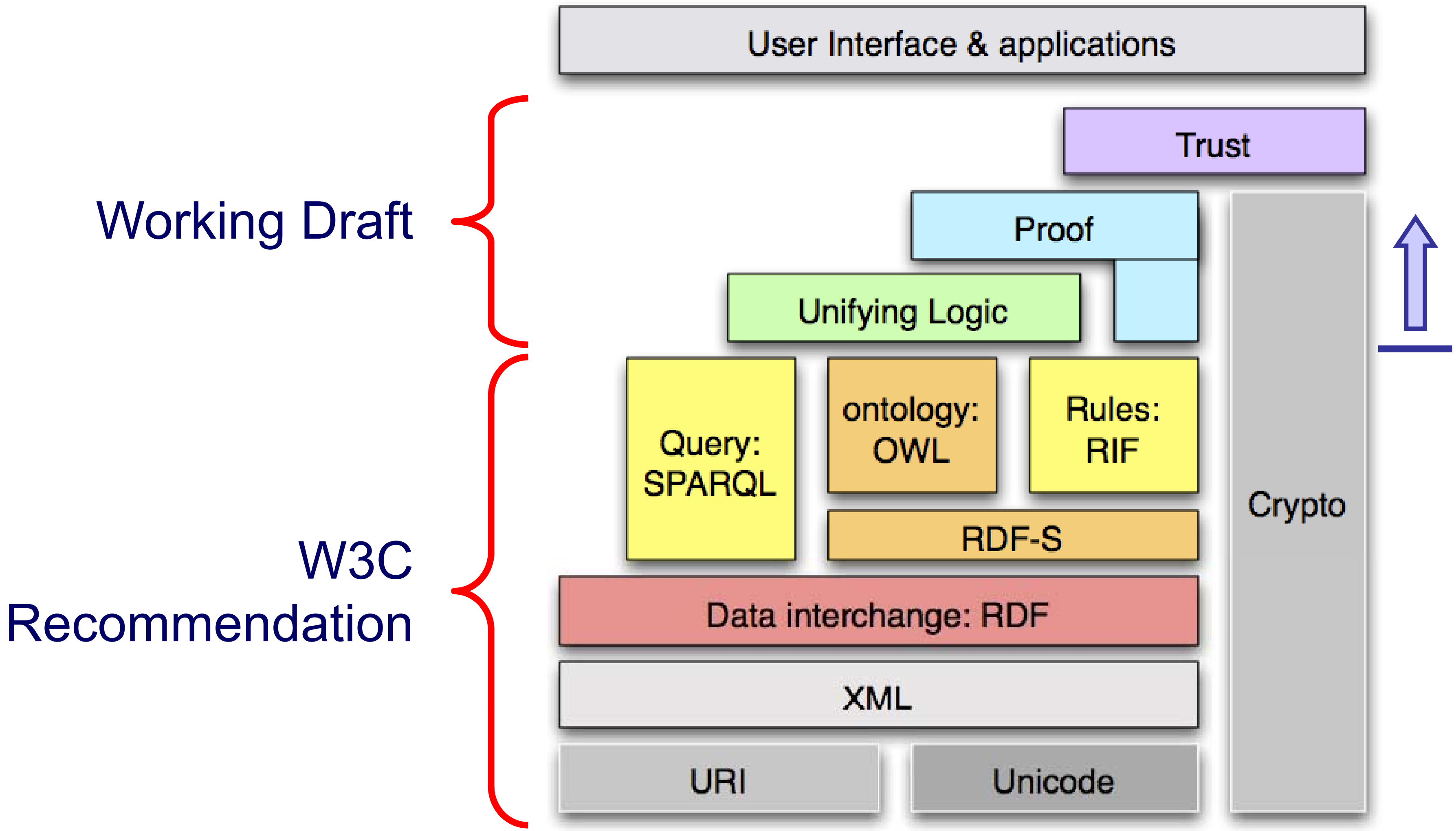
Intelligent Systems Principles and Programming

