Semantic-Driven Enforcement of Rights Delegation Policies via the Combination of Rules and Ontologies

Prof.(Dr.) Yuh-Jong Hu

December-13-2007

hu@cs.nccu.edu.tw
Emerging Network Technology(ENT) Lab.
Department of Computer Science
National Chengchi University, Taipei, Taiwan

Outline

- *Introduction*
- *Research Goal*
- *Our Approach*
- *Related Work*
- *License Agreement for Usage (or Transfer) Rights*
  - ✔ *Usage Rights Delegation*
  - ✔ *A Rights Delegation Ontology*
- *Transfer Rights Delegation*
  - ✔ *Prerequisites Expressions*
  - ✔ *Rights Transfer Delegation Rules*
  - ✔ *A Usage Rights Delegation Scenario*
- *Discussion*
- *Conclusion*

Introduction

- *XML-based digital right expression (or preference) languages, such as ODRL, XrML, P3P, lack machine understandable formal semantics of license agreements for automatic agent processing.*

- *Generic First Order Logic (FOL) provides formal semantics for the above underlying XML-based standards but it is machine unfriendly.*

- *Several Ontologies+Rules combinations provide semantic-driven enforcement of access rights control and delegation policies for the permissible service agreements but which is the right one?*

Research Goal

- *To resolve the problem of license agreements written in XML-based ODRL rights expression language that lacks of formal semantics.*

- *To construct an abstract formal semantic layer overlaid on ODRL for license agreement semantics instead of using semantic ambiguity natural language, such as English.*

- *To explore the possible semantic-driven enforcement of digital rights management (DRM) access control and delegation policies via one of the ontologies+rules combinations, i.e., SWRL.*

- *To generalize our results to other digital access rights control and delegation domains , such as privacy protection.*

Our Approach

- *Exploiting XML-based ODRL specifications, including expression language, data dictionary elements, and XML syntax.*

- *Designing rights expression and delegation ontology overlaid on ODRL specifications.*

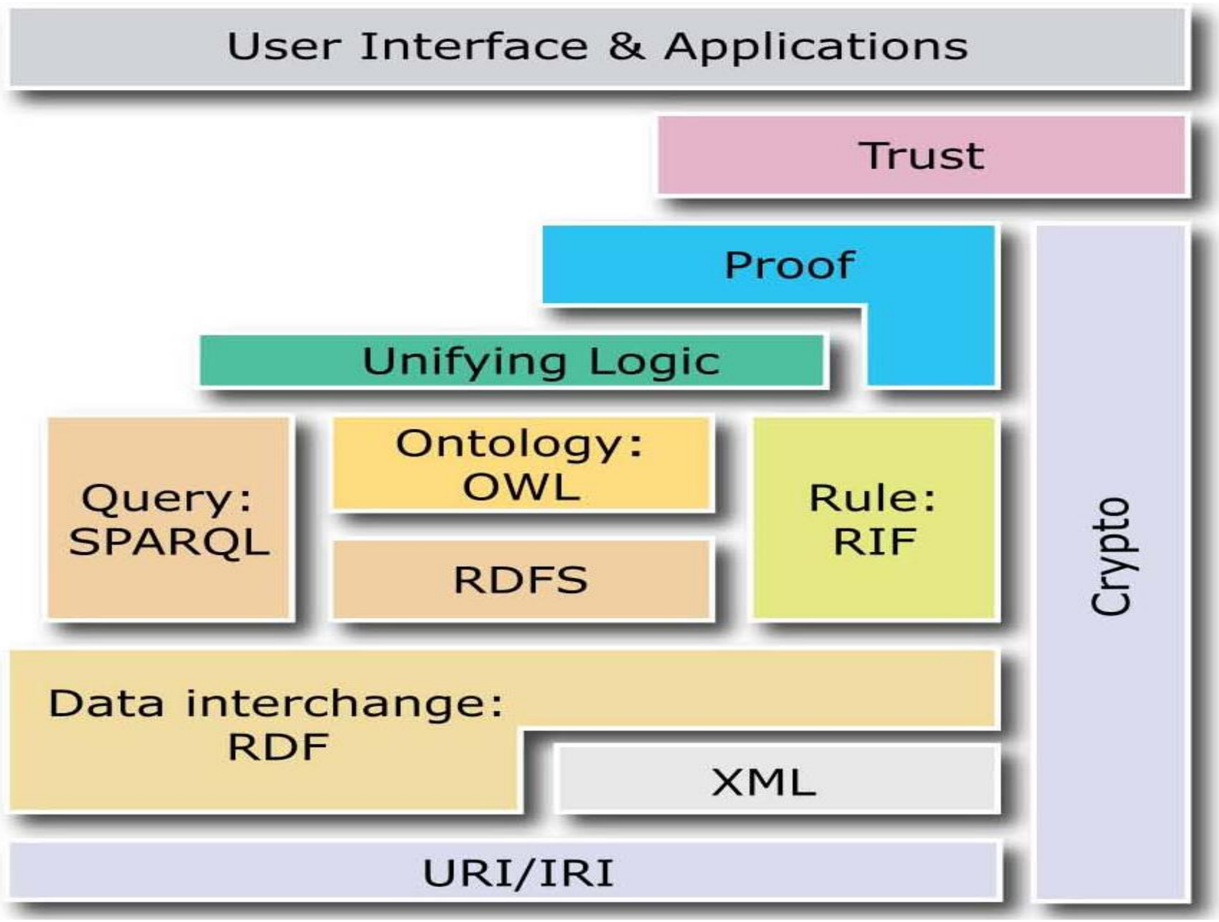
- *Proposing usage rights and transfer rights delegation policies as SWRL rules.*

