# Combining safe rules and ontologies by interfacing of reasoners

Uwe Aßmann, Jakob Henriksson, Jan Małuszyński

PPSWR06, Budva, Montenegro, 10th June 2006

# The objective

- Define a scheme that

  from given
  - Rule language R (e.g. Datalog, Xcerpt)
  - Logical language S (e.g. OWL-DL, ...)

  constructs
  - A language $R_S$ integrating R and S:
    - Syntax, Semantics of $R_S$: from syntax and semantics of R and S
    - A (complete) reasoner for $R_S$ by interfacing the reasoners of R and S

# Outline

- Motivating example
- The scheme
  - Principles and restrictions
  - An instance:
    - Datalog + OWL-DL
    - Prototype: interfacing XSB and a DL reasoner
- Conclusions
- Related work

# Motivating example

**Rule component $\prod$:**

$r_1$: price-in-usa(X,high) ←
made-by(X,Y),
NoFellowCompany(Y).

$r_2$: price-in-usa(X,high) ←
made-by(X,Y),
Associate(Y,Z),
American(Z),
monopoly-in-usa(Y,X).

$r_3$: made-by(a,b).

$r_4$: monopoly-in-usa(b,a).

**DL component $\sum$:**

*T-Box:*

| | |
|---|---|
| European ∩ American | ⊆ ⊥ |
| EuropeanAssociate | ≡ ∃Associate.European |
| AmericanAssociate | ≡ ∃Associate.American |
| NoFellowCompany | ≡ ∀Associate.¬American |
| InternationalCompany | ≡ EuropeanAssociate ∪ AmericanAssociate |

*A-Box:*

InternationalCompany(b)

**Ref:** A.Levy and M C.Rousset.CARIN:A Representation Language Combining Horn rules and Description Logics. Artificial Intelligence 104(1 2):165 –209, 1998.

# Motivating example

**Rule component $\Pi$:**

$r_1$: price-in-usa(X,high) ←
made-by(X,Y),
NoFellowCompany(Y).

$r_2$: price-in-usa(X,high) ←
made-by(X,Y),
Associate(Y,Z),
American(Z),
monopoly-in-usa(Y,X).

$r_3$: made-by(a,b).
$r_4$: monopoly-in-usa(b,a).

**DL component $\Sigma$:**

*T-Box:*

| | |
|---|---|
| European ∩ American | ⊆ ⊥ |
| EuropeanAssociate | ≡ ∃Associate.European |
| AmericanAssociate | ≡ ∃Associate.American |
| NoFellowCompany | ≡ ∀Associate.¬American |
| InternationalCompany | ≡ EuropeanAssociate ∪ AmericanAssociate |

*A-Box:*

InternationalCompany(b)

Constraining the extent of the head predicate in models of the rule-base

With constraint domain

# Motivating example

- $\prod \cup \sum$ |= price-in-usa(a,high) ?

**Rule component $\prod$:**

$r_1$: price-in-usa(X,high) ←
made-by(X,Y),
NoFellowCompany(Y).

$r_2$: price-in-usa(X,high) ←
made-by(X,Y),
Associate(Y,Z),
American(Z),
monopoly-in-usa(Y,X).

$r_3$: made-by(a,b).
$r_4$: monopoly-in-usa(b,a).

**DL component $\sum$:**

*T-Box:*

| | |
|---|---|
| European ∩ American | ⊆ ⊥ |
| EuropeanAssociate | ≡ ∃Associate.European |
| AmericanAssociate | ≡ ∃Associate.American |
| NoFellowCompany | ≡ ∀Associate.¬American |
| InternationalCompany | ≡ EuropeanAssociate ∪ AmericanAssociate |

*A-Box:*

InternationalCompany(b)

# Motivating example

- $\Pi \cup \Sigma \models$ price-in-usa(a,high) ?

**Rule component $\Pi$:**

$r_1$: price-in-usa(**a**,high) ←
made-by(**a**,**b**),
NoFellowCompany(**b**).

$r_2$: price-in-usa(X,high) ←
made-by(X,Y),
Associate(Y,Z),
American(Z),
monopoly-in-usa(Y,X).

$r_3$: made-by(a,b).

