NICTA

Heuristics for Licenses Composition

Guido Governatori[1], Ho-Pun Lam[1], Antonino Rotolo[2], Serena Villata[3], Fabien Gandon[3]

[1] NICTA
[2] CIRSFID, and Law Faculty, University of Bologna
[3] INRIA Sophia Antipolis

JURIX 2013, Bologna

Australian Government
Department of Broadband, Communications and the Digital Economy
Australian Research Council

NICTA Funding and Supporting Members and Partners

License Composition Problem

The Tragedy of App A.U. Thor

License Composition Problem

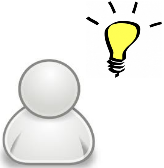
The Tragedy of App A.U. Thor

License Composition Problem

The Tragedy of App A.U. Thor

License Composition Problem

The Tragedy of App A.U. Thor

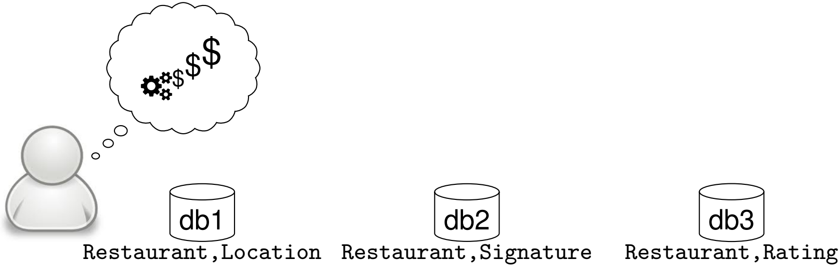


create a mobile app for Jurix 2013 delegates to find

1. best restaurant for tortellini
2. best restaurant for spaghetti bolognaise

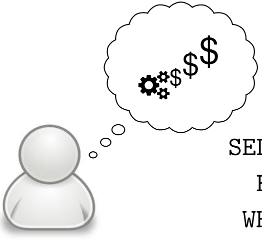
License Composition Problem

The Tragedy of App A.U. Thor



License Composition Problem

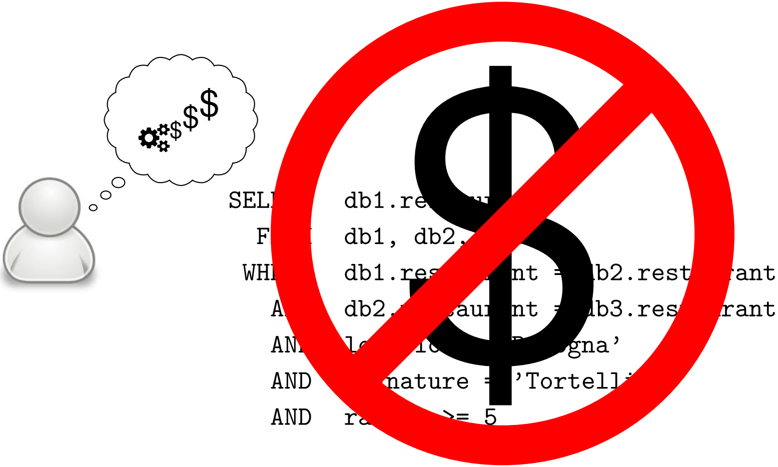
The Tragedy of App A.U. Thor



```
SELECT  db1.restaurant
  FROM  db1, db2, db3
 WHERE  db1.restaurant = db2.restaurant
   AND  db2.restaurant = db3.restaurant
   AND  location = 'Bologna'
   AND  signature = 'Tortellini'
   AND  rating >= 5
```

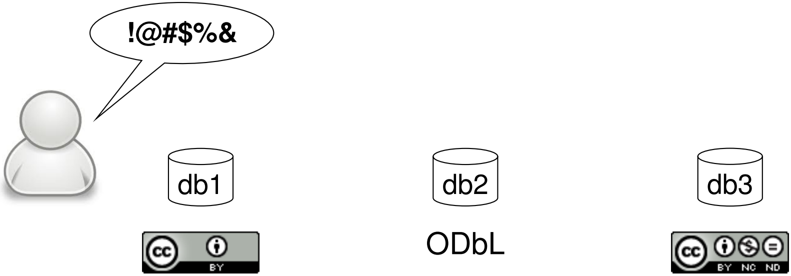
License Composition Problem

The Tragedy of App A.U. Thor



License Composition Problem

The Tragedy of App A.U. Thor



Research Question

NICTA

How to express the licensing terms associated to data coming from heterogeneous sources?