- *How about the other hybrid integration approaches, such as AL-log, CARIN, hybrid $MKNF$, etc instead of SWRL?*

Related Work

- ☛ *A Formal Foundation for ODRL [Pucella04]* $\Longleftarrow$ *pure FOL semantics*
- ☛ *A Formal Semantics for P3P [Yu04]* $\Longleftarrow$ *data-centric relational semantics*
- ☛ *Flexible Authorization Framework (FAF)[Jajodia01]* $\Longleftarrow$ *LP semantics*
- ☛ *E-P3P and its successor EPAL [Ashley03]* $\Longleftarrow$ *FAF semantics*
- ☛ *Rei, KAoS[Tonti03]* $\Longleftarrow$ *DL-based FOL semantics*
- ☛ *XACML[OASIS]* $\Longleftarrow$ *XML so no semantics*

Semantic Web Well-Known Layer Cake (2007/03)



License Agreement for Usage (Transfer) Rights

- A license agreement indicates the policies (rules) under which a principal $Prin_o$ allows another principal $Prin_{u_i}$ to use an asset r presumably owned by $Prin_o$, where $Prin_o$ is an asset owner, $Prin_{u_i}$ is one of n asset users, where $i \in (1, \cdots, n)$.

- A license agreement refers as a policy set showing any number of prerequisites and policies. A prerequisite is either a constraint, a requirement, or a condition. If all of the prerequisites are met, then policies say that the agreement's users may perform the action for the license agreement's assets.

Usage Rights Delegation

- We define $hasUsageRights$ as an abstract property describing the generic usage rights for a principal x to use an asset r.
- The domain class of $hasUsageRights$ property is $Party$, and the range class is $Asset$.
- The domain class of $delegate$ property is $Prin_o$ and the range class is $Prin_u$, where the delegate does have $subPropertyOf\ (delegate_g, delegate_t, \cdots)$.
- The $delegate_g$ represents generic usage rights delegation property and the $delegate_t$ represents rights transfer delegation property.

