unrestricted ways for the purposes of teaching and research. Then, we require the content distributor to abide by a content user's opt-in and opt-out privacy rights. The data usage policy forces the distributor to comply with privacy protection laws on collecting, using, and disclosing of each user's profile and digital trace.

7 A scenario of rights protection

A scenario of the rights protection use case is extended from [Hu, 2007], and shown as Figure 4, where a license agreement for content usage is signed between a DRM server `Charlie` and two DRM clients, `Alice` and `Bob` to facilitate the content usage rights for a server and privacy and fair use rights for each client.

A license agreement in different expressions are shown as follows:

1. **Natural Language (NL)**:

   A DRM content distributor server, `Charlie`, makes a license agreement with two content consumer clients, `Alice` and `Bob`. After each paying thirty dollars and receiving acknowledgement from `Charlie`, `Alice` and `Bob` are each given personal usage rights and may display an *eBook*, , `TheSemanticWebPrimer`, up to five times in each client's side DRM controller box. They may each print it only once. The total number of actions, either displays or prints, done by `Alice` and `Bob` together, may be at most ten. The usage rights validity period is from `2008/05/07/00 : 00` to `2008/06/06/24 : 00`.

   However, if either `Alice` or `Bob` uses this `eBook` for teaching and research, then the usage rights constraint may be relaxed for fair use to comply with the Copyright law. In this case, a maximum of 25 consecutive pages of each `eBook` can be printed and an unrestricted number of pages can be displayed for an unlimited number of times with unlimited validity period. Fair use is not allowed if the usage purposes are not successfully verified. Furthermore, to protect the privacy rights of `Alice` and `Bob`, we allow each one to specify usage options for respective profiles and digital traces

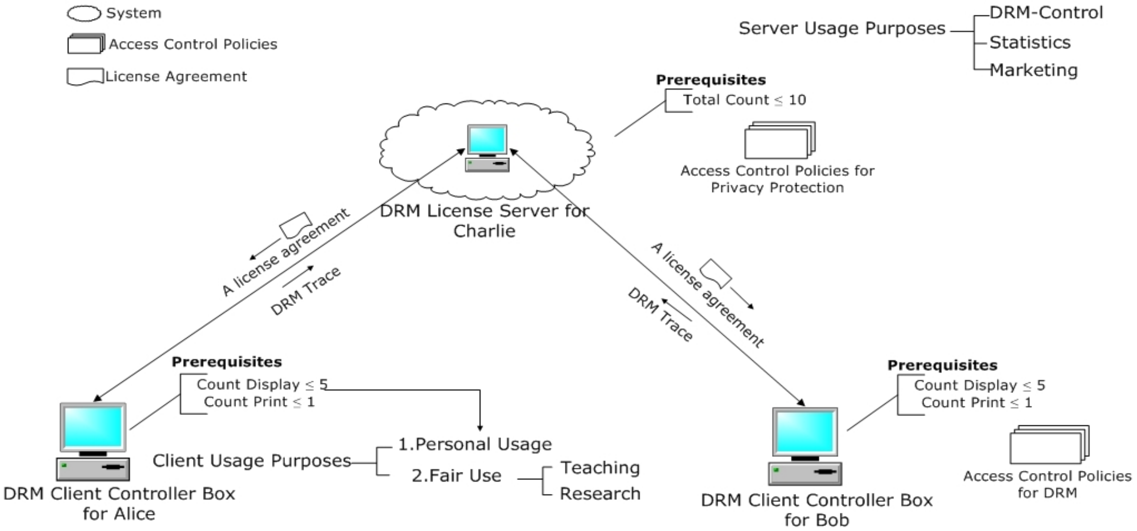

Figure 4: A scenario of rights protection as a license agreement between a license server, `Charlie`, and two clients `Alice`, and `Bob`, to enforce the respective semantics-enabled rights protection web policies

before the DRM server can collect them. In this case, `Alice` only allows her DRM client's controller to disclose her personal profile and online digital trace to a server for DRM control purposes but for no other purposes. On the other hand, `Bob` allows his DRM client's controller to disclose his personal profile and online digital trace to a server, for DRM control as well as non-DRM control purposes, such as marketing and statistics.

2. **Human readable abstract syntax**:

```
agreement
between Charlie and {Alice,Bob}
about The Semantic Web Primer
with inSequence[prePay[30.00],
attribution[Charlie]]

Access control applied to a client for DRM in a client:

clientUsagePurpose:

case Non-FairUse{personal}:
|==> not[and[Time < 2008/05/07/00:00,
Time > 2008/06/06/24:00]]
|==> with usageCount[10] ==>
and[forEachMember[{Alice,Bob};displayCount[5]]
    ==> display,
    forEachMember[{Alice,Bob};printCount[1]]
    ==> print]
```

```
case FairUse{teaching,research}:
|==> forEachMember[{Alice,Bob}] ==> display
|==> forEachMember[{Alice,Bob};
not [and [printPage# > endPage#,
     printPage# < startPage#]]
|==> forEachMember[{Alice,Bob};
     printPageCount[25]]
|==> forEachMember[{Alice,Bob}] ==> print

Access control applied to a server for privacy protection in a client:

serverUsagePurpose:

case DRMControl{DRMControl}:
|==> forEachMember[{Alice, Bob},
clientAllowPurpose[DRMControl]]
|==> forEachMember[{Alice,Bob},
profileDiscloseAllowed[personalProfile]==> disclose]
|==> forEachMember[{Alice,Bob},
traceDiscloseAllowed[digitalTrace]==> disclose]

case Non-DRMControl{Marketing,Statistics}:
|==> forEachMember[{Bob},
clientAllowPurpose[Non-DRMControl]]
|==> forEachMember[{Bob},
profileDiscloseAllowed[personalProfile]==> disclose]
|==> forEachMember[{Bob},
traceDiscloseAllowed[digitalTrace]==> disclose]
```

3. **First Order Logic (FOL)**:

   - Access control applied to a client for DRM in a client:

     $$\begin{array}{l} \forall x((x = Alice \vee x = Bob) \Rightarrow \\ (\exists t_1 \exists t_2 (t_1 < t_2 \wedge Paid(30, t_1) \wedge Attributed(Charlie, t_2)) \\ \Rightarrow \exists y((y = teaching \vee y = research) \wedge HasClientUsagePurpose(x, y)) \\ \Rightarrow \mathbf{Permitted}(x, display, eBook)) \end{array}$$

     $$\begin{array}{l} \Rightarrow \forall p \exists sp \exists ep \\ ((hasPrintPage\#(eBook, pg) \geq startPage\#(eBook, sp) \\ \wedge hasPrintPage\#(eBook, pg) \leq endPage\#(eBook, ep) \\ \Rightarrow hasPrintPageCount(eBook, sub(ep, sp)) \leq 25 \\ \Rightarrow \mathbf{Permitted}(x, display, eBook)) \end{array}$$

     $$\begin{array}{l} \Rightarrow \exists y((y = personal) \wedge hasClientUsagePurpose(x, y)) \\ \Rightarrow \forall t(hasUsageDateTime(t) \geq 2008/05/07/00:00 \\ \wedge\ hasUsageDateTime(t) \leq 2008/06/06/24:00) \end{array}$$

$$\begin{array}{l}
\Rightarrow hasDisplayCount(Alice, id_1) + hasDisplayCount(Alice, id_2) \\
+\, hasPrintCount(Bob, id_1) + hasPrintCount(Bob, id_2) < 10 \\
\Rightarrow (hasDisplayCount(Alice, id_1) < 5\ \wedge hasDisplayCount(Bob, id_1) < 5 \\
\Rightarrow \mathbf{Permitted}(x, display, eBook))
\end{array}$$

$$\begin{array}{l}
\Rightarrow hasPrintCount(Alice, id_2) < 1 \wedge hasPrintCount(Bob, id_2) < 1 \\
\Rightarrow \mathbf{Permitted}(x, print, eBook))))
\end{array}$$

- Access control applied to a server for privacy protection in a client:

$$\begin{array}{l}
\forall x((x = Alice) \Rightarrow \\
\Rightarrow \exists p \exists y \exists f \exists d((p = DRM - control) \wedge y = Charlie) \\
\Rightarrow serverUsagePurpose(p) \wedge personalProfile(f) \\
\Rightarrow clientAllowPurpose(x, p) \wedge serverRequestPurpose(y, p) \\
\Rightarrow ProfileDiscloseAllowed(f, p) \\
\Rightarrow \mathbf{Permitted}(y, disclose, f) \\
\Rightarrow TraceDiscloseAllowed(d, p) \\
\Rightarrow \mathbf{Permitted}(y, disclose, d))
\end{array}$$

$$\begin{array}{l}
\forall x((x = Bob) \Rightarrow \\
\Rightarrow \exists p \exists y \exists f \exists d((p = DRM - control \vee p = Marketing \vee p = Statistics) \wedge y = Charlie) \\
\Rightarrow serverUsagePurpose(p) \wedge personalProfile(f) \\
\Rightarrow clientAllowPurpose(x, p) \wedge serverRequestPurpose(y, p) \\
\Rightarrow ProfileDiscloseAllowed(f, p) \\
\Rightarrow \mathbf{Permitted}(y, disclose, f) \\
\Rightarrow TraceDiscloseAllowed(d, p) \\
\Rightarrow \mathbf{Permitted}(y, disclose, d))
\end{array}$$

7.1 Semantic web policies as $\mathcal{O} + \mathcal{R}$ for a license

Three types of ontology are proposed to describe the concepts of data user, data type, and data usage purpose. In the data user ontology (see Figure 5), $\mathtt{DRM} - \mathtt{Client}$ class is the set of content users for a client $\mathtt{c} \in \mathtt{DRM} - \mathtt{Client}$ class asks for content usage rights and a server $\mathtt{s} \in \mathtt{DRM} - \mathtt{server}$ class through enforcing DRM web policies in a client side's DRM controller box to make its decision. On the other hand, a server $\mathtt{s} \in \mathtt{DRM} - \mathtt{Server}$ class who asks for customer data usage rights and the permission granting is also enforced by privacy protection web policies in a DRM controller box.

The data usage purpose ontology (see Figure 6) provides the classification concepts of data usage purpose for a DRM client and a DRM server. The $\mathtt{Client} - \mathtt{Usage} - \mathtt{Purpose}$ class provides data usage purposes for a DRM client to indicate whether it asks for a fair use or a not-fair use (such as a personal use) data usage. The $\mathtt{Server} - \mathtt{Usage} - \mathtt{Purpose}$ class constrains a DRM server so that it can only disclose data satisfied previous client's opt-in purposes, such as DRM control or marketing,.

In the data type ontology (see Figure 7), the $\mathtt{Digital} - \mathtt{Content}$ class provides the classification concepts of digital media content for users and $\mathtt{Customer} - \mathtt{Data}$ class provides the classification concepts of client data for a selected server to access. In a client's DRM controller box, most of the vocabularies used for describing the concepts of data user and data type ontologies for semantic DRM web policies are imported directly from the DRM server's access control ontologies. The data usage purpose ontology

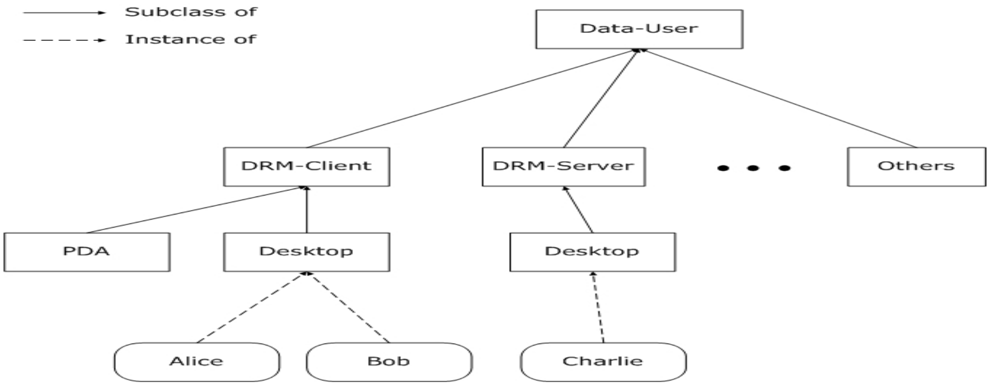

Figure 5: The data user ontology for the concepts of DRM client and DRM server taxonomy

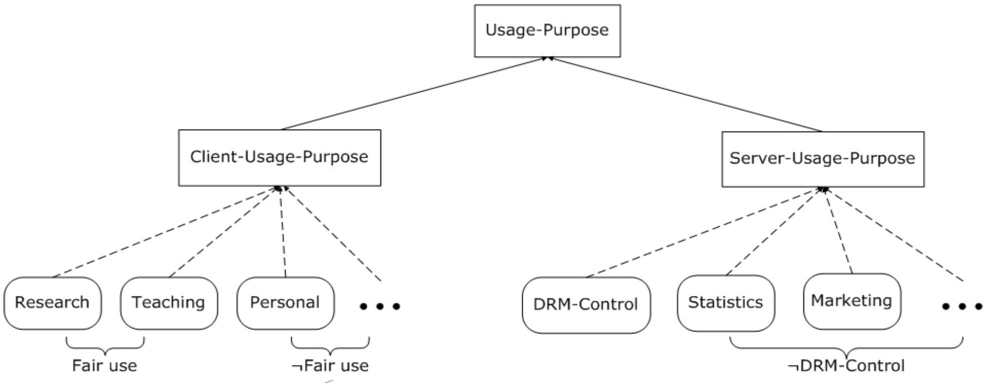

Figure 6: The data usage purpose ontology for the concepts of client's usage purposes and server's usage purposes taxonomy

(see Figure 6) is the exception, where the concepts for describing data usage to achieve the fair use and privacy protection purposes are outside the DRM's $\mathcal{O} + \mathcal{R}$ representations (see Figure 3) [3].

7.2 Properties for data usage purposes

Properties in the data usage purpose ontology in Figure 6 for a DRM client's to constraint the fair use and privacy protection purposes are shown as follows:

- $\texttt{HasClientUsagePurpose} \sqsubseteq \texttt{HasUsagePurpose}$
- $\texttt{HasServerUsagePurpose} \sqsubseteq \texttt{HasUsagePurpose}$

- $\texttt{T} \sqsubseteq \forall\ \texttt{HasUsagePurpose.Data} - \texttt{User}$ [4]
- $\texttt{T} \sqsubseteq \forall\ \texttt{HasUsagePurpose}^{-}\texttt{.Usage} - \texttt{Purpose}$ [5]
- $\texttt{T} \sqsubseteq \forall\ \texttt{HasClientUsagePurpose.DRM} - \texttt{Client}$

[3] The first capital character in the predicates is a marker to indicate that they are directly created in the datalog rule.
[4] The $\texttt{Data} - \texttt{User}$ class is defined as the domain of property $\texttt{HasUsagePurpose}$, as are as the following specifications.
[5] The $\texttt{Usage} - \texttt{Purpose}$ class is defined as the range of property $\texttt{HasUsagePurpose}$, as are the following specifications.

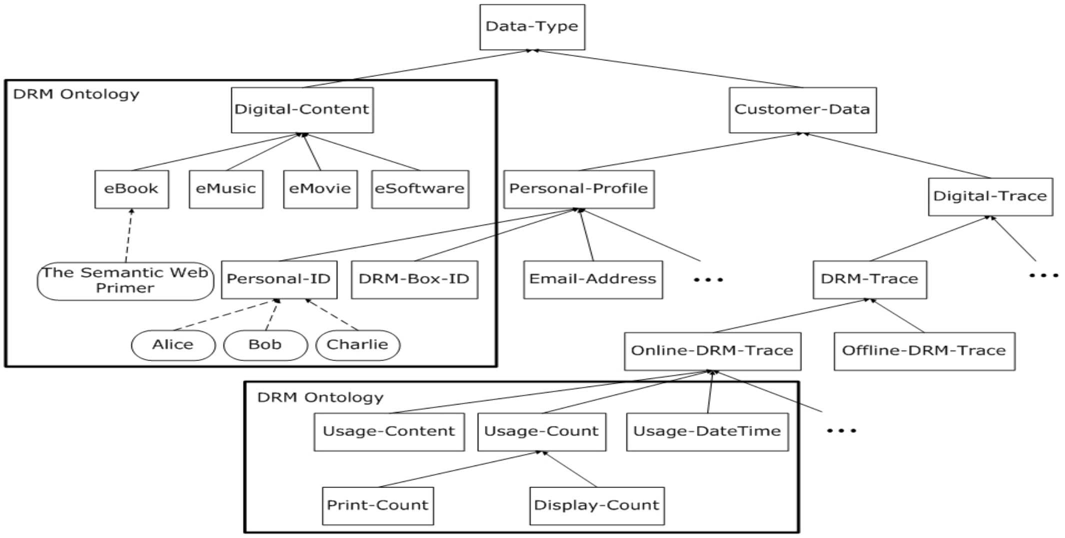

Figure 7: The data type ontology for the concepts of digital content and customer data taxonomy

- $\mathtt{T} \sqsubseteq \forall$ $\mathtt{HasClientUsagePurpose^{-}.Client - Usage - Purpose}$
- $\mathtt{T} \sqsubseteq \forall$ $\mathtt{HasServerUsagePurpose.DRM - Server}$
- $\mathtt{T} \sqsubseteq \forall$ $\mathtt{HasServerUsagePurpose^{-}.Server - Usage - Purpose}$
- $\mathtt{T} \sqsubseteq \forall$ $\mathtt{HasResourceFairUse.Digital - Content}$
- $\mathtt{T} \sqsubseteq \forall$ $\mathtt{HasResourceFairUse^{-}.Client - Usage - Purpose}$

The domain class and range class of a property HasUsagePurpose and its sub-properties, such as HasClientUsagePurpose and HasServerUsagePurpose are imported from the data user and the data usage purpose ontologies. The datalog rules specified for the DRM control and privacy protection policies reuse these imported predicates to ensure all of the permission for information disclosure are satisfied.

7.3 O+R Representations

The $\mathcal{O}+\mathcal{R}$ representations for a DRM server, Charlie, and a DRM client, Alice are shown as the following sections. The given ontology modules are shown as TBox axioms, ABox instances; the rule modules are shown as rules and facts. The enforcement of unifying semantic privacy protection web policies for the DRM system are explicitly demonstrated:

Let $\Pi = (\Gamma, \Delta)$ are the $\mathcal{O}+\mathcal{R}$ knowledge representations of semantics-enabled web policies for privacy protection in the DRM system, where $\Gamma = \mathcal{O} = (\mathtt{axioms}, \mathtt{instances})$, $\Delta = \mathcal{R} = (\mathtt{rules}, \mathtt{facts})$.