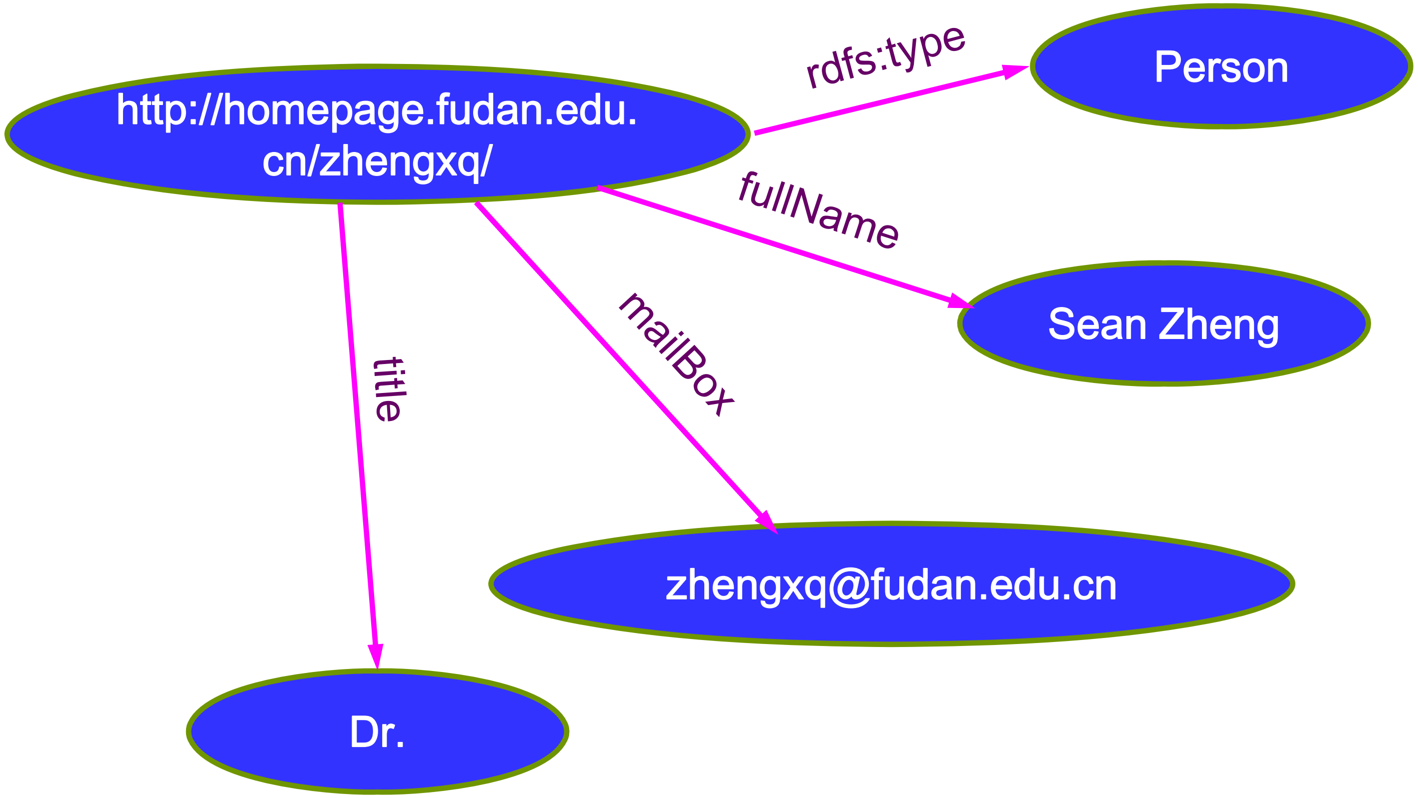
Xiaoqing Zheng

zhengxq@fudan.edu.cn

Semantic Web Stack

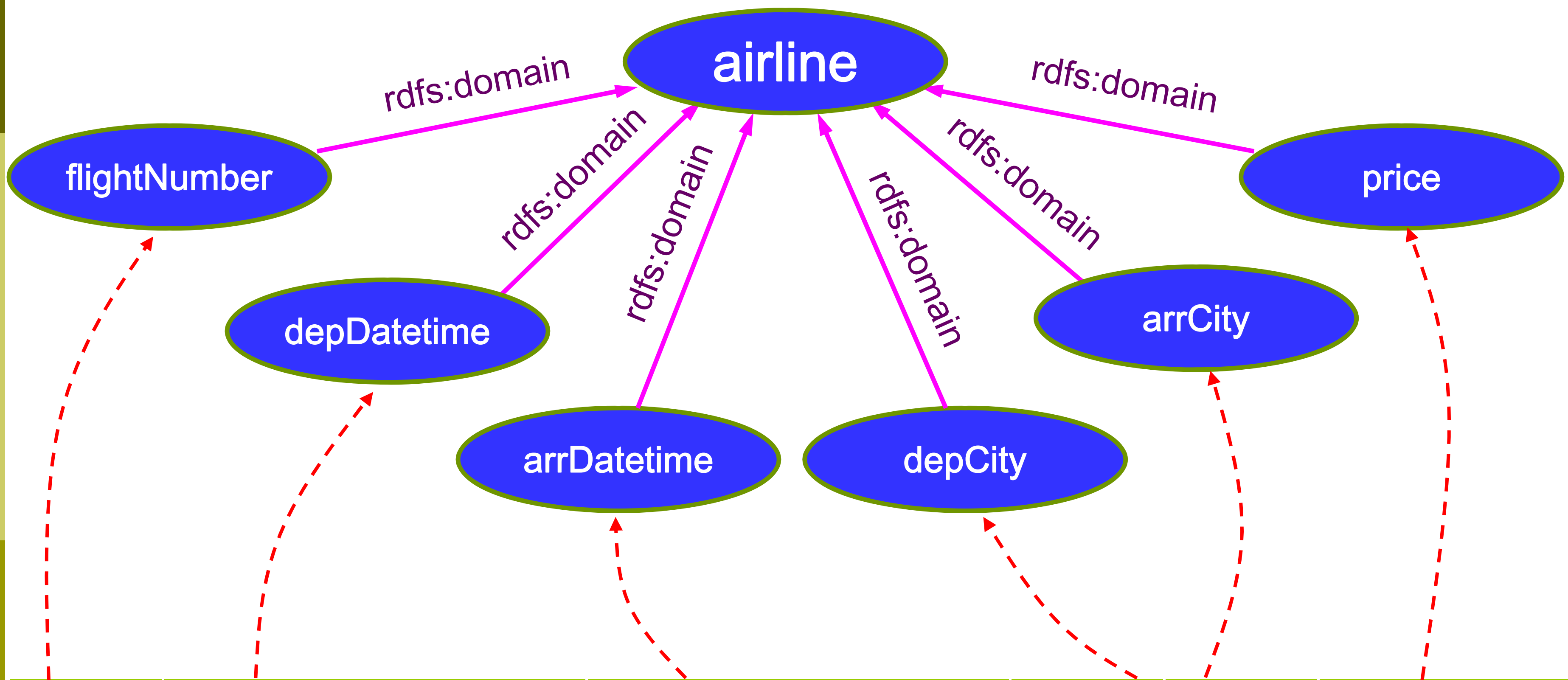


Resource Description Framework



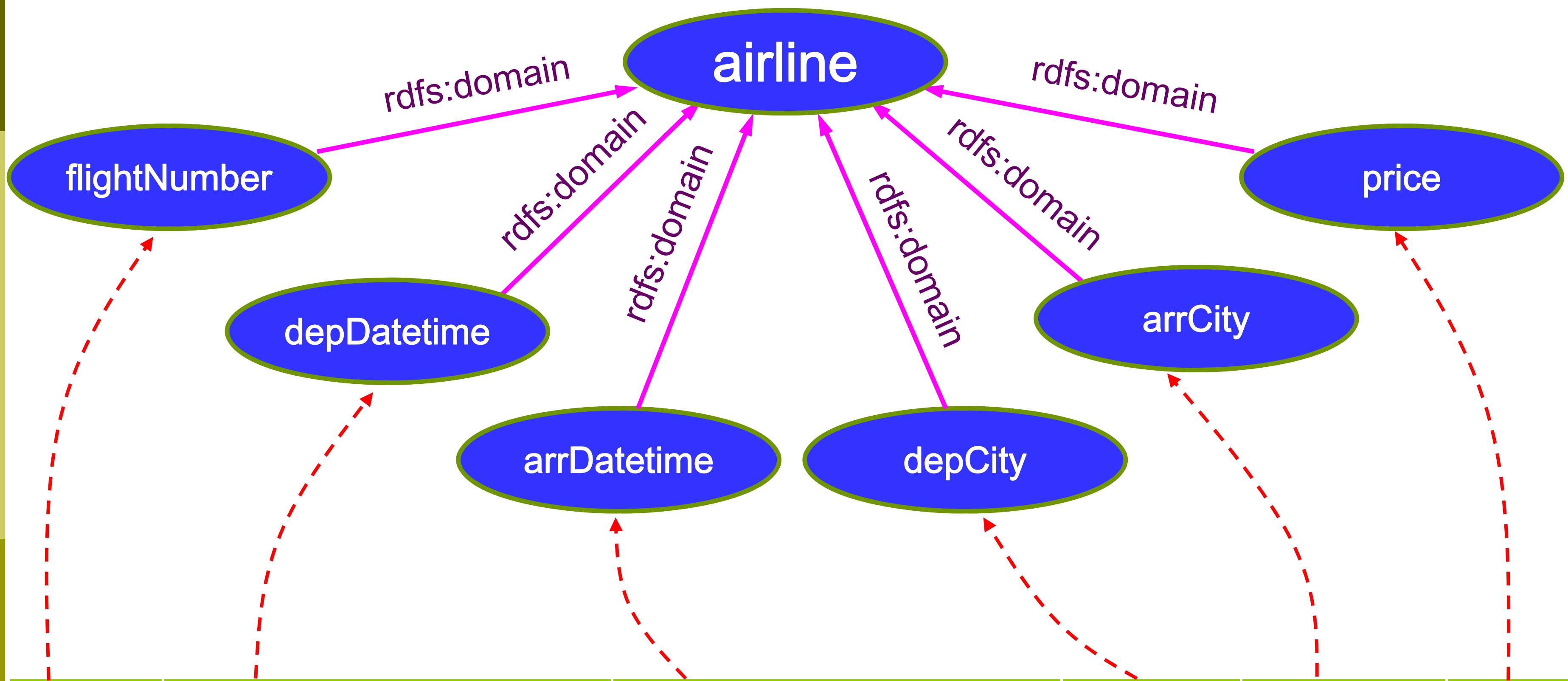
http://homepage.fudan.edu.cn/zhengxq/ rdfs:type Person ;
fullName "Sean Zheng" ;
mailBox "zhengxq@fudan.edu.cn" ;
title "Dr." .

Schema Mapping



flight	departure	arrival	from	to	ticketprice
us339	12:30 PM 02/09/2011	7:25 AM 02/10/2011	BOS	TYO	950
us512	9:45 AM 02/10/2011	10:30 PM 02/10/2011	TYO	SHA	380

Schema Mapping



ID	depdate	arrdate	depcity	arrcity	price
jp241	2011-02-10T09:25:00Z	2011-02-10T22:05:00Z	Tokyo	Shanghai	25

Heterogeneity

```
# Named graph: http://usairline.com/flights

us339  depDateTime  "12:30 PM 02/09/2011" .
us339  arrDateTime  "7:25 AM 02/10/2011"  .
us339  depCity  "BOS" .
us339  arrCity  "TYO" .
us339  price  950 .

us512  depDateTime  "9:45 AM 02/10/2011" .
us512  arrDateTime  "10:30 PM 02/10/2011" .
us512  depCity  "TYO" .
us512  arrCity  "SHA" .
us512  price  380 .
```

```
# Named graph: http://japanairline.com/flights

jp241  depDateTime  "2011-02-10T09:25:00Z"^^xsd:dateTime .
jp241  arrDateTime  "2011-02-10T22:05:00Z"^^xsd:dateTime .
jp241  depCity  "Tokyo" .
jp241  arrCity  "Shanghai" .
jp241  price  25 .
```

```
SELECT  ?airline1  ?airline2  ?total
WHERE  {
  GRAPH  ?graph1
    {  ?airline1   depDateTime  ?depDateTime1 ;
                   arrDateTime  ?arrDateTime1 ;
                   depCity  "Boston" ;
                   arrCity  "Tokyo" ;
                   price  ?price1 .  }
  GRAPH  ?graph2
    {  ?airline2  depDateTime  ?depDateTime2 ;
                   arrDateTime  ?arrDateTime2 ;
                   depCity  "Tokyo" ;
                   arrCity  "Shanghai" ;
                   price  ?price2 .  }
  FILTER ( ?depDateTime1 >= "9:30 AM 02/09/2011" )  .
  FILTER ( ?arrDateTime2 <= "11:30 PM 02/10/2011" )  .
  FILTER ( ?arrDateTime1 < ?depDateTime2 )  .
  LET ( ?total  :=  ?price1 + ?price2 )  }
  ORDER BY ASC(?total)
  LIMIT 1
```

Encoding Heterogeneity

Heterogeneity

```
# Named graph: http://usairline.com/flights

us339  depDateTime  "12:30 PM 02/09/2011" .
us339  arrDateTime  "7:25 AM 02/10/2011"  .
us339  depCity  "BOS" .
us339  arrCity  "TYO" .
us339  price  950 .

us512  depDateTime  "9:45 AM 02/10/2011" .
us512  arrDateTime  "10:30 PM 02/10/2011" .
us512  depCity  "TYO" .
us512  arrCity  "SHA" .
us512  price  380 .
```

```
# Named graph: http://japanairline.com/flights

jp241  depDateTime  "2011-02-10T09:25:00Z"^^xsd:dateTime .
jp241  arrDateTime  "2011-02-10T22:05:00Z"^^xsd:dateTime .
jp241  depCity  "Tokyo" .
jp241  arrCity  "Shanghai" .
jp241  price  25 .
```

```
SELECT  ?airline1  ?airline2  ?total
WHERE  {
  GRAPH  ?graph1
    {  ?airline1   depDateTime  ?depDateTime1 ;
                   arrDateTime  ?arrDateTime1 ;
                   depCity  "Boston" ;
                   arrCity  "Tokyo" ;
                   price  ?price1 .  }
  GRAPH  ?graph2
    {  ?airline2  depDateTime  ?depDateTime2 ;
                  arrDateTime  ?arrDateTime2 ;
                  depCity  "Tokyo" ;
                  arrCity  "Shanghai" ;
                  price  ?price2 .  }
  FILTER ( ?depDateTime1 >= "9:30 AM 02/09/2011" )  .
  FILTER ( ?arrDateTime2 <= "11:30 PM 02/10/2011" )  .
  FILTER ( ?arrDateTime1 < ?depDateTime2 )  .
  LET ( ?total  :=  ?price1 + ?price2 )  }
  ORDER BY ASC(?total)
  LIMIT 1
```

Format Heterogeneity

Heterogeneity

```
# Named graph: http://usairline.com/flights

us339  depDateTime  "12:30 PM 02/09/2011" .
us339  arrDateTime  "7:25 AM 02/10/2011"  .
us339  depCity  "BOS" .
us339  arrCity  "TYO" .
us339  price  950 .

us512  depDateTime  "9:45 AM 02/10/2011" .
us512  arrDateTime  "10:30 PM 02/10/2011" .
us512  depCity  "TYO" .
us512  arrCity  "SHA" .
us512  price  380 .
```

```
# Named graph: http://japanairline.com/flights

jp241  depDateTime  "2011-02-10T09:25:00Z"^^xsd:dateTime .
jp241  arrDateTime  "2011-02-10T22:05:00Z"^^xsd:dateTime .
jp241  depCity  "Tokyo" .
jp241  arrCity  "Shanghai" .
jp241  price  25 .
```

```
SELECT  ?airline1  ?airline2  ?total
WHERE  {
  GRAPH  ?graph1
    {  ?airline1   depDateTime  ?depDateTime1 ;
                   arrDateTime  ?arrDateTime1 ;
                   depCity  "Boston" ;
                   arrCity  "Tokyo" ;
                   price  ?price1 .  }
  GRAPH  ?graph2
    {  ?airline2  depDateTime  ?depDateTime2 ;
                  arrDateTime  ?arrDateTime2 ;
                  depCity  "Tokyo" ;
                  arrCity  "Shanghai" ;
                  price  ?price2 .  }
  FILTER ( ?depDateTime1 >= "9:30 AM 02/09/2011" )  .
  FILTER ( ?arrDateTime2 <= "11:30 PM 02/10/2011" )  .
  FILTER ( ?arrDateTime1 < ?depDateTime2 )  .
  LET ( ?total  :=  ?price1 + ?price2 )  }
  ORDER BY ASC(?total)
  LIMIT 1
```