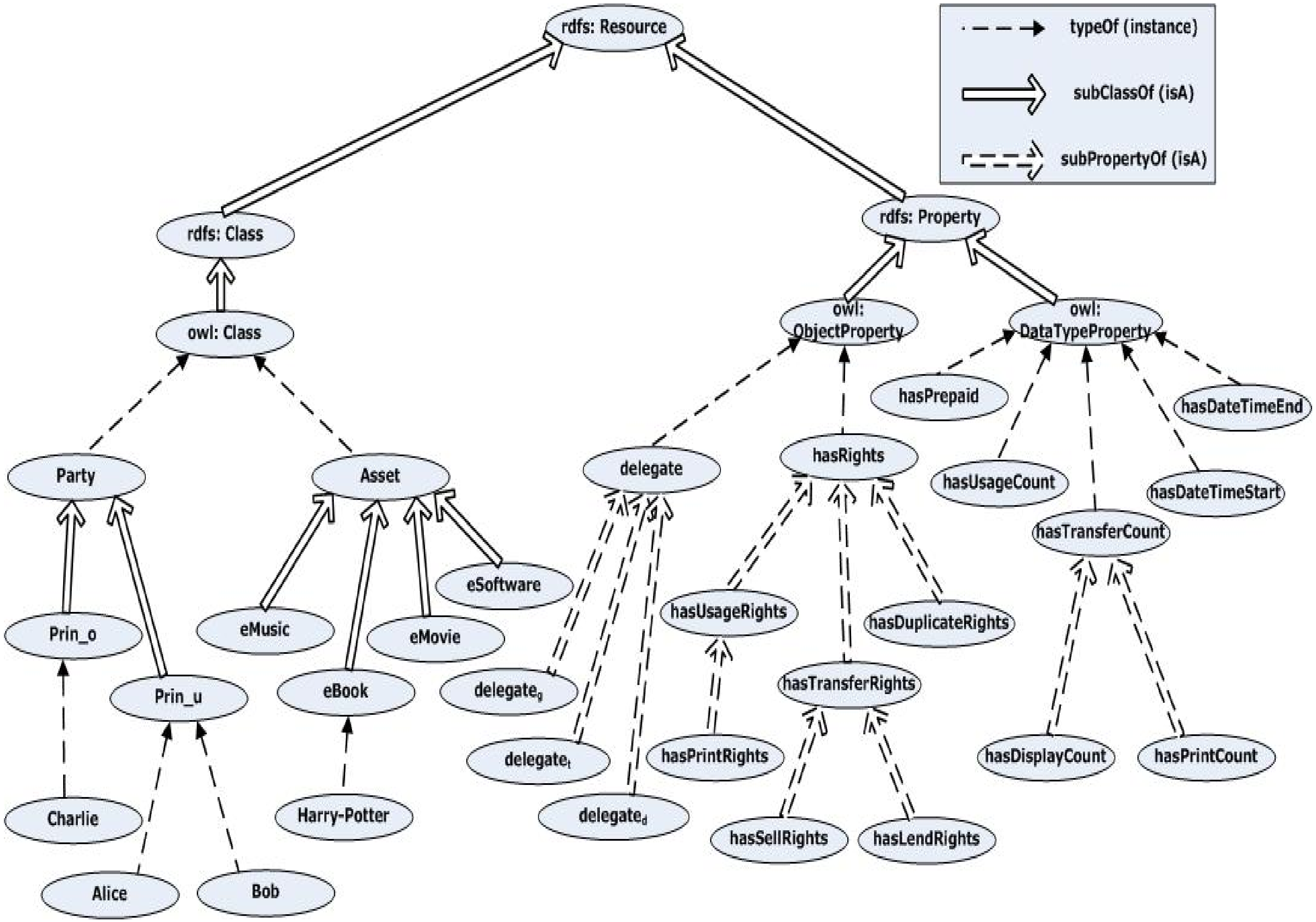
Usage Rights Delegation (conti.)

- *ODRL does not enforce or mandate any policies for DRM, but provides mechanisms to express such policies.*