- At the DRM license server Charlie's site:

  - $\Gamma = \mathcal{O}$, ontology module at the Charlie site:

*Axioms in the ontology module for DRM:

$$\begin{array}{l}
\mathtt{hasDisplayRights} \sqsubseteq \mathtt{hasUsageRights} \\
\mathtt{hasPrintRights} \sqsubseteq \mathtt{hasUsageRights} \\
\mathtt{eBook} \sqsubseteq \mathtt{Digital-Content} \\
\mathtt{DRM-Client} \sqsubseteq \mathtt{Data-User} \\
\mathtt{DRM-Server} \sqsubseteq \mathtt{Data-User} \\
\mathtt{Print-Count} \sqsubseteq \mathtt{Usage-Count} \\
\mathtt{Display-Count} \sqsubseteq \mathtt{Usage-Count} \\
\mathtt{hasDisplayRights} \sqsubseteq \mathtt{hasUsageRights} \\
\mathtt{hasPrintRights} \sqsubseteq \mathtt{hasUsageRights}
\end{array}$$

$\mathtt{T} \sqsubseteq \forall\ \mathtt{hasUsageCount.Data-User}$ [6]

$$\begin{array}{l}
\mathtt{T} \sqsubseteq \forall\ \mathtt{hasUsageCount}^{-}\mathtt{.Digital-Content} \\
\mathtt{T} \sqsubseteq \forall\ \mathtt{hasDisplayCount.Data-User} \\
\mathtt{T} \sqsubseteq \forall\ \mathtt{hasDisplayCount}^{-}\mathtt{.Digital-Content} \\
\mathtt{T} \sqsubseteq \forall\ \mathtt{hasPrintCount.Data-User} \\
\mathtt{T} \sqsubseteq \forall\ \mathtt{hasPrintCount}^{-}\mathtt{.Digital-Content} \\
\mathtt{T} \sqsubseteq \forall\ \mathtt{hasUsageDateTime.Data-User} \\
\mathtt{T} \sqsubseteq \forall\ \mathtt{hasUsageDateTime}^{-}\mathtt{.Digital-Content}
\end{array}$$

*Facts in the ontology module for DRM:

$$\begin{array}{l}
\mathtt{DRM-Client(Alice)} \\
\mathtt{DRM-Client(Bob)} \\
\mathtt{DRM-Server(Charlie)}, \\
\mathtt{Teacher(Alice)} \\
\mathtt{Researcher(Bob)} \\
\mathtt{eBook(TheSemanticWebPrimer)} \\
\mathtt{hasDisplayRights(Alice, TheSemanticWebPrimer)}
\end{array}$$

*Axioms in the ontology module for privacy protection:

$$\begin{array}{l}
\mathtt{Personal-Profile} \sqsubseteq \mathtt{Customer-Data} \\
\mathtt{Digital-Trace} \sqsubseteq \mathtt{Customer-Data} \\
\mathtt{DRM-Trace} \sqsubseteq \mathtt{Digital-Trace} \\
\mathtt{Online-DRM-Trace} \sqsubseteq \mathtt{DRM-Trace}
\end{array}$$

$$\begin{array}{l}
\mathtt{T} \sqsubseteq \forall\ \mathtt{ClientAllowPurpose.DRM-Client} \\
\mathtt{T} \sqsubseteq \forall\ \mathtt{ClientAllowPurpose}^{-}\mathtt{.Server-Usage-Purpose} \\
\mathtt{T} \sqsubseteq \forall\ \mathtt{ServerRequestPurpose.DRM-Server} \\
\mathtt{T} \sqsubseteq \forall\ \mathtt{ServerRequestPurpose}^{-}\mathtt{.Server-Usage-Purpose} \\
\mathtt{T} \sqsubseteq \forall\ \mathtt{ProfileDiscloseAllowed.Personal-Profile} \\
\mathtt{T} \sqsubseteq \forall\ \mathtt{ProfileDiscloseAllowed}^{-}\mathtt{.Server-Usage-Purpose}
\end{array}$$

[6] In $\mathtt{OWL-DL}$, $\mathtt{maxCardinalityQ}$ is shown as $\leqslant_{\mathtt{n}} \mathtt{P.C}$, where $\mathtt{n}$ is an integer number, $\mathtt{P}$ is a property and $\mathtt{C}$ is a class. So $\leqslant_{5}$ $\mathtt{hasUsageCount.Usage-Count}(?\mathtt{r}, ?\mathtt{uc})$ indicates that a particular resource $\mathtt{r}$, such as $\mathtt{eBook}$ is bound to a variable $?\mathtt{r}$, and the current usage count $\mathtt{uc}$ is bound to a variable $?\mathtt{uc}$ with maximum number 5.

$\mathtt{T} \sqsubseteq \forall\ \mathtt{TraceDiscloseAllowed.Digital} - \mathtt{Trace}$
$\mathtt{T} \sqsubseteq \forall\ \mathtt{TraceDiscloseAllowed}^{-}\mathtt{.Server} - \mathtt{Usage} - \mathtt{Purpose}$

*Facts in the ontology module for privacy protection:

$\mathtt{Personal} - \mathtt{Profile}(\mathtt{AliceProfile})$
$\mathtt{Personal} - \mathtt{Profile}(\mathtt{BobProfile})$
$\mathtt{DRM} - \mathtt{Trace}(\mathtt{AliceDRMTrace})$
$\mathtt{DRM} - \mathtt{Trace}(\mathtt{BobDRMTrace})$
$\mathtt{Server} - \mathtt{Usage} - \mathtt{Purpose}(\mathtt{DRM} - \mathtt{Control})$
$\mathtt{Server} - \mathtt{Usage} - \mathtt{Purpose}(\mathtt{Marketing})$
$\mathtt{ClientAllowPurpose}(\mathtt{Alice}, \mathtt{DRM} - \mathtt{Control})$
$\mathtt{ClientAllowPurpose}(\mathtt{Bob}, \mathtt{DRM} - \mathtt{Control})$
$\mathtt{ClientAllowPurpose}(\mathtt{Bob}, \mathtt{Marketing})$
$\mathtt{ProfileDiscloseAllowed}(\mathtt{AliceProfile}, \mathtt{DRM} - \mathtt{Control})$
$\mathtt{TraceDiscloseAllowed}(\mathtt{AliceDRMTrace}, \mathtt{DRM} - \mathtt{Control})$
$\mathtt{ProfileDiscloseAllowed}(\mathtt{BobProfile}, \mathtt{Marketing})$
$\mathtt{TraceDiscloseAllowed}(\mathtt{BobDRMTrace}, \mathtt{DRM} - \mathtt{Control})$
$\mathtt{ProfileDiscloseAllowed}(\mathtt{BobProfile}, \mathtt{DRM} - \mathtt{Control})$
$\mathtt{TraceDiscloseAllowed}(\mathtt{BobDRMTrace}, \mathtt{Marketing})$

– $\Delta = \mathcal{R}$ rule module at the `Charlie` site:

*Rules in the rule module for DRM:

$\mathtt{hasDisplayRights}(?\mathrm{x}, ?\mathrm{r}) \wedge \mathtt{hasSell}_{\mathtt{d}}\mathtt{Rights}(?\mathrm{x}, ?\mathrm{r})$
$\Longrightarrow \mathtt{hasDisplaySell}_{\mathtt{d}}\mathtt{Rights}(?\mathrm{x}, ?\mathrm{r}) \leftarrow (\mathrm{c1})$

$\mathtt{hasDisplaySell}_{\mathtt{d}}\mathtt{Rights}(?\mathrm{x}, ?\mathrm{r}) \wedge \mathtt{delegate}_{\mathtt{g}}(?\mathrm{x}, ?\mathrm{y}) \wedge \mathtt{hasPrepaid}(?\mathrm{y}, ?\mathrm{a})$
$\Longrightarrow \mathtt{hasDisplayRights}(?\mathrm{y}, ?\mathrm{r}) \leftarrow (\mathrm{c2})$
$\cdots\cdots$

*Rules in the rule module of privacy protection:

$\mathtt{DRM} - \mathtt{Client}(?\mathrm{x}) \wedge \mathtt{DRM} - \mathtt{Server}(?\mathrm{y}) \wedge \mathtt{Server} - \mathtt{Usage} - \mathtt{Purpose}(?\mathrm{p})$
$\wedge \mathtt{Personal} - \mathtt{Profile}(?\mathrm{f}) \wedge \mathtt{ClientAllowPurpose}(?\mathrm{x}, ?\mathrm{p})$
$\wedge \mathtt{ServerRequestPurpose}(?\mathrm{y}, ?\mathrm{p}) \wedge \mathtt{ProfileDiscloseAllowed}(?\mathrm{f}, ?\mathrm{p})$
$\Longrightarrow \mathtt{Permitted}_{\mathtt{Charlie}}(\mathtt{Disclose}, ?\mathrm{f})) \leftarrow (\mathrm{c3})$

$\mathtt{DRM} - \mathtt{Client}(?\mathrm{x}) \wedge \mathtt{DRM} - \mathtt{Server}(?\mathrm{y}) \wedge \mathtt{Server} - \mathtt{Usage} - \mathtt{Purpose}(?\mathrm{p})$
$\wedge \mathtt{Digital} - \mathtt{Trace}(?\mathrm{d}) \wedge \mathtt{ClientAllowPurpose}(?\mathrm{x}, ?\mathrm{p})$
$\wedge \mathtt{ServerRequestPurpose}(?\mathrm{y}, ?\mathrm{p}) \wedge \mathtt{TraceDiscloseAllowed}(?\mathrm{d}, ?\mathrm{p})$
$\Longrightarrow \mathtt{Permitted}_{\mathtt{Charlie}}(\mathtt{Disclose}, ?\mathrm{d}) \leftarrow (\mathrm{c4})$

Rules (c1) and (c2) are $\mathtt{Datalog} - \mathtt{Safe}$ DRM control rules, where all of the variables appearing in each rule's head also appear in the rule's body. Moreover, all of the predicates in these rules are imported from DRM ontologies. More detailed descriptions can refer to [Hu, 2007]. Rules ($\texttt{c3}$) and ($\texttt{c4}$) are privacy protection rules that satisfy the $\texttt{DL}-\texttt{Safe}$ conditions, where all of the rule variables occur at least in one of the datalog predicates in each rule's body. When server Charlie requests a DRM controller box in Alice's site to enforce privacy protection policies for disclosing Alice's profile or a digital trace under the purpose of DRM control, it will be permitted, i.e., the following facts will be derived by rules ($c3$) and ($c4$):

$\texttt{Permitted}_{\texttt{Charlie}}(\texttt{Disclose}, \texttt{AliceProfile})$
$\texttt{Permitted}_{\texttt{Charlie}}(\texttt{Disclose}, \texttt{AliceDRMTrace})$

Similarly, when server Charlie asks for disclosure of Bob's profile or digital trace for DRM control purpose, it is also permitted. However, when server Charlie asks for the disclosure of Alice's profile and digital trace for marketing purposes, it will not be permitted. In fact, we cannot explicitly obtain the following two facts from rules ($\texttt{c3}$) and ($\texttt{c4}$):

$\texttt{Permitted}_{\texttt{Charlie}}(\texttt{Disclose}, \texttt{AliceProfile})$
$\texttt{Permitted}_{\texttt{Charlie}}(\texttt{disclose}, \texttt{AliceDRMTrace})$

The fact is that Alice does not explicitly allow her profile and DRM digital trace to be shown as facts for marketing purposes in the ontologies module. Therefore server Charlie cannot obtain a positive permission from rules ($\texttt{c3}$) and ($\texttt{c4}$).

- At content consumer client Alice's site:

  - $\Gamma = \mathcal{O}$, the ontology module at the Alice site:

    Most of the axioms and facts for privacy protection in the ontology module in an Alice DRM controller box are the same as the results we have shown at server Charlie's site except for fair use access control policies shown as the following:

    *Facts in the ontology module for DRM:

    $\texttt{Teacher}(\texttt{Alice}), \texttt{Researcher}(\texttt{Alice})$
    $\texttt{HasClientUsagePurpose}(\texttt{Alice}, \texttt{Teaching})$
    $\texttt{HasClientUsagePurpose}(\texttt{Alice}, \texttt{Research})$
    $\texttt{HasResourceFairUse}(\texttt{TheSemanticWebPrimer})$
    $\geq_3 \texttt{HasStartPage}\#(\texttt{TheSemanticWebPrimer}, 20)$
    $\leq_{24} \texttt{HasEndPage}\#(\texttt{TheSemanticWebPrimer}, 20)$
    $\leq_{25} \texttt{HasPrintPageCount}(\texttt{TheSemanticWebPrimer}, 17)$

  - $\Delta = \mathcal{R}$, rules in the rule module for DRM to enforce fair use right:

    *Rules in the rule module for DRM to enforce fair use:

$\mathtt{Teacher(?x)} \wedge \mathtt{DRM - Client(?x)} \wedge \mathtt{Client - Usage - Purpose(Teaching)}$
$\Longrightarrow \mathtt{HasFairUseAllowed(?x, Teaching)} \leftarrow \mathtt{(a1)}$

$\mathtt{Researcher(?x)} \wedge \mathtt{DRM - Client(?x)} \wedge \mathtt{Client - Usage - Purpose(Research)}$
$\Longrightarrow \mathtt{HasFairUseAllowed(?x, Research)} \leftarrow \mathtt{(a2)}$

$\mathtt{hasDisplayRights(?x, ?r)} \wedge \mathtt{eBook(?r)} \wedge \mathtt{HasFairUseAllowed(?x, ?p)}$
$\wedge \mathtt{HasClientUsagePurpose(?x, ?p)} \wedge \mathtt{HasResourceFairUse(?r, ?p)}$
$\Longrightarrow \mathtt{Permitted_{Alice}(Display, ?r, ?p)} \leftarrow \mathtt{(a3)}$

$\mathtt{hasDisplayRights(?x, ?r)} \wedge \mathtt{eBook(?r)} \wedge \mathtt{HasFairUseAllowed(?x, ?p)}$
$\wedge \mathtt{HasClientUsagePurpose(?x, ?p)} \wedge \geq_{\mathsf{sp}} \mathtt{HasStartPage\#(?r, ?pg)}$
$\wedge \leq_{\mathsf{ep}} \mathtt{HasEndPage\#(?r, ?pg)} \wedge \leq_{25} \mathtt{HasPrintPageCount(?r, ?c)}$
$\wedge \mathtt{HasResourceFairUse(?r, ?p)} \Longrightarrow \mathtt{Permitted_{Alice}(Print, ?r, ?p)} \leftarrow \mathtt{(a4)}$

$\mathtt{hasDisplayRights(?x, ?r)} \wedge <_{10} \mathtt{hasUsageCount.Usage - Count(?r, ?uc)}$
$\wedge <_{5} \mathtt{hasDisplayCount.Display - Count(?r, ?dc)}$
$\wedge \geq_{\mathsf{2008/05/07/00:00}} \mathtt{hasUsageDateTime.Usage - DateTime(?r, ?ut)}$
$\wedge \leq_{\mathsf{2008/06/06:24:00}} \mathtt{hasUsageDateTime.Usage - DateTime(?r, ?ut)}$
$\Longrightarrow \mathtt{Permitted_{Alice}(Display, ?r)} \leftarrow \mathtt{(a5)}$

$\mathtt{hasPrintRights(?x, ?r)} \wedge <_{10} \mathtt{hasUsageCount.Usage - Count(?r, ?uc)}$
$\wedge <_{1} \mathtt{hasPrintCount.Print - Count(?r, ?pc)}$
$\wedge \geq_{\mathsf{2008/05/07/00:00}} \mathtt{hasUsageDateTime.Usage - DateTime(?r, ?ut)}$
$\wedge \leq_{\mathsf{2008/06/06:24:00}} \mathtt{hasUsageDateTime.Usage - DateTime(?r, ?ut)}$
$\Longrightarrow \mathtt{Permitted_{Alice}(Print, ?r)} \leftarrow \mathtt{(a6)}$

*Facts in the rule module for DRM to enforce fair use right:

$\mathtt{HasFairUseAllowed(Alice, Teaching)} \leftarrow$ derived by $\mathtt{(a1)}$
$\mathtt{HasFairUseAllowed(Alice, Research)} \leftarrow$ derived by $\mathtt{(a2)}$

The DRM system derives fair use rights of teaching and research for Alice by using rules (a1) and (a2) if *Alice* can provide her teacher or researcher's digital certificate to the Charlie server and this certificate is verified successfully by a trusted third party (TTP) to endorse this fair use right. In this case, a maximum of 25 consecutive pages of TheSemanticWebPrimer eBook can be printed and an unrestricted number of pages can be displayed for an unlimited number of times when Alice asks for a request and is derived by ($a3$) and ($a4$) rules.

In another case, when Alice asks for content usage rights for TheSemanticWebPrimer of the eBook by using her personal digital certificate, a non-fair use right of this eBook for Alice is derived by the rules ($a5$) and ($a6$). All of the rules in ($a1$)-($a6$) are $\mathtt{DL - Safe}$ because they satisfy the conditions that all of the variables occurred within the datalog predicate, i.e., the non-DL predicate of the rule's body. The $\mathtt{DL - Safe}$ conditions ensure a decidable computation time for each permission decision of a request.

8 Conclusions

RELs, such as ODRL and P3P, provide an information model and vocabularies for designing a license agreement through the integration of web protection policies from both client and server. However, we sometimes face a semantic ambiguity problem when we use REL-based web protection policies to represent and enforce the access rights of data use. In this chapter, we proposed a unifying semantic model of REL to unambiguously express and enforce fair use and privacy protection rights for digital content users. This formal semantic model of REL is based on the homogeneous (or tight) integration of ontologies and rules, i.e., SWRL-based $\mathcal{O} + \mathcal{R}$ from the semantic web. A real-life scenario was given to demonstrate how to ensure the DRM server's content usage rights and a DRM client's fair use and privacy protection rights. This rights protection scenario, we believe, cannot be easily achieved by other semantic models, such as FOL, DL, and LP.

Appendix

This research was partially supported by the NSC of Taiwan under Grant No. NSC 98-2221-E-004-009.

References

[Anderson, 2006] Anderson, A. H. (2006). A comparison of two privacy policy languages: EPAL and XACML. In *Proceedings of the 3rd ACM Workshop on Secure Web Services (SWS'06)* (pp. 53–60).: ACM.

[Antón et al., 2007] Antón, I. A. et al. (2007). A roadmap for comprehensive online for privacy policy management. *Comm. of the ACM*, 50(7), 109–116.

[Antoniou et al., 2007] Antoniou, G. et al. (2007). Rule-based policy specification. In T. Yu & S. Jajodia (Eds.), *Secure Data Management in Decentralized Systems* (pp. 169–216). Springer.

[Arnab & Hutchison, 2005] Arnab, A. & Hutchison, A. (2005). Fair usage contracts for DRM. In *DRM '05: Proceedings of the 5th ACM workshop on Digital rights management* (pp. 1–7).: ACM.

[Berstel et al., 2007] Berstel, B. et al. (2007). Reactive rules on the web. In *Reasoning Web 2007, Third International Summer School*, LNCS4636 Dresden, Germany: Springer.

[Boley et al., 2007] Boley, H. et al. (2007). Rule interchange on the web. In *Reasoning Web 2007, Third International Summer School*, LNCS 4636 Dresden, Germany: Springer.

[Bonatti et al., 2006] Bonatti, A. P. et al. (2006). Semantic web policies - a discussion of requirements and research issues. In *3rd Eurpoean Semantic Web Conference (ESWC 2006)* Budva, Montenergro.

[Cohen, 2003] Cohen, E. J. (2003). DRM and privacy. *Commun. ACM*, 46(4), 47–49.

[ContentGuard, 2002] ContentGuard, I. (2002). *XrML: The digital rights language for trusted content and services.* Technical report, ContentGuard Inc. http://www.xrml.org/index.asp.

[Cranor et al., 2002] Cranor, L. et al. (2002). The platform for privacy preferences (P3P) 1.0 (p3p 1.0) specification. http://www.w3.org/P3P/.

[Donini et al., 1998] Donini, M. F. et al. (1998). *AL*-log: Integrating datalog and description logics. *Journal of Intelligent Information Systems*, 10(3), 227–252.

[Eiter & Ianni, 2008] Eiter, T. & Ianni, G. (2008). Rules and ontologies for the semantics web. In *Reasoning Web 2008*, LNCS 5224 (pp. 1–53).: Springer.

[Erickson, 2003] Erickson, S. J. (2003). Fair use, DRM, and trusted computing. *Commun. ACM*, 46(4), 34–39.

[Feigenbaum et al., 2002] Feigenbaum, J. et al. (2002). Privacy engineering for digital rights management systems. In *Digital Rights Management (DRM) Workshop 2002*, volume 2320 of *LNCS 2320* (pp. 76–105).: Springer.

[Garcia et al., 2005] Garcia, R., Gallego, I., & Delgado, J. (2005). Formalising ODRL semantics using web ontologies. In *2nd International ODRL Workshop* Lisbon, Portugal. `http://odrl.net/workshop2005/`.

[Grau et al., 2008] Grau, C. B. et al. (2008). OWL 2: The next step for OWL. *Web Semantics: Science, Services and Agents on the World Wide Web 3*, (pp. 309–322).

[Grosof et al., 2003] Grosof, N. B. et al. (2003). Description logic programs: Combining logic programs with description logic. In *World Wide Web 2003* (pp. 48–65). Budapest, Hungary.

[Guth & Iannella, 2005a] Guth, S. & Iannella, R. (2005a). *ODRL V2.0 - Requirements*. Working draft, The ODRL Initiative. `http://odrl.net/2.0/v2req.html`.

[Guth & Iannella, 2005b] Guth, S. & Iannella, R. (2005b). *Open Digital Rights Language (ODRL) Version 2*. Odrl initiative working draft, The ODRL Initiative. `http://odrl.net/2.0/v2req.html`.

[Guth & Iannella, 2007] Guth, S. & Iannella, R. (2007). *ODRL V2.0 - Model Semantics*. Working draft, The ODRL Initiative. `http://odrl.net/2.0/v2req.html`.