1. How to compose in a **compliant** way the licensing terms associated to a set of heterogeneous data to produce a single **composite license**?
2. How to determine what are the **legal effects** of the **composite license**?
3. How to produce in an **automated** way the composite license adopting different **composition heuristics**?

Framework features

1. Combination of **Semantic Web languages** (machine-readable licenses) – **defeasible deontic logic**,

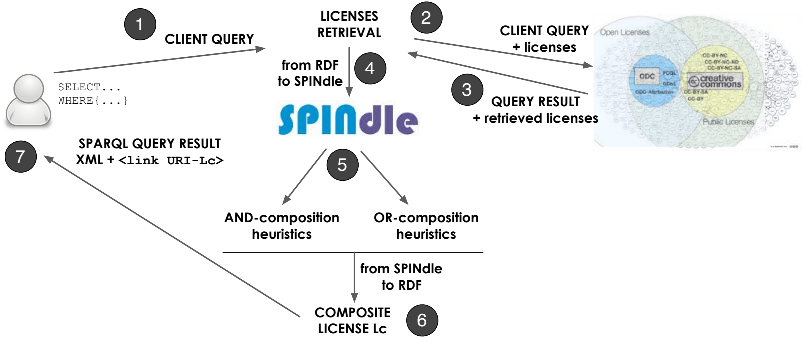
Framework features

1. Combination of **Semantic Web languages** (machine-readable licenses) – **defeasible deontic logic**,
2. Extension of existing proposals for licenses compatibility and composition in **service license analysis and CC licenses**,

Framework features

1. Combination of **Semantic Web languages** (machine-readable licenses) – **defeasible deontic logic**,
2. Extension of existing proposals for licenses compatibility and composition in **service license analysis and CC licenses**,
3. **Heuristics** for licenses combination.

Architecture



The Logic

1. describe ontology of concepts involved in LOD licenses,
2. capture the deontic component of those licenses.

The Rules

- **Ontology rules**:
  regular defeasible logic rules for deriving plain literals,

$$a_1, \dots, a_n \Rightarrow_c^{l_1} b$$

- **Logic of deontic rules**:
  constructive account of basic deontic modalities (obligation, prohibition, permission),

$$a_1, \dots, a_n, \mathsf{O}b_1, \dots, \mathsf{O}b_m, \mathsf{P}c_1, \dots, \mathsf{P}c_n \Rightarrow_{\mathsf{O}}^{l_2} p$$

The Rules

- **Ontology rules**:
  regular defeasible logic rules for deriving plain literals,

$$a_1, \ldots, a_n \Rightarrow_c^{l_1} b$$

- **Logic of deontic rules**:
  constructive account of basic deontic modalities (obligation, prohibition, permission),

$$a_1, \ldots, a_n, \mathrm{O}b_1, \ldots, \mathrm{O}b_m, \mathrm{P}c_1, \ldots, \mathrm{P}c_n \Rightarrow_{\mathrm{O}}^{l_2} p$$

if a_i are the case, b_j are obligatory (forbidden) and c_k are permitted then Op holds in license l_2 $(+\partial^{\mathrm{O}^{l_2}} p)$

Composition heuristics

- **OR-composition:** if *at least one* of the licenses involved in the composition owns a clause, then also l_c owns it;

Composition heuristics

- **OR-composition:** if *at least one* of the licenses involved in the composition owns a clause, then also l_c owns it;
- **AND-composition:** if *all* the licenses involved in the composition own a clause, then also l_c owns it;

PANDORO

PANDORO

- **OR heuristic for obligations**
  Something is obligatory (forbidden) if there is at least one license that makes it obligatory (prohibits it)

PANDORO

- **OR heuristic for obligations**
  Something is obligatory (forbidden) if there is at least one license that makes it obligatory (prohibits it)
- **AND heuristic for permissions**
  Something is permitted if it is permitted by all licenses.