$r_4$: monopoly-in-usa(b,a).

**DL component $\Sigma$:**

*T-Box:*

European ⊓ American ⊑ ⊥
EuropeanAssociate ≡ ∃Associate.European
AmericanAssociate ≡ ∃Associate.American
NoFellowCompany ≡ ¬∃Associate.American
InternationalCompany ≡ EuropeanAssociate ⊔ AmericanAssociate

*A-Box:*

InternationalCompany(b)

**$\Sigma \not\models$ NoFellowCompany(b)**

# Motivating example

- $\prod \cup \sum$ |= price-in-usa(a,high) ?

**Rule component $\prod$:**

$r_1$: price-in-usa(X,high) ←
made-by(X,Y),
NoFellowCompany(Y).

$r_2$: price-in-usa(**a**,high) ←
made-by(**a**,**b**),
Associate(**b**,**_Z**),
American(**_Z**),
monopoly-in-usa(**b**,**a**).

$r_3$: made-by(a,b).
$r_4$: monopoly-in-usa(b,a).

**DL component $\sum$:**

*T-Box:*

European ∩ American ⊆ ⊥
EuropeanAssociate ≡ ∃Associate.European
AmericanAssociate ≡ ∃Associate.American
NoFellowCompany ≡ ¬∃Associate.(European ∪ American)
InternationalCompany ≡ ∃Associate.(European ∪ American)

$\sum \not\models \exists$(**Associate(b,_Z) ∧ American(_Z)**)

*A-Box:*

InternationalCompany(b)

# Motivating example

- $\Pi \cup \Sigma \models$ price-in-usa(a,high) ?

**Rule component $\Pi$:**

$r_1$: price-in-usa(**a**,high) ←
made-by(**a**,**b**),
NoFellowCompany(**b**).

$r_2$: price-in-usa(**a**,high) ←
made-by(**a**,**b**),
Associate(**b**,**_Z**),
American(**_Z**),
monopoly-in-usa(**b**,**a**).

$r_3$: made-by(a,b).
$r_4$: monopoly-in-usa(b,a).

**DL component $\Sigma$:**

*T-Box:*

European ∩ American ⊑ ⊥
EuropeanAssociate ≡ ∃Associate.European
AmericanAssociate ≡ ∃Associate.American
NoFellowCompany ≡ ∀Associate.¬American
InternationalCompany ≡ EuropeanAssociate ∪ AmericanAssociate

*A-Box:*

InternationalCompany(b)

But:

**$\Sigma \models$ NoFellowCompany(b)**

**∨**

**∃(Associate(b, _Z) ∧ American(_Z))**

# Motivating example

- $\Pi \cup \Sigma \models$ price-in-usa(a,high)

**Rule component $\Pi$:**

$r_1$: price-in-usa(**a**,high) ←
made-by(**a**,**b**),
NoFellowCompany(**b**).

$r_2$: price-in-usa(**a**,high) ←
made-by(**a**,**b**),
Associate(**b**,**_Z**),
American(**_Z**),
monopoly-in-usa(**b**,**a**).

$r_3$: made-by(a,b).
$r_4$: monopoly-in-usa(b,a).

**DL component $\Sigma$:**

*T-Box:*

European ⊓ American ⊑ ⊥
EuropeanAssociate ≡ ∃Associate.European
AmericanAssociate ≡ ∃Associate.American
NoFellowCompany ≡ ∀Associate.¬American
InternationalCompany ≡ EuropeanAssociate ⊔ AmericanAssociate

*A-Box:*

InternationalCompany(b)

Thus:

**$\Pi \cup \Sigma \models$ price-in-usa(a, high)**

# Rules we consider

**HEAD ← BODY**

- HEAD is some basic construct (*atom*)
- BODY is a set of atoms
- **Safety**: head variables appear in the body
- Examples:
  - Datalog: atomic formulae
  - Xcerpt: *Query terms* and *Construct terms*

# Semantics of rules

- Fixpoint semantics
  - Rules derive ground atoms from given ground atoms
    - *matching* of body atoms vs. given atoms gives substitution $\theta$
    - $\theta$ applied to head ⇒ derived atom