[Halpern, 2008] Halpern, Y. J. V. W. (2008). A formal foundation for XrML. *Journal of the ACM*, 55(1), 1–42.

[Hitzler et al., 2010] Hitzler, P. et al. (2010). *Foundations of Semantic Web Technologies*. CRC Press.

[Horrocks et al., 2005] Horrocks, I. et al. (2005). OWL rules: A proposal and prototype implementation. *Web Semantics: Science, Services and Agents on the World Wide Web 3*, (1), 23–40.

[Hu, 2007] Hu, Y. J. (2007). Semantic-driven enforcement of rights delegation policies via the combination of rules and ontologies. In *Workshop on Privacy Enforcement and Accountability with Semantics in conjunction with ISWC+ASWC'07*.

[Hu et al., 2008] Hu, Y. J., Guo, H. Y., & Lin, G. D. (2008). Semantic enforcement of privacy protection policies via the combination of ontologies and rules. In *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC 2008)* Taichung, Taiwan.

[Jajodia et al., 2001] Jajodia, S. et al. (2001). Flexible support for multiple access control policies. *ACM Trans. on Database Systems*, 26(2), 214–260.

[Karjoth & Schunter, 2002] Karjoth, G. & Schunter, M. (2002). A privacy policy model for enterprises. In *15th IEEE Computer Security Foundations Workshop (CSFW)*: IEEE.

[Li et al., 2006] Li, N., Yu, T., & Antón, A. I. (2006). A semantics-approach to privacy languages. *Computer Systems and Engineering (CSSE)*, 21(5).

[Motik et al., 2004] Motik, B., Sattler, U., & Studer, R. (2004). Query answering for OWL-DL with rules. In *3rd International Semantic Web Conference (ISWC) 2004*, LNCS 3298 (pp. 549–563).: Springer.

[Park & Sandhu, 2004] Park, J. & Sandhu, R. T. (2004). The UCON$_{ABC}$ usage control model. *ACM Trans. on Information and System Security*, 7(1), 128–174.

[Patel-Schneider & Siméon, 2002] Patel-Schneider, F. P. & Siméon, J. (2002). Building the semantic web on XML. In *ISWC 2002*, LNCS2342 (pp. 147–161).: Springer.

[Pucella & Weissman, 2006] Pucella, R. & Weissman, V. (2006). *A Formal Foundation for ODRL*. arXiv:cs/0601085v1, Cornell University. `http://arxiv.org/abs/cs/0601085`.

[Rosati, 2006] Rosati, R. (2006). Integrating ontologies and rules: Semantic and computional issues. In *Reasoning Web 2006*, LNCS 4126 (pp. 128–151).

[Tonti et al., 2003] Tonti, G. et al. (2003). Semantic web languages for policy representation and reasoning: A comparison of KAoS, Rei, and Ponder. In *2nd International Semantic Web Conference (ISWC) 2003*, LNCS 2870 (pp. 419–437).: Springer.

[Vimercati et al., 2007] Vimercati, S. D. C. d. et al. (2007). Access control policies and languages in open environments. In T. Yu & S. Jajodia (Eds.), *Secure Data Management in Decentralized Systems* (pp. 21–58). Springer.

[Yu et al., 2004] Yu, T., N. Li, A., & Antón, I. (2004). A formal semantics for P3P. In *ACM Workshop on Secure Web Services* Fairfax, VA, USA. `http://citeseer.ist.psu.edu/750176.html`.

A Semantic Privacy-Preserving Model for Data Sharing and Integration

Yuh-Jong Hu
ENT Lab., Dept. of CS
National Chengchi University
Taipei, Taiwan, 11605
hu@cs.nccu.edu.tw

Jiun-Jan Yang
ENT Lab., Dept. of CS
National Chengchi University
Taipei, Taiwan, 11605
98753036@nccu.edu.tw

ABSTRACT

In this paper, we encompass and extend previous ontology-based data integration system. A semantic privacy-preserving model provides authorized view-based query answering over a widespread multiple servers for data sharing and integration. The combined semantics-enabled privacy protection policies are used to empower the data integration and access control services at the virtual platform ($\mathcal{VP}$). The ontology mapping and merging algorithm creates a global ontology schema at the $\mathcal{VP}$ by integrating multiple local ontology schemas for data sharing. The perfect rules integration of datalog rules enforces the data query and protection services. Semantics-enable policies are combined together at the $\mathcal{VP}$, but the access control criteria specified in each server are still satisfied. The soundness and completeness of data sharing and protection criteria are ensured to support the validity of policy combination. This guarantees the trustworthiness of data sharing and protection services in multiple servers.

Categories and Subject Descriptors

H.4 [**Web Technologies**]:

General Terms

Semantic Web

Keywords

data sharing and integration, semantics-enabled policy, privacy protection,query rewriting, ontology and rule

1. INTRODUCTION

Large enterprises spend a great deal of time and money on data (or information) integration [3]. Data integration is the problem of combining the data from autonomous and heterogeneous sources, and providing users with a unified view of these data through so called global (or mediated) schema. The global schema, which is a reconciled view of the information, that provides query services to end users. The design of a data integration system is a very complex task, which includes several different issues: heterogeneity of the data sources, relation between the global schema and the data sources, limitations on the mechanisms for accessing the sources, and how to process queries expressed on the global schema, etc [11].

Three approaches have been proposed to model a set of *source descriptions* that specify the semantic mapping between the source schema and the global schema. The first one, called global-as-view (GAV), requires that the each concept in the global schema is expressed in terms of query over the data sources. The GAV deals with the case when the stable data source contains details not present in the global schema so it is not used for dynamically adding or deleting data sources.

The second one, called local-as-view (LAV), requires the global schema to be specified independently from the sources, and the source descriptions between the stable global schema, such as ontology and the dynamic data sources are established by defining each concept in the data sources as a view over the global schema [10] [26]. LAV descriptions handle the case in which the global schema contains details that are not present in every data sources.

The third one, called global-local-as-view (GLAV), a source description that combines the expressive power of both GAV and LAV, allowing flexible schema definitions independent of the particular details of the data sources [14]. The data integration system uses these different source descriptions to reformulate a user query into a query over the source schemas. However, data sharing and integration are hampered by legitimate and widespread privacy concerns so it is critical to develop techniques that enables the integration and sharing of data without losing a user's privacy [12].

Privacy protection policies represent a long-term promise made by an enterprise to its users and are determined by business practice and legal concerns. It is undesirable to change an enterprise's promises to customers every time an internal access control rule changes. If possible, we should enable the integration of Platform for Privacy Preferences (P3P) and Enterprise Privacy Authorization Language (EPAL) policies to provide accountable and transparent information processing for data owners to revise their data usage permissions [2].

Although many organizations post online privacy policies, they must realize that simply posting a privacy policy on their websites does not guarantee true compliance with existing legislation. Following the OECD's Fair Information Principles (FIPs)[1], an organization should provide norms of personal information process for its data collection, retention, use, disclosure, and destruction. An organization must also be accountable for its information possession and should declare the purposes of information usage before collection. Moreover, an organization should collect personal information with an individual's consent and disclose personal information only for previously identified purposes [34].

In this paper we are addressing the following research issues. More detailed modelling and implementation will be shown in the later sections.

- Data sharing and protection services are considered in a large number of servers. The incentives for using the virtual platform ($\mathcal{VP}$) is to avoid solving the complex pair-wise problem of ontology matching and rule integration between these servers. Therefore a unified global data sharing and protection service can be achieved at the $\mathcal{VP}$.

- Privacy protection policies are expressed as a combination ontology and rule, i.e. $\mathcal{O} + \mathcal{R}$, where ontology $\mathcal{O}$ includes `TBox` schema and `ABox` instances, and rules $\mathcal{R}$ include deductive rule set ($\mathcal{RS}$) and facts ($\mathcal{F}$). Data sharing and protection in multiple servers are achieved through a combination of semantics-enabled formal protection policy ($\mathcal{FPP}$).

- The challenge of this study is to ensure whether we have a *soundness* and a *completeness* of data sharing and protection by this semantic privacy protection model. For the *sound* criterion, we do not allow unintended data being released to the data users through the global policy schema ($\mathcal{GPS}$) at the $\mathcal{VP}$. Otherwise, it violates the privacy protection policies. As for the *complete* criterion, we do not miss any eligible shared data when a user asks for a data request service at the $\mathcal{VP}$. Therefore, sharable data obtained at the $\mathcal{VP}$ should equal data obtained directly from each server.

Each enterprise server declares its P3P privacy protection policies that takes into account the FIPs criteria (see Figure 1). Then, EPAL policies are established in each site, corresponding to the P3P [24]. For each data request, the data handling and usage controls are based on the EPAL policies. However, P3P and EPAL lack formal and unambiguous semantics to specify privacy protection policies, which are limited in the policy enforcement and auditing support for the software agents. One of the research challenges for the online privacy protection problem is to develop a privacy management framework and a formal semantics language to empower agents to enforce privacy protection policies. Agents must avoid any policy violation of each data request. We attempt to establish a semantic privacy protection model to address this issue. Each server shares its collected data with other servers but without breaking the original data usage commitment to its clients [25].

The contributions of this paper are twofold. We first offer a three layers semantic privacy-preserving model which encompasses and extends the existing work on data sharing and integration using a combination of ontology and rule for the representation of privacy protection policies. In particular, we define a formal policy using ontology for concept descriptions and rule for data query and access control services. Then we focus on solving the soundness and completeness of query rewriting problem using a perfect ontology merging and a perfect rule integration from the local formal protection policies. Followed by each possible data query at the $\mathcal{VP}$, we briefly demonstrate how the sound and complete criteria for privacy protection data integration can be achieved using this semantics-enabled privacy-preserving model.

The paper is organized as follows. In section 2, we present a semantic privacy-preserving model as a framework for data sharing and integration services. In section 3, we define a formal policy combination as an integration of formal policies from autonomous data sources. Each formal policy is composed of ontologies and rules for each independent data source. A privacy protection policy is a type of formal policy used for specifying a data usage constraint from a data owner. In section 4, we formally define a formal policy combination in terms of ontology mapping, alignment, and merging. Then we demonstrate how a perfect rule integration is used for query rewriting at the $\mathcal{VP}$ corresponding to each local schema. In section 6, we briefly prove the soundness and completeness of privacy-preserving data sharing and integration based on this semantic privacy-preserving model. Finally, in the last two sections, we conclude with related work and discussion.

2. A PRIVACY-PRESERVING MODEL

A semantic privacy protection model is proposed with three layers, where the bottom layer provides the data sources from the relational databases, the middle layer provides a semantics-enabled local schema for each independent service domain. The top layer is served at the $\mathcal{VP}$, which provides a unified global view of privacy-preserving data sharing and integration services (see Figure 2).

We have a merged global ontology schema created by mapping and aligning local ontology schemas from multiple local schemas in the middle layer. The idea of using description logic (DL) to model the local and global schemas is to empower the ontology's abstract concept representation and reasoning capabilities. A query is defined as an SQWRL datalog rule in the SWRL-based policy to access to a global ontology [31]. Each SQWRL data service query for a global ontology at the $\mathcal{VP}$ is mapped to multiple queries as SQWRL datalog rules for each local schema. This is a LAV query rewriting service which has been investigated in databases but it is largely unexplored in the context of DL-based ontologies [14].

2.1 Formal Privacy Protection Policy

A policy's explicit representation in terms of ontologies or rules depends on what the underlying logic foundation of your policy language is. If your policies are created from DL-

[1] See http://www.privacyrights.org/ar/fairinfo.htm

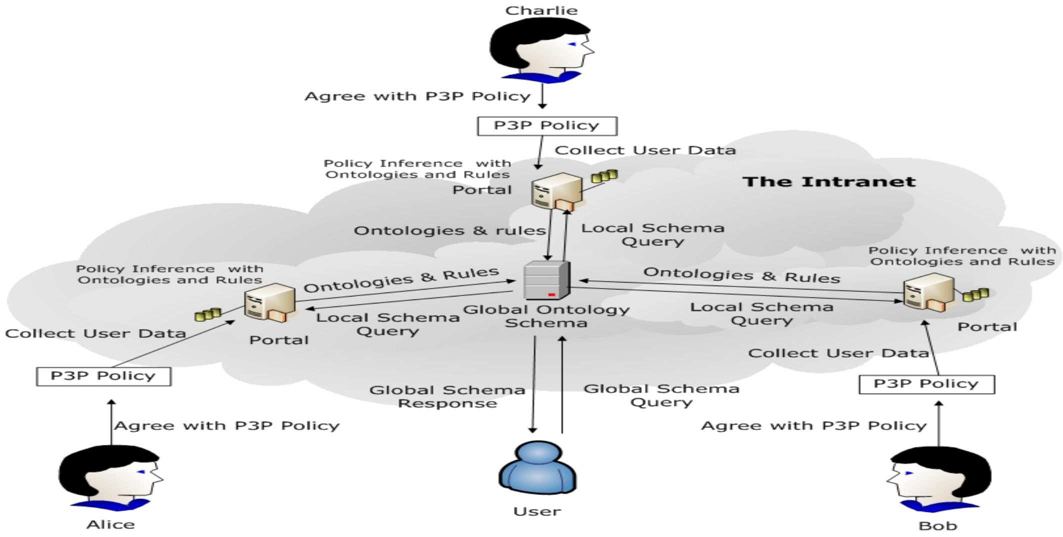

Figure 1: A semantic privacy protection model extended from the integration of P3P and EPAL for data sharing and protection in multiple servers

based policy language, such as Rein or KAoS, then ordinary policies are shown as TBox schema and ABox instances. Otherwise, policies created from LP-based policy language, such as EPAL or Protune ordinary policies are a set of rules with predicates of unary, binary, or ternary variables and facts [5].

In the SemPIF framework [21], we define Policy Interchange Format (PIF) to follows W3C $\mathcal{O} + \mathcal{R}$ standards [6] and strives to provide a mechanism for agents to preserve different policy syntax and semantics throughout its policy integration and interchange. In addition, agents can use meta-PIF, providing further management and reconciliation services of PIF-enabled multiple policies across various domains. In this paper, we apply the SemPIF framework for the privacy-preserving data integration through a combination of formal policies.

A formal policy ($\mathcal{FP}$) is a declarative expression corresponding to a human legal norm that can be executed in a computer system without causing any semantic ambiguity. An $\mathcal{FP}$ is created from a policy language ($\mathcal{PL}$), and this $\mathcal{PL}$ is shown as a combination of ontology language and rule language . Therefore, an $\mathcal{FP}$ is composed of ontologies $\mathcal{O}$ and rules $\mathcal{R}$, where ontologies are created from an ontology language and rules are created from a rule language.

A formal protection policy ($\mathcal{FPP}$) is an $\mathcal{FP}$ that aims at representing and enforcing resource protection principles, where the structure of resources is modelled as ontologies $\mathcal{O}$ but the resources protection is shown as rules $\mathcal{R}$.

A privacy protection policy shown as an $\mathcal{FPP}$ is a combination of ontologies and rules, e.g., $\mathcal{O} + \mathcal{R}$, where DL-based ontologies, such as OWL-DL ontologies provide a well-defined structure data model for data sharing, while Logic Program (LP)-based rules, such as datalog rules provide further expressive power for data query and protection. There are numerous $\mathcal{O} + \mathcal{R}$ combinations available for designing privacy protection policies, such as DLP [17], SWRL [20], and OWL2 RL [16]. Each $\mathcal{O} + \mathcal{R}$ combination implies what expressive power we can extract from ontologies for the rules and vice versa.

The SWRL is one of the $\mathcal{O} + \mathcal{R}$ semantic web languages suitable for a policy representation in the privacy protection model. But this is not an exclusive selection. Other $\mathcal{O} + \mathcal{R}$ combinations, such as CARIN, OWL2 RL are also possible for modeling formal privacy protection policy when their underlying theoretical foundations and development tools are available. We fully utilize the SWRLTab development tools and SQWRL OWL-DL query language [31] in the Protégé to model and further enforce our semantic privacy protection policies.

We face a research challenge of combining SWRL-based privacy protection policies from multiple servers to ensure the soundness and completeness of data sharing and protection criteria. Another challenge is to solve the policy's syntax and semantics incompatibility when we allow policy combination in multiple servers. SWRL is based on the classical first order logic (FOL) semantics that mitigates a possible semantic and syntax inconsistency when policies come from different servers. But we still face a background policy inconsistency problem when default policy assumptions vary between different servers. For example, one server uses open policy assumption, where no explicit option-out for data usage means option-in, but the other server uses closed policy assumption, where no explicit option-in for data usage means option-out. We avoid this kind of policy inconsistency by requesting all sites to use a uniform policy assumption, and to collect option-in data usage choices from users when multiple policies are integrated.

Previous studies for policy combination did not consider solving the problem of merging multiple schemas and integrating access control rules from multiple servers [4] [28]. In this paper we propose a semantic privacy protection model

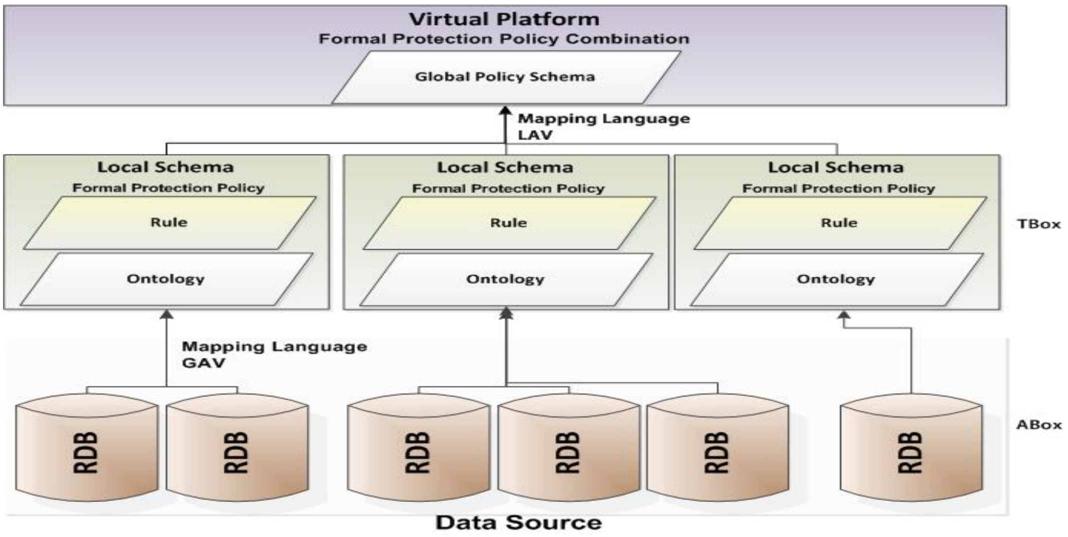

Figure 2: A semantic privacy protection model

that allows flexibly combining `TBoxes` of privacy protection policies without moving `ABox` instances from its original data source until a data request service is initiated (see Figure 3). Therefor the global ontology `TBox` schema and rules created at the $\mathcal{VP}$ have the latest updated incoming data from each server when a user asks a query.

Data integration aims at providing unified and transparent access to a set of autonomous and heterogeneous data sources. The semantic privacy protection model providing global ontology schema for data sharing is similar to the data integration problem solved by $DL-Lite_A$ ontologies shown in [8]. Here we are focusing on data protection besides data sharing and integration.

The goal of ontology-based data integration in $DL-Lite_A$ is to provide a uniform access mechanism to a set of heterogeneous relational database sources, freeing the user from the knowledge about where the data are, what they are stored, and how they can be accessed. The idea is based on decoupling information access from its relational data storage so users only access the conceptual layer shown as ontology, while the relational data layer, hidden to users, manages the data. Compared with $DL-Lite_A$, we have extended and used it as a part of our semantic privacy protection model. We have three layers of data sharing and integration infrastructure instead of two layers in $DL-Lite_A$ so we face a research challenge of ontology merging and rule integration from the middle layer to the top layer when we enforce a privacy protection policy (see Figure 3).

A semantic privacy protection model composed of three main components:

- In the top layer at the $\mathcal{VP}$, we have a global policy schema ($\mathcal{GPS}$), including a global ontology schema ($\mathcal{GS}$) aligned and merged from several local schemas ($\mathcal{LS}$), e.g. `TBox` and a set of rule integration at the middle layer. The $\mathcal{VP}$ provides conceptual data access and protection services that give users a unified conceptual "global view" with access control power for each data request.