Unit Heterogeneity

Heterogeneity

```
# Named graph: http://usairline.com/flights

us339  depDateTime  "12:30 PM 02/09/2011" .
us339  arrDateTime  "7:25 AM 02/10/2011"  .
us339  depCity  "BOS" .
us339  arrCity  "TYO" .
us339  price  950 .

us512  depDateTime  "9:45 AM 02/10/2011" .
us512  arrDateTime  "10:30 PM 02/10/2011" .
us512  depCity  "TYO" .
us512  arrCity  "SHA" .
us512  price  380 .
```

```
SELECT  ?airline1  ?airline2  ?total
WHERE  {
  GRAPH  ?graph1
    {  ?airline1   depDateTime  ?depDateTime1 ;
                   arrDateTime  ?arrDateTime1 ;
                   depCity  "BOS" ;
                   arrCity "TYO" ;
                   price  ?price1 .  }
  GRAPH  ?graph2
    {  ?airline2  depDateTime  ?depDateTime2 ;
                  arrDateTime  ?arrDateTime2 ;
                  depCity "TYO" ;
                  arrCity  "SHA" ;
                  price  ?price2 .  }
  FILTER ( ?depDateTime1 >= "9:30 AM 02/09/2011" )  .
  FILTER ( ?arrDateTime2 <= "11:30 PM 02/10/2011" )  .
  FILTER ( ?arrDateTime1 < ?depDateTime2 )  .
  LET ( ?total  :=  ?price1 + ?price2 )  }
  ORDER BY ASC(?total)
  LIMIT 1
```

```
# Named graph: http://japanairline.com/flights

jp241  depDateTime  "2011-02-10T09:25:00Z"^^xsd:dateTime .
jp241  arrDateTime  "2011-02-10T22:05:00Z"^^xsd:dateTime .
jp241  depCity  "Tokyo" .
jp241  arrCity  "Shanghai" .
jp241  price  25 .
```

It still does not work.

Query Mediation

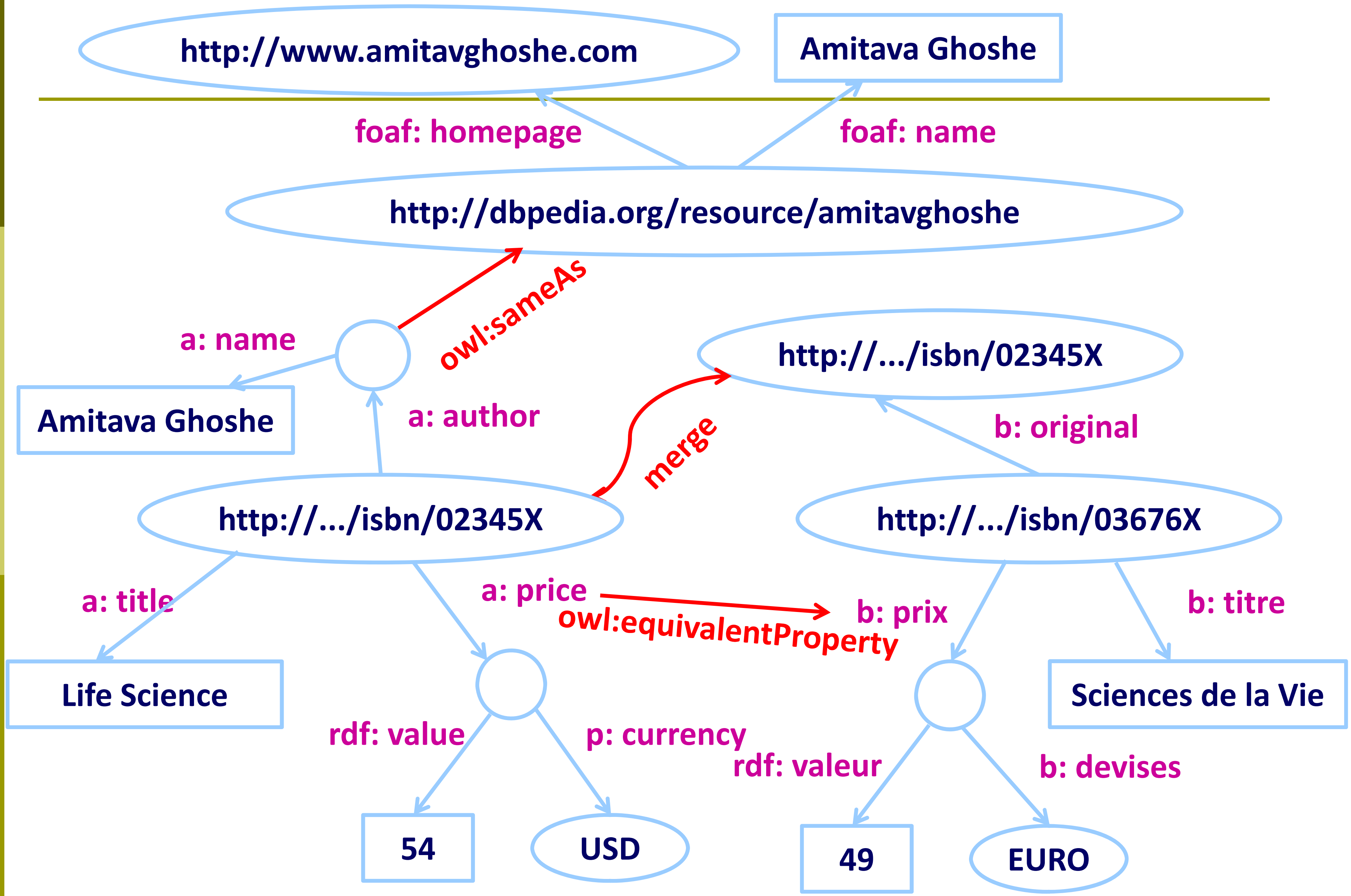
```
SELECT  ?airline1  ?airline2  ?total
WHERE  {
  GRAPH  ?graph1
   {  ?airline1   depDateTime  ?depDateTime1 ;
                  arrDateTime  ?arrDateTime1 ;
                  depCity  "Boston" ;
                  arrCity  "Tokyo" ;
                  price  ?price1 .  }
  GRAPH  ?graph2
   {  ?airline2  depDateTime  ?depDateTime2 ;
                  arrDateTime  ?arrDateTime2 ;
                  depCity  "Tokyo" ;
                  arrCity  "Shanghai" ;
                  price  ?price2 .  }
  FILTER ( ?depDateTime1 >= "9:30 AM 02/09/2011" )  .
  FILTER ( ?arrDateTime2 <= "11:30 PM 02/10/2011" )  .
  FILTER ( ?arrDateTime1 < ?depDateTime2 )
  LET ( ?total  :=  ?price1 + ?price2 )  }
  ORDER BY ASC(?total)
```

The answers returned should be further transformed so that they conform to the context of the receiver.

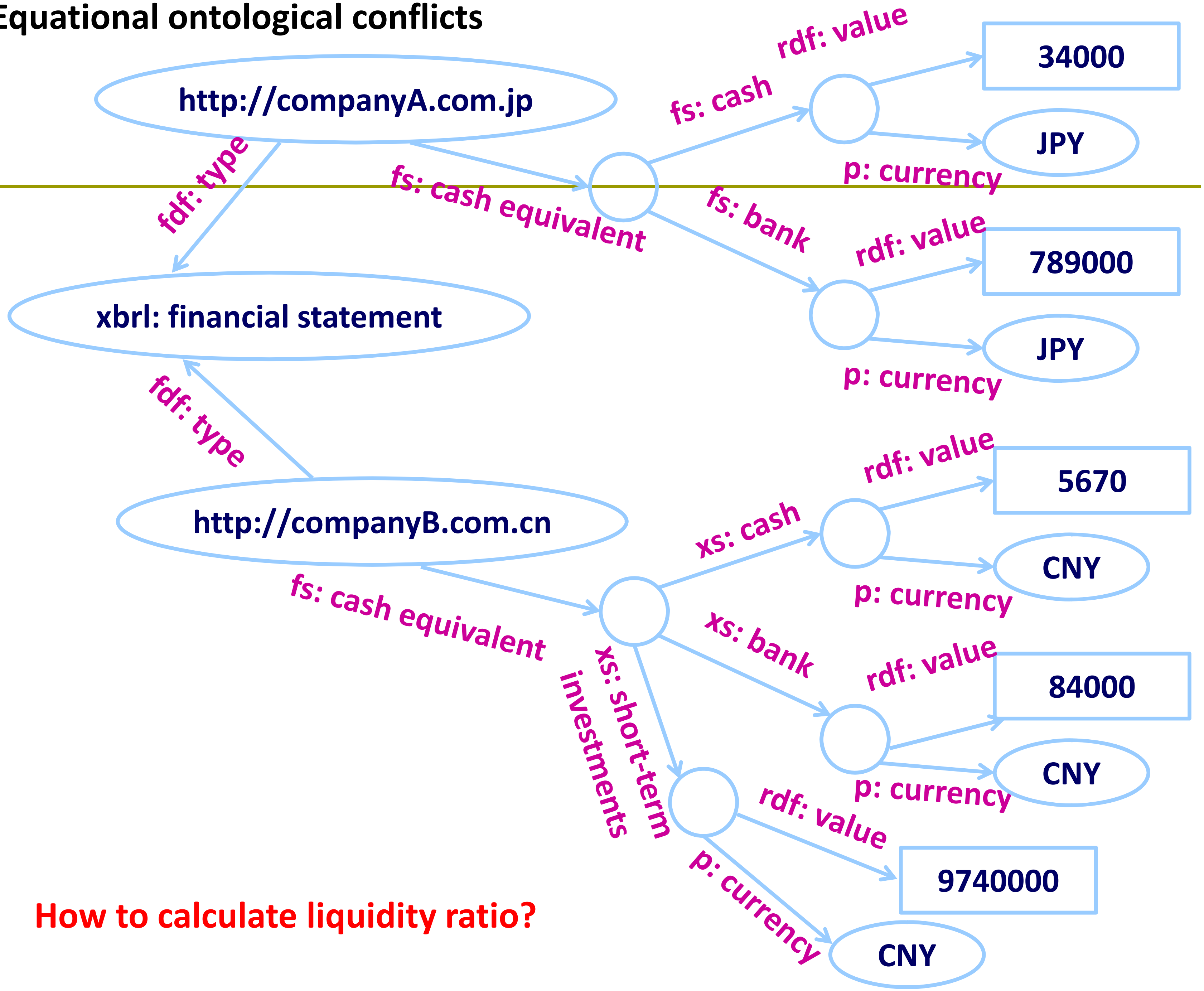
The constants should be transformed to comply with assumptions in the source contexts.