Proof Conditions for O

If $P(n+1) = +\partial^{\mathrm{O}^{l_c}} q$ then

(1) $+\Delta^{\mathrm{O}^{l_c}} q \in P(1..n)$ or

(2) $-\Delta^{\mathrm{O}^{l_c}} {\sim}q \in P(1..n)$ and $\exists l_i \in \mathrm{Lic}$ such that

(1) $\exists r \in R_{sd}^{\mathrm{O}^{l_c}}[q] : \forall a, Xb, \neg Yd \in A(r)$:
$+\partial^{c} a, +\partial^{X^{l_i}} b, -\partial^{Y^{l_c}} d \in P(1..n)$ and

(2) $\forall l_j \in \mathrm{Lic}, \forall s \in R^{\mathrm{O}^{l_j}}[{\sim}q]$, either

(1) $\exists a \in A(s)$ or $Xb \in A(s)$ or $\neg Y \in A(s)$:
$-\partial^{c} a \in P(1..n)$, or $-\partial^{X^{l_c}} b \in P(1..n)$, or $+\partial^{Y^{l_c}} d \in P(1..n)$; or

(2) $\exists l_k \in \mathrm{Lic}, \exists t \in R^{\mathrm{O}^{l_k}}[q]$: $\forall a, Xb, \neg Yd \in A(t)$,
$+\partial^{c} a, +\partial^{l_c} b, -\partial^{l_c} d \in P(1..n)$, and $t \succ s$.

Proof Conditions for O explained

We can prove Op if

- there is a license having an applicable rule for p and
- for all licenses all rules for $\sim p$ are either
  - the rule is discarded (i.e., not applicable)
  - the rule is defeated by a stronger rule for p

Permission

- Permissions as lack of the obligation to the contrary

Permission

- Permissions as lack of the obligation to the contrary
- Permissions as exceptions to obligations to the contrary

$$a_1, \ldots, a_n \rightsquigarrow^{l_i}_{O} p$$

Proof Conditions for P (strong permission)

We can prove Pp

- there is a license with an applicable defeater for p and

Proof Conditions for P (strong permission)

We can prove Pp

- there is a license with an applicable defeater for p and
- all rules for $\sim p$ in all licenses are either

Proof Conditions for P (strong permission)

We can prove Pp

- there is a license with an applicable defeater for p and
- all rules for $\sim p$ in all licenses are either
  - discarded (non applicable)

Proof Conditions for P (strong permission)

We can prove Pp

- there is a license with an applicable defeater for p and
- all rules for $\sim p$ in all licenses are either
  - discarded (non applicable)
  - defeated (weaker than an applicable rule for p)

Evaluation: SPINdle

SPINdle

http://spin.nicta.org.au/spindle/

SPINdle

- SPINdle handle natively a single license
- encode each license in SPINdle
- transformations to handle AND- and OR-heuristics

Encoding Licenses

NICTA

$$
\begin{aligned}
F &= \{\mathit{Open}\} \\
L &= \{l_{OGL}, l_{ODbL}, l_{BY\text{–}NC\text{–}ND}\} \\
R^{O^{l_{OGL}}} &= \{r_1 :\Rightarrow^{l_{OGL}}_{O} \mathit{Attribution}, \quad r_2 : \mathit{Open} \leadsto^{l_{OGL}}_{O} \mathit{Publishing}, \\
&\quad\; r_3 : \mathit{Open} \leadsto^{l_{OGL}}_{O} \mathit{Distribution}, \quad r_4 : \mathit{Open} \leadsto^{l_{OGL}}_{O} \mathit{Derivative}, \\
&\quad\; r_5 : \mathit{Open} \leadsto^{l_{OGL}}_{O} \mathit{Commercial}\} \\
R^{O^{l_{ODbL}}} &= \{r_6 :\Rightarrow^{l_{ODbL}}_{O} \mathit{ShareAlike}, \quad r_7 :\Rightarrow^{l_{ODbL}}_{O} \mathit{Attribution}, \\
&\quad\; r_8 :\leadsto^{l_{ODbL}}_{O} \mathit{Sharing}, \quad r_9 :\leadsto^{l_{ODbL}}_{O} \mathit{Derivative}\} \\
R^{O^{l_{BY\text{–}NC\text{–}ND}}} &= \{r_{10} :\Rightarrow^{l_{BY\text{–}NC\text{–}ND}}_{O} \mathit{Attribution}, \quad r_{11} :\Rightarrow^{l_{BY\text{–}NC\text{–}ND}}_{O} \sim\mathit{Commercial}, \\
&\quad\; r_{12} :\Rightarrow^{l_{BY\text{–}NC\text{–}ND}}_{O} \sim\mathit{Derivative}, \quad r_{13} :\leadsto^{l_{BY\text{–}NC\text{–}ND}}_{O} \mathit{Sharing}\} \\
\succ &= \{l_{ODbL} \succ l_{BY\text{–}NC\text{–}ND}\}
\end{aligned}
$$