- Ontology-based data sources are external, independent, and heterogeneous, and each local ontology was combined with logic program ($\mathcal{LP}$)-based rules for each server in the middle layer.

- Mapping language ($\mathcal{ML}$), which semantically links a $\mathcal{GS}$ and integrated rule set in the top layer to each server's ontology $\mathcal{LS}$ and privacy protection rules in the middle layer.

3. A FORMAL POLICY COMBINATION

A formal policy combination ($\mathcal{FPC}$) in a global policy schema ($\mathcal{GPS}$) allows data sharing as integration of $\mathcal{FP}$ from a variety of servers.

Each $\mathcal{FP}$ is shown as $\mathcal{K} = \mathcal{O} + \mathcal{R}$, where ontology
$\mathcal{O} = (\mathcal{T}, \mathcal{A})$ and rule $\mathcal{R} = (\mathcal{RS}, \mathcal{F})$, $\mathcal{T}$ is `TBox`, and $\mathcal{A}$ is `ABox`;
$\mathcal{RS}$ is a set of rules, and $\mathcal{F}$ is a set of facts.

$$\mathcal{FPC} = \underset{i}{\oplus}\mathcal{K}_i = (\underset{i}{\diamond}\mathcal{O}_i, \underset{i}{\odot}\mathcal{R}_i) = (\underset{i}{\diamond}(\mathcal{T},\mathcal{A})_i, \underset{i}{\odot}(\mathcal{RS},\mathcal{F})_i)$$
$$=((\underset{i}{\diamond}\mathcal{T}_i, \underset{i}{\diamond}\mathcal{A}_i), (\underset{i}{\odot}\mathcal{RS}_i, \underset{i}{\odot}\mathcal{F}_i))$$

where
i is the index of a server i.
$\oplus$ is an operator for formal policy combination,
$\diamond$ is an operator for ontology mapping and merging,
$\odot$ is an operator for rule integration.

In a semantic privacy protection model, a formal protection policy combination ($\mathcal{FPPC}$) allows data sharing and protection from $\mathcal{FPC} = \underset{i}{\oplus}\mathcal{K}_i = (\underset{i}{\diamond}\mathcal{O}_i, \underset{i}{\odot}\mathcal{R}_i)$, where $\underset{i}{\odot}\mathcal{R}_i = (\underset{i}{\odot}\mathcal{RS}_i, \underset{i}{\odot}\mathcal{F}_i)$ provides data query and protection services in $\underset{i}{\diamond}\mathcal{O}_i$.

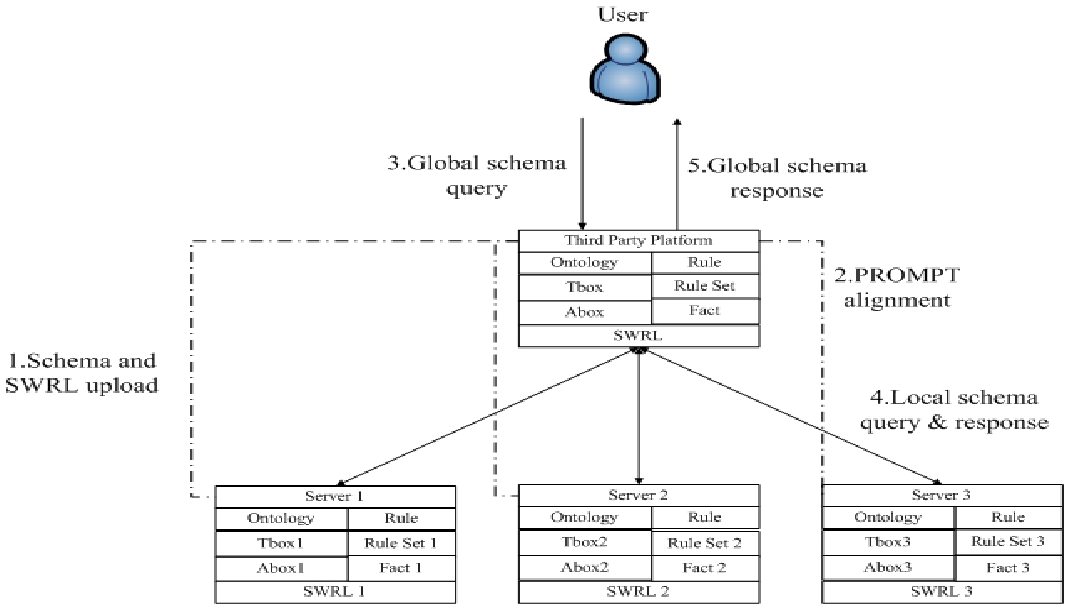

Figure 3: A virtual platform for ontology mapping, merging, and rule integration from multiple servers

3.1 $\mathcal{FPP}$ for Privacy Protection

A privacy protection policy is a type of $\mathcal{FPP}$. We designed an ontology that declares the FIPs' attributes as classes in an $\mathcal{FPP}$ (see Figure 4). The attributes, `purpose`, `datauser`, `data`, `obligation`, and `action` that allow people to specify the constraints of privacy protection policies using related property chains.

Constraint properties is a type of `owl` : `ObjectProperty` that specify what are the feasible domain and range classes of the above attributes. For example, a property `hasOptInPurpose` has its domain and range classes shown as follows:
$\mathtt{T} \sqsubseteq \forall$ `hasOptInPurpose.Data`,
$\mathtt{T} \sqsubseteq \forall$ `hasOptInPurpose`$^{-}$`.Purpose`.

Then a datalog rule, in the SWRL-based policy representation, allows us to use a property chain to combine the two feasible classes together:
`hasOptInPurpose.Data`$(?data)$
$\wedge$ `hasOptInPurpose`$^{-}$`.Purpose`$(?purpose)$
$\longrightarrow$ `hasOptInPurpose`$(?data, ?purpose) \longleftarrow (1)$

Similarly, a `hasOptInDatauser` property has its domain and range classes shown as follows:

$\mathtt{T} \sqsubseteq \forall$ `hasOptInDatauser.Data`,
$\mathtt{T} \sqsubseteq \forall$ `hasOptInDatauser`$^{-}$`.Datauser`.

Then another datalog rule allows us to use another property chain to combine another two feasible classes together:
`hasOptInDatauser.Data`$(?data)$
$\wedge$ `hasOptInDatauser`$^{-}$`.Datauser`$(?datauser)$
$\longrightarrow$ `hasOptInDatauser`$(?data, ?datauser) \longleftarrow (2)$

Based on (1) and (2), we have a feasible set of `ABox` instances with `data`, `purpose`, and `datauser` combinations of an attribute set that was permitted from the original `dataowner` to allow a particular type of `datauser` to ask for a `data` set with a permissive `purpose`. When a server collects a customer's data, the promise of data usage will be ensured if a data user's identity and usage purpose are verified successfully. Otherwise, the data will be kept secret without a data user's awareness.

The specifications are easily extended to the other two attributes, `action` and `obligation`, to complete the FIPs' privacy protection criteria. An ordinary data user is allowed to ask a query service with `action` = `read` at the $\mathcal{VP}$. The other actions, such as `deletion` or `modify`, are only allowed for a system administrator in the middle layer when (s)he asks to delete a user's data to satisfy the obligation of data retention period or for a data owner updates his or her own profile data.

3.2 Data Request Services

A server declares its privacy policy in P3P before a data owner's data is collected. Once a user accepts a server's privacy declaration policy, the data usage constraints are specified as Figure 5, where FIP's five attributes $(?d, ?p, ?du, ?a, ?o)$ for `data`, `purpose`, `datauser`, `action`, and `obligation`, are classes, and `hasOptInDatauser`, `hasOptInPurpose`, etc., are properties proposed as chains of usage constraints for attributes.

For each data request service, an initial feasible parameter input set is $\mathcal{FS} = input(?du, ?r, ?p)$, where $?du \in$ `Datauser`, $?r = read \in$ `Action`, $?p \in$ `Purpose` and output dataset with associated obligations is $output(?d, ?o)$, where $?d \in$ `Data`, $?o \in$ `Obligation`. The feasible dataset shown as `ABox` instances will be discovered by using SQWRL datalog rules. Further permissible actions will be activated when the following data protection policies are satisfied.

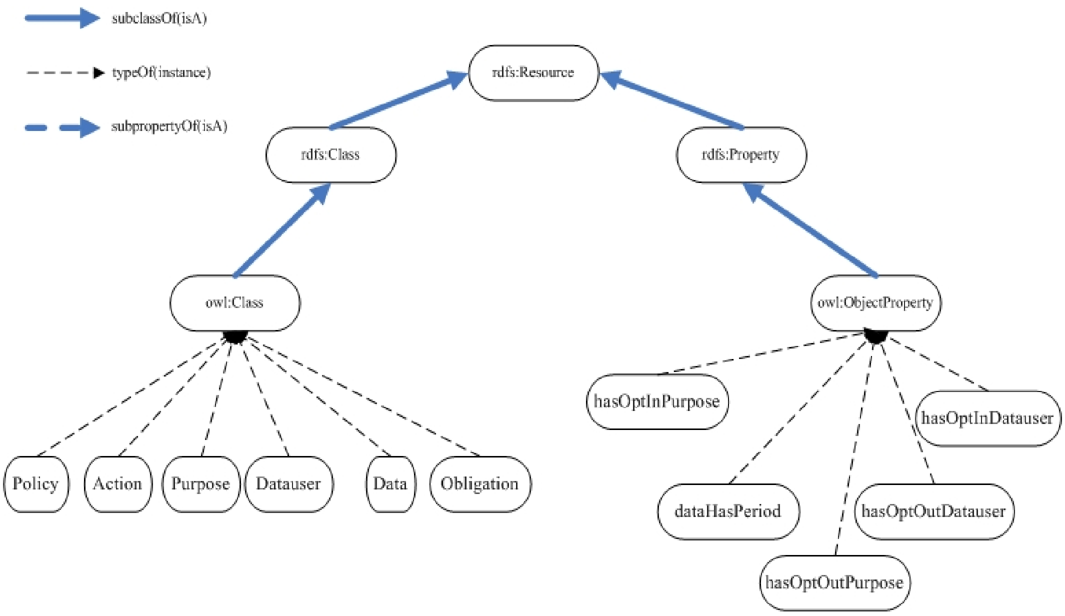

Figure 4: A partial ontology schema for OECD FIPs' attributes shown as $\texttt{owl}:\texttt{Class}$, and constraints shown as $\texttt{owl}:\texttt{Property}$

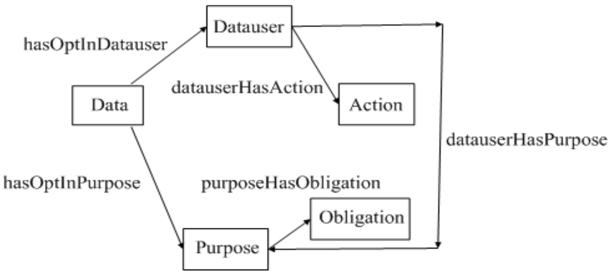

Figure 5: Five major FIP's attributes, such as data, purpose, etc are shown as $\texttt{owl}:\texttt{class}$ and chained by associated $\texttt{owl}:\texttt{Property}$, such as $\texttt{hasOptInDatauser}, \texttt{hasOptInPurpose}$, etc.

3.3 $\mathcal{FPPC}$ at the $\mathcal{VP}$

A data user still possibly collects a sharable data by asking each server individually without using a formal privacy protection policy combination ($\mathcal{FPPC}$). But the high complexity of using query services for all of data sources hinders people from using this data sharing approach. The other possible approach to collect a sharable data is to combine pair-wise servers' policies. Then, we face another scalability problem when more than two servers are intending to share their data.

In this semantic privacy-preserving model, we propose the $\mathcal{VP}$ infrastructure to allow a server in each data source to offer its $\mathcal{FPP}$ at the $\mathcal{VP}$ to enforce $\mathcal{FPPC}$. $\mathcal{FPP}$ in each data source is shown as $\mathcal{K} = \mathcal{O} + \mathcal{R}$, where ontology $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ and rule $\mathcal{R} = (\mathcal{RS}, \mathcal{F})$. At the $\mathcal{VP}$, we only map and merge $\mathcal{T}$, e.g. `TBox` but leave $\mathcal{A}$, e.g. `ABox` instances in its original RDB data source. Similarly, we only integrate $\mathcal{RS}$, a set of rules at the $\mathcal{VP}$ but leave $\mathcal{F}$, a set of facts in its original RDB data source. The benefit of using this approach is to map and merge the `TBoxes` and to integrate the $\mathcal{RS}$ only once.

4. ONTOLOGY SCHEMA MERGING

A merged ontology come from mapping and alignment that provides data integration services. In particular, data integration through ontologies, such as LAV is possible for multiple servers if a mapping language $\mathcal{ML}$ provides a semantic mapping descriptions between the $\mathcal{GS}$ and the underlying $\mathcal{LS}$ for each server [14]. In LAV, the relationships between the $\mathcal{GS}$ and the $\mathcal{LS}$ are established by making LAV assertions. Every assertions has the form $Q_{LS} \rightsquigarrow Q_{GS}$, where Q_{LS} provides the views of the conjunctive query ($\mathcal{CQ}$) over the global schema $\mathcal{GS}$ for each data source, and Q_{GS} is a $\mathcal{CQ}$ over the global schema $\mathcal{GS}$ at the $\mathcal{VP}$.

A $\mathcal{CQ}$ for Q_{LS} can be defined as a privacy-aware authorized view of each server so we do not disclose any non-sharable data to the $\mathcal{VP}$ whenever each server submits its $\mathcal{FPP}$ for ontology merging and rule integration. A $\mathcal{CQ}$ can be defined as a subset of Datalog program, i.e. $\mathcal{CQ}$ containment problem, for query the relational database. This problem was previously investigated in [36]. On the other hand, the connection between the problem of answering queries using extensions of views and the problem rewriting queries using views were studied before through an ontology expressed in DL [15]. In [8], a relational data integration was obtained by mapping each ontology element, e.g. class and property, in the $\mathcal{GS}$ into an SQL query of a relational data source. This is a GAV approach, focusing on mapping the elements of the $\mathcal{GS}$ to a view (SQL query) over the sources. However, our approach is more like LAV , where each term in a SQWRL query for each $\mathcal{LS}_i$ is defined as a view for a SQWRL query in the $\mathcal{GS}$.

4.1 Ontology Mapping and Alignment

A mapping can be shown as (uid, e_1, e_2, n, ρ), where uid is a unique identity for the mapping, e_1, e_2 are entity names, such as class or property, and in the vocabulary of $\mathcal{O}_1$, $\mathcal{O}_2$, n is a numeric confidence measure between 0 and 1, and ρ is a relation such as subsumption ($\sqsubseteq$), equivalence ($\equiv$), or disjointness ($\perp$) between e_1 and e_2 [23].

In this study, the entity names for describing the ontology's class and property, and the structure of using these entity names in the root of the ontology schema for $\mathcal{O}_i$ to define the FIPs' privacy protection criteria (see Figure 5) that are required to be the same. This is a strict constraint to achieve a perfect ontology alignment of this study. Moreover, a perfect mapping language $\mathcal{ML}$ provides semantic mapping services for each entity $e \in \mathcal{GS}$ at the $\mathcal{VP}$ to the corresponding entities $e_i \in \mathcal{LS}_i$.

A perfect ontology alignment obtained via a mapping (uid, e_i, e_j, n, ρ) and merging between $\mathcal{T}_i$ in $\mathcal{O}_i$ and $\mathcal{T}_j$ in $\mathcal{O}_j$ satisfies the following conditions:

- $e_i \in \mathcal{T}_i$ and $e_j \in \mathcal{T}_j$ entity names are either defined for describing the root class names which corresponding to the privacy protection concepts, such as `purpose`, `action`, `datauser`, `data`, and `obligation` or property names, such as `hasOptInDatauser`, `hasOptInPurpose`, etc; Furthermore entity names below the root class and root property are also defined for the descriptions of the underlying subclass and subproperty names.
- A numeric confidence measure n is always equal 1.
- ρ is either equivalence ($\equiv$) or subsumption ($\sqsubseteq$) between entity names of $\mathcal{T}_i$ and $\mathcal{T}_j$ schemas. In an equivalent ($\equiv$) case, we can find a pair of one-to-one corresponding entity names for $e_i \in \mathcal{T}_i$ and $e_j \in \mathcal{T}_j$ in the same layer of the respective ontology schema with $n = 1$; In a subsumption ($\sqsubseteq$) case, there are subclass or subproperty entity names not in the same layer so $e_i \in \mathcal{T}_i$ and $e_i \sqsubseteq e_j \in \mathcal{T}_j$, and vice versa.

4.2 Query Rewriting Services

SWRL combines OWL-DL's ontology language with an additional datalog rule language, where a datalog rule language is shown as an axiom of ontology, a little extension of the OWL-DL language that overcomes the limitations of property chaining in the OWL-DL language [20]. The computation complexity of answering SWRL-based policies might be undecidable regarding the verification of rights access permission unless these policies satisfy the $\mathtt{DL-Safe}$ conditions [29].

SPARQL is a query language for the RDF(S)-based ontologies. OWL2 QL is another query language for the OWL2-based ontologies. We did not use SPARQL query language or OWL2 QL, since our current local and global ontologies are modelled as the OWL-DL ontology language. In fact, SPARQL might not be able to query the complete semantics of the OWL-DL's ontologies. The OWL-DL's ontology queries can be shown as the SQWRL datalog rules, where the $\mathcal{CQ}$ conditions are shown as the rule's body and the query results, i.e., views are shown as the rule's conclusion. SQWRL uses SWRL's strong FOL semantic foundation as its formal semantics so this query language provides a small but powerful array of operators that allows users to construct queries over OWL-DL ontologies [31].

For each data request query service, a perfect mapping language $\mathcal{ML}$ should provide the semantically linking service of an entity name $e \in \mathcal{GS}$ in the datalog rule at the $\mathcal{VP}$ to the entity name $e_i \in \mathcal{LS}_i$ in the datalog rule at $server_i$, where $\mathcal{LS}_i$ is the $\mathtt{TBox}$ of $\mathcal{O}_i$, and e is a class or a property name. If there does not exist an $e_i \in \mathtt{TBox}_i$ in a subtree of the $\mathcal{LS}_i$ on the same layer as $e \in \mathtt{TBox}$ in the global tree of $\mathcal{GS}$, then we can recursively find a superclass or superproperty of e'_i with $e \sqsubseteq e'_i$ as the corresponding entity name, with a confidence measure of $n = 1$.

To successfully fulfill the semantically linking service of any entity name $e \in \mathcal{GS}$ via $\mathcal{ML}$, an ontology schema designer must follow the principles we propose using the specifications of concepts and relations for the FIPs on the root layer of each ontology's local schema's $\mathcal{LS}_i$. But we still allow the designer to use different entity name string, $e_i \in \mathcal{LS}_i$ below the root layer of each local schema and to have an entirely different underlying subtree structure. We use *Prompt* ontology mapping algorithm first to synchronize the entity names between $\mathcal{LS}_i$ and further perform the ontology mappings and aligning operations. Finally, we can perfectly merge their schemas, even if the subtrees of the local schemas are variant.

We use $\mathcal{ML}$ to map the name of a class entity $c \in \mathcal{GS}$ to one of the equivalent local ontology schema's class entity name in a deeper subtree, say $c_j \in \mathcal{LS}_j$, i.e., $c \rightsquigarrow c_j$ in the datalog rule's conditions of each data request service. When the class semantics for c is $c \sqsubseteq c_i$ in the $\mathcal{LS}_i$, i.e., we do not have a corresponding class $c'_i \in \mathcal{LS}_i$ on the same lower layer of a schema tree as $c \in \mathcal{GS}$. All of the $\mathtt{ABox}$ instances a_i in the class name entity c_i, i.e., $a_i \in c_i$ are still feasibly collected for this data request. Because class c_i is a legal domain class or range class for a particular property in the datalog rule for enforcing its privacy protection.

Similarly, a property $p \in \mathcal{GS}$ is mapped to another equivalent property $p_j \in \mathcal{LS}_j$ for the associated datalog rule's body conditions. Then property $p \rightsquigarrow p_j$ might be on a lower layer in the schema tree when compared with property $p_i \in \mathcal{LS}_i$. We still regard property p_i as feasible for its enforcement of the datalog rule on data sharing and protection.

