DAMLJessKB/OWLJessKB:
Tools for Reasoning with the Semantic Web

Joseph B. Kopena  
joe@plan.cs.drexel.edu

William C. Regli  
regli@drexel.edu

October, 2003

http://gicl.cs.drexel.edu/
Motivating Application: Design Repositories

Engineering firms maintain large collections of complex **design knowledge**

- **Part/Design Libraries**
  - Used in: assembly design, variant design, process/analysis re-use

- **Design Databases**
  - Used in: collaborative design, life-cycle management (e.g. maintenance, disassembly)

- **Legacy Data**
  - Weakly organized (if at all)

**Design repositories**—an evolution of traditional design databases

- **Capture** and **utilize** more design knowledge
- **Enable enhanced services**
Knowledge-based repository process:

- **Input Artifacts**
- **Extract Semantics**
- **Build Knowledge Base**
- **Categorize/Browse**
- **Compare/Search**

Major **services** of interest that a repository may provide:

- **Search**: Finding designs that meet criteria
- **Classification**: Placing designs into a hierarchy
- **Categorization**: Inducing a hierarchy automatically
Function and flow signatures: DL-oriented function modeling approach

Simple DL core ontology (~13 statements)
- Function, flow, and artifact classes
- Relations between them

Non-conservative descriptive extensions
- ~75 functions, ~50 flows
- Based on Functional Basis/NIST SBF
- Conservatively domain extensible

Break-Beam Encoder

F & F Signature

http://edge.cs.drexel.edu/assemblies/
Reasoning: DAMLJessKB (OWLJessKB)

DAMLJessKB: expert system-based description logic reasoner

- RDF XML documents are converted to triples in KB
- Rules in KB implement RDF(S), XSD, and DAML semantics
- Applications query for information/apply application-specific reasoning
Simple Example: `rdfs:subClassOf`

Necessary implication entailed by **RDF-S semantics:**

\[ \forall a, b, i \left( \text{rdfs:subClassOf } a \ b \right) \land \left( \text{rdf:type } i \ a \right) \supset \left( \text{rdf:type } i \ b \right) \]

**Implementation in Jess’ rule language:**

```jess
(defrule rdfs-subclass-instances
    "An instance of a subclass is an instance of the parent class."

    (triple (predicate "http://www.w3.org/2000/01/rdf-schema#subClassOf")
        (subject ?child)
        (object ?parent))
    (triple (predicate "http://www.w3.org/1999/02/22-rdf-syntax-ns#type")
        (subject ?instance)
        (object ?child))

    =>
    (assert (triple (predicate "http://www.w3.org/1999/02/22-rdf-syntax-ns#type")
        (subject ?instance)
        (object ?parent)))
)
```

*(this rule from **OWLJessKB**)*
More Complicated Inference: Intersection Subsumption

Conjunctive class description subsumption inference:

(deffct intersection-of-subsumption
    "Implements subsumption between classes defined by intersectionOf elements."
    (declare (salience -50))

    (triple (predicate "http://www.w3.org/2002/07/owl#intersectionOf")
        (subject ?topClass) (object ?topList))
    (triple (predicate "http://www.w3.org/2002/07/owl#intersectionOf")
        (subject ?botClass&~?topClass) (object ?botList))

; There does not exist a member of ?topList which does not either
; appear in ?botlist or have a subclass in ?botList.
    (not (and (list-item ?topList ?y)
        (not (or (list-item ?botList ?y)
            (and (list-item ?botList ?x)
                (triple
                    (predicate "http://www.w3.org/2000/01/rdf-schema#subClassOf")
                    (subject ?x) (object ?y)))))))))

=> (assert (triple (predicate "http://www.w3.org/2000/01/rdf-schema#subClassOf")
    (subject ?botClass) (object ?topClass))))

(this rule from OWLJessKB)
Datatype Reasoning

**Literals** are encapsulated in `daml:Datatype` objects

- Enables a **consistent interface** for rules, querying, typing

**XSD Datatypes** are encoded in RDF as subclasses of `rdfs:class`

- Enables **classification** of literals; **subsumption** reasoning, e.g.:

```
(defrule mininclusive-subclassing
  (PropertyValue http://www.w3.org/1999/02/22-rdf-syntax-ns#type
    ?dt1 http://www.daml.org/2001/03/daml+oil#Datatype)
  (PropertyValue http://www.w3.org/2000/10/XMLSchema#minInclusive ?dt1 ?anon1)
  (PropertyValue http://www.w3.org/1999/02/22-rdf-syntax-ns#value ?anon1 ?value1)

  (PropertyValue http://www.w3.org/1999/02/22-rdf-syntax-ns#type
  (PropertyValue http://www.w3.org/2000/10/XMLSchema#minInclusive ?dt2 ?anon2)
  (PropertyValue http://www.w3.org/1999/02/22-rdf-syntax-ns#value ?anon2 ?value2)

  (test (and (integerp ?value1) (integerp ?value2) (>= ?value1 ?value2)))
  =>
  (assert (PropertyValue http://www.w3.org/2000/01/rdf-schema#subClassOf ?dt1 ?dt2)))
```

(this rule from DAMLJessKB)
Analysis—Theoretical

Major properties:

- **Sound**—modulus closed world assumption
  - Often useful to assume/not require explicit closure in practice
  - Need to do thorough analysis/proof; empirical evidence for now
- **Incomplete**—handles most inferences likely to be encountered
  - Could be complete?
  - Need to strongly characterize coverage
  - **Configurability**—tailor to performance/reasoning needs
- **Complexity**—performance seems good, but…
  - What is Jess’ complexity? Are results for Rete still applicable?
  - Can bounds be made tighter for this reasoning?
- **Equivalence**—icky
  - Not a problem in terms of class, property extensions
  - Messy for intensional/instance equivalency, e.g. `owl:sameClassAs`, `owl:sameIndividualAs`
DAMLJessKB: useful, capable

- Handles most of the *commonly used inferences*
- **Performance** acceptable, better in OWLJessKB
- Java interface & Jess backend: *solid application framework*
- Open-source; supported & in active development

Jess: extremely powerful

- **Commercial quality**, government supported
- **Excellent Java interaction**; useful scripting language
- Free license for research; source released to licensees
- Very well supported & in **active development**
  - Upcoming: Improved API; debugging tools and rule editor

Comparison to “proper” DL reasoners, other OWL/DAML inference engines?
Summary

This talk has presented:

- **Design repositories**: Application of a wide range of DL inferences
- **DJKB/OJKB**: Practical, in use, Semantic Web reasoning tools

**Topics for future thought:**

- Database backend—one way to address scalability
- Configurable reasoning
  - What does that mean in terms of interoperability?
- Working with rules (e.g. OWLRules)
- Non-standard DL reasoning (msc, lcs)
In Closing...

Major support:

- **National Science Foundation (NSF)**
  Knowledge & Distributed Intelligence (KDI) Grant CISE/IIS-9873005
- **National Science Foundation (NSF) CAREER Award** CISE/IIS-9733545
- **Office of Naval Research (ONR) Grant** N00014-01-1-0618
- **National Institute of Standards and Technology (NIST)**
  Grant #70NAN33H1026 (funded by NSF)

More information:

- **Project homepage:** [http://edge.cs.drexel.edu/assemblies/](http://edge.cs.drexel.edu/assemblies/)
- **eMail:** joe@plan.cs.drexel.edu, regli@drexel.edu

Questions?