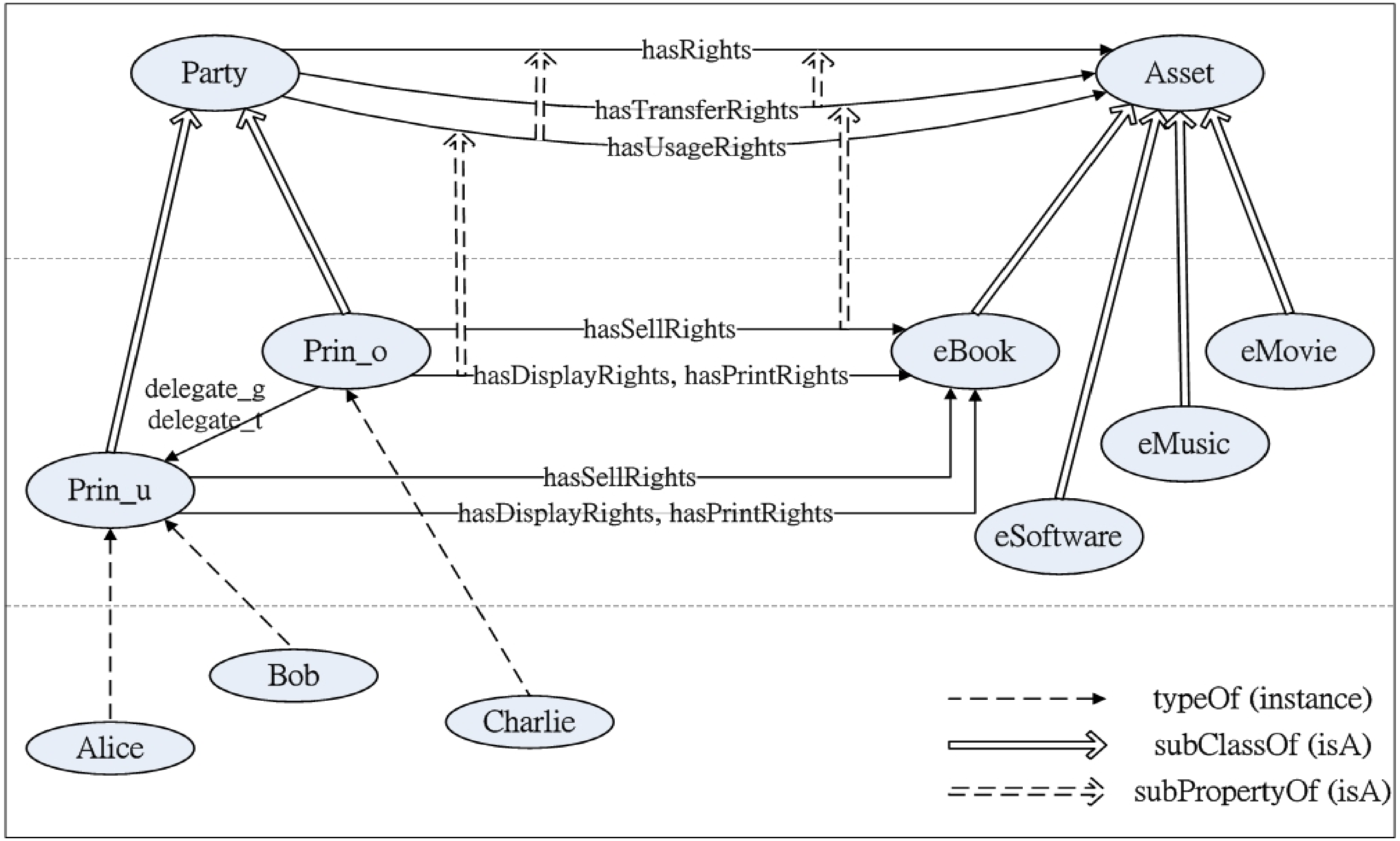
- *Using ODRL expression language and data dictionary elements as rights delegation ontology's entities.*

- *The class and property terms in this rights delegation ontology will be considered as antecedents or conclusion(s) in the usage and tranfer rights delegation policies (or rules) to enforce real rights delegation inference.*

A Rights Delegation Ontology



A Rights Delegation Snapshot



Transfer Rights Delegation

- The $hasTransferRights$ is an abstract property describing the transfer rights delegation of usage rights for a principal x for an asset r.

- The domain class of property $hasTransferRights$ is $Party$ and the range class is $Asset$.

- $Prin_o$ might use $delegate_g$ to transfer usage rights only to $Prin_{u_i}$, where $i \in (1, \cdots, n)$, but does not delegate his transfer rights to $Prin_{u_i}$, where transfer rights $\in (hasSell_tRights, \cdots)$.

- $Prin_o$ might use $delegate_t$ property, then any one of the transfer rights permissions $\in (hasSell_tRights, \cdots)$ and usage rights can be further propagated.