Finally, if we consider mappings for binding property and class from the aligning ontology schema $\mathcal{GS}$ to $\mathcal{LS}_i$ and $\mathcal{LS}_j$ for the respective datalog rules, then we have the following semantically linking relationships by using $\mathcal{ML}$'s mapping services to align the class and property shown as follows:

Property $\mathtt{p} \in \mathcal{GS}$ with its domain class $\mathtt{dc}$ and range class $\mathtt{rc}$ that are mapped to property $\mathtt{p}_i \in \mathcal{LS}_i$ with its domain class $\mathtt{dc}_i$ and its range class $\mathtt{rc}_i$. For each data request service using a perfect mapping language $\mathcal{ML}$, when $\mathtt{p} \sqsubseteq \mathtt{p}_i$, we use property $\mathtt{p}_i$. Otherwise, when $\mathtt{p}_i \sqsubseteq \mathtt{p}$, we use property $\mathtt{p}$ for the datalog rule $\mathtt{r}_i$. When $\mathtt{dc} \sqsubseteq \mathtt{dc}_i$ and $\mathtt{rc} \sqsubseteq \mathtt{rc}_i$, we use class $\mathtt{dc}_i$ and $\mathtt{rc}_i$. Otherwise, when $\mathtt{dc}_i \sqsubseteq \mathtt{dc}$ and $\mathtt{rc}_i \sqsubseteq \mathtt{rc}$, we use class $\mathtt{dc}$ and $\mathtt{rc}$ for the datalog rule $\mathtt{r}_i$.

Example 1. In Figure 6, after we map and align two local partial ontology schemas, $\mathcal{LS}_1$ and $\mathcal{LS}_2$, into a merged partial ontology global schema $\mathcal{GS}$, we receive a data request service with class $\mathtt{P}_{212}$. In the purpose class $\mathtt{P}$, $\mathtt{P}_{111} \leftrightsquigarrow \mathtt{P}_{211}$, but $\mathtt{P}_{212} \in \mathcal{GS}$ does not have a corresponding subclass in $\mathcal{LS}_1$, since $\mathtt{P}_{212} \sqsubseteq \mathtt{P}_{21}$ and $\mathtt{P}_{21} \leftrightsquigarrow \mathtt{P}_{11}$. When a data request service asks for class $\mathtt{P}_{212} \in \mathcal{GS}$, mapping language $\mathcal{ML}$ will map $\mathtt{P}_{212}$ to $\mathtt{P}_{11}$ for the datalog rule $\mathbf{r}_i$ to query the $\mathcal{LS}_1$.

5. PERFECT RULE INTEGRATION

In $\mathcal{FPPC}$, we define an integrated rule set $\underset{i}{\odot}\mathcal{R}_i = (\underset{i}{\odot}\mathcal{RS}_i, \underset{i}{\odot}\mathcal{F}_i)$ to enforce data query and protection services in $\underset{i}{\diamond}\mathcal{O}_i$. In fact, an integrated rule set $\underset{i}{\odot}\mathcal{RS}_i$ is a part of $\mathcal{FPC}$ that was created by collecting the datalog rules, e.g. SQWRL queries, in the formal policies $\mathcal{FP}_i$, from local servers. A datalog rule $\mathbf{r}_i$ in the $\mathcal{R}_i$ of $\mathcal{FP}_i$ is shown as [2]:

$$\mathcal{H} \longleftarrow \mathcal{B}_1 \wedge \mathcal{B}_2 \wedge, \cdots, \wedge \mathcal{B}_n,$$

where $\mathcal{H}$, the query results (or views) are expressed as SQWRL built-ins, such as $\mathtt{sqwrl:select}$ and the rule antecedent $\mathcal{B}_i$, are defined as a pattern matching specifications, i.e., query conditions that are either SQWRL built-ins or class and property predicates from the ontology schema.

A perfect rule integration is defined for the integration of any datalog rules as: $\exists \mathbf{r}_i \in \mathcal{RS}_i$ in $\mathcal{FP}_i$, for the purpose of data sharing and protection without causing conflicts with $\exists \mathbf{r}'_i \in \underset{i}{\odot}\mathcal{R}_i$, $\lambda_i \in \underset{i}{\diamond}\mathcal{O}_i$, i.e., conditions do not exist for $\exists \mathbf{r}_i \models \lambda_i \Rightarrow \exists \mathbf{r}'_i \not\models \lambda_i$, or $\exists \mathbf{r}_i \not\models \lambda_i \Rightarrow \exists \mathbf{r}'_i \models \lambda_i$. Then, $\exists \mathbf{r}'_i \in \underset{i}{\odot}\mathcal{R}_i$ at the $\mathcal{VP}$ can be activated and mapped by the perfect mapping language $\mathcal{ML}$ to $\exists \mathbf{r}_i$, for enabling a global data query and protection service of multiple servers.

Example 2. A rule $\mathbf{r}'_i$ is one of the rules within the integrated rule set at the $\mathcal{VP}$. It asks for a data set ?d with related obligations ?o under the feasible parameter input set $\mathcal{FS}_i = (\mathtt{M1}, \mathtt{TMarketing6}, \mathtt{Read2})$, where data user M1 is a marketing staff with a purpose of achieving telephone marking TMarketing, A rule $\mathbf{r}'_i$ is mapped to a rule $\mathbf{r}_i$ and a rule $\mathbf{r}_j$ using the rule mapping processes when we have done an upward perfect ontology mapping, alignment, merging and a perfect rule integration. A downward perfect mapping language $\mathcal{ML}$ operation maps the $\mathbf{r}'_i$'s predicates, such as class, property to the corresponding predicates in a rule $\mathbf{r}_i$ and a rule $\mathbf{r}_i$ with $\mathtt{MUser(M1)} \sqsubseteq \mathtt{Datauser(M1)}$, $\mathtt{TMarketing(TMarketing6)} \sqsubseteq \mathtt{Purpose(TMarketing6)}$. Therefore, real data query and protection services requested by a rule $\mathbf{r}'_i$ are performed by a rule $\mathbf{r}_i$ and a rule $\mathbf{r}_j$.

A rule $\mathbf{r}'_i$ queries at the $\forall i\ \underset{i}{\diamond}\mathcal{O}_i$:
$\underline{\mathtt{MUser(M1)}} \wedge \underline{\mathtt{TMarketing(TMarketing6)}}$
$\wedge \mathtt{datauserHasPurpose(M1, TMarketing6)}$
$\wedge \mathtt{datauserHasAction(M1, Read2)}$
$\wedge\ \mathtt{hasOptInPurpose(?d, TMarketing6)}$
$\wedge \mathtt{hasOptInDataUser(?d, M1)}$
$\wedge\ \mathtt{purposeHasObligation(TMarketing6, ?o)}$
$\longrightarrow \mathtt{sqwrl:selectDistinct(?d, M1, TMarketing6, Read2, ?o)}$

A rule $\mathbf{r}_i$ queries at the $\mathcal{O}_i$:
$\underline{View(\mathtt{Datauser(M1)})} \wedge \underline{View(\mathtt{TMarketing(TMarketing6)})}$
$\wedge\ \mathtt{datauserHasPurpose(M1, TMarketing6)}$
$\wedge\ \mathtt{datauserHasAction(M1, Read2)}$
$\wedge\ \mathtt{hasOptInPurpose(?d, TMarketing6)}$
$\wedge\ \mathtt{hasOptInDataUser(?d, M1)}$
$\wedge\ \mathtt{purposeHasObligation(TMarketing6, ?o)}$
$\longrightarrow \mathtt{sqwrl:selectDistinct(?d, M1, TMarketing6, Read2, ?o)}$

A rule $\mathbf{r}_j$ queries at the $\mathcal{O}_j$:
$\underline{View(\mathtt{MUser(M1)})} \wedge \underline{View(\mathtt{Purpose(TMarketing6)})}$
$\wedge\ \mathtt{datauserHasPurpose(M1, TMarketing6)}$
$\wedge\ \mathtt{datauserHasAction(M1, Read2)}$
$\wedge\ \mathtt{hasOptInPurpose(?d, TMarketing6)}$
$\wedge\ \mathtt{hasOptInDataUser(?d, M1)}$
$\wedge\ \mathtt{purposeHasObligation(TMarketing6, ?o)}$
$\longrightarrow \mathtt{sqwrl:selectDistinct(?d, M1, TMarketing6, Read2, ?o)}$

Example 3. Under the data protection law, two hospitals, A and B, have allowed to share their patients' Electronic Health Records (EHRs) after patients give their consents for the medication purpose . A patient was hospitalized in the hospital A for a surgery. After that, this patient went to the hospital B for an outpatient medication. A physician in the hospital B was authorized to query this patient's sharable EHR at the $\mathcal{VP}$ collected from hospital A and hospital B's RDB data sources. The vocabularies of partial ontology schemas for hospital A's local schema LS_A, hospital B's local schema LS_B, and the global schema GS at the $\mathcal{VP}$ are shown as Figure 7.

Hospital A has the following terms as its ontology's local schema LS_A vocabularies:

Class: Clinic and HealthData with subClass SurgeryData and HospitalizationData
Property: create with domain class as Hospital and range class as HealthData, i.e.,
$\mathtt{T} \sqsubseteq \forall\ \mathtt{create.Hospital}$
$\mathtt{T} \sqsubseteq \forall\ \mathtt{create^{-}.HealthData}$

Hospital B has the following terms as its ontology's local schema LS_B vocabularies:

Class: Person, HealthCenter, and PatientData with subClass OutPatientData
Property: own, beMedicared with their respective domain and range class are shown as follows:

$\mathtt{T} \sqsubseteq \forall$ own.Person, $\mathtt{T} \sqsubseteq \forall\ \mathtt{Own^{-}.PatientData}$.
$\mathtt{T} \sqsubseteq \forall$ beMedicated.Person,
$\mathtt{T} \sqsubseteq \forall\ \mathtt{beMedicated^{-}.HealthCenter}$.

[2] This datalog rule is related to a $\mathcal{CQ}$ of the form: $v_i \leftarrow conj_i(\overrightarrow{x}_i, \overrightarrow{x}_i)$ [9]

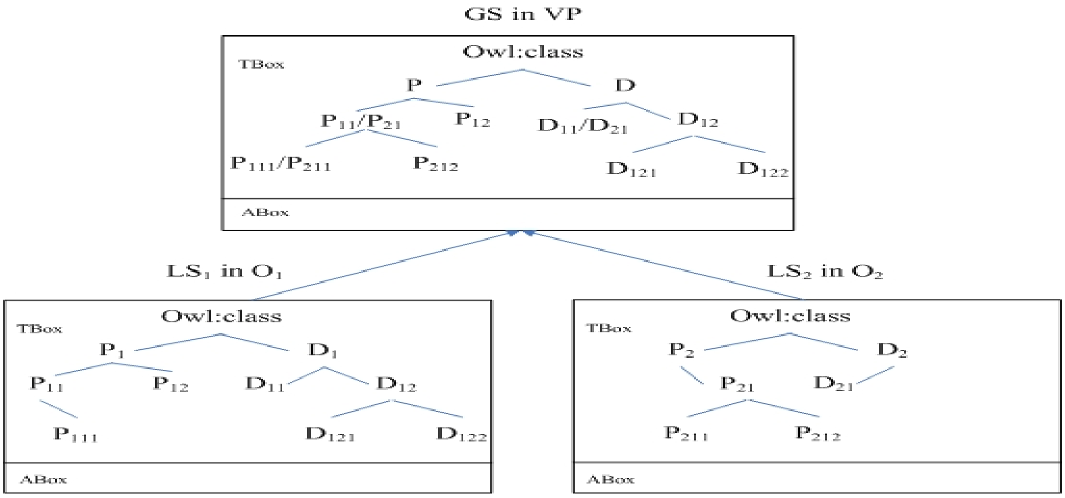

Figure 6: Partial ontology mapping for class alignment and ontology merging

The $\mathcal{VP}$ offers the following vocabularies:
$\texttt{Class:}$ $\texttt{Patient}$, $\texttt{Hospital}$, $\texttt{Surgery}$, and $\texttt{HealthRecord}$
$\texttt{Property:}$ $\texttt{beCured}$, $\texttt{hasHealthRecord}$, $\texttt{generate}$ with their respective domain and range class are shown as follows:

$$\texttt{T} \sqsubseteq \forall\ \texttt{beCured.Patient},\ \texttt{T} \sqsubseteq \forall\ \texttt{beCured}^{-}\texttt{.Hospital}$$
$$\texttt{T} \sqsubseteq \forall\ \texttt{hasHealthRecord.Patient}$$
$$\texttt{T} \sqsubseteq \forall\ \texttt{hasHealthRecord}^{-}\texttt{.HealthRecord}$$
$$\texttt{T} \sqsubseteq \forall\ \texttt{generate.Hospital}$$
$$\texttt{T} \sqsubseteq \forall\ \texttt{generate}^{-}\texttt{.HealthRecord}$$

Use LAV approach to define each class and property in these two hospital local schemas as views in terms of the global schema's vocabularies shown as follows:

Views use at the $\mathcal{VP}$ created from the hospital $\texttt{A}$ local schema's vocabularies are:

$$\texttt{def}(\texttt{V1}_{\texttt{Clinic}}) = \texttt{Hospital}$$
$$\texttt{def}(\texttt{V2}_{\texttt{HealthData}}) = \texttt{HealthRecord}$$
$$\texttt{def}(\texttt{V3}_{\texttt{SuregeryData}})$$
$$= \texttt{HealthRecord} \wedge \forall \texttt{hasMedType.Surgery}$$
$$\texttt{def}(\texttt{V4}_{\texttt{HospitalizationData}})$$
$$=\texttt{HealthRecord} \wedge \forall\ \texttt{hasMedType.Hospitalization}$$
$$\texttt{def}(\texttt{V5}_{\texttt{create}}) = \texttt{generate}$$

Views use at the $\mathcal{VP}$ created from the hospital $\texttt{B}$ local schema's vocabularies are:

$$\texttt{def}(\texttt{V6}_{\texttt{Person}}) = \texttt{Patient}$$
$$\texttt{def}(\texttt{V7}_{\texttt{HealthCenter}}) = \texttt{Hospital}$$
$$\texttt{def}(\texttt{V8}_{\texttt{PatientData}}) = \texttt{HealthRecord}$$
$$\texttt{def}(\texttt{V9}_{\texttt{OutPatientData}})$$
$$= \texttt{HealthRecord} \wedge \forall\ \texttt{hasMedType.OutPatient}$$
$$\texttt{def}(\texttt{V10}_{\texttt{beMedicated}}) = \texttt{beCured}$$
$$\texttt{def}(\texttt{V11}_{\texttt{own}}) = \texttt{hasHealthRecrod}$$

A physician queries a patient's surgery record at the $\mathcal{VP}$ by using a merged global ontology schema based on LAV query rewriting instead of directly requesting each hospital. An original datalog-based SQWRL rule for a query $\texttt{q}$ at the $\mathcal{VP}$ is shown as:

$$\texttt{Patient}(?\texttt{x}) \wedge \texttt{beCured}(?\texttt{x}, ?\texttt{y}) \wedge \texttt{hasHealthRecrod}(?\texttt{x}, ?\texttt{r})$$
$$\wedge\ \texttt{HealthRecord}(?\texttt{r}) \wedge \texttt{hasMedType}(?\texttt{r}, \texttt{Surgery})$$
$$\wedge\ \texttt{generate}(?\texttt{y}, ?\texttt{r}) \longrightarrow \texttt{sqwrl} : \texttt{select}(?\texttt{x}, ?\texttt{r})$$

Query rewriting of the $\texttt{q}$ in terms of two $\mathcal{CQ}$s, e.g., $\texttt{q}_{\texttt{va}}$ and $\texttt{q}_{\texttt{vb}}$, uses views defined at the $\mathcal{VP}$:

$$\texttt{V6}_{\texttt{Person}} \wedge \texttt{V10}_{\texttt{beMedicated}} \wedge \texttt{V11}_{\texttt{own}} \wedge \texttt{V9}_{\texttt{OutPatientData}} \wedge \texttt{V5}_{\texttt{create}}$$
$$\longrightarrow \texttt{sqwrl} : \texttt{select}(?\texttt{x}, ?\texttt{r}) \longleftarrow (\texttt{q}_{\texttt{va}})$$

Above $\texttt{q}_{\texttt{va}}$ query is corresponding to a query as:

$$\texttt{B} : \texttt{Person}(?\texttt{p}) \wedge \texttt{B} : \texttt{beMedicated}(?\texttt{p}, ?\texttt{c}) \wedge \texttt{B} : \texttt{own}(?\texttt{p}, ?\texttt{d})$$
$$\wedge\ \texttt{B} : \texttt{OutPatientData}(?\texttt{od}) \wedge \texttt{A} : \texttt{create}(?\texttt{h}, ?\texttt{hd})$$
$$\longrightarrow \texttt{sqwrl} : \texttt{select}(?\texttt{p}, ?\texttt{od})$$

$$\texttt{V6}_{\texttt{Person}} \wedge \texttt{V10}_{\texttt{beMedicated}} \wedge \texttt{V11}_{\texttt{own}} \wedge \texttt{V3}_{\texttt{SuregeryData}} \wedge \texttt{V5}_{\texttt{create}}$$
$$\longrightarrow \texttt{sqwrl} : \texttt{select}(?\texttt{x}, ?\texttt{r}) \longleftarrow (\texttt{q}_{\texttt{vb}})$$

Above $\texttt{q}_{\texttt{vb}}$ query is corresponding to a query as:

$$\texttt{B} : \texttt{Person}(?\texttt{p}) \wedge \texttt{B} : \texttt{beMedicated}(?\texttt{p}, ?\texttt{c}) \wedge \texttt{B} : \texttt{own}(?\texttt{p}, ?\texttt{d})$$
$$\wedge\ \texttt{A} : \texttt{SuregeryData}(?\texttt{sd}) \wedge \texttt{A} : \texttt{create}(?\texttt{h}, ?\texttt{hd})$$
$$\longrightarrow \texttt{sqwrl} : \texttt{select}(?\texttt{p}, ?\texttt{sd})$$

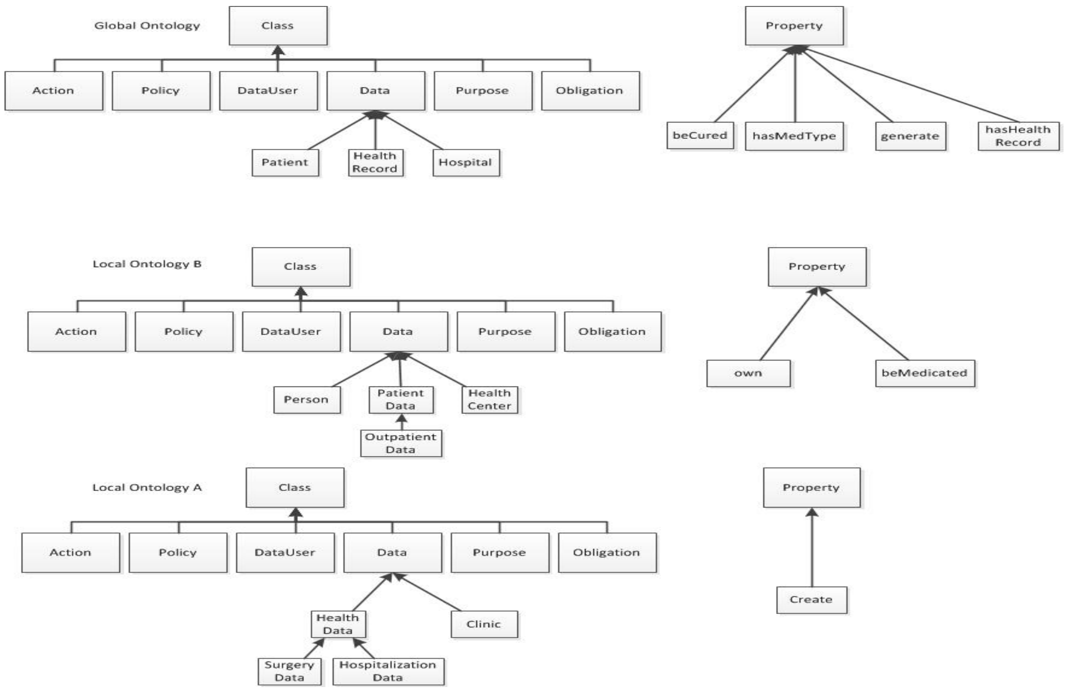

Figure 7: A partial ontology for Electronic Health Record (EHR) sharing and privacy protection

6. SOUNDNESS AND COMPLETENESS

In this section, we briefly demonstrate how the exact query rewriting service satisfies the sound and complete criteria by using the LAV source descriptions based on the $\mathcal{GPS} = (\underset{i}{\diamond}\mathcal{O}_i, \underset{i}{\odot}\mathcal{R}_i)$ at the $\mathcal{VP}$: If $\mathtt{q(x)}$ is a $\mathcal{CQ}$ over $\underset{i}{\diamond}\mathcal{O}_i$ at the $\mathcal{VP}$ and $\mathtt{q_{vi}(x)}$ is a $\mathcal{CQ}$ over $\mathcal{O}_i$ using LAV source descriptions from a data $server_i$, then $\forall x \;\; q(x) \longleftrightarrow \bigsqcup_i q_{vi}(x)$. In [15], authors showed that when a query has a finite number of *maximally contained conjunctive rewritings*, then the complete set of its answers can be obtained as the union of the answer sets of its rewritings. The *datalog-rewriting* was introduced, in which query language is a *hybrid language* with CARIN as its combination of $\mathcal{O} + \mathcal{R}$, and the rewriting language is a *relational language*. They also provided a rewriting algorithm, and showed that the *RewriteQuery* is sound and complete. In comparison, we use LAV for rewriting queries using views and use SWRL as a combination of $\mathcal{O} + \mathcal{R}$. A perfect ontology merging and a rule integration ensure the sound and complete of data sharing and integration in a semantic privacy-preserving model.

