Towards a RIF(RAF) Ontology

Axel Polleres

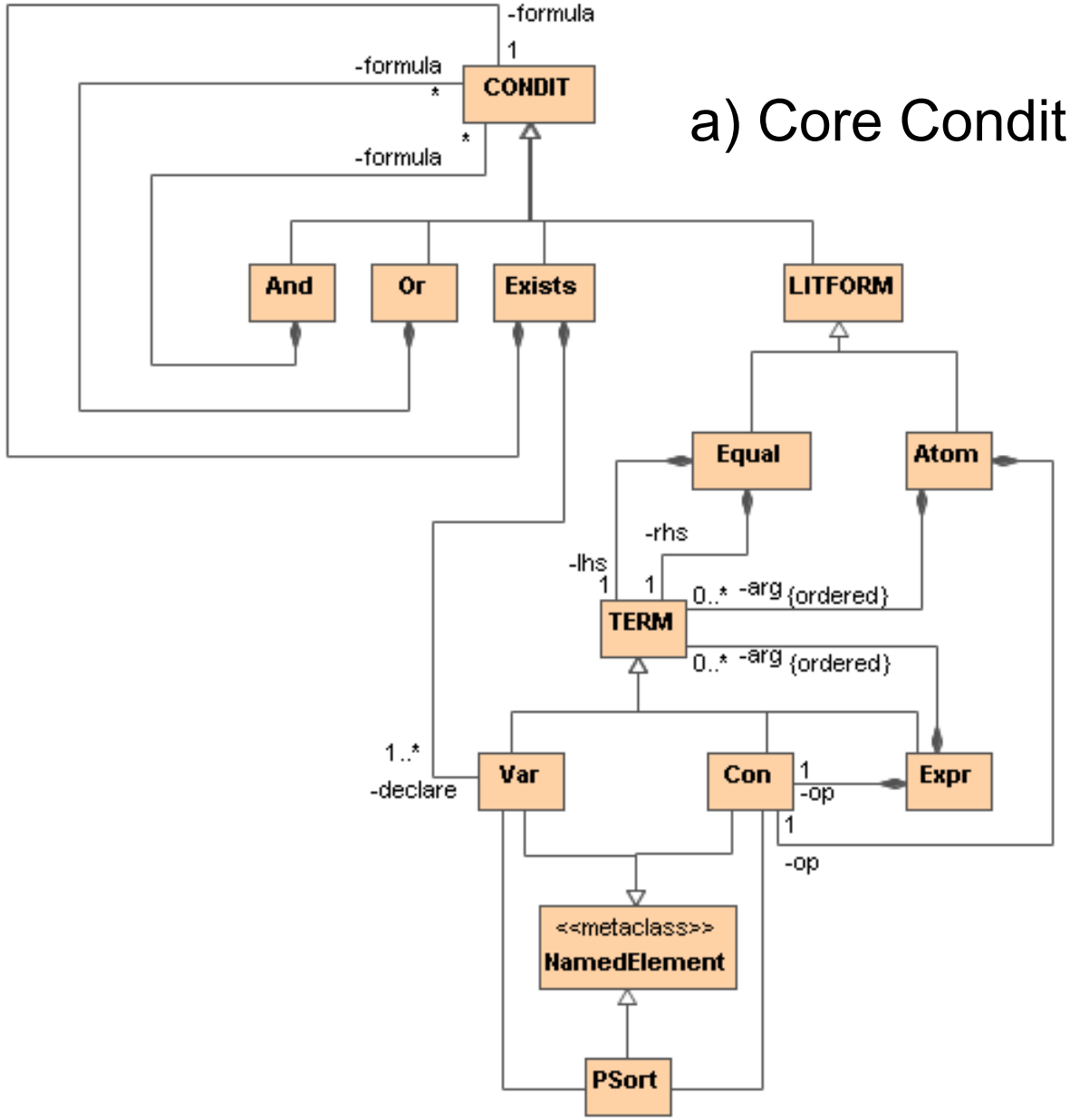
Why an Ontology for RIFRAF

- Many discriminators in the RAF questionnaire
- Not clear how to proceed on that
- Idea/approach form last F2F:
  - “Ontologize” the questionnaire
  - and thereby formalize the discriminators
- Goal: Get a classification scheme for rule systems and supported RIF dialects, ie.:
  - Given a ruleset R, determine whether it falls into RIF dialect D, or whether it can be processed by rulesystem S.