One of two arguments in expressions should be transformed so that the two arguments conform to the same context.

SELECT ?author ?titre ?homepage

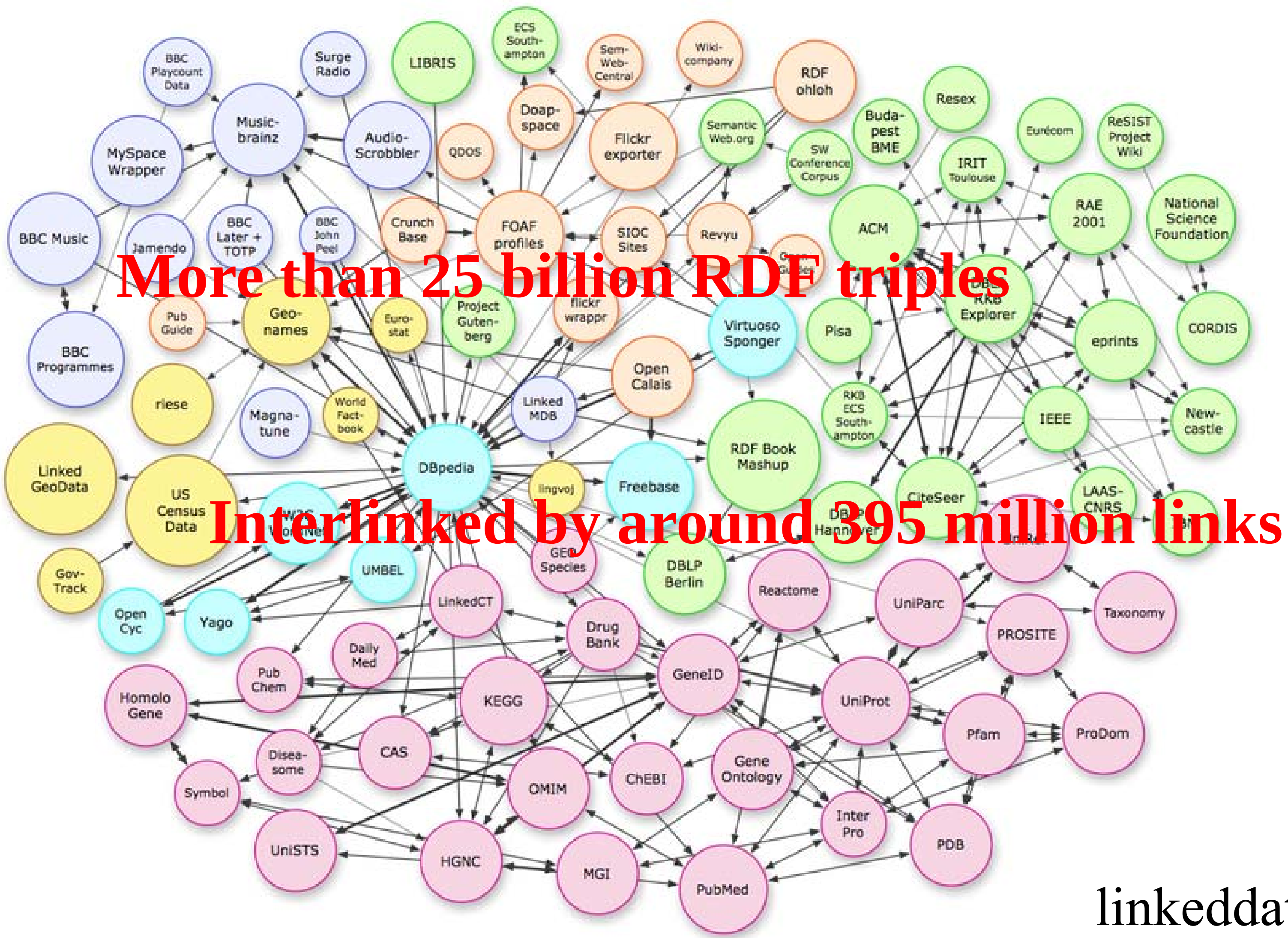


Equational ontological conflicts



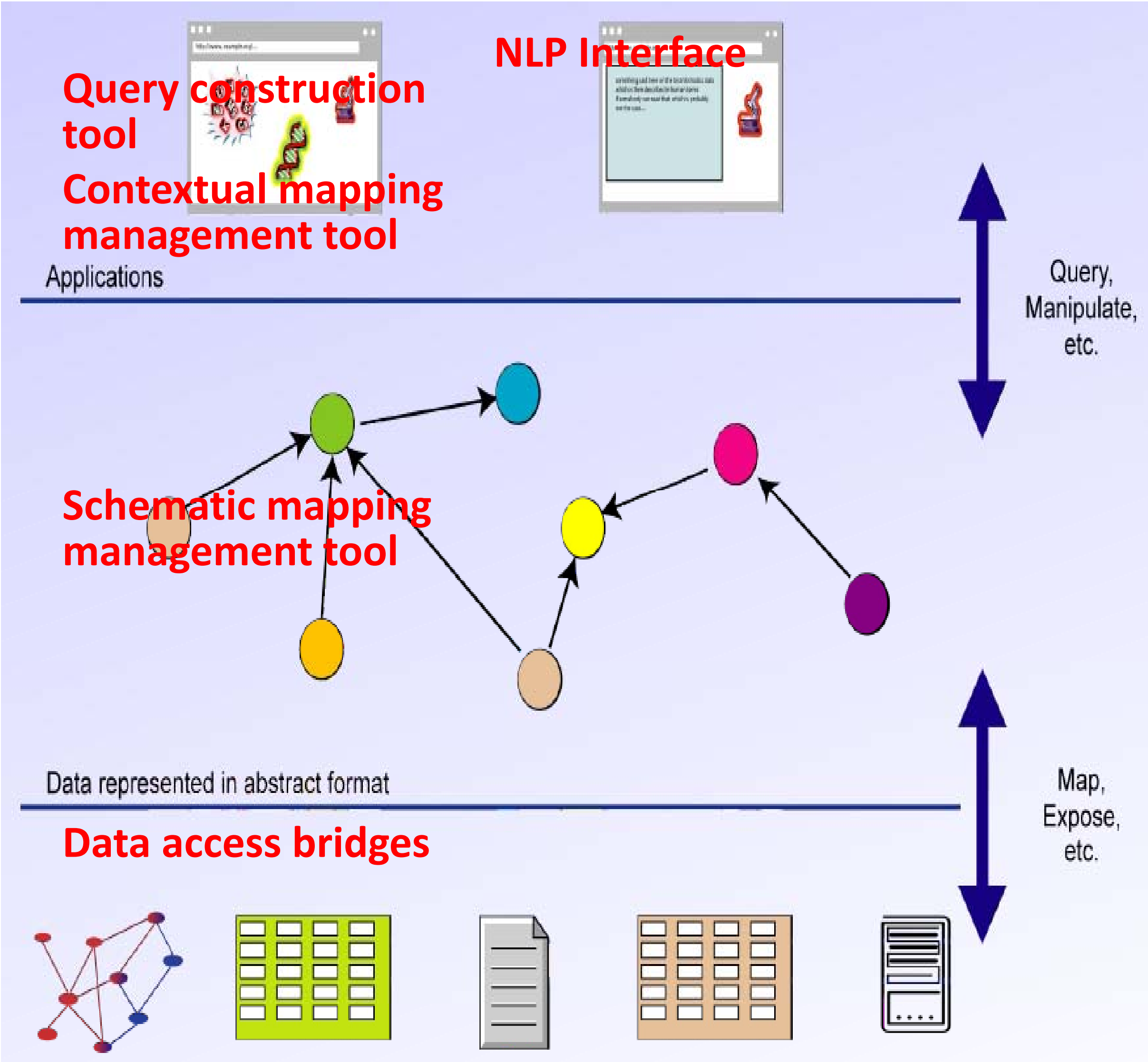
How to calculate liquidity ratio?

Linked Data



linkeddata.org

Data Integration System



High-level Applications

SPARQL query mediation system

SPARQL query engine

Mapping of the schema of underlying data sources to the common ontology

Access data sources by their RDF views

RDF databases, relational databases, and other non-RDF sources

Description Logic

- Description logics (DL) are the descendants of so-called "***structured inheritance networks***", which were introduced to overcome the ambiguities of early ***semantic networks*** and ***frames***. In DL, ***concepts*** are defined by the sets of ***objects*** as unary predicates, and ***roles*** are defined by the relationships between objects as binary predicates.
- We name each $\mathcal{AL}$-language by a string of the form