6.1 [Soundness]

For the *soundness* criterion, we do not allow any unintentionally released (or protected) data for a user by using a query rewriting service from a rule (query) $\mathtt{r'}_i \in \underset{i}{\odot}\mathcal{R}_i$ at the $\mathcal{VP}$ to direct the query services as rules (queries) $\mathtt{r}_i \in \mathcal{R}_i$ in multiple servers.

THEOREM 1. *After a perfect ontology alignment and rule integration with $\mathcal{FPPC}$, $\exists\mathcal{GPS} = (\underset{i}{\diamond}\mathcal{O}_i, \underset{i}{\odot}\mathcal{R}_i)$ at the $\mathcal{VP}$, Under a particular feasible parameter input set $\mathcal{FS}_i$, if $\lambda_j \in \mathcal{O}_i$ is protected by a $\mathcal{FPP}_i$ at the local $server_i$, i.e., $r_i \in \mathcal{R}_i \not\models \lambda_j$, then $\mathtt{r'}_i \in \underset{i}{\odot}\mathcal{R}_i \not\models \lambda_j$ for the same $\mathcal{FS}_i$, where λ_j is a protective data set in $\mathcal{O}_i$.*

PROOF. (Sketch) If $\mathtt{q(x)}$ is a query over $\underset{i}{\diamond}\mathcal{O}_i$ at the $\mathcal{VP}$ and $\mathtt{q_{vi}(x)}$ is a query over $\mathcal{O}_i$ in a $server_i$, then we need to prove the statement $\forall x \;\; q(x) \longrightarrow \bigsqcup_i q_{vi}(x)$. This statement is equivalent to the original argument: If $r_i \in \mathcal{R}_i \not\models \lambda_j$, then $\mathtt{r'}_i \in \underset{i}{\odot}\mathcal{R}_i \not\models \lambda_j$. The $\mathcal{CQ}$ $\mathtt{q(x)}$ is a query containment of datalog rule $\mathtt{r'}_i$ and the $\mathcal{CQ}$ $\mathtt{q_{vi}(x)}$ is a query containment of datalog rule $r_i \in \mathcal{R}_i$. The statement $\forall x \;\; q(x) \longrightarrow \bigsqcup_i q_{vi}(x)$ is true because based on a perfect rule integration and a perfect ontology merging method, we ensure to avoid the following condition: $\exists \mathtt{r}_i \not\models \lambda_j \Rightarrow \exists \mathtt{r'}_i \models \lambda_j$. □

6.2 [Completeness]

As for the *completeness* criterion, we do not allow any eligible shared data being missed for a query through query rewriting services from a rule (query) $\mathtt{r'}_i \in \underset{i}{\odot}\mathcal{R}_i$ at the $\mathcal{VP}$ to direct query services as rules (queries) $\mathtt{r}_i \in \mathcal{R}_i$ in multiple servers.

THEOREM 2. *After a perfect ontology alignment and rule integration with $\mathcal{FPPC}$, $\exists\mathcal{GPS} = (\underset{i}{\diamond}\mathcal{O}_i, \underset{i}{\odot}\mathcal{R}_i)$ at the $\mathcal{VP}$, Under a particular feasible parameter input set $\mathcal{FS}_i$, if $\lambda_j \in \mathcal{O}_i$ is sharable by a $\mathcal{FPP}_i$ at the local $server_i$, i.e., $r_i \in \mathcal{R}_i \models \lambda_j$, then $\mathtt{r'}_i \in \underset{i}{\odot}\mathcal{R}_i \models \lambda_j$ for the same $\mathcal{FS}_i$, where λ_j is a sharable data set in $\mathcal{O}_i$.*

Proof. (Sketch) If $\mathtt{q(x)}$ is a query over $\underset{i}{\diamond}\mathcal{O}_i$ at the $\mathcal{VP}$ and $\mathtt{q_{vi}(x)}$ is a query over $\mathcal{O}_i$ in a $server_i$, then we need to prove the statement $\forall x \;\; q(x) \longleftarrow \bigsqcup_i q_{vi}(x)$. This statement is equivalent to the original argument: If $r_i \in \mathcal{R}_i \models \lambda_j$, then $\mathtt{r'}_i \in \underset{i}{\odot}\mathcal{R}_i \models \lambda_j$. The $\mathcal{CQ}$ $\mathtt{q(x)}$ is a query containment of datalog rule $\mathtt{r'}_i$ and the $\mathcal{CQ}$ $\mathtt{q_{vi}(x)}$ is a query containment of datalog rule $r_i \in \mathcal{R}_i$. The statement $\forall x \;\; q(x) \longleftarrow \bigsqcup_i q_{vi}(x)$ is true because based on a perfect rule integration and a perfect ontology merging method, we ensure to avoid the following condition: $\exists \mathtt{r}_i \models \lambda_j \Rightarrow \exists \mathtt{r'}_i \not\models \lambda_j$. □

7. RELATED WORK

Data integration is a pervasive challenge faced in the applications that need to query across multiple autonomous and heterogeneous data sources. This problem has been received considerable attention from researchers in the fields of Artificial Intelligence and Database System more than a decade [18] [27]. A logic of the Description Logic family is used to model the ontology managed by the integration system, to formulate queries posed to the system, and to perform several types of automated reasoning supporting both the modeling, and the query answering process [11]. The ontology expresses the domain of interest of the information system at a high level of abstraction, and the relationship between data at the sources and instances of concepts and roles in the ontology is expressed by means of mappings, such as GLAV, GAV, LAV [7] [33].

Unfortunately, data integration and sharing are hampered by legitimate and widespread privacy concerns so it is critical to develop a technique to enable the integration and sharing of data without losing privacy. We face a challenge to develop a privacy framework for data integration that is flexible and clear to the end users [12]. View-based query answering over DL provides a framework to answer a query under the assumption that the only accessible information consists of the precomputed answers to a set of queries, called views. Privacy-aware access to data, each user is associated with a set of views, called authorization views, which specify the information that the user is allowed to access [9].

We encompass and extend previous ontology-based data integration system. A semantic privacy-preserving model provides authorized view-based query answering over a widespread multiple servers for data sharing and integration. The combined semantics-enabled privacy protection policies are used to empower the data integration and access control services at the virtual platform.

The role-based access control (RBAC) model is used to enforce the access control policies with a static role assignment for a stand-alone system. It is therefore not useful for solving the privacy protection problem. In fact, the RBAC model did not consider the prime elements of the FIPs, so it is not intended for a privacy protection problem. In [32], the $UCON_{ABC}$ might be useful for the privacy protection problem, but it did not explicitly allow the data sharing and protection in multiple sites. The EFAF access control model is an extension of the FAF that provided the solution for privacy protection [22] [24]. This is the closest method to our solution, but its privacy protection control is more on the logic program and less on the ontology schema for the structure data modelling. This also prevents the data sharing and protection in multiple sites. The other similar models for enforcing the enterprise privacy protection go to the following EPAL [25] [37]. OASIS XACML is a policy language for privacy and digital rights protection. But it is an XML-based policy language so the policies based on XACML possibly have ambiguous semantics that prevent using a flexible policy combination in multiple servers [1].

8. CONCLUSION

In this study, we propose a semantic privacy protection model to solve the data sharing, integration and protection problem through a combination of ontology and rule language, e.g. SWRL. The perfect ontology alignment and merging algorithm creates a global ontology schema at the $\mathcal{VP}$ by integrating multiple local ontology schemas in the multiple servers. The perfect rule integration avoids the possible conflicts between the datalog rules at the $\mathcal{VP}$ and in the multiple servers when we use these rules to enable data integration and protection. Semantics-enabled policies are combined together at the $\mathcal{VP}$, so we simplify the data sharing and protection processes, but the soundness and completeness criteria specified in each data collection server for access control are still satisfied. This supports the trustworthiness of a policy combination in multiple servers.

Acknowledgements

This research was partially supported by the NSC Taiwan under Grant No. NSC 99-2221-E-004-010.

9. REFERENCES

[1] A. H. Anderson. A comparison of two privacy policy languages: EPAL and XACML. In *Proceedings of the 3rd ACM Workshop on Secure Web Services (SWS'06)*, pages 53–60. ACM, 2006.

[2] I. A. Antón et al. A roadmap for comprehensive online for privacy policy management. *Comm. of the ACM*, 50(7):109–116, July 2007.

[3] A. P. Bernstein and L. M. Haas. Information integration in the enterprise. *Comm. of the ACM*, 51(8):72–79, July 2008.

[4] A. P. Bonatti et al. An algebra for composing access control policies. *ACM Trans. on Information and Systems Security*, 5(1):1–35, February 2002.

[5] P. Bonatti and D. Olmedilla. Policy language specification, enforcement, and integration. project deliverable D2, working group I2. Technical report, REWERSE, 2005.

[6] J. d. Bruijn. RIF RDF and OWL compatibility. Technical report, W3C, Oct. 2009.

[7] D. Calvanese et al. Description logic framework for information integration. In *Proc. of the 6th Int. Conf. on Principles of Knowledge Representation and Reasoning*, pages 2–13. Morgan Kaufmann, 1998.

[8] D. Calvanese et al. Data integration through $DL - Lite_A$ ontologies. In *3rd Int. Workshop on*

Semantics in Data and Knowledge Base (SDKB), volume 4925, pages 26–47. Springer, 2008.

[9] D. Calvanese et al. View-based query answering over description logic ontologies. In *Proc. of KR-2008*. AAAI Press, 2008.

[10] D. Calvanese and G. D. Giacomo. Data integration: A logic-based perspective. *AI Magazine*, 26(1):59–70, 2005.

[11] D. Calvanses et al. Description logics for information integration. In *Computational Logic*, LNAI 2408, pages 41–60. Springer, 2002.

[12] C. Clifton et al. Privacy-preserving data integration and sharing. In *Data Mining and Knowledge Discovery*, pages 19–26. ACM, 2004.

[13] J. Euzenat and P. Shvaiko. *Ontology Matching*. Springer-Verlag, 2007.

[14] M. Friedman et al. Navigational plans for data integration. In *Proc. of the Sixteen National Conference on Artificial Intelligence (AAAI'99)*, pages 67–73. AAAI/MIT Press, 1999.

[15] F. Goasdoué and M.-C. Rousset. Answering queries using views: a KRDB perspective for the semantic web. *ACM Trans. on Internet Technology*, 4(3):255–288, August 2004.

[16] C. B. Grau et al. OWL2: The next step for OWL. *Web Semantics: Science, Services and Agents on the World Wide Web 3*, pages 309–322, 2008.

[17] N. B. Grosof et al. Description logic programs: Combining logic programs with description logic. In *World Wide Web 2003*, pages 48–65, Budapest, Hungary, 2003.

[18] A. Halevy, A. Rajaraman, and J. Ordille. Data integration: The teenage years. In *VLDB'06*, pages 9–16. ACM, 2006.

[19] Y. A. Halevy. Answering queries using views: A survey. *The VLDB Journal*, 10(4):270–294, 2001.

[20] I. Horrocks et al. OWL rules: A proposal and prototype implementation. *Web Semantics: Science, Services and Agents on the World Wide Web 3*, 3(1):23–40, 2005.

[21] Y. J. Hu and H. Boley. Sempif: A semantic meta-policy interchange format for multiple web policies. In *2010 IEEE/WIC/ACM Int. Conference on Web Intelligence and Intelligent Agent Technology*, pages 302–307. IEEE, 2010.

[22] S. Jajodia et al. Flexible support for multiple access control policies. *ACM Trans. on Database Systems*, 26(2):214–260, June 2001.

[23] E. Jiménez-Ruiz et al. Ontology integration using mappings: Towards getting the right logical consequences. In *ESWC 2009*, LNCS 5554, pages 173–187. Springer, 2009.

[24] G. Karjoth and M. Schunter. A privacy policy model for enterprises. In *15th IEEE Computer Security Foundations Workshop (CSFW)*. IEEE, June 2002.

[25] G. Karjoth, M. Schunter, and E. V. Herreweghen. Translating privacy practices into privacy promises - how to promise what you can keep. In *Proceedings of the 4th International Workshop on Policies for Distributed Systems and Networks (POLICY'03)*. IEEE, 2003.

[26] M. Lenzerini. Data integration: A theoretical perspective. In *Proceedings of the ACM Symposium on Principles of Database Systems (PODS)*, pages 233–246. ACM, 2002.

[27] Y. A. Levy. Logic-based techniques in data integration. In T. Yu and S. Jajodia, editors, *Logic-based Artificial Intelligence*, pages 1–27. Kulwer, 2001.

[28] P. Mazzoleni et al. XACML policy integration algorithms. *ACM Trans. on Information and System Security*, 11(1), 2008.

[29] B. Motik, U. Sattler, and R. Studer. Query answering for OWL-DL with rules. In *3rd International Semantic Web Conference (ISWC) 2004*, LNCS 3298, pages 549–563. Springer, 2004.

[30] A. Nash and A. Deutsch. Privacy in glav information integration. In *ICDT 2007*, LNCS 4353, pages 89–103. Springer, 2007.

[31] J. M. O'Connor and A. K. Das. SQWRL: a query language for OWL. In *OWL: Experiences and Directions (OWLED)*, volume 529. CEUR Workshop Proceedings, 2009.

[32] J. Park and R. T. Sandhu. The UCON$_{ABC}$ usage control model. *ACM Trans. on Information and System Security*, 7(1):128–174, 2004.

[33] A. Poggi et al. Linking data to ontologies. *Journal on Data Semantics X*, 4900:133–173, 2008.

[34] D. C. Raab. The future of privacy protection, 2004.

[35] R. Rosati. On the decidability and complexity of integrating ontologies and rules. *Web Semantics: Science, Services and Agents on the World Wide Web 3*, pages 61–73, 2005.

[36] D. J. Ullman. Information integration using logical views. *Theoretical Computer Science*, 239:189–210, 2000.

[37] S. D. C. d. Vimercati et al. Access control policies and languages in open environments. In T. Yu and S. Jajodia, editors, *Secure Data Management in Decentralized Systems*, pages 21–58. Springer, 2007.

國科會補助專題研究計畫項下出席國際學術會議心得報告

日期:99_年12_月07_日

計畫編號	NSC 98－2221－E－004－009		
計畫名稱	給隱私權保護與著作權管理使用的語意式web規範互通語言		
出國人員姓名	胡毓忠	服務機構及職稱	國立政治大學資訊科學系教授
會議時間	2010年8月31日至 2010年9月3日	會議地點	加拿大多倫多市
會議名稱	(中文)IEEE/WIC/ACM Web智慧與智慧型代理者技術國際研討會 （英文）IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology		
發表論文題目	(中文)提供多重Web規範交換的語意式meta規範格式 (英文)SemPIF: A Semantic Meta-Policy Interchange Format for Multiple Web Policies		

一、參加會議經過

本研討會於2010年8月31-9月3日於加拿大多倫多市的York大學舉行。York大學位於多倫多市近郊約1個多小時車程的郊區，附近的大學則有位於多倫多市世界知名的多倫多大學。

8月31日的研討會首日進行的是Workshop。9月1日至9月3日則是正式的研討會。研討會分成兩部分來進行：Web Intelligence, Intelligent Agent。在這個基礎上論文的發表大致上分成幾個track來進行論文的發表。Web Information Retrieval and Filtering, Web Mining, Semantics and Ontology Engineering, Social Network Analysis, Web Services, Distributed Problem Solving, etc。從這些tracks的觀察我們可以知道這個研討會涵蓋面非常的廣，但是主要著重的還是Web Mining以及Web Information Retrieval。計畫主持人發表論文SemPIF: A Semantic Meta-Policy Interchange Format for Multiple Web Policies所排定的時段是在9月3日早上的Semantics and Ontology Engineering IV的場次。我也順便主持了這個場次其他演講者的論文發表。本場次的演講共有5位的論文發表者，其中有一位大陸東北大學的研究學者並未出席，另外德州大學的論文發表者則請人代打。另外則有京都大學、多倫多大學、及本人（代表政治大學)。本研討會在9月2日晚上舉辦餐宴於多倫多港上的遊輪上，台灣也有不少的學者參與此盛會。

二、與會心得

對於本研討(IEEE/WIC/ACM Web 智慧與智慧型代理者技術國際研討會)會計畫主持人在過去幾年都是這個研討會 Web Intelligence (WI)的論文評審委員(Program Committee)。這個研討會雖然在語意網和智慧型代理者的知名度沒有 International Semantic Web Conference (ISWC), World Wide Web (WWW), 以及 Autonomous Agent and Multi-Agent System (AAMAS)來得大。但是根據往年和今年的經驗總論文的投稿數量也都可以達到 300-400 篇左右。主要的投稿來源還是來自於大陸的學者以及分佈在世界各地大學與研究單位的大陸學者。這個國際研討會的論文有被收錄到 IEEE Explore 的數位圖書資料庫，因此這也是吸引人投稿的因素之一。除此之外，本國際研討會也能夠每年分別在世界的五大洲選定國家來舉辦。整體來說研討會的論文主要議題還是在 Web Intelligence (WI)的部分，因此對於 Intelligent Agent 這一部份的學術成果發表顯得比較弱勢。實際上論文審稿作業也是 Web Intelligence, Intelligent Agent 分開審稿的方式來進行。因此其審稿委員的聘任和論文的挑選與評定也是分別去進行的。除此之外，在這個研討會發表的論文如果有獲得前幾名的殊榮還有機會在修稿並延伸之後在 WIAT 的期刊刊登。

三、考察參觀活動(無是項活動者略)

無。

四、建議

這幾年隨著大陸經濟起飛與學者國際化參與意願的提升，使得各項研討會都可以看到大陸學者的大量參與並且發表論文。除了他們在過去幾年在海外求學之後定居於當地形成一個有利的學術網絡之外，大陸的內陸學者只要有意願並且經費許可的情況之下都會主動的出國並且參加。除此之外，世界知名的電腦科學研討會也陸陸續續在組織運作上佔有很重要的一席之地，這是我們台灣學者必須要體認的一項事實。

五、攜回資料名稱及內容

研討會論文議事錄一本與全論文集光碟片一張。

六、其他

2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology

SemPIF: A Semantic Meta-Policy Interchange Format for Multiple Web Policies

Yuh-Jong Hu
Dept. of Computer Science
NCCU, Taipei, Taiwan
hu AT cs.nccu.edu.tw

Harold Boley
IIT – e-Business,
NRC-CNRC, Canada,
Harold.Boley AT nrc.gc.ca

***Abstract*—We propose a semantics-enabled layered policy architecture for the exchange and management of multiple policies created by different policy languages on the Web. This architecture consists of four layers: Unifying Logic (UNL), Policy Interchange Format (PIF), Privacy Protection/DRM (PPD), and Domain Specific Applications (DSA). A meta-Policy Interchange Format (meta-PIF) layer is also introduced, side by side with the corresponding PIF layer, allowing agents in the facilitator to provide uniform services of interchange, reconciliation, and combination of policies. This SemPIF architecture extends W3C's Semantic Web architecture to permit the reuse of earlier work. A scenario of agents in the facilitator employing SemPIF for Digital Rights Management (DRM) and privacy protection policies on digital library subscription services will be demonstrated.**

***Keywords*-semantic web; ontology and rule; computer policy; privacy protection; digital rights management**

I. Introduction

In the Semantic Web, information is given well-defined meaning to better enable computers and people to work in cooperation. The well-known Semantic Web layered architecture[1] has undergone revisions reflecting the evolution of layers such as the Description Logic (DL)-based ontology language OWL [1], the Horn Logic (HL)-based rule language RIF [2], and their relationship. On the other hand, policy languages, such as Rei [3], KAoS [4], Protune [5], have also been proposed – on the basis of DL and LP – to allow agents understand policies and to enforce these policies as intended by their semantics. However, the semantic bases of policy languages vary considerably, ranging from DL to HL to Logic Programming (LP), e.g. leading to different stances w.r.t. the unique name assumption (UNA) and the closed world assumption (CWA) [6]. This makes policies created in these policy languages hard to interchange and combine with each other.