$T_P(S) = \{ H\theta \mid (H \leftarrow B_1, ..., B_n) \in P$ and
$(B_1, ..., B_n)$ *matches* some $A_1, ..., A_n$ in S with result $\theta$ $\}$

  - $T_P$ monotonic, $T_P(S) \subseteq T_P(S')$ for any $S \subseteq S'$
  - **Semantics of program P**: least fixpoint of $T_P$

# Examples of rule languages

- The class includes:
  - Logical rule languages, e.g.
    - **Datalog** (without negation)
    - Sematics of program: set of Datalog atoms
    - least Herbrand model
  - Rule languages lacking logical semantics, e.g
    - **Xcerpt** (negation-free subset)
    - Semantics of program: set of Xcerpt data terms

# Extended rules

HEAD ← BODY,**C**

- **C** formula of an ***external theory*** in logical language ***L***
- Ground atoms associated with a constraint
  - **A;C** where **A** ground atom, **C** formula of ***L***
- Extend $T_P$ operator

$$T_P(S) = \{ H\theta; (\mathbf{C}\boldsymbol{\theta} \wedge C_1 \wedge \ldots \wedge C_n) \mid (H \leftarrow B_1, \ldots, B_n, \mathbf{C}) \in P \text{ and for some } A_1;C_1, \ldots, A_n;C_n \text{ in } S \ (B_1, \ldots, B_n) \textit{ matches } A_1, \ldots, A_n \text{ with result } \theta \ \}$$

# Semantics of extended rules

- Restrict model of underlying rule program
  - A *constraint C*, wrt. an external theory $\Sigma$, can be:
    1. True in all models of $\Sigma$ $(\Sigma \models C)$
    2. False in all models of $\Sigma$ $(\Sigma \models \neg C)$
    3. None of above:

       satisfiable, but false in some models of $\Sigma$

$$M(P) = \{ A \mid A \in lfp(T_P) \text{ and } \Sigma \models C_A \}$$

  - $C_A$ is the disjunction of all constraints of $A$

# Instance: Datalog + OWL-DL

- Restrictions:
  - Only OWL concepts
- Requirements

**(1)** Collect constraints from Datalog in XSB

**(2)** Solve disjunctive DL constraints in existing reasoner

# (1) Collecting constraints

- Existing rule reasoners not aware of "external" predicates
  - How re-use rule reasoners?
  - How collect constraints?
  - Must be solved specifically for each language and rule reasoner
  - Here: **Datalog** in **XSB**

$\Pi$

$r_1$: price-in-usa(X,high) ←
made-by(X,Y),
NoFellowCompany(Y).

$r_2$: price-in-usa(X,high) ←
made-by(X,Y),
Associate(Y,Z),
American(Z),
monopoly-in-usa(Y,X).

$r_3$: made-by(a,b).
$r_4$: monopoly-in-usa(b,a).

# (1) Collecting constraints

- Collecting constraints in XSB

Π

```
price-in-usa(X,high) :-
      made-by(X,Y),
      NoFellowCompany(Y).

price-in-usa(X,high) :-
      made-by(X,Y),
      Associate(Y,Z),
      American(Z),
      monopoly-in-usa(Y,X).

made-by(a,b).
monopoly-in-usa(b,a).
```

Π'

```
price-in-usa(X,high,[NoFellowCompany(Y)|A]) :-
      made-by(X,Y,A).

price-in-usa(X,high,[Associate(Y,Z),American(Z)|A]) :-
      made-by(X,Y,A1),
      monopoly-in-usa(Y,X,A2),
      append(A1,A2,A).

made-by(a,b,[]).
monopoly-in-usa(b,a,[]).
```

# (1) Collecting constraints

- Query ← *price-in-usa(a,high,**C**)* wrt. $\Pi$':

  **C** = [NoFellowCompany(b)]
  **C** = [Associate(b,_Z),American(_Z)]

$\Pi$'

```
price-in-usa(X,high,[NoFellowCompany(Y)|A]) :-
      made-by(X,Y,A).

price-in-usa(X,high,[Associate(Y,Z),
                     American(Z)|A]) :-
      made-by(X,Y,A1),
      monopoly-in-usa(Y,X,A2),
      append(A1,A2,A).

made-by(a,b,[]).
monopoly-in-usa(b,a,[]).
```

ground($\Pi$)

$r_1$: price-in-usa(**a**,high) ←
made-by(**a**,**b**),
NoFellowCompany(**b**).