SPINdle Representation

```
» Open
r1: =>[Oc]Attribution
r2: Open =>[-Oc] -Publishing
r3: Open =>[-Oc] -Distribution
r4: Open =>[-Oc] -Derivative
r5: Open =>[-Oc] -CommercialExpl

r6: =>[Oc] ShareAlike
r7: =>[Oc] Attribution
r8: =>[-Oc] -Share
r9: =>[-Oc] -Derivative

r10: =>[Oc] Attribution
r11: =>[Oc] -CommercialExpl
r12: =>[Oc] -Derivative
r13: =>[-Oc] -Share

r9 > r12
```

Transformations (OR)

$$tor(r) = \begin{cases} r : A(r) \hookrightarrow p & r \in R^c \\ r : A(r) \rightarrow_{O^c} p & r \in R_s^{O^{l_i}}, l_i \in \text{Lic} \\ r : A(r) \Rightarrow_{O^c} p & r \in R_d^{O^{l_i}}, l_i \in \text{Lic} \\ r : A(r) \Rightarrow_{-O^c} \sim p & r \in R_{dft}^{O^{l_i}}, l_i \in \text{Lic} \end{cases}$$

Transformations (AND)

NICTA

$$tando(r) = \{r_{ij} : A(r) \leadsto_{\mathrm{O}^i} C(r) | r \in R^{\mathrm{O}^{l_i}}\} \cup \{r : A(r) \Rightarrow_{-\mathrm{O}^i} \sim C(r) | r \in R^{\mathrm{O}^{l_i}}_{dft}\} \cup$$
$$\{r | r \in R^{O^{l_i}}_{sd}\} \cup \{o_q : \mathrm{O}^{l_1} q, \ldots, \mathrm{O}^{l_n} q \Rightarrow_{\mathrm{O}^c} q \mid l_i \in \mathrm{Lic}, \exists r \in R^{\mathrm{O}^{l_i}}, C(r) = q\}$$

$$tandsp(r) = \{p_q : \mathrm{P}^* q, \mathrm{P}^{l_1} q, \ldots, \mathrm{P}^{l_n} q \Rightarrow_{\mathrm{P}^c} q, f^i_q : \neg \mathrm{O}^{l_i} q \Rightarrow_{P^{l_i}} q$$
$$p^{i*}_q : -\mathrm{O}^{l_i} \sim q \Rightarrow_{P^*} q, \; p_q : -\mathrm{O}^{l_i} \sim q \Rightarrow_{P^{l_i}} q \mid l_i \in \mathrm{Lic}, \exists r \in R^{\mathrm{O}^{l_i}}, C(r) = q\}$$

$$tandwp(r) = \{p_q : \neg \mathrm{O}^{l_1} \sim q, \ldots, \neg \mathrm{O}^{l_n} \sim q \Rightarrow_{P^c} q \mid l_i \in \mathrm{Lic}, q \in \mathrm{Lit}\}$$

Experiment

- CC PDM
  - Permissions: Reproduction, Distribution, Derivative Works.
- OS OpenData
  - Permissions: Reproduction, Distribution, Derivative Works.
  - Obligations: Notice, Attribution.
- CC-BY-ND
  - Permissions: Reproduction, Distribution, Derivative Works.
  - Obligations: Notice, Attribution, Share Alike.
  - Prohibitions: Non Commercial.
- CC-BY-NC-SA
  - Permissions: : Reproduction, Distribution.
  - Obligations: Notice, Attribution.

Experiment

Heuristics	Compose (ms)	Reasonig	Total time (ms)
NO	6.6	47.8	54.4
OR-composition	11.6	59.6	71.2
AND-composition	25	210.8	235.8
P-AND/OR-O-composition	17.3	92.4	109.7
Both AND- and OR-composition	29	226.2	255.2

1. Data obtained by **inference** from one or several licensed datasets, i.e., queries going beyond basic `SELECT` queries, where **aggregations** are present, e.g., *average*, *sum*;
2. **Temporal terms** of the licenses.

Thanks for your attention!