Policies are formulated and treated as knowledge bases, i.e. ontologies and rules ($\mathcal{O}+\mathcal{R}$). Many operations can be automated, thereby reducing ad-hoc program coding to a minimum and enabling automated documentation [5]. Policy frameworks also need to support interoperability. Moreover, the context of a policy is itself described in a machine-understandable way. Therefore, we propose a semantics-enabled policy architecture consisting of four layers: Unifying Logic (UNL), Policy/meta-Policy Interchange Format (PIF/meta-PIF), Privacy Protection/DRM (PPD), and Domain Specific Applications (DSA). Here UNL directly corresponds to the layer "Unifying Logic" of the most recent version of the Semantic Web architecture. We also introduce a meta-PIF layer, side by side with the corresponding PIF layer, allowing software agents in the facilitator, to provide the management functions of interchange, reconciliation, and combination of policies. The Policy Web architecture can be viewed as an extension of the Semantic Web architecture shown as Fig. 1.

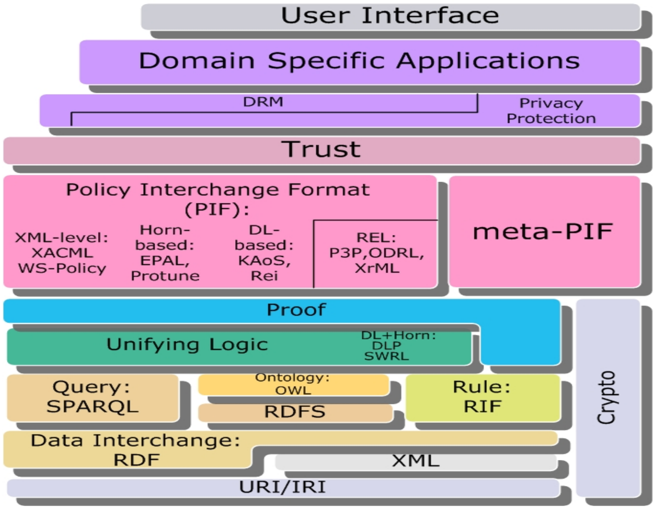

Fig. 1. SemPIF: A semantics-enabled layered policy model centered on semantic policy interchange format (PIF) and meta-PIF.

PIF is built on DL-based ontologies and LP-based rules, i.e. $\mathcal{O}+\mathcal{R}$, that allow agents in the facilitator to support interchange services of policies created from different policy languages. In addition, we may use meta-PIF to specify meta-policies for managing policies created from different policy languages. A meta-policy is a policy about policies that provides a set of rules for realizing services needed for the management of policies. Moreover, a meta-policy consists of a set of rules for setting up the priority of polices to coordinate,

[1] http://www.w3.org/2007/03/layerCake.svg

978-0-7695-4191-4/10 $26.00 © 2010 IEEE
DOI 10.1109/WI-IAT.2010.238

enforce, and even negotiate policies [7].

In a particular policy language framework, policy management services could be implemented as meta-policies as shown in the Rei framework [3] or it could be implemented as policy administration tools as shown in KAoS [4]. In the Protune framework, the role of meta-policies is in governing the behavior to reduce ad-hoc programming efforts and to improve policy readability and maintainability. However, policy management services in these frameworks were only allowed to operate within their own environments. For added flexibility, SemPIF allows agents in the facilitator to use meta-policies providing the management services of policy interchange, combination, and negotiation across multiple heterogeneous domains.

In contrast to other policy languages, such as KAoS, Rei, and Protune, PIF follows W3C $\mathcal{O}+\mathcal{R}$ standards [8] and strives to provide a mechanism for agents to preserve different policy syntaxes and semantics throughout its policy integration and interchange. In addition, agents can use meta-PIF, providing further management and reconciliation services of PIF-enabled multiple policies across various domains (see Fig. 2).

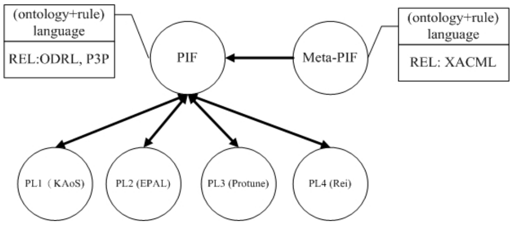

Fig. 2. Policy integration and interchange for various policy languages (PL) through PIF and meta-PIF in the SemPIF

An XML-based Rights Expression Language (REL) lacks semantic expressive power so it is a restricted form of policy language in the PIF layer. Currently, there are three RELs available, i.e. P3P for privacy protection as well as ODRL and XrML for DRM. Unfortunately, policies created from these XML-level RELs lack formal semantics, which prevents agents from automatically and accurately interpreting and processing these policies.

A formal semantic model for policies could be expressed and enforced as a combination of $\mathcal{O}+\mathcal{R}$. Obviously, if we do not know what are the available expressive features of each $\mathcal{O}+\mathcal{R}$ combination, then we cannot decide which combination will be the best one to represent the formal semantics of RELs. We have shown the semantics of DRM policies in PPD as a homogeneous combination of $\mathcal{O}+\mathcal{R}$, i.e. SWRL, where both $\mathcal{O}+\mathcal{R}$ are embedded in a logical language $\mathcal{L}$, to structure the semantics of ODRL [9]. We also have shown the semantics of privacy protection policies in PPD as a hybrid combination of $\mathcal{O}+\mathcal{R}$, i.e. DL+log, where a strict separation between the rule predicates and ontology predicates, to formalize the semantics of P3P in the PIF layer [10].

Another issue addressed by our investigation of the DRM vs. privacy usage control problem is the following. While DRM systems are collecting personal information for usage control, it is quite possible that they might also invade privacy rights. To reconcile this conflict, the $(\mathcal{O}+\mathcal{R})$ language in meta-PIF permits agents to enforce the fine-grained mapping and merging of ontologies with interchangeable rules from policies on privacy protection and DRM. This paves the way for accomplishing the objective of unifying multiple Web policies through SemPIF.

Finally, we will show a scenario for digital library subscription services in a client-server model and demonstrate how to use it by agents in the facilitator to eliminate possible conflicts between a server's DRM policies and a client's privacy protection policies.

II. Semantic Web Layered Architecture

We have both Web markup languages and Semantic Web languages in the Semantic Web Layered Architecture (SWLA) (see Fig. 1). XML / XML Schema and URI/IRI references constitute the foundation, which provides interoperable syntax for RELs at the PIF layer. The semantics of RELs need to be formalized with one of the $\mathcal{O}+\mathcal{R}$ combinations from the UNL, to provide meanings for policies. For Semantic Web languages, we have ontology languages, rule languages, and a combination of ontology and rule languages, i.e. $\mathcal{O}+\mathcal{R}$ languages. The ontology languages include the graph-based RDF(S) and the DL-based OWL. The Horn-based rule languages and their extensions to LP-based rule languages include RIF and RuleML. SWRL is a Semantic Web language using a combination of OWL-DL ontologies and Datalog RuleML rules so it is an $\mathcal{O}+\mathcal{R}$ language [11]. OWL 2 RL and its combination with RIF is another emerging $\mathcal{O}+\mathcal{R}$ language that can be compared with DLP[2].

A. Unifying Logic

PIF and Sem-PIF are built on the unifying logic of DL-based ontologies and LP-based rules [6]. In the UNL layer, DL is a subset of the First Order Logic (FOL). DL provides a basic logic foundation for an ontology language, such as OWL 2. Similarly, Horn logic and LP provides a basic logic foundation for rule languages, such as RIF or RuleML. One of LP's characteristics, procedural attachment, is not included in DL (or FOL) but this feature is very important for the execution of policy's actions.

DLP was introduced as the intersection of DL and LP [12], which has quite limited expressive power when compared to other $\mathcal{O}+\mathcal{R}$ combinations, such as $\mathcal{AL}$-log, $\mathcal{DL}$-log, etc. [13] [14]. The homogeneous $\mathcal{O}+\mathcal{R}$ combination of DL and Datalog provides the logic foundation of SWRL. These combinations of $\mathcal{O}+\mathcal{R}$ have much more expressive power than DLP regarding $\mathcal{O}+\mathcal{R}$, which is needed for Sem-PIF. However, given the ongoing $\mathcal{O}+\mathcal{R}$ research[3], we have not fixed yet any one combination for SemPIF's representation and enforcement requirements.

[2] The combination of OWL 2 and RIF has been shown in http://www.w3.org/2005/rules/wiki/OWLRL.

[3] See, e.g., ONTORULE http://ontorule-project.eu/.

B. Policy Interchange Format

The PIF layer consists of regular DL-based policy languages, such as Rei, KAoS; or Horn-based policy languages, such as EPAL [15], and XML-syntax policy languages, such as XACML [16]. P3P and EPAL were proposed as policy languages for privacy protection in the corresponding client-server and server-server models [17] [18]. As REL sublayer, ODRL and XrML were proposed for designing DRM policies [19] [20]. DL-based or Horn-based logic can be used as foundations to underpin these RELs with explicitly defined semantics for policy languages.

A policy's explicit representation in terms of ontologies or rules depends on what the underlying logic foundation of your policy language is. If policies are created from a DL-based policy language, such as Rei or KAoS, then ordinary policies are shown as $\mathcal{TB}$ox ontology schemas and $\mathcal{AB}$ox instances. Otherwise, with policies created from LP-based policy languages, such as EPAL, ordinary policies are sets of rules and facts using unary or binary predicates.

These policy languages in the PIF layer do not fully utilize the syntax and semantics expressive power of OWL or RDF(S) shown in the SWLA. Therefore, we do not expect these policy languages to be able to leverage the power of existing ontology or rule languages. Another restriction is policies created from different policy languages might not be able to interchange or negotiate with each other. This calls for the use of SemPIF to achieve policy interchange, combination, reconciliation, and negotiation.

C. Privacy Protection and DRM

Privacy protection and DRM are introduced as independent but intertwined layers on top of PIF / meta-PIF and the Trust layer (see Fig. 1). This relationship reflects that access rights enforcements for these two domains are closely related with each other. In [21], the authors proposed that a DRM system should consider user-desirable privacy rights indicated in the Fair Information Principles (FIP), such as data collection, retention, use, disclosure, and destruction, etc., when it enforces privacy protection policies. Otherwise, user privacy rights might be violated. A scenario will be demonstrated to show how agents in the facilitator employ SemPIF to integrate DRM and privacy protection policies (see section III).

III. A SCENARIO OF DIGITAL LIBRARY SUBSCRIPTION

Protection policies are created from various policy languages, such as ODRL, P3P, XACML, and EPAL, for enforcing DRM and privacy protection. This access-control scenario is extended from policy-aware access control for the open Web environment [22]. Agents in the facilitator use PIF-based policies to provide services of integrating semantics-enabled protection policies between a client and a server. Moreover, Agents use meta-PIF-based policies to manage policies, which permits clients and a server to compromise on their respective rights and obligations(see Fig. 3).

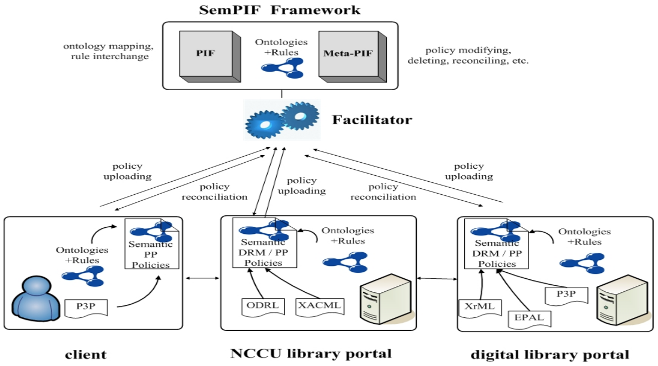

Fig. 3. Agents in the facilitator provide policy interchange services with PIF-based policies and policy management services with meta-PIF-based policies

A. Web server's policies

The $NCCU$ university library has subscribed to $IEEE$, ACM, and $Springer$ digital library services, which provide a set of eJournal article access rights for authorized students and staff. There are two types of policy for an IEEE Web server: one for DRM and the other for the declaration of privacy statements.

1) Policies in the IEEE server: The IEEE publisher has two PIF-based policies: $\texttt{policy}(\texttt{drm1}-\texttt{IEEE})$ for DRM and $\texttt{policy}(\texttt{pp1}-\texttt{IEEE})$ for privacy declaration. $\texttt{policy}(\texttt{drm1}-\texttt{IEEE})$ indicates that the policy's name is drm1-IEEE, which corresponds to a URI as a policy indicator for agents to apply a meta-PIF policy to manage it. The predicates of each RIF rule are specified in PIF-based ontologies (see Fig. 4 and Fig. 5).

- $\texttt{policy}(\texttt{drm1}-\texttt{IEEE})$:
  If a student owns a valid student ID issued by a department of a university, e.g. a registrar, and its library is one of the subscribers on the IEEE publisher's list, then the student is endowed with DRM usage rights $\{\texttt{download},\texttt{view},\texttt{print}\}$ for eJournals from a Web server of the IEEE publisher's delegation.

$$\begin{aligned}
&?\texttt{st}\#\texttt{Student}\wedge ?\texttt{id}\#\texttt{StudentID}\wedge ?\texttt{st}[\texttt{own}\rightarrow ?\texttt{id}]\\
&\wedge ?\texttt{uni}[\texttt{nccuHasPartR}\rightarrow ?\texttt{rg}]\wedge ?\texttt{st}[\texttt{enrolledAt}\rightarrow ?\texttt{uni}]\\
&\wedge ?\texttt{rg}[\texttt{issue}\rightarrow ?\texttt{id}]\wedge ?\texttt{uni}[\texttt{nccuhasPartN}\rightarrow ?\texttt{lib}]\\
&\wedge ?\texttt{lib}[\texttt{subscribedTo}\rightarrow \texttt{IEEE}]\\
&\wedge \texttt{IEEE}[\texttt{hasPublished}\rightarrow ?\texttt{ejr}]\\
&\wedge \texttt{IEEE}[\texttt{endowedWith}\rightarrow ?\texttt{rgt}]\wedge ?\texttt{rgt}[\texttt{appliedTo}\rightarrow ?\texttt{ejr}]\\
&\wedge \texttt{IEEE}[\texttt{delegate}\rightarrow ?\texttt{st}] \Longrightarrow ?\texttt{st}[\texttt{endowedWith}\rightarrow ?\texttt{d}]\\
&\wedge ?\texttt{st}[\texttt{endowedWith}\rightarrow ?\texttt{v}]\ \wedge ?\texttt{st}[\texttt{endowedWith}\rightarrow ?\texttt{p}]\\
&\wedge ?\texttt{d}\#\texttt{Download}\wedge ?\texttt{d}[\texttt{appliedTo}\rightarrow ?\texttt{ejr}]\\
&\wedge ?\texttt{v}\#\texttt{View}\wedge ?\texttt{v}[\texttt{appliedTo}\rightarrow ?\texttt{ejr}]\\
&\wedge ?\texttt{p}\#\texttt{Print}\wedge ?\texttt{p}[\texttt{appliedTo}\rightarrow ?\texttt{ejr}].
\end{aligned}$$

- $\texttt{policy}(\texttt{pp1}-\texttt{IEEE})$:
  If a person is endowed with DRM usage rights from a Web server of the IEEE's publisher and this publisher has the purpose of enforcing DRM control for collecting,

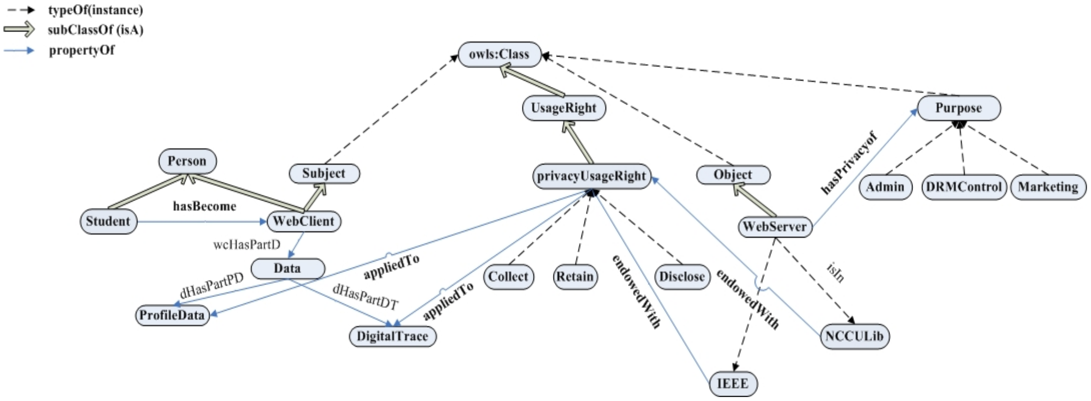

Fig. 4. A PIF-based ontology for privacy protection policies

retaining, and disclosing a person's data, then the IEEE publisher is endowed with privacy usage rights $\{\texttt{collect}, \texttt{retain}, \texttt{disclose}\}$ on this data from a person's delegation, including profiles and digital traces in the Web server under condition of a retention period of two months after the data are first collected.

$$
\begin{aligned}
&?\texttt{per}[\texttt{endowedWith} \rightarrow ?\texttt{drmr}] \wedge ?\texttt{drmr}[\texttt{appliedTo} \rightarrow ?\texttt{ejr}]\\
&\wedge \texttt{IEEE}[\texttt{hasPublished} \rightarrow ?\texttt{ejr}]\\
&\wedge \texttt{IEEE}[\texttt{hasPrivacyOf} \rightarrow \texttt{DRMControl}]\\
&\wedge ?\texttt{per}[\texttt{dHasPartPD} \rightarrow ?\texttt{prf}] \wedge ?\texttt{per}[\texttt{dHasPartDT} \rightarrow ?\texttt{dif}]\\
&\wedge ?\texttt{per}[\texttt{endowedWith} \rightarrow ?\texttt{ppr}] \wedge ?\texttt{per}[\texttt{delegate} \rightarrow \texttt{IEEE}]\\
&\wedge \texttt{Retain}[\texttt{hasDuration} \rightarrow =\text{2Month}]\\
&\wedge ?\texttt{sdtime}[\texttt{dHasPartD} \rightarrow ?\texttt{dtime}]\\
&\wedge ?\texttt{edtime}[\texttt{dHasPartD} \rightarrow ?\texttt{dtime}]\\
&\wedge \text{subtract-dateTimes}(?\texttt{edtime}, ?\texttt{sdtime}) \leq \texttt{Retain}\\
&\Longrightarrow \texttt{IEEE}[\texttt{endowedWith} \rightarrow ?\texttt{ppr}]\\
&\wedge ?\texttt{ppr}[\texttt{appliedTo} \rightarrow ?\texttt{prf}] \wedge ?\texttt{ppr}[\texttt{appliedTo} \rightarrow ?\texttt{dit}].
\end{aligned}
$$

In $\texttt{policy}(\texttt{drm1} - \texttt{IEEE})$, we use ODRL basic primitive vocabularies $\texttt{principal}, \texttt{asset}, \texttt{right}$, or $\texttt{obligation}$ to define a license agreement between principals, i.e., library, registrar, university, and publisher. Similarly, in $\texttt{policy}(\texttt{pp1} - \texttt{IEEE})$, we use P3P basic primitive vocabularies, such as $\texttt{user}, \texttt{owner}, \texttt{purpose}, \texttt{rights}, \texttt{obligation}$ to define a privacy protection agreement between data user and data owner. All of the basic vocabularies are defined in the DRM or privacy protection ontology's schema (see Fig. 4 and Fig. 5), so the semantics of ODRL and P3P RELs are formalized. Furthermore, policies specified in other policy languages for DRM and privacy protection can be mapped to the PIF-based policies for the purposes of policy interchange and integration.