Prerequisites Expressions

- *MaxCardinality:* $\leq_{\exists u}\ hasUsageCount_{\exists p}.Asset$
- *MaxCardinality:* $\leq_{\exists t}\ hasTransferCount_{\exists p}.Asset$
- *Cardinality:* $=_{\exists a}\ hasPrepaid_{\exists p}.Party$
- *Validity of time interval* $\forall Time \in (t_1, t_2)$*:*
  $\geq_{\exists t_1}\ hasDateTime_{\exists p}.Time \wedge \exists \leq_{t_2}\ hasDateTime_{\exists p}.Time$

Rights Transfer Delegation Rules

- $hasUsageRights(?x, ?r) \wedge hasTransferRights(?x, ?r) \Longrightarrow hasUsageTransferRights(?x, ?r) \longleftarrow (o1)$

- $hasUsageTransferRights(?x, ?r) \wedge delegate_g(?x, ?y) \wedge hasPrepaid(?y, ?a) \wedge <_{\exists u} hasUsageCount(?r) \Longrightarrow hasUsageRights(?y, ?r) \longleftarrow (o2)$

- $hasUsageRights(?x, ?r) \wedge <_{\exists u} hasUsageCount(?r) \wedge \geq_{\exists t_1} hasDateTime(?t) \wedge \leq_{\exists t_2} hasDateTime(?t) \Longrightarrow Permitted(Usage, ?r) \longleftarrow (o3)$

- $hasUsageTransferRights(?x, ?r) \wedge delegate_t(?x, ?y) \wedge hasPrepaid(?y, ?a) \wedge \geq_1 hasTransferCount(?r) \Longrightarrow hasUsageTransferRights(?y, ?r) \longleftarrow (o4)$

A Usage Rights Delegation Scenario

- Natural Language for license agreement:

Content distributor $Charlie$ c makes an agreement with two content consumers, $Alice$ a and Bob b. After each paying five dollars, and then both receiving acknowledgement from $Charlie$, $Alice$ and Bob are given the usage rights and may each display an $eBook$ asset, Harry Potter and the Deathly Hallows, up to five times. They may each print it only once. However, the total number of actions, either displays or prints done by $Alice$ and Bob, may be at most ten. The usage rights validity period is between 2007/05/07/09:00 - 2007/05/10/24:00.

A Usage Rights Delegation Scenario

- *Abstract Syntax for license agreement:*

agreement

between Charlie and $\{$*Alice,Bob*$\}$

about Harry Potter and the Deathly Hallows

with inSequence[prePay[5.00],attribution[Charlie]]

$\Longrightarrow$ *not[and[Time* $<$ *2007/05/07/09:00,Time* $>$ *2007/05/10/24:00]]*

$\Longrightarrow$ *with count[10]* $\Longrightarrow$

and[forEachMember[Alice,Bob;count[5]] $\Longrightarrow$ *display,*

forEachMember[Alice,Bob;count[1]] $\Longrightarrow$ *print]*

A Usage Rights Delegation Scenario

- First Order Logic (FOL) for license agreement:

$$\forall x((x = Alice \vee x = Bob) \Longrightarrow$$
$$\exists t_1 \exists t_2 (t_1 < t_2 \wedge Paid(5, t_1) \wedge Attributed(Charlie, t_2))) \Longrightarrow$$
$$\forall t \wedge hasDateTime(t) \geq 2007/05/07/09:00 \wedge$$
$$hasDateTime(t) \leq 2007/05/10/24:00 \Longrightarrow$$
$$count(Alice, id_1) + count(Alice, id_2) + count(Bob, id_1)$$
$$+ count(Bob, id_2) < 10 \Longrightarrow$$
$$(count(Alice, id_1) < 5 \wedge count(Bob, id_1) < 5 \Longrightarrow \mathbf{Permitted}(x, display, ebook))$$
$$\wedge (count(Alice, id_2) < 1 \wedge count(Bob, id_2) < 1 \Longrightarrow \mathbf{Permitted}(x, print, ebook)))$$

A Usage Rights Delegation Scenario

- Ontologies+Rules(SWRL) for license agreement:

Ontology for content distributor Charlie's:

$hasDisplayRights \sqsubseteq hasUsageRights$
$hasPrintRights \sqsubseteq hasUsageRights$
$\leq (hasDisplayCount_{\{a,b\}}.eBook,\ hasUsageCount_c.eBook)$
$\leq (hasPrintCount_{\{a,b\}}.eBook,\ hasUsageCount_c.eBook)$
$\{Alice, Bob\} \overset{domain}{\longleftarrow} hasUsageRights \overset{range}{\longrightarrow} R_1,$
where $R_1 =\leq_{10} hasUsageCount_c$
$\wedge \geq_{2007/05/07/0900} hasDateTime_c.Time$
$\wedge \leq_{2007/05/10/2400} hasDateTime_c.Time$
$\exists =_{\alpha} \exists = sum(\exists \leq_5 hasDisplayCount_i.\{HarryPotter\}),\ i \in \{a, b\},$
where α: $\exists hasDisplayCount_c.\{HarryPotter\} \longleftarrow (c1)$
$\exists =_{\beta} \exists = sum(\exists \leq_1 hasPrintCount_i.\{HarryPotter\}),\ i \in \{a, b\},$
where β: $\exists hasPrintCount_c.\{HarryPotter\} \longleftarrow (c2)$
$\exists =_{\delta} sum(\alpha, \beta),$
where $\delta : \exists hasUsageCount_c\{HarryPotter\} \longleftarrow (c3)$

A Usage Rights Delegation Scenario

- Ontologies+Rules(SWRL) for license agreement:

Rules for content distributor Charlie's:

$$hasDisplayRights(?x, ?r) \wedge hasSell_d Rights(?x, ?r)$$
$$\Longrightarrow hasDisplaySell_d Rights(?x, ?r) \longleftarrow (c4)$$

$$hasPrintRights(?x, ?r) \wedge hasSell_d Rights(?x, ?r)$$
$$\Longrightarrow hasPrintSell_d Rights(?x, ?r) \longleftarrow (c5)$$

$$hasDisplaySell_d Rights(?x, ?r) \wedge delegate_g(?x, ?y)$$
$$\wedge\, hasPrepaid(?y, ?a) \wedge \Longrightarrow hasDisplayRights(?y, ?r) \longleftarrow (c6)$$

$$hasPrintSell_d Rights(?x, ?r) \wedge delegate_g(?x, ?y)$$
$$\wedge\, hasPrepaid(?y, ?a) \Longrightarrow hasPrintRights(?y, ?r) \longleftarrow (c7)$$

A Usage Rights Delegation Scenario

- Ontologies+Rules(SWRL) for license agreement:

  Facts for content distributor Charlie's:

  $eBook(HarryPotter)$

  $hasDisplayRights(Charlie, HarryPotter)$

  $hasPrintRights(Charlie, HarryPotter)$

  $hasSell_dRights(Charlie, HarryPotter)$

  $hasDisplaySell_dRights(Charlie, HarryPotter)$

  $hasPrintSell_dRights(Charlie, HarryPotter)$

  $\exists =_5 hasPrepaid(Alice)$

  $hasDisplayRights(Alice, HarryPotter) \longleftarrow (c8)$

  $hasPrintRights(Alice, HarryPotter) \longleftarrow (c9)$

  $\exists =_5 hasPrepaid(Bob)$

  $hasDisplayRights(Bob, HarryPotter) \longleftarrow (c10)$

  $hasPrintRights(Bob, HarryPotter) \longleftarrow (c11)$

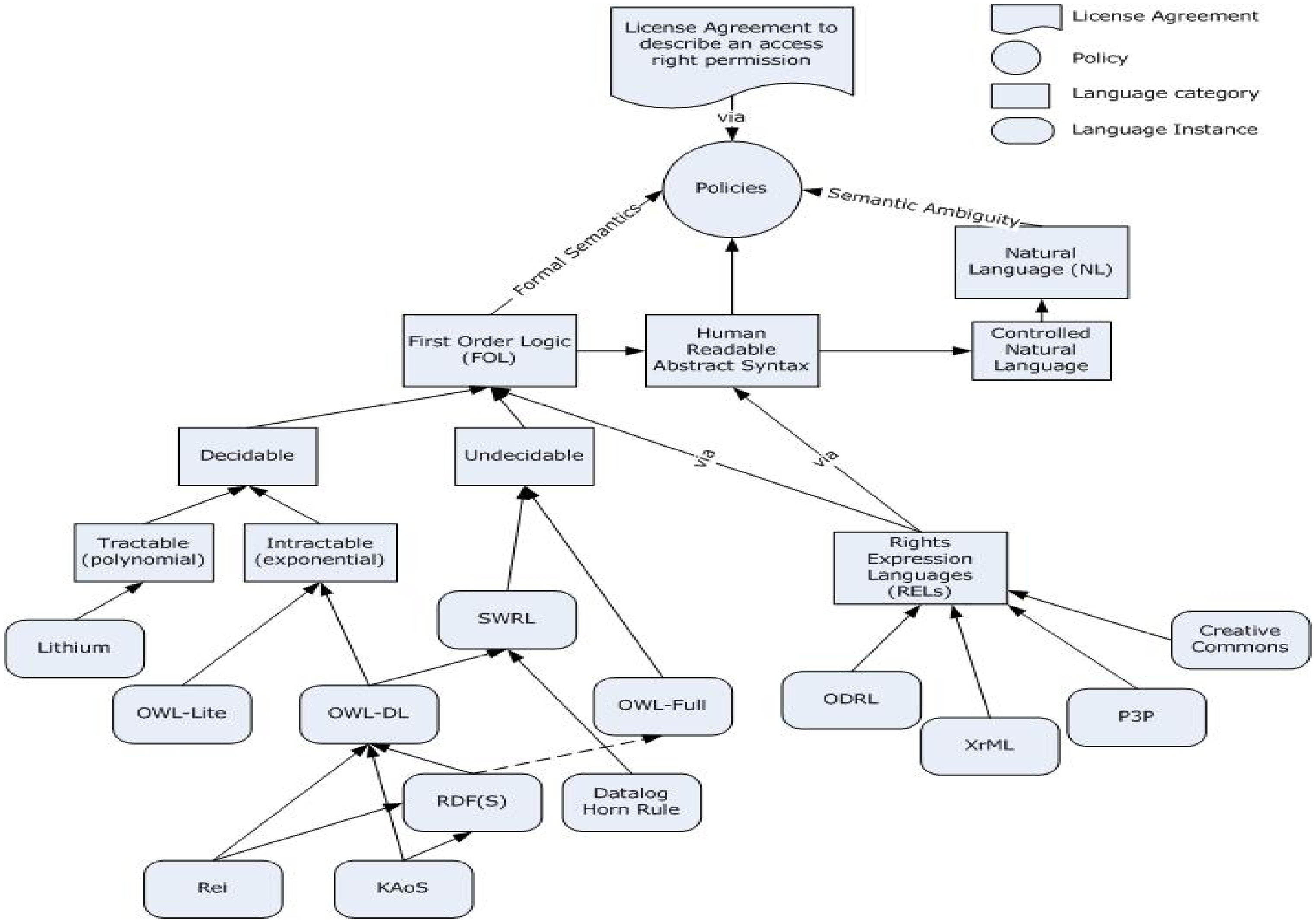
  $\cdots \cdots \cdots$

Discussion

☛ *The Pros and Cons of different license agreement expression languages:*

- ✔ *Natural Language: Pros human readable and understandable but Cons machine unfriendly, no formal semantics.*
- ✔ *Pure FOL: Pros formal and clear syntax and semantics but Cons machine unfriendly, possibly undecidable computation complexity, and policy writer (reader) needs to be a logician.*
- ✔ *Ontologies+Rules: Pros formal semantics for automatic machine processing and understanding but Cons limited expressing power, such as negation-free, function-free, and with limited number of parameter parities.*
- ✔ *Rights Expression Languages: Pros XML-based for machine processing but Cons no formal semantics.*

Policy Languages for Access Rights Permission



Conclusion

- *The semantic formal model for a license agreement is an ODRL-based rights delegation policy that can be enforced as a combination of ontologies and rules.*
- *A rights delegation ontology is proposed based on ODRL's expressions and data dictionary,*
- *The rights delegation policies are proposed as a set of rules for usage and transfer (or duplicate) rights delegations.*
- *A real usage rights delegation scenario is demonstrated to justify our formal semantic model.*