$r_2$: price-in-usa(**a**,high) ←
made-by(**a**,**b**),
Associate(**b**,**_Z**),
American(**_Z**),
monopoly-in-usa(**b**,**a**).

$r_3$: made-by(a,b).
$r_4$: monopoly-in-usa(b,a).

# (2) Disjunctive DL constraints

- Conjunctive query languages available
  - RacerPro, DQLServer, KAON2, Pellet etc.
- Disjunctive:
  - Service not directly supported

$\Sigma \models \text{AmericanAssociate}(a) \vee \text{NoFellowCompany}(b)$

$\Rightarrow$

$\Sigma \cup \{ a : \neg\text{AmericanAssociate}, b : \neg\text{NoFellowCompany} \}$
unsatisfiable?

**Ref**: Horrocks, I, Sattler U. Tessaris S and Tobies S. Query containment using a DLR Abox. LTCS-Report 99-15, LuFG Theoretical Computer Science, RWTH Aachen, Germany.

# (2) Disjunctive DL constraints

- Disjunctions of conjunctive queries

  $\Sigma \models \text{NoFellowCompany}(a) \vee (\text{EuropeanAssociate}(b) \wedge \text{American}(b))$

  – DNF ⇒ CNF:

  $\Sigma \models (\text{NoFellowCompany}(a) \vee \text{EuropeanAssociate}(b)) \wedge (\text{NoFellowCompany}(a) \vee \text{American}(b))$

**(1)** $\Sigma \cup \{ a:\neg\text{NoFellowCompany}, b:\neg\text{EuropeanAssociate}\}$

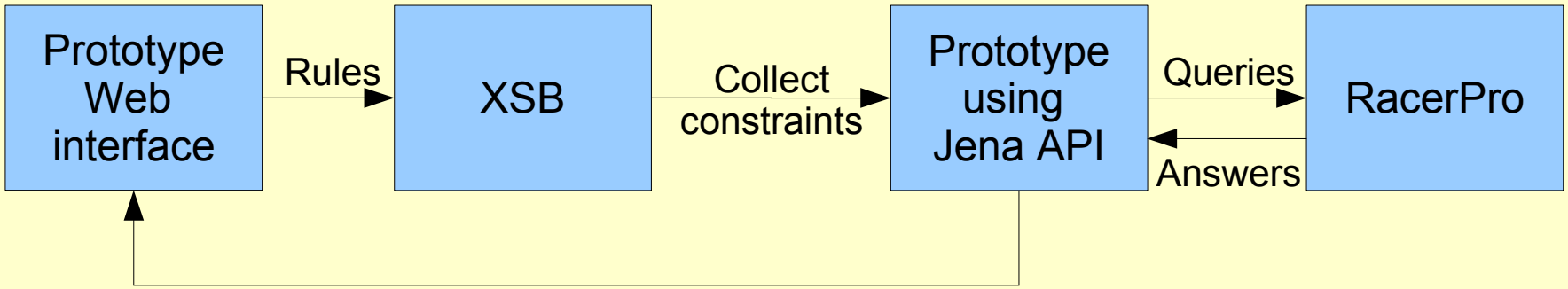
**(2)** $\Sigma \cup \{ a:\neg\text{NoFellowCompany}, b:\neg\text{American}\}$

  – Answer **"yes"** if **(1)** and **(2)** are unsatisfiable

**Ref**: Horrocks, I, Sattler U. Tessaris S and Tobies S. Query containment using a DLR Abox. LTCS-Report 99-15, LuFG Theoretical Computer Science, RWTH Aachen, Germany.