2) Policies in a Web client: A student, *John*, as a Web client has privacy protection policies, i.e., $\texttt{policy}(\texttt{pp3} - \texttt{John})$, $\texttt{policy}(\texttt{pp4} - \texttt{John})$ to address how and what of his personal data can (or cannot) be collected, retained, or disclosed from a Web server. Here we show the policies in natural language only.

- $\texttt{policy}(\texttt{pp3} - \texttt{John})$:
  If an $\texttt{eJournal}$ distributor from {ACM, IEEE, Springer} has the purpose of enforcing DRM control by collecting, retaining, and disclosing data on John as the Web client, then it is endowed with privacy usage rights $\{\texttt{collect}, \texttt{retain}\}$ on the profiles of John as the Web client under the condition of a retention period of less than thirty days after the profiles are first collected.
- $\texttt{policy}(\texttt{pp4} - \texttt{John})$:
  If the distributor IEEE $\texttt{Journal}$ has the purpose of enforcing DRM control by collecting, retaining, and disclosing data on John as the Web client, then IEEE is endowed with the privacy usage rights $\{\texttt{collect}, \texttt{retain}\}$ from the digital traces of John as the Web client, where the data retention period is less than fourteen days after the trace data are first collected.

IV. SemPIF for Multiple Web Policies

Agents in the facilitator provide policy interchange to avoid possibly inconsistent or ambiguous syntax and semantics between source and target policies.

A. Meta-PIF

We envision several important issues in the design of agents while using SemPIF as a mediation architecture to enforce policy management services, such as policy sequencing, adding, deleting, merging, etc.

- In the SemPIF architecture, agents use PIF to provide basic interchange services of various policy languages.
- The basic vocabularies of PIF for interchange of policies are specified in the various REL policy languages, such as P3P or ODRL. They are $\texttt{principal}, \texttt{subject}(\texttt{owner}), \texttt{object}(\texttt{user})$, $\texttt{resource}(\texttt{asset})$, $\texttt{right}, \texttt{obligation}, \texttt{purpose}$, and $\texttt{condition}$. In addition, access right vocabularies for privacy protection and DRM are different. We have $\{\texttt{collect}, \texttt{retain}, \texttt{disclose}\}$, etc. for privacy protection and we have $\{\texttt{download}, \texttt{view}, \texttt{print}\}$, etc. for DRM.
- Most of the basic vocabularies for meta-PIF are the same as PIF's except some of them are directly related to

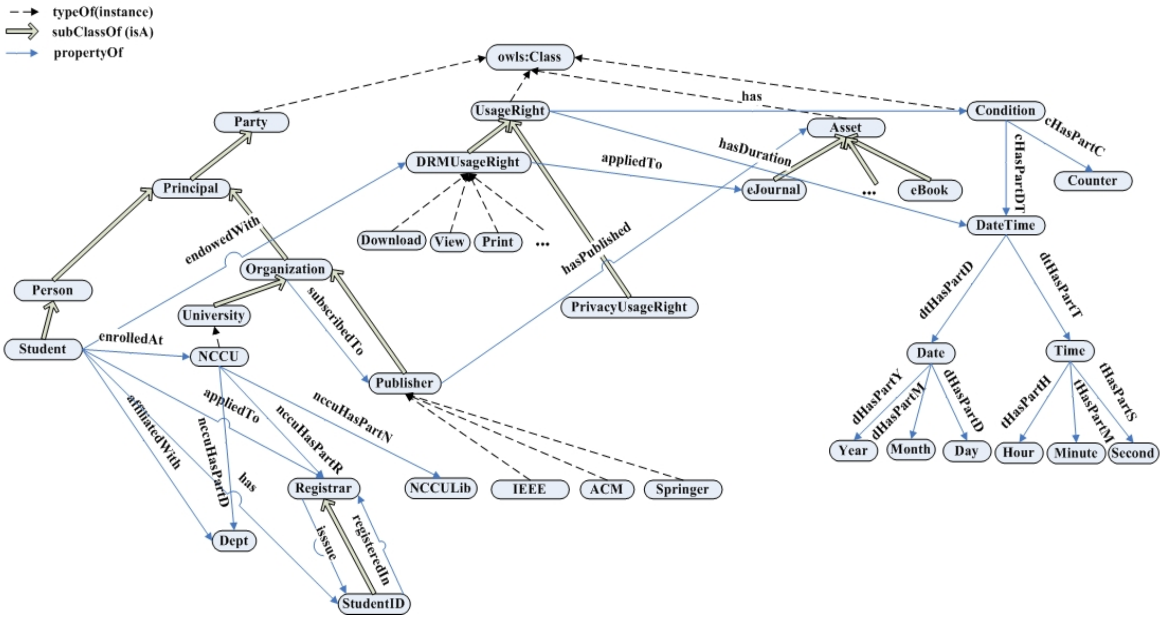

Fig. 5. A PIF-based ontology for DRM policies

PIF-based policies. The policy itself is introduced as a resource with respective users, rights, and conditions, etc. for agents to enforce its policy management services.

- Meta-PIF is a meta-policy language for PIF and only provides management services for PIF-based policies. If meta-PIF attempts to provide an interchange format for different meta-policies in PIF, then we also have to provide policy management interoperability services for different policy languages. This requires further study.

B. Agents in the facilitator enable meta-PIF policies

We use P3P basic vocabularies to specify data owner (or subject), user (or object), type, right, obligation as ontology classes with associated properties to formalize the semantics of privacy protection [10]. Furthermore, we also use Datalog-based rules to decide whether a Web server is allowed to collect, retain, or disclose a particular client's profiles and digital traces. Policies are defined as a combination of $\mathcal{O} + \mathcal{R}$. In order to unify policies from a client and a server, we allow agents in the facilitator to collect client and server's ontologies, and to enable SemPIF policy transformation and management services shown as follows:

1) **Ontologies mapping and aligning**
We map and align vocabularies from domain dependent ontologies of DRM and privacy protection policies. In Section III scenario, the vocabularies of class "Student, Publisher" in policy(drm1 − IEEE) and policy(pp1 − IEEE) correspond to "WebClient, WebServer" vocabularies of class in policy(pp3 − John) to policy(pp4 − John). Furthermore, we align the ontology schemas constructed with the vocabularies of class and property.

2) **Semantics mediation and unification**
We mediate and unify the semantic differences of vocabularies and schemas in the ontologies belonging to different protection policies. For example, a condition of Retain[hasDuration → =2Month] in the policy(pp1 − IEEE) corresponds to a condition of "retention period less than fourteen days" in the policy(pp4 − John).

3) **Conflicts resolving**
Agents initiate the reconciliation processes between conflicting policies using the meta-PIF framework. In this example, IEEE declares its intention to collect, retain, and disclose a Web user's data in the policy(pp1 − IEEE) for two months. The data include a Web user's profiles and digital traces. Web user John does not allow an IEEE Web server to disclose his personal profile to the other partners. Thus, policies between policy(pp1 − IEEE) and policy(pp3 − John), policy(pp4 − John) are inconsistent. Agents enable a policy priority-setting with meta-PIF-based policies to avoid the policy conflicts. In this example, an agent gives a higher priority to a client's policy(pp3 − John) and policy(pp5 − John) than to a server's policy(pp1 − IEEE). The defeasible logic of a meta-PIF's expression, Overrides(policy(?pid1), policy(?pid2)), for resolving conflicts of policies is a possible solution, where policy(?pid1) is bound to policy(pp3 − John) and policy(pp4 − John); policy(?pid2) is bound to policy(pp1 − IEEE). This negotiation protocol requires further study.

V. Related Work

REL is a subset of the PIF layers. FOL-semantics-enabled policy models for RELs have been proposed to specify the semantics of ODRL, XrML, and P3P [23] [24] [25]. However, it is still unclear how to design semantics-enabled policy languages from the FOL-enabled RELs to allow policies to be machine readable and understandable on the Web.

Tonti et. al. compared the three policy languages KAoS, Rei, and Ponder w.r.t. the representation and reasoning of specific policies [4]. The policy semantics of KAoS and Rei came from DL-based ontology. Rei has a policy management services framework for agents to manage policies but agents still cannot interoperate and cooperate with other agents across different frameworks. Moreover, policies created from LP-based policy languages, such as EPAL [15], were not able to interoperate and cooperate with the DL-based policies. We need a *de facto* standard policy interchange language as attempted by the W3C PLING[4] and with OMG's SBVR[5] to achieve policy interoperability.

The idea of meta-policies was proposed almost two decades ago [7]. It was used for policy management services in the Rei and Protune frameworks [5] [3]. In the Rei framework, the authors tried to propose a policy interchange mechanism instead of using a single policy language for describing all policies. Thus, SemPIF can be seen as bringing the objective of the Rei framework is close to the Semantic Web. In the Protune framework, meta-policies provide a simple means to specify which parts of a policy are sensitive, and how application-specific atomic conditions are to be verbalized in the documentation. However, predating SemPIF, the Rei and Protune frameworks did not show yet how a semantics-enabled policy layered architecture can be compatible with the current Semantic Web architecture.

VI. Conclusion and Future Work

We propose a semantics-enabled policy architecture, SemPIF, which extends W3C's Semantic Web layered architecture. We have introduced the SemPIF architecture as a 4-layer framework, i.e., UNL, PIF, PPD, and DSA. A meta-PIF layer is also introduced, side by side with the corresponding PIF, allowing agents in the facilitator to provide uniform services of interchange, reconciliation, and combination of policies from various domains on the Web. A scenario of employing SemPIF for DRM and privacy protection policies on digital library subscription services is described to demonstrate the feasibility of the SemPIF architecture. Future work include refining the PIF and meta-PIF languages to enable a multiple Web policies system on the Web.

Acknowledgements

This research was partially supported by the NSC Taiwan under Grant No. NSC 95-2221-E004-001-MY3, NSC 98-2918-E-004-003, and NSC 98-2221-E-004-009.

References

[1] S. Bechhofer et al., "OWL web ontology language reference", Tech. Rep., W3C, Feb. 2004.

[2] H. Boley et al., "Rule interchange on the web", in *Reasoning Web 2007, Third International Summer School*, Dresden, Germany, Sep. 2007, LNCS 4636, Springer.

[3] L. Kagal et al., "Using semantic web technologies for policy management on the web", in *21st National Conference on Artificial Intelligence (AAAI)*. July 2006, AAAI.

[4] G. Tonti et al., "Semantic web languages for policy representation and reasoning: A comparison of KAoS, Rei, and Ponder", in *2nd International Semantic Web Conference (ISWC) 2003*, 2003, LNCS 2870, pp. 419–437.

[5] P. Bonatti and D. Olmedilla, "Policy language specification, enforcement, and integration. project deliverable D2, working group I2", Tech. Rep., REWERSE, 2005.

[6] F. P. Patel-Schneider and I. Horrocks, "A comparison of two modelling paradigms in the semantic web", *Journal of Web Semantics*, pp. 240–250, 2007.

[7] H. Hilary Hosmer, "Metapolicies I", *ACM SIGSAC Review*, vol. 10, no. 2-3, pp. 18–43, 1992.

[8] Jos de Bruijn, "RIF RDF and OWL compatibility", Tech. Rep., W3C, Oct. 2009, http://www.w3.org/TR/rif-rdf-owl/.

[9] Y. J. Hu, "Semantic-driven enforcement of rights delegation policies via the combination of rules and ontologies", in *Workshop on Privacy Enforcement and Accountability with Semantics in conjunction with ISWC+ASWC'07*, 2007.

[10] Y. J. Hu, H. Y. Guo, and G. D. Lin, "Semantic enforcement of privacy protection policies via the combination of ontologies and rules", in *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC 2008)*, Taichung, Taiwan, June 2008.

[11] I. Horrocks et al., "SWRL: A semantic web rule language combining OWL and RuleML", 2004.

[12] N. B. Grosof et al., "Description logic programs: Combining logic programs with description logic", in *World Wide Web 2003*, Budapest, Hungary, 2003, pp. 48–65.

[13] M. F. Donini et al., "*AL*-log: Integrating datalog and description logics", *Journal of Intelligent Information Systems*, vol. 10, no. 3, pp. 227–252, 1998.

[14] B. Motik, U. Sattler, and R. Studer, "Query answering for OWL-DL with rules", in *3rd International Semantic Web Conference (ISWC) 2004*. 2004, LNCS 3298, pp. 549–563, Springer.

[15] G. Karjoth and M. Schunter, "A privacy policy model for enterprises", in *15th IEEE Computer Security Foundations Workshop (CSFW)*. June 2002, IEEE.

[16] E. Rissanen, "eXtensible Access Control Markup Language (XACML) ver. 3.0", May 2007.

[17] A. H. Anderson, "A comparison of two privacy policy languages: EPAL and XACML", in *Proceedings of the 3rd ACM Workshop on Secure Web Services (SWS'06)*. 2006, pp. 53–60, ACM.

[18] L. Cranor et al., "The platform for privacy preferences (P3P) 1.0 (P3P 1.0) specification", 2002, http://www.w3.org/P3P/.

[19] Inc. ContentGuard, "XrML: The digital rights language for trusted content and services", Tech. Rep., ContentGuard Inc., 2002.

[20] S. Guth and R. Iannella, "Open Digital Rights Language (ODRL) version 2", ODRL initiative working draft, The ODRL Initiative, February 2005.

[21] E. J. Cohen, "DRM and privacy", *Commun. ACM*, vol. 46, no. 4, pp. 47–49, 2003.

[22] D. J. Weitzner et al., "Creating a policy-aware web: Discretionary, rule-based access for the world wide web", in *Web and Information Security*, E. Ferrari and B. Thuraisingham, Eds., pp. 1–31. Idea Group Inc., 2006.

[23] Y. J.and Vicky Weissman Halpern, "A formal foundation for XrML", *Journal of the ACM*, vol. 55, no. 1, pp. 1–42, 2008.

[24] N. Li, T. Yu, and A. I. Antón, "A semantics-approach to privacy languages", *Computer Systems and Engineering (CSSE)*, vol. 21, no. 5, Sep. 2006.

[25] R. Pucella and V. Weissman, "A formal foundation for ODRL", arXiv:cs/0601085v1, Cornell University, January 2006, http://arxiv.org/abs/cs/0601085.

[4] Policy Interchanges Interest Group - PLING, http://www.w3.org/Policy/pling.

[5] Semantics of Business Vocabulary and Business Rules (SBVR), http://www.omg.org/spec/SBVR/1.0/.

國科會補助計畫衍生研發成果推廣資料表

日期:2010/12/08

國科會補助計畫	計畫名稱：給隱私權保護與著作權管理使用的語意式web規範互通語言
	計畫主持人：胡毓忠
	計畫編號：98-2221-E-004-009- 學門領域：WEB 技術

無研發成果推廣資料

98 年度專題研究計畫研究成果彙整表

計畫主持人：胡毓忠	計畫編號：98-2221-E-004-009-
計畫名稱：給隱私權保護與著作權管理使用的語意式 web 規範互通語言	

成果項目			量化			單位	備註（質化說明：如數個計畫共同成果、成果列為該期刊之封面故事...等）
			實際已達成數（被接受或已發表）	預期總達成數(含實際已達成數)	本計畫實際貢獻百分比		
國內	論文著作	期刊論文	0	0	100%	篇	
		研究報告/技術報告	0	0	100%		
		研討會論文	0	0	100%		
		專書	0	0	100%		
	專利	申請中件數	0	0	100%	件	
		已獲得件數	0	0	100%		
	技術移轉	件數	0	0	100%	件	
		權利金	0	0	100%	千元	
	參與計畫人力（本國籍）	碩士生	0	0	100%	人次	
		博士生	0	0	100%		
		博士後研究員	0	0	100%		
		專任助理	0	0	100%		
國外	論文著作	期刊論文	0	0	100%	篇	
		研究報告/技術報告	0	0	100%		
		研討會論文	2	0	100%		參與 RuleML-2009(擔任 PC member)以及RuleML-Challenge-2009(擔任Co-Chairs)並且發表論文一篇於 RuleML-2009 並且擔任 RuleML-Challenge-2009 的論文編輯。 擔任 Web Intelligence (WI)-2010 的 PC member，並且發表論文一篇並且負責主持一個論文發表會的session。
		專書	1	0	100%	章/本	發表專書篇章於 Introduction to the Semantic Web: Concepts, Technologies and Applications, iConcept, 2010(以經被接受)
	專利	申請中件數	0	0	100%	件	
		已獲得件數	0	0	100%		
	技術移轉	件數	0	0	100%	件	
		權利金	0	0	100%	千元	

	參與計畫人力（外國籍）	碩士生	4	0	100%	人次	郭弘毅、吳建輝、楊竣展、楊協達
		博士生	0	0	100%		
		博士後研究員	0	0	100%		
		專任助理	0	0	100%		
其他成果(無法以量化表達之成果如辦理學術活動、獲得獎項、重要國際合作、研究成果國際影響力及其他協助產業技術發展之具體效益事項等，請以文字敘述填列。)		辦理 RuleML-Challenge-2009 的國際研討會競賽，並且出版論文集於 CEUR,請參考： http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-549/					

🟥	成果項目	量化	名稱或內容性質簡述
科教處計畫加填項目	測驗工具(含質性與量性)	0	
	課程/模組	0	
	電腦及網路系統或工具	0	
	教材	0	
	舉辦之活動/競賽	0	
	研討會/工作坊	0	
	電子報、網站	0	
	計畫成果推廣之參與（閱聽）人數	0	

國科會補助專題研究計畫成果報告自評表

請就研究內容與原計畫相符程度、達成預期目標情況、研究成果之學術或應用價值（簡要敘述成果所代表之意義、價值、影響或進一步發展之可能性）、是否適合在學術期刊發表或申請專利、主要發現或其他有關價值等，作一綜合評估。

1. 請就研究內容與原計畫相符程度、達成預期目標情況作一綜合評估

■達成目標

□未達成目標（請說明，以 100 字為限）

□實驗失敗

□因故實驗中斷

□其他原因

說明：

2. 研究成果在學術期刊發表或申請專利等情形：

論文：■已發表 □未發表之文稿 □撰寫中 □無

專利：□已獲得 □申請中 ■無

技轉：□已技轉 □洽談中 ■無

其他：（以 100 字為限）

3. 請依學術成就、技術創新、社會影響等方面，評估研究成果之學術或應用價值（簡要敘述成果所代表之意義、價值、影響或進一步發展之可能性）（以 500 字為限）

解決 Web(全球資訊網) 資料流通、分享、與保護的問題。並且能夠透過語意網的本體論和規則兩大知識系統表達與具體落實執行能力讓電腦軟體可以正確解讀出具有語意式的 Web 規範。利用標準的 Web 語言如 OWL, RIF 等來表示隱私權保護與著作權保護的規範可以讓規範所需要表達語意可以更明確以避免混淆不清的窘境。除了可以提升原有 XML 為基礎的權力表達語言如 ODRL, XrML, P3P, XACML 的缺失之外也可以讓現有的規範語言如 Rein, KaOS 可以透過這個 SemPIF 互通語言的架構來達成互通與規範整合的目的。語意式電腦規範的價值在於找出人類法治規範可以落實到Web環境中自動化執行的概念與規則讓電腦系統可以有效且自動化的來解讀和執行以避免完全用人類手動式處理規範的困境。除此之外可以達成資訊有效分享與保護的目的，系統可以在資料分享與保護時在事先與進行中的程序來加以檢驗，避免事後的冗長訴訟程序與資料不當揭露與使用所產生的副作用。本研究案的進一步發展可以將參照個人資料保護法與著作權保護法等法治規範將相對應的語意式電腦規範運用到雲端運算的環境中，來解決雲端環境資料分享與保護的目的，讓雲端中資料的個人資料保護與數位內容的著作權保護可以透過此電腦規範的表示與執行得到適當的解決。