$$\mathcal{AL}[\mathcal{U}][\mathcal{E}][\mathcal{N}][\mathcal{C}]$$

Description Logic Concept Constructors

Name	Syntax	Semantics	Symbol
Top	$\top$	$\Delta^{\mathcal{I}}$	$\mathcal{AL}$
Bottom	$\bot$	$\emptyset$	$\mathcal{AL}$
Intersection	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$	$\mathcal{AL}$
Union	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$	$\mathcal{U}$
Negation	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$	$\mathcal{C}$
Value restriction	$\forall R.C$	$\{a \in \Delta^{\mathcal{I}} \mid \forall b.\ (a,b) \in R^{\mathcal{I}} \to b \in C^{\mathcal{I}}\}$	$\mathcal{AL}$
Existential quant.	$\exists R.C$	$\{a \in \Delta^{\mathcal{I}} \mid \exists b.\ (a,b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\}$	$\mathcal{E}$
Unqualified number restriction	$\geqslant n\, R$ $\leqslant n\, R$ $= n\, R$	$\{a \in \Delta^{\mathcal{I}} \mid \lvert\{b \in \Delta^{\mathcal{I}} \mid (a,b) \in R^{\mathcal{I}}\}\rvert \geq n\}$ $\{a \in \Delta^{\mathcal{I}} \mid \lvert\{b \in \Delta^{\mathcal{I}} \mid (a,b) \in R^{\mathcal{I}}\}\rvert \leq n\}$ $\{a \in \Delta^{\mathcal{I}} \mid \lvert\{b \in \Delta^{\mathcal{I}} \mid (a,b) \in R^{\mathcal{I}}\}\rvert = n\}$	$\mathcal{N}$
Qualified number restriction	$\geqslant n\, R.C$ $\leqslant n\, R.C$ $= n\, R.C$	$\{a \in \Delta^{\mathcal{I}} \mid \lvert\{b \in \Delta^{\mathcal{I}} \mid (a,b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\}\rvert \geq n\}$ $\{a \in \Delta^{\mathcal{I}} \mid \lvert\{b \in \Delta^{\mathcal{I}} \mid (a,b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\}\rvert \leq n\}$ $\{a \in \Delta^{\mathcal{I}} \mid \lvert\{b \in \Delta^{\mathcal{I}} \mid (a,b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\}\rvert = n\}$	$\mathcal{Q}$
Role-value-map	$R \subseteq S$ $R = S$	$\{a \in \Delta^{\mathcal{I}} \mid \forall b.(a,b) \in R^{\mathcal{I}} \to (a,b) \in S^{\mathcal{I}}\}$ $\{a \in \Delta^{\mathcal{I}} \mid \forall b.(a,b) \in R^{\mathcal{I}} \leftrightarrow (a,b) \in S^{\mathcal{I}}\}$	
Agreement and disagreement	$u_1 \doteq u_2$ $u_1 \not\doteq u_2$	$\{a \in \Delta^{\mathcal{I}} \mid \exists b \in \Delta^{\mathcal{I}}.\ u_1^{\mathcal{I}}(a) = b = u_2^{\mathcal{I}}(a)\}$ $\{a \in \Delta^{\mathcal{I}} \mid \exists b_1, b_2 \in \Delta^{\mathcal{I}}.\ u_1^{\mathcal{I}}(a) = b_1 \neq b_2 = u_2^{\mathcal{I}}(a)\}$	$\mathcal{F}$
Nominal	I	$I^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ with $\lvert I^{\mathcal{I}}\rvert = 1$	$\mathcal{O}$

Family

$$
\begin{aligned}
\mathsf{Woman} &\equiv \mathsf{Person} \sqcap \mathsf{Female}\\
\mathsf{Man} &\equiv \mathsf{Person} \sqcap \neg(\mathsf{Person} \sqcap \mathsf{Female})\\
\mathsf{Mother} &\equiv (\mathsf{Person} \sqcap \mathsf{Female}) \sqcap \exists \mathsf{hasChild}.\mathsf{Person}\\
\mathsf{Father} &\equiv (\mathsf{Person} \sqcap \neg(\mathsf{Person} \sqcap \mathsf{Female})) \sqcap \exists \mathsf{hasChild}.\mathsf{Person}\\
\mathsf{Parent} &\equiv ((\mathsf{Person} \sqcap \neg(\mathsf{Person} \sqcap \mathsf{Female})) \sqcap \exists \mathsf{hasChild}.\mathsf{Person})\\
&\quad \sqcup ((\mathsf{Person} \sqcap \mathsf{Female}) \sqcap \exists \mathsf{hasChild}.\mathsf{Person})\\
\mathsf{Grandmother} &\equiv ((\mathsf{Person} \sqcap \mathsf{Female}) \sqcap \exists \mathsf{hasChild}.\mathsf{Person})\\
&\quad \sqcap \exists \mathsf{hasChild}.(((\mathsf{Person} \sqcap \neg(\mathsf{Person} \sqcap \mathsf{Female}))\\
&\qquad\qquad \sqcap \exists \mathsf{hasChild}.\mathsf{Person})\\
&\qquad\qquad \sqcup ((\mathsf{Person} \sqcap \mathsf{Female})\\
&\qquad\qquad\quad \sqcap \exists \mathsf{hasChild}.\mathsf{Person}))\\
\mathsf{MotherWithManyChildren} &\equiv ((\mathsf{Person} \sqcap \mathsf{Female}) \sqcap \exists \mathsf{hasChild}.\mathsf{Person}) \sqcap \geqslant 3\, \mathsf{hasChild}\\
\mathsf{MotherWithoutDaughter} &\equiv ((\mathsf{Person} \sqcap \mathsf{Female}) \sqcap \exists \mathsf{hasChild}.\mathsf{Person})\\
&\quad \sqcap \forall \mathsf{hasChild}.(\neg(\mathsf{Person} \sqcap \mathsf{Female}))\\
\mathsf{Wife} &\equiv (\mathsf{Person} \sqcap \mathsf{Female})\\
&\quad \sqcap \exists \mathsf{hasHusband}.(\mathsf{Person} \sqcap \neg(\mathsf{Person} \sqcap \mathsf{Female}))
\end{aligned}
$$

The Oedipus

$\mathsf{hasChild}(\mathsf{IOKASTE}, \mathsf{OEDIPUS})$

$\mathsf{hasChild}(\mathsf{OEDIPUS}, \mathsf{POLYNEIKES})$

$\mathsf{Patricide}(\mathsf{OEDIPUS})$

$\mathsf{hasChild}(\mathsf{IOKASTE}, \mathsf{POLYNEIKES})$

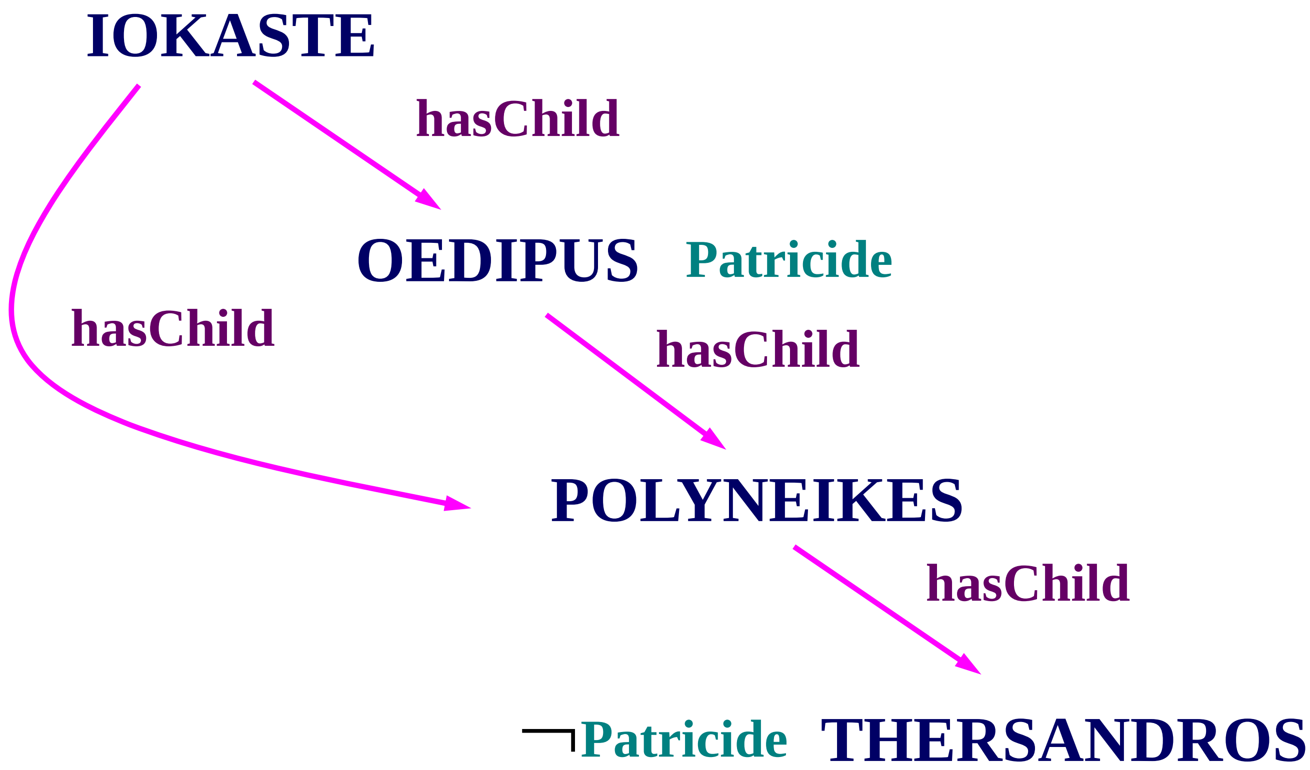
$\mathsf{hasChild}(\mathsf{POLYNEIKES}, \mathsf{THERSANDROS})$

$\neg\mathsf{Patricide}(\mathsf{THERSANDROS})$

Question:

$(\exists \mathsf{hasChild}.(\mathsf{Patricide} \sqcap \exists \mathsf{hasChild}.\neg\mathsf{Patricide}))(\mathsf{IOKASTE})\ ?$

The Oedipus



$$(\exists \mathsf{hasChild}.(\mathsf{Patricide} \sqcap \exists \mathsf{hasChild}.\neg\mathsf{Patricide}))(\mathsf{IOKASTE})\ ?$$

Reasoning Problems

Reasoning Problems

1. *KB-satisfiability* : Σ is *satisfiable*, if it has a model;

2. *Concept Satisfiability* : C is *satisfiable* w.r.t Σ, if there exists a model $\mathcal{I}$ of Σ such that $C^{\mathcal{I}} \neq \emptyset$;

3. *Subsumption* : C is *subsumed* by D w.r.t. Σ, if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for every model $\mathcal{I}$ of Σ;

4. *Instance Checking* : a is an instance of C, written $\Sigma \models C(a)$, if the assertion $C(a)$ is satisfied in every model of Σ.