# Prototype

- Interfaces existing reasoners
  - Rule reasoner: **XSB**
  - Ontology reasoner: **DIG compliant DL reasoner**




  - Available at: **http://www.ida.liu.se/hswrl**
- Work in progress:
  - Allow roles in constraints through "*rolling-up*"

# Conclusions

- Combining general class of rules with constraints
  - Rules are negation-free, fixpoint semantics
- Non-logical rule languages
  - E.g. Xcerpt
- Re-using existing reasoners
- Prototype integration:
  - Datalog + OWL-DL
  - Using: XSB + RacerPro

# Related work

- Motivated by and extends $\mathcal{AL}$-Log
- ASP + DL [Eiter et. al.]
  - Negation
  - Bi-directional flow of information
- Safe hybrid KBs [Rosati]
  - Disjunctive Datalog
  - Ontological predicates in rule heads
- Different objectives from *language extensions*
  - E.g. SWRL [Horrocks et. al.], OWL-DL [Motik et. al.]

# Future work

- How re-use existing rule reasoners?
- Eager interaction
- Other constraint languages
- Rules with negation

# Combining safe rules and ontologies by interfacing of reasoners

Uwe Aßmann, Jakob Henriksson, Jan Małuszyński

PPSWR06, Budva, Montenegro, 10th June 2006

# The objective

- Define a scheme that
  from given
  - Rule language R (e.g. Datalog, Xcerpt)
  - Logical language S (e.g. OWL-DL, ...)

  constructs
  - A language $R_S$ integrating R and S:
    - Syntax, Semantics of $R_S$: from syntax and semantics of R and S
    - A (complete) reasoner for $R_S$
      by interfacing the reasoners of R and S

# Outline

- Motivating example
- The scheme
  - Principles and restrictions
  - An instance:
    - Datalog + OWL-DL
    - Prototype: interfacing XSB and a DL reasoner
- Conclusions
- Related work

# Motivating example

**Rule component $\prod$:**

$r_1$: price-in-usa(X,high) ←
made-by(X,Y),
NoFellowCompany(Y).

$r_2$: price-in-usa(X,high) ←
made-by(X,Y),
Associate(Y,Z),
American(Z),
monopoly-in-usa(Y,X).

$r_3$: made-by(a,b).
$r_4$: monopoly-in-usa(b,a).

**DL component $\sum$:**

*T-Box:*

| | |
|---|---|
| European ∩ American | ⊆ ⊥ |
| EuropeanAssociate | ≡ ∃Associate.European |
| AmericanAssociate | ≡ ∃Associate.American |
| NoFellowCompany | ≡ ∀Associate.¬American |
| InternationalCompany | ≡ EuropeanAssociate ∪ AmericanAssociate |

*A-Box:*

InternationalCompany(b)

**Ref:** A.Levy and M C.Rousset.CARIN:A Representation Language Combining Horn rules and Description Logics. Artificial Intelligence 104(1 2):165 –209, 1998.

# Motivating example

**Rule component $\Pi$:**

$r_1$: price-in-usa(X,high) ←
made-by(X,Y),
NoFellowCompany(Y).

$r_2$: price-in-usa(X,high) ←
made-by(X,Y),
Associate(Y,Z),
American(Z),
monopoly-in-usa(Y,X).

$r_3$: made-by(a,b).
$r_4$: monopoly-in-usa(b,a).

**DL component $\Sigma$:**

*T-Box:*

| | |
|---|---|
| European ∩ American | ⊆ ⊥ |
| EuropeanAssociate | ≡ ∃Associate.European |
| AmericanAssociate | ≡ ∃Associate.American |
| NoFellowCompany | ≡ ∀Associate.¬American |
| InternationalCompany | ≡ EuropeanAssociate ∪ AmericanAssociate |

*A-Box:*

InternationalCompany(b)

Constraining the extent of the head predicate in models of the rule-base

With constraint domain

# Motivating example

- $\prod \cup \sum$ |= price-in-usa(a,high) ?

**Rule component $\prod$:**

$r_1$: price-in-usa(X,high) ←
made-by(X,Y),
NoFellowCompany(Y).

$r_2$: price-in-usa(X,high) ←
made-by(X,Y),
Associate(Y,Z),
American(Z),
monopoly-in-usa(Y,X).

$r_3$: made-by(a,b).
$r_4$: monopoly-in-usa(b,a).

**DL component $\sum$:**

*T-Box:*

| | |
|---|---|
| European ∩ American | ⊆ ⊥ |
| EuropeanAssociate | ≡ ∃Associate.European |
| AmericanAssociate | ≡ ∃Associate.American |
| NoFellowCompany | ≡ ∀Associate.¬American |
| InternationalCompany | ≡ EuropeanAssociate ∪ AmericanAssociate |

*A-Box:*

InternationalCompany(b)

# Motivating example

- $\prod \cup \sum$ |= price-in-usa(a,high) ?