Base Ontology

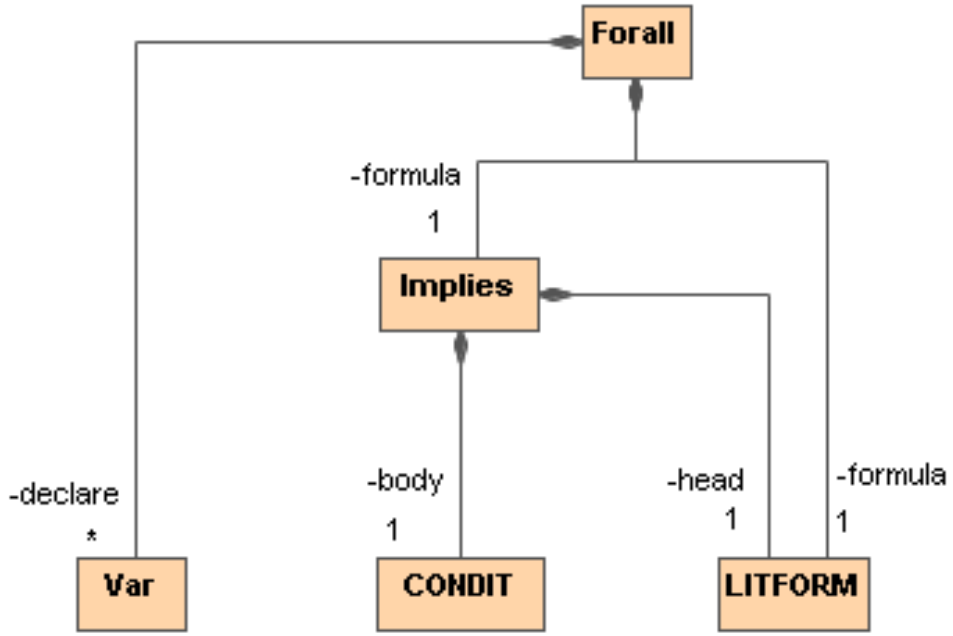
- Basic Ingredients very similar to the concepts and properties discussed for thw RIF Core Metamodel, Discriminators talk about:
  - Condition, Action, Event
  - Rule
  - Ruleset
  - RuleSystem
  - etc.
- So, let's start with that…

Base Ontology vs RIF Core Metamodel



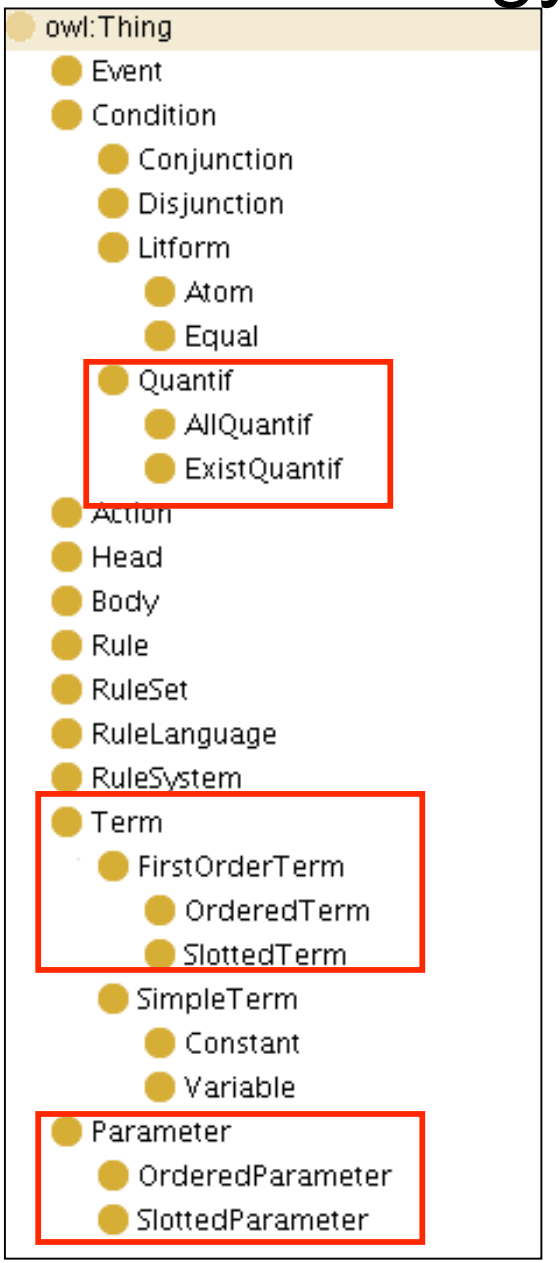
a) Core Condition metamodel

Base Ontology vs RIF Core Metamodel



b) Core Rule metamodel

Base Ontology - Classes and Properties:



Find latest version at:
http://lists.w3.org/Archives/Public/public-rif-wg/2007Feb/att-0082/rifraf20070221.owl

Base Ontology - Differences to RIF Core Metamodel:

- Caters for slotted and ordered Terms.
- Functors of Terms can be Terms again (e.g. HiLog, not catered for in basic model so far)
- No distinction between term and expression on ontological level
- Pure OWL ontology likely not enough for formalization?
  - we will need integrity constraints/CWA to restrict certain dialects/rulesets, e.g. to express: all rules in a PROPOSITIONAL ruleset contain only propositional atoms.
  - Non-DL expressible axioms necessary?
- (How deep) do we want to go into this exercise?
- What's the benefit?

Formalizing/Testing Discriminators on rulesets - Examples

1 Syn: Syntactic Discriminators

1.1 Restricted vs. Unrestricted Use of Logic Variables

1.1.1 Single-occurrence variables (no implicit variable equality) vs. Multiple-occurrence variables (implicit variable equality)

- **Multiple variable occurrences per predicate allowed:**
- **Only single occurrence of variables per predicate:**

```
recHasCondition(X,C1) :- hasCondition(X,C).
recHasCondition(X,C1) :- hasCondition(X,C0), hasCondition(C,C1).

MULTIPLEVAROCCURR     :-    ruleset(RS), hasRule(RS,R), hasBody(R,B),
                            recHasCondition(B,C), Atom(C), hasTerm(C,T),
                            recHasParameter(T,P1), recHasParameter(T,P2), P1 != P2,
                            hasTerm(P1,V), hasTerm(P2,V), Variable(V).
```

- Equality test predicate available

```
EQUALTESTOCCUR     :-    ruleset(RS), hasRule(RS,R), hasBody(R,B),
                         recHasCondition(B,C), Equal(C).
```

Forgive me to have used prolog syntax here, replace it by whatever rule language you like. What I want to show here is that using e.g. a logic program and the vocabulary defined in the RIFRAF Base Ontology, I can test the disrciminators on a given ruleset.

- 1.1.2 Range-restricted Rules (No Head-Only Variables) vs. non-Range-restricted Rules.
  - Head-only variables allowed
  - Head variables must occur in any form in the body/contition

```
recHasCondition(X,C1) :- hasCondition(X,C).
recHasCondition(X,C1) :- hasCondition(X,C0), hasCondition(C,C1).

HAEDONLYVAROCCURR     :-    ruleset(RS), hasRule(RS,R), headVar(V,R), naf bodyVar(V,R).

bodyVar(V,R) :-  hasBody(R,B), recHasCondition(B,C), Atom(C), hasTerm(C,T),
                 recHasParameter(T,P), hasTerm(P,V), Variable(V).

headVar(V,R) :-  hasBody(R,H), recHasCondition(H,C), Atom(C), hasTerm(C,T),
                 recHasParameter(T,P), hasTerm(P,V), Variable(V).
```

would need slight modification, if Action part was also considered.

  - Head variables must occur positively in the body/contition (safety):

    need to extend base ontology for Litforms, kinds of negation, etc. … Phase 2, I'd say, I'd rather wait for the respective suggestions in the RIF metamodel, or resp. we develop it further jointly!

etc. …
Summary:

- Exercise for formalizing the discriminators is not complete, I just started in a one shot attempt, but:

Bottom-line:

- Many of the dscriminators can be formalized by similar logic programs/rules over the proposed base ontology.

- Important: I'd appreciate some people to join, if this is of interest at all to the WG (Is it? Does it make sense?)

- Stable RIF Core Ontology and agreement on the Base ontology is a prerequisite. **I do not see much difference between the base ontology and the RIF Metamodel, actually it should be the same.**