The Above Problem can be reduced to KB-satisfiability

$$
\begin{array}{rcl}
C \text{ is satisfiable w.r.t } \Sigma & \text{iff} & \langle \mathcal{T}, \mathcal{A} \cup \{C(b)\}\rangle \text{ is satisfiable.} \\
C \text{ is subsumed by } D \text{ w.r.t. } \Sigma & \text{iff} & \langle \mathcal{T}, \mathcal{A} \cup \{(C \sqcap \neg D)(b)\}\rangle \text{ is not satisfiable.} \\
\Sigma \models C(a) & \text{iff} & \langle \mathcal{T}, \mathcal{A} \cup \{(\neg C)(a)\}\rangle \text{ is not satisfiable.}
\end{array}
$$

Description Logic $\mathcal{ALCNR}$

$C, D \rightarrow A$	$\mid$		(concept name)
$\top$	$\mid$		(top concept)
$\bot$	$\mid$		(bottom concept)
$C \sqcap D$	$\mid$		(conjunction)
$C \sqcup D$	$\mid$		(disjunction)
$\neg C$	$\mid$		(complement)
$\forall R.C$	$\mid$		(universal quantification)
$\exists R.C$	$\mid$		(existential quantification)
$(\geq n\ R)$	$\mid$	$(\leq n\ R)$	(number restrictions)
$R \rightarrow P_1 \sqcap \ldots \sqcap P_k$			(role conjunction)

Description Logic $\mathcal{ALCNR}$

$$
\begin{aligned}
\top^{\mathcal{I}} &= \Delta^{\mathcal{I}} \\
\bot^{\mathcal{I}} &= \emptyset \\
(C \sqcap D)^{\mathcal{I}} &= C^{\mathcal{I}} \cap D^{\mathcal{I}} \\
(C \sqcup D)^{\mathcal{I}} &= C^{\mathcal{I}} \cup D^{\mathcal{I}} \\
(\neg C)^{\mathcal{I}} &= \Delta^{\mathcal{I}} \backslash C^{\mathcal{I}} \\
(\forall R.C)^{\mathcal{I}} &= \{ d_1 \in \Delta^{\mathcal{I}} \mid \forall d_2 : (d_1, d_2) \in R^{\mathcal{I}} \rightarrow d_2 \in C^{\mathcal{I}} \} \\
(\exists R.C)^{\mathcal{I}} &= \{ d_1 \in \Delta^{\mathcal{I}} \mid \exists d_2 : (d_1, d_2) \in R^{\mathcal{I}} \wedge d_2 \in C^{\mathcal{I}} \} \\
(\geq n R)^{\mathcal{I}} &= \{ d_1 \in \Delta^{\mathcal{I}} \mid \# \{ d_2 : \mid (d_1, d_2) \in R^{\mathcal{I}} \} \geq n \} \\
(\leq n R)^{\mathcal{I}} &= \{ d_1 \in \Delta^{\mathcal{I}} \mid \# \{ d_2 : \mid (d_1, d_2) \in R^{\mathcal{I}} \} \leq n \} \\
(P_1 \sqcap \ldots \sqcap P_k)^{\mathcal{I}} &= P_1^{\mathcal{I}} \cap \ldots \cap P_k^{\mathcal{I}}
\end{aligned}
$$

Constraint system

Knowledge Base

$\mathcal{T} = \{\mathtt{Italian} \sqsubseteq \exists \mathtt{FRIEND.Italian}\}$

$$\begin{aligned}\mathcal{A} = \{&\mathtt{FRIEND(peter, susan)},\\ &\forall \mathtt{FRIEND}.\neg \mathtt{Italian(peter)},\\ &\exists \mathtt{FRIEND.Italian(susan)}\}\end{aligned}$$

Constraint System

$s{:}C, \quad sPt, \quad \forall x.x{:}C, \quad s \neq t,$

The Corresponding Constraint System

$$\begin{aligned}S_\Sigma = \{&\forall x.x{:}\neg \mathtt{Italian} \sqcup \exists \mathtt{FRIEND.Italian},\\ &\mathtt{peterFRIENDsusan},\\ &\mathtt{peter}{:}\forall \mathtt{FRIEND}.\neg \mathtt{Italian},\\ &\mathtt{susan}{:}\exists \mathtt{FRIEND.Italian}\\ &\mathtt{peter} \neq \mathtt{susan}\}\end{aligned}$$

$$S_\Sigma = \{\forall x.x{:}\ \neg\texttt{Italian} \sqcup \exists\texttt{FRIEND.Italian},$$
$$\texttt{peterFRIENDsusan},$$
$$\texttt{peter}{:}\ \forall\texttt{FRIEND.}\neg\texttt{Italian},$$
$$\texttt{susan}{:}\ \exists\texttt{FRIEND.Italian}$$
$$\texttt{peter} \not\doteq \texttt{susan}\}$$

$(\rightarrow_\forall\text{-rule})$

$$\texttt{susan}{:}\ \neg\texttt{Italian}$$

$(\rightarrow_{\forall x}\text{-rule})$

$$\texttt{peter}{:}\ \neg\texttt{Italian} \sqcup \exists\texttt{FRIEND.Italian}$$
$$\texttt{susan}{:}\ \neg\texttt{Italian} \sqcup \exists\texttt{FRIEND.Italian}$$

$(\rightarrow_\sqcup\text{-rule})$

$\texttt{peter}{:}\ \neg\texttt{Italian}$ **Clash**

$(\rightarrow_\sqcup\text{-rule})$

$$\texttt{susan}{:}\ \exists\texttt{FRIEND.Italian}$$

$\rightarrow_\exists\text{-rule}$

$$\texttt{susanFRIENDx},\ \texttt{x}{:}\ \texttt{Italian}$$

$(\rightarrow_{\forall}\text{-rule})$

$\mathtt{x}: \neg\mathtt{Italian} \sqcup \exists\mathtt{FRIEND.Italian}$

$(\rightarrow_{\sqcup}\text{-rule})$

Clash

$\mathtt{x}: \exists\mathtt{FRIEND.Italian}$

$\rightarrow_{\exists}\text{-rule}$

$\mathtt{xFRIENDy}, \mathtt{y}: \mathtt{Italian}$

$(\rightarrow_{\forall}\text{-rule})$

$\mathtt{y}: \neg\mathtt{Italian} \sqcup \exists\mathtt{FRIEND.Italian}$

$(\rightarrow_{\sqcup}\text{-rule})$

Clash

$\mathtt{y}: \exists\mathtt{FRIEND.Italian}$

$$\Delta^{\mathcal{I}} = \{\mathtt{peter}, \mathtt{susan}, \mathtt{x}, \mathtt{y}\}$$
$$\mathtt{Italian}^{\mathcal{I}} = \{\mathtt{x}, \mathtt{y}\}$$
$$\mathtt{FRIEND}^{\mathcal{I}} = \{(\mathtt{peter}, \mathtt{susan}), (\mathtt{susan}, \mathtt{x}), (\mathtt{x}, \mathtt{y}), (\mathtt{y}, \mathtt{y})\}$$

Sequence of applications of the rules

The Corresponding Constraint System

$$
\begin{aligned}
S_\Sigma = \{&\forall x.x\colon \neg\mathtt{Italian} \sqcup \exists\mathtt{FRIEND.Italian},\\
&\mathtt{peterFRIENDsusan},\\
&\mathtt{peter}\colon \forall\mathtt{FRIEND}.\neg\mathtt{Italian},\\
&\mathtt{susan}\colon \exists\mathtt{FRIEND.Italian}\\
&\mathtt{peter} \neq \mathtt{susan}\}
\end{aligned}
$$

$$
\begin{aligned}
S_1 &= S_\Sigma \cup \{\mathtt{susan}\colon \neg\mathtt{Italian}\}\ (\rightarrow_\forall\text{-rule})\\
S_2 &= S_1 \cup \{\mathtt{peter}\colon \neg\mathtt{Italian} \sqcup \exists\mathtt{FRIEND.Italian}\}\ (\rightarrow_{\forall x}\text{-rule})\\
S_3 &= S_2 \cup \{\mathtt{susan}\colon \neg\mathtt{Italian} \sqcup \exists\mathtt{FRIEND.Italian}\}\ (\rightarrow_{\forall x}\text{-rule})\\
S_4 &= S_3 \cup \{\mathtt{peter}\colon \neg\mathtt{Italian}\}\ (\rightarrow_\sqcup\text{-rule})\\
S_5 &= S_4 \cup \{\mathtt{susanFRIENDx}, \mathtt{x}\colon \mathtt{Italian}\}\ (\rightarrow_\exists\text{-rule})\\
S_6 &= S_5 \cup \{\mathtt{x}\colon \neg\mathtt{Italian} \sqcup \exists\mathtt{FRIEND.Italian}\}\ (\rightarrow_{\forall x}\text{-rule})\\
S_7 &= S_6 \cup \{\mathtt{x}\colon \exists\mathtt{FRIEND.Italian}\}\ (\rightarrow_\sqcup\text{-rule})\\
S_8 &= S_7 \cup \{\mathtt{xFRIENDy}, \mathtt{y}\colon \mathtt{Italian}\}\ (\rightarrow_\exists\text{-rule})\\
S_9 &= S_8 \cup \{\mathtt{y}\colon \neg\mathtt{Italian} \sqcup \exists\mathtt{FRIEND.Italian}\}\ (\rightarrow_{\forall x}\text{-rule})\\
S_{10} &= S_9 \cup \{\mathtt{y}\colon \exists\mathtt{FRIEND.Italian}\}\ (\rightarrow_\sqcup\text{-rule})
\end{aligned}
$$

Propagation rules

1. $S \rightarrow_{\sqcap} \{s{:}C_1,\ s{:}C_2\} \cup S$

   if 1. $s{:}C_1 \sqcap C_2$ is in S,
   2. $s{:}C_1$ and $s{:}C_2$ are not both in S

2. $S \rightarrow_{\sqcup} \{s{:}D\} \cup S$

   if 1. $s{:}C_1 \sqcup C_2$ is in S,
   2. neither $s{:}C_1$ nor $s{:}C_2$ is in S,
   3. $D = C_1$ or $D = C_2$

3. $S \rightarrow_{\forall} \{t{:}C\} \cup S$

   if 1. $s{:}\forall R.C$ is in S,
   2. t is an R-successor of s,
   3. $t{:}C$ is not in S

4. $S \rightarrow_{\exists} \{sP_1y, \ldots, sP_ky,\ y{:}C\} \cup S$

   if 1. $s{:}\exists R.C$ is in S,
   2. $R = P_1 \sqcap \ldots \sqcap P_k$,
   3. y is a new variable,
   4. there is no t such that t is an R-successor of s in S and $t{:}C$ is in S,
   5. if s is a variable there is no variable w such that $w \prec s$ and $s \equiv_s w$

Propagation rules

5. $S \rightarrow_{\geq} \{sP_1 y_i, \ldots, sP_k y_i \mid i \in 1..n\} \cup \{y_i \neq y_j \mid i,j \in 1..n, i \neq j\} \cup S$

   if
   1. $s\colon(\geq n\, R)$ is in S,
   2. $R = P_1 \sqcap \ldots \sqcap P_k$,
   3. $y_1, \ldots, y_n$ are new variables,
   4. there do not exist n pairwise separated R-successors of s in S,
   5. if s is a variable there is no variable w such that $w \prec s$ and $s \equiv_s w$

6. $S \rightarrow_{\leq} S[y/t]$

   if
   1. $s\colon(\leq n\, R)$ is in S,
   2. s has more than n R-successors in S,
   3. y, t are two R-successors of s which are not separated

7. $S \rightarrow_{\forall x} \{s\colon C\} \cup S$

   if
   1. $\forall x.x\colon C$ is in S,
   2. s appears in S,
   3. $s\colon C$ is not in S.