**Rule component $\prod$:**

$r_1$: price-in-usa(**a**,high) ←
made-by(**a**,**b**),
NoFellowCompany(**b**).

$r_2$: price-in-usa(X,high) ←
made-by(X,Y),
Associate(Y,Z),
American(Z),
monopoly-in-usa(Y,X).

$r_3$: made-by(a,b).
$r_4$: monopoly-in-usa(b,a).

**DL component $\sum$:**

*T-Box:*

European ⊓ American ⊑ ⊥
EuropeanAssociate ≡ ∃Associate.European
AmericanAssociate ≡ ∃Associate.American
NoFellowCompany ≡ ¬∃Associate.American
InternationalCompany ≡ EuropeanAssociate ⊔ AmericanAssociate

*A-Box:*

InternationalCompany(b)

**$\sum \not\models$ NoFellowCompany(b)**

# Motivating example

- $\prod \cup \sum$ |= price-in-usa(a,high) ?

**Rule component $\prod$:**

$r_1$: price-in-usa(X,high) ←
made-by(X,Y),
NoFellowCompany(Y).

$r_2$: price-in-usa(**a**,high) ←
made-by(**a**,**b**),
Associate(**b**,**_Z**),
American(**_Z**),
monopoly-in-usa(**b**,**a**).

$r_3$: made-by(a,b).
$r_4$: monopoly-in-usa(b,a).

**DL component $\sum$:**

*T-Box*:

European ⊓ American ⊑ ⊥
EuropeanAssociate ≡ ∃Associate.European
AmericanAssociate ≡ ∃Associate.American
NoFellowCompany ≡ ¬∃Associate.American
InternationalCompany ≡ ∃Associate ⊔ ∃Associate.Associate

$\sum \not\models \exists$(**Associate(b, _Z) ∧ American(_Z)**)

*A-Box*:

InternationalCompany(b)

# Motivating example

- $\prod \cup \sum$ |= price-in-usa(a,high) ?

**Rule component $\prod$:**

$r_1$: price-in-usa(**a**,high) ←
made-by(**a**,**b**),
NoFellowCompany(**b**).

$r_2$: price-in-usa(**a**,high) ←
made-by(**a**,**b**),
Associate(**b**,**_Z**),
American(**_Z**),
monopoly-in-usa(**b**,**a**).

$r_3$: made-by(a,b).
$r_4$: monopoly-in-usa(b,a).

**DL component $\sum$:**

*T-Box:*

European ⊓ American ⊑ ⊥
European ⊑ ¬American
AmericanAssociate ≡ ∃Associate.American
NoFellowCompany ≡ ∀Associate.¬American
International ≡ ∀Associate.American ⊔ ∀Associate.∃Associate

*A-Box:*

InternationalCompany(b)

But:

**$\sum$ |= NoFellowCompany(b)**

**∨**

**∃(Associate(b, _Z) ∧ American(_Z))**

# Motivating example

- $\prod \cup \sum$ |= price-in-usa(a,high)

**Rule component $\prod$:**

$r_1$: price-in-usa(**a**,high) ←
made-by(**a**,**b**),
NoFellowCompany(**b**).

$r_2$: price-in-usa(**a**,high) ←
made-by(**a**,**b**),
Associate(**b**,**_Z**),
American(**_Z**),
monopoly-in-usa(**b**,**a**).

$r_3$: made-by(a,b).
$r_4$: monopoly-in-usa(b,a).

**DL component $\sum$:**

*T-Box:*

European ⊓ American ⊑ ⊥
NoFellowCompany ≡ ∀Associate.¬American
InternationalCcompany ≡ EuropeanAssociate ⊔ AmericanAssociate

*A-Box:*

InternationalCompany(b)

Thus:

$\prod \cup \sum$ |= price-in-usa(a, high)

# Rules we consider

**HEAD ← BODY**

- HEAD is some basic construct (*atom*)
- BODY is a set of atoms
- **Safety**: head variables appear in the body
- Examples:
  - Datalog: atomic formulae
  - Xcerpt: *Query terms* and *Construct terms*

## Semantics of rules

- Fixpoint semantics
  - Rules derive ground atoms from given ground atoms
    - *matching* of body atoms vs. given atoms gives substitution $\theta$
    - $\theta$ applied to head ⇒ derived atom

$T_P(S) = \{ H\theta \mid (H \leftarrow B_1, ..., B_n) \in P$ and $(B_1, ..., B_n)$ *matches* some $A_1, ..., A_n$ in S with result $\theta \}$

  - $T_P$ monotonic, $T_P(S) \subseteq T_P(S')$ for any $S \subseteq S'$
  - **Semantics of program P**: least fixpoint of $T_P$



1. Matching depends on the rule language

# Examples of rule languages

- The class includes:
  - Logical rule languages, e.g.
    - **Datalog** (without negation)
    - Sematics of program: set of Datalog atoms
    - least Herbrand model
  - Rule languages lacking logical semantics, e.g
    - **Xcerpt** (negation-free subset)
    - Semantics of program: set of Xcerpt data terms

# Extended rules

HEAD ← BODY,**C**

- **C** formula of an ***external theory*** in logical language ***L***
- Ground atoms associated with a constraint
  - **A;C** where **A** ground atom, **C** formula of ***L***
- Extend $T_P$ operator

$T_P(S) = \{ H\theta; (\mathbf{C}\theta \wedge C_1 \wedge \ldots \wedge C_n) \mid (H \leftarrow B_1, \ldots, B_n, \mathbf{C}) \in P$ and
for some $A_1;C_1, \ldots, A_n;C_n$ in S
$(B_1, \ldots, B_n)$ *matches* $A_1, \ldots, A_n$ with result $\theta$ $\}$

## Semantics of extended rules

- Restrict model of underlying rule program
  - A *constraint C*, wrt. an external theory $\Sigma$, can be:
    1. True in all models of $\Sigma$ $(\Sigma \models C)$
    2. False in all models of $\Sigma$ $(\Sigma \models \neg C)$
    3. None of above:
       satisfiable, but false in some models of $\Sigma$

    $M(P) = \{ A \mid A \in lfp(T_P) \text{ and } \Sigma \models C_A \}$
  - $C_A$ is the disjunction of all constraints of A



1. Refer to introductory example

# Instance: Datalog + OWL-DL

- Restrictions:
  - Only OWL concepts
- Requirements

  **(1)** Collect constraints from Datalog in XSB

  **(2)** Solve disjunctive DL constraints in existing reasoner

# (1) Collecting constraints

- Existing rule reasoners not aware of "external" predicates
  - How re-use rule reasoners?
  - How collect constraints?
  - Must be solved specifically for each language and rule reasoner
  - Here: **Datalog** in **XSB**

$\Pi$

$r_1$: price-in-usa(X,high) ←
made-by(X,Y),
NoFellowCompany(Y).

$r_2$: price-in-usa(X,high) ←
made-by(X,Y),
Associate(Y,Z),
American(Z),
monopoly-in-usa(Y,X).

$r_3$: made-by(a,b).
$r_4$: monopoly-in-usa(b,a).

# (1) Collecting constraints

- Collecting constraints in XSB

Π

```
price-in-usa(X,high) :-
      made-by(X,Y),
      NoFellowCompany(Y).

price-in-usa(X,high) :-
      made-by(X,Y),
      Associate(Y,Z),
      American(Z),
      monopoly-in-usa(Y,X).

made-by(a,b).
monopoly-in-usa(b,a).
```

Π'

```
price-in-usa(X,high,[NoFellowCompany(Y)|A]) :-
      made-by(X,Y,A).

price-in-usa(X,high,[Associate(Y,Z),American(Z)|A]) :-
      made-by(X,Y,A1),
      monopoly-in-usa(Y,X,A2),
      append(A1,A2,A).

made-by(a,b,[]).
monopoly-in-usa(b,a,[]).
```

# (1) Collecting constraints

- Query ← *price-in-usa(a,high,**C**)* wrt. $\Pi$':

  **C** = [NoFellowCompany(b)]
  **C** = [Associate(b,_Z),American(_Z)]

$\Pi$'

```
price-in-usa(X,high,[NoFellowCompany(Y)|A]) :-
    made-by(X,Y,A).

price-in-usa(X,high,[Associate(Y,Z),
                     American(Z)|A]) :-
    made-by(X,Y,A1),
    monopoly-in-usa(Y,X,A2),
    append(A1,A2,A).

made-by(a,b,[]).
monopoly-in-usa(b,a,[]).
```

ground($\Pi$)

$r_1$: price-in-usa(**a**,high) ←
    made-by(**a**,**b**),
    NoFellowCompany(**b**).