Strategy for the application of rules

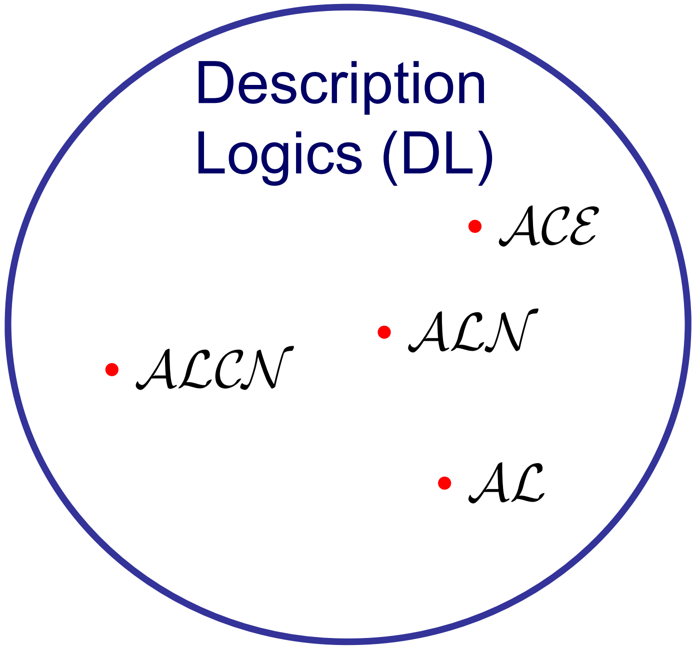
1. apply a rule to a variable only if no rule is applicable to individuals;
2. apply a rule to a variable x only if no rule is applicable to a variable y such that $y \prec x$;
3. apply generating rules only if no nongenerating rule is applicable.

Clash

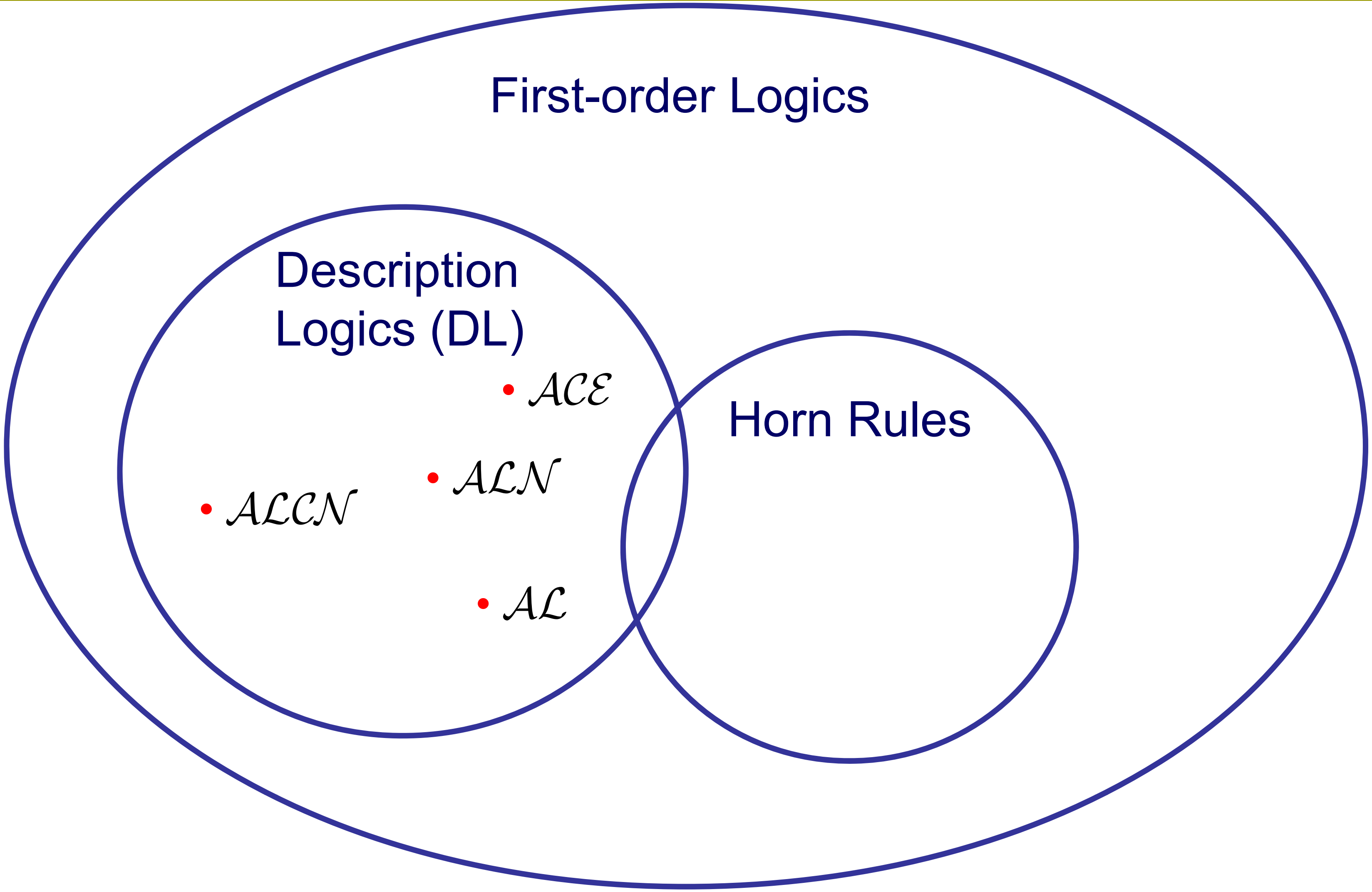
- $\{s\colon \bot\}$
- $\{s\colon A,\ s\colon \neg A\}$, where A is a concept name.
- $$\{s\colon (\leq n\ R)\} \cup \{sP_1t_i, \ldots, sP_kt_i \mid i \in 1..n+1\} \\ \cup \{t_i \not\doteq t_j \mid i, j \in 1..n+1, i \neq j\},$$

  where $R = P_1 \sqcap \ldots \sqcap P_k$.

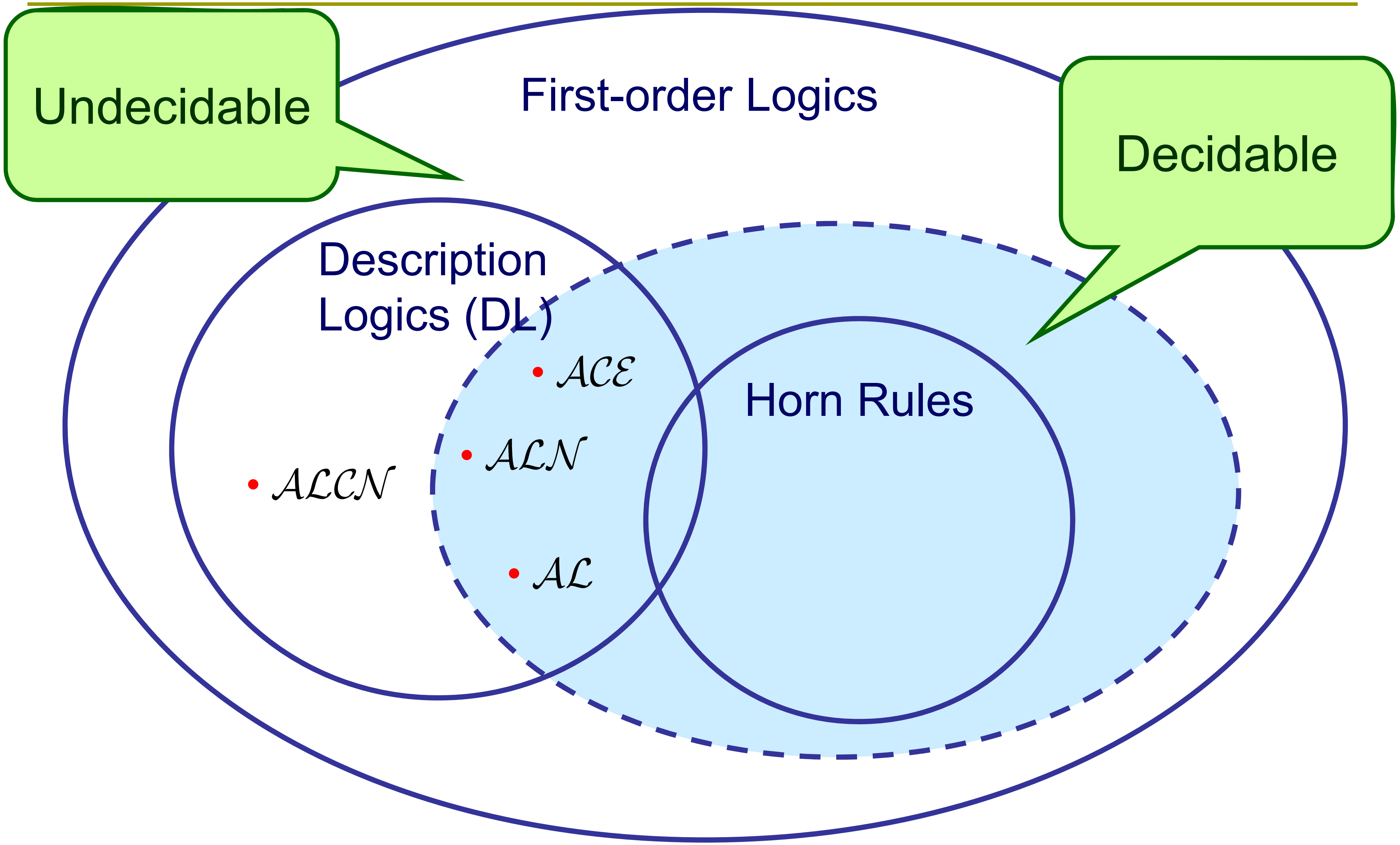
Description Logics & Horn Rules



Description Logics & Horn Rules



Description Logics & Horn Rules



Description Logics & Horn Rules

- Horn rules lack the possibility to express the existence of individuals whose identities might not be expressed explicitly.