$r_2$: price-in-usa(**a**,high) ←
    made-by(**a**,**b**),
    Associate(**b**,**_Z**),
    American(**_Z**),
    monopoly-in-usa(**b**,**a**).

$r_3$: made-by(a,b).
$r_4$: monopoly-in-usa(b,a).

# (2) Disjunctive DL constraints

- Conjunctive query languages available
  - RacerPro, DQLServer, KAON2, Pellet etc.
- Disjunctive:
  - Service not directly supported

$\Sigma \models AmericanAssociate(a) \vee NoFellowCompany(b)$

$\Rightarrow$

$\Sigma \cup \{ a : \neg AmericanAssociate, b : \neg NoFellowCompany \}$
unsatisfiable?

**Ref**: Horrocks, I, Sattler U. Tessaris S and Tobies S. Query containment using a DLR Abox. LTCS-Report 99-15, LuFG Theoretical Computer Science, RWTH Aachen, Germany.

# (2) Disjunctive DL constraints

- Disjunctions of conjunctive queries

  $\Sigma \models \text{NoFellowCompany}(a) \vee (\text{EuropeanAssociate}(b) \wedge \text{American}(b))$

  - DNF $\Rightarrow$ CNF:

  $\Sigma \models (\text{NoFellowCompany}(a) \vee \text{EuropeanAssociate}(b)) \wedge (\text{NoFellowCompany}(a) \vee \text{American}(b))$

**(1)** $\Sigma \cup \{ a:\neg\text{NoFellowCompany}, b:\neg\text{EuropeanAssociate}\}$

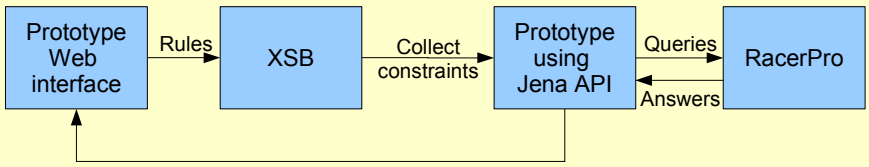
**(2)** $\Sigma \cup \{ a:\neg\text{NoFellowCompany}, b:\neg\text{American}\}$

- Answer **"yes"** if **(1)** and **(2)** are unsatisfiable

**Ref**: Horrocks, I, Sattler U. Tessaris S and Tobies S. Query containment using a DLR Abox. LTCS-Report 99-15, LuFG Theoretical Computer Science, RWTH Aachen, Germany.

# Prototype

- Interfaces existing reasoners
  - Rule reasoner: **XSB**
  - Ontology reasoner: **DIG compliant DL reasoner**




    - Available at: **http://www.ida.liu.se/hswrl**
- Work in progress:
  - Allow roles in constraints through "*rolling-up*"

# Conclusions

- Combining general class of rules with constraints
  - Rules are negation-free, fixpoint semantics
- Non-logical rule languages
  - E.g. Xcerpt
- Re-using existing reasoners
- Prototype integration:
  - Datalog + OWL-DL
  - Using: XSB + RacerPro

# Related work

- Motivated by and extends $\mathcal{AL}$-Log
- ASP + DL [Eiter et. al.]
  - Negation
  - Bi-directional flow of information
- Safe hybrid KBs [Rosati]
  - Disjunctive Datalog
  - Ontological predicates in rule heads
- Different objectives from *language extensions*
  - E.g. SWRL [Horrocks et. al.], OWL-DL [Motik et. al.]

# Future work

- How re-use existing rule reasoners?
- Eager interaction
- Other constraint languages
- Rules with negation