It is impossible to state that every person has a mother (known or unknown).

$$Person \sqsubseteq \exists Mother.\top$$

Description Logics & Horn Rules

- Negation is not allowed within the body or head of a definite Horn rule .

It is impossible to represent that all persons are either male or female (but not both).

$$Person \sqsubseteq Male \sqcup Female$$

$$Male \sqsubseteq \neg Female$$

Description Logics & Horn Rules

- Description logics require that the quantified variable must occur in a binary predicate along with the free variable.

It is impossible to describe a concept, "home workers", whose individuals live and work at the same location.

$$work(x, y) \wedge live(x, z) \wedge locate(y, w) \wedge locate(z, w)$$
$$\longrightarrow houseWorkder(x)$$

Components of Hybrid Language

A knowledge base is a triple $\mathcal{K} = < \Sigma, \mathcal{R}, \mathcal{F} >$

- $\Sigma = <\mathcal{T}, \mathcal{A}>$ is an $\mathcal{ALCNR}$ knowledge base;
- $\mathcal{R}$ is a set of Horn rules;
- $\mathcal{F}$ is a set of ground facts;

Combining DL with Horn Rules

- Description Logic KB

$ForeignCompany \sqcap DomesticCompany \sqsubseteq \bot$

$CompeteWithForeign \equiv \exists SameIndustry.ForeignCompany$

$CompeteWithDomestic \equiv \exists SameIndustry.DomesticCompany$

$ProtectedCompany \equiv \forall SameIndustry.\neg ForeignCompany$

$NonMonopoly \equiv CompeteWithForeign \sqcup CompeteWithDomestic$

$ForeignCompany \sqcap ProtectedCompany \sqsubseteq \bot$

$NonMonopoly(b)$

- Horn Rules

r1: $serviceBy(x, y) \wedge ProtectedCompany(y) \longrightarrow price(x, high)$

r2: $serviceBy(x, y) \wedge SameIndustry(y, z) \wedge ForeignCompany(z) \wedge highQuality(x, y) \longrightarrow price(x, high)$

- Ground Facts

$serviceBy(a, b)$, $highQuality(a, b)$ **Query:** $price(a, high)$**?**

Two Interpretations

$\mathcal{I}_1$:	$\mathcal{I}_2$:
$\Delta^{\mathcal{I}_1} = \{b, v_1\}$	$\Delta^{\mathcal{I}_2} = \{b, v_1\}$
$ForeignCompany^{\mathcal{I}_1} = \{v_1\}$	$ForeignCompany^{\mathcal{I}_2} = \{\emptyset\}$
$DomesticCompany^{\mathcal{I}_1} = \{\emptyset\}$	$DomesticCompany^{\mathcal{I}_2} = \{v_1\}$
$CompeteWithForeign^{\mathcal{I}_1} = \{b\}$	$CompeteWithForeign^{\mathcal{I}_2} = \{\emptyset\}$
$CompeteWithDomestic^{\mathcal{I}_1} = \{\emptyset\}$	$CompeteWithDomestic^{\mathcal{I}_2} = \{b\}$
$ProtectedCompany^{\mathcal{I}_1} = \{\emptyset\}$	$ProtectedCompany^{\mathcal{I}_2} = \{b, v_1\}$
$NonMonopoly^{\mathcal{I}_1} = \{b\}$	$NonMonopoly^{\mathcal{I}_2} = \{b\}$
$SameIndustry^{\mathcal{I}_1} = \{(b, v_1)\}$	$SameIndustry^{\mathcal{I}_2} = \{(b, v_1)\}$

For the First Interpretation

$$r_1: \quad \neg serviceBy(x, y) \vee price(x, high) \| ProtectedCompany(y) \qquad (1)$$

$$r_2: \quad \neg serviceBy(x, y) \vee \neg highQuality(y, x) \vee price(x, high) \| ForeignCompany(z) \wedge SameIndustry(y, z) \qquad (2)$$

$$f_1: \quad serviceBy(a, b) \| \Box \qquad (3)$$

$$f_2: \quad highQuality(b, a) \| \Box \qquad (4)$$

$$Q: \quad \neg price(a, high) \| \Box \qquad (5)$$

For the model $\mathcal{M}_1$, we can deduce the following constrained refutation.

$$\neg serviceBy(a, y) \vee \neg highQuality(y, a) \| ForeignCompany(z) \wedge SameIndustry(y, z) \qquad (6) \quad \text{by (5) and (2) with } \{a/x\}$$

$$\neg highQuality(b, a) \| ForeignCompany(z) \wedge SameIndustry(b, z) \qquad (7) \quad \text{by (6) and (3) with } \{b/y\}$$

$$\Box \| ForeignCompany(z) \wedge SameIndustry(b, z) \qquad (8) \quad \text{by (7) and (4) with } \{\emptyset\}$$

and by mapping variable z to v_1, $\mathcal{M}_1 \models (ForeignCompany(z) \wedge SameIndustry(b, z))$ holds.

For the Second Interpretation

$r_1: \quad \neg serviceBy(x, y) \vee price(x, high) \| ProtectedCompany(y)$ (1)

$r_2: \quad \neg serviceBy(x, y) \vee \neg highQuality(y, x) \vee price(x, high) \|$ (2)
$ForeignCompany(z) \wedge SameIndustry(y, z)$

$f_1: \quad serviceBy(a, b) \| \Box$ (3)

$f_2: \quad highQuality(b, a) \| \Box$ (4)

$Q: \quad \neg price(a, high) \| \Box$ (5)

For the model $\mathcal{M}_2$, we also have another constrained refutation.

$\neg serviceBy(a, y) \| ProtectedCompany(y)$ (9) by (5) and (1) with $\{a/x\}$

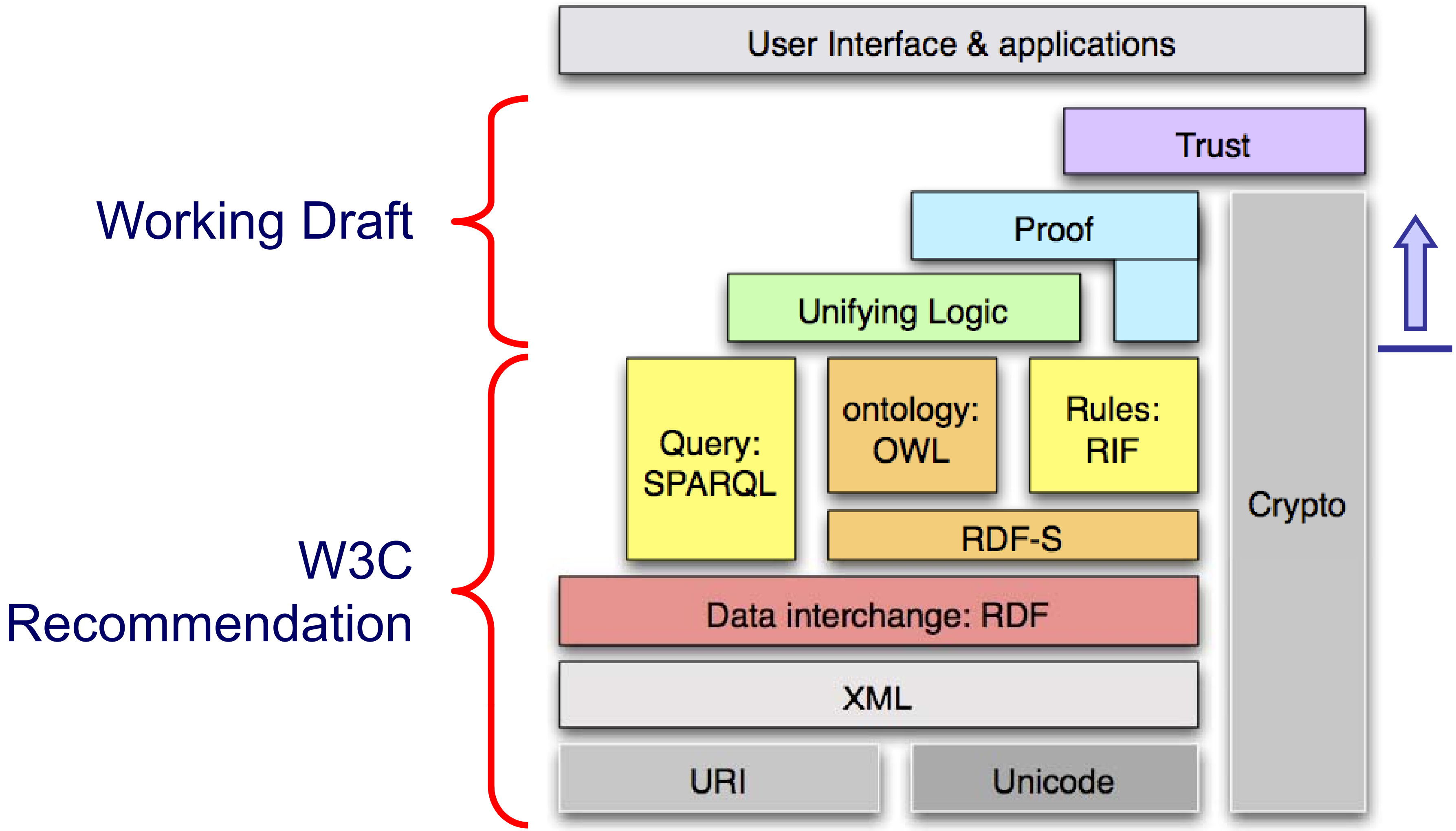
$\Box \| ProtectedCompany(b)$ (10) by (9) and (3) with $\{b/y\}$

and $\mathcal{M}_2 \models ProtectedCompany(b)$ holds.

Reasoning Algorithm

- Build all the ***canonical interpretations*** for a DL knowledge base and represent them by Herbrand structures;
- Collect all the ***SLD-derivations*** ending with the empty clauses for a set of function-free recursive Horn rules;
- For every canonical interpretation of DL knowledge base, check if there is a ***constrained SLD-refutation***.

Semantic Web Stack



Any question?

Xiaoqing Zheng

Fundan University