The Semantic Web: (Ontology) Languages and Reasoning

Ian Horrocks
horrocks@cs.man.ac.uk

University of Manchester
Manchester, UK
Semantic Web Ontology Languages

US DAML programme (in cooperation with W3C and a cast of thousands) aim to develop so-called Semantic Web

☞ Most existing Web resources only human understandable
  ● Markup (HTML) provides rendering information
  ● Textual/graphical information for human consumption

☞ Semantic Web aims at machine understandability
  ● Semantic markup will be added to web resources
  ● Markup will use Ontologies for shared understanding

☞ Requirement for a suitable ontology language
  ● Compatible with existing Web standards (XML, RDF, RDFS)
  ● Captures common KR idioms
  ● Formally specified and of adequate expressive power
  ● Can provide reasoning support

☞ DAML-ONT language developed to meet these requirements
OIL and DAML+OIL

Meanwhile, somewhere in darkest Europe...

- OIL language already developed to meet similar requirements
  - Extends existing Web standards (XML, RDF, RDFS)
  - Intuitive (frame) syntax plus high expressive power
  - Well defined semantics via mapping to SHIQ DL
  - Can use DL systems to reason with OIL ontologies

- Two efforts merged to produce single language, DAML+OIL

- Detailed specification agreed by Joint EU/US Committee on Agent Markup Languages

- Proposed W3C Ontology Language WG will take DAML+OIL as starting point (?)
DAML+OIL Language Overview

DAML+OIL is an ontology language

- Describes **structure** of the domain (i.e., a Tbox)
  - RDF used to describe specific **instances** (i.e., an Abox)
- Structure described in terms of **classes** (concepts) and **properties** (roles)
- Ontology consists of set of **axioms**
  - E.g., asserting class subsumption/equivalence
- Classes can be names or **expressions**
  - Various **constructors** provided for building class expressions
- **Expressive power** determined by
  - Kinds of axiom supported
  - Kinds of class (and property) constructor supported
### DAML+OIL Overview: Class Constructors

<table>
<thead>
<tr>
<th>Constructor</th>
<th>DL Syntax</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>intersectionOf</td>
<td>$C_1 \sqcap \cdots \sqcap C_n$</td>
<td>Human $\sqcap$ Male</td>
</tr>
<tr>
<td>unionOf</td>
<td>$C_1 \sqcup \cdots \sqcup C_n$</td>
<td>Doctor $\sqcup$ Lawyer</td>
</tr>
<tr>
<td>complementOf</td>
<td>$\neg C$</td>
<td>$\neg$Male</td>
</tr>
<tr>
<td>oneOf</td>
<td>${x_1 \ldots x_n}$</td>
<td>${john, mary}$</td>
</tr>
<tr>
<td>toClass</td>
<td>$\forall P.C$</td>
<td>$\forall$hasChild.Doctor</td>
</tr>
<tr>
<td>hasClass</td>
<td>$\exists P.C$</td>
<td>$\exists$hasChild.Lawyer</td>
</tr>
<tr>
<td>hasValue</td>
<td>$\exists P.{x}$</td>
<td>$\exists$citizenOf.{USA}</td>
</tr>
<tr>
<td>minCardinalityQ</td>
<td>$\geq n P.C$</td>
<td>$\geq 2$hasChild.Lawyer</td>
</tr>
<tr>
<td>maxCardinalityQ</td>
<td>$\leq n P.C$</td>
<td>$\leq 1$hasChild.Male</td>
</tr>
<tr>
<td>cardinalityQ</td>
<td>$= n P.C$</td>
<td>$= 1$hasParent.Female</td>
</tr>
</tbody>
</table>

- XMLS **datatypes** as well as classes
- Arbitrarily complex **nesting** of constructors
  - E.g., $\forall$hasChild.(Doctor $\sqcup$ $\exists$hasChild.Doctor)
### DAML+OIL Overview: Axioms

<table>
<thead>
<tr>
<th>Axiom</th>
<th>DL Syntax</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>subClassOf</td>
<td>$C_1 \sqsubseteq C_2$</td>
<td>Human $\sqsubseteq$ Animal $\sqcap$ Biped</td>
</tr>
<tr>
<td>sameClassAs</td>
<td>$C_1 \equiv C_2$</td>
<td>Man $\equiv$ Animal $\sqcap$ Male</td>
</tr>
<tr>
<td>subPropertyOf</td>
<td>$P_1 \sqsubseteq P_2$</td>
<td>hasDaughter $\sqsubseteq$ hasChild</td>
</tr>
<tr>
<td>samePropertyAs</td>
<td>$P_1 \equiv P_2$</td>
<td>cost $\equiv$ price</td>
</tr>
<tr>
<td>sameIndividualAs</td>
<td>${x_1} \equiv {x_2}$</td>
<td>${\text{President}<em>{-}\text{Bush}} \equiv {\text{G}</em>{-}\text{W}_{-}\text{Bush}}$</td>
</tr>
<tr>
<td>disjointWith</td>
<td>$C_1 \sqsubseteq \neg C_2$</td>
<td>Male $\sqsubseteq \neg$Female</td>
</tr>
<tr>
<td>differentIndividualFrom</td>
<td>${x_1} \sqsubseteq \neg {x_2}$</td>
<td>${\text{john}} \sqsubseteq \neg {\text{peter}}$</td>
</tr>
<tr>
<td>inverseOf</td>
<td>$P_1 \equiv P_2^-$</td>
<td>hasChild $\equiv$ hasParent$^-$</td>
</tr>
<tr>
<td>transitiveProperty</td>
<td>$P^+ \sqsubseteq P$</td>
<td>ancestor$^+$ $\sqsubseteq$ ancestor</td>
</tr>
<tr>
<td>uniqueProperty</td>
<td>$\top \sqsubseteq \leq 1 P$</td>
<td>$\top \sqsubseteq \leq 1$hasMother</td>
</tr>
<tr>
<td>UnambiguousProperty</td>
<td>$\top \sqsubseteq \leq 1 P^-$</td>
<td>$\top \sqsubseteq \leq 1$isMotherOf$^-$</td>
</tr>
</tbody>
</table>

* Axioms (mostly) **reducible to subClass/PropertyOf**
DAML+OIL

- Is a Description Logic (but don’t tell anyone)
- More precisely, DAML+OIL is $SHIQ$
  - Plus nominals
  - Plus datatypes (simple concrete domains)
  - With RDFS based syntax
- $SHIQ$/DAML+OIL was not built in a day (or even a year)
  - $SHIQ$ is based on 15+ years of DL research
- Can use DL reasoning with DAML+OIL
  - Existing $SHIQ$ implementations support (most of) DAML+OIL
Why Reasoning Services?

Reasoning is important for:

☞ Ontology **design**
  - Check class consistency and (unexpected) implied relationships
  - Particularly important with large ontologies/multiple authors

☞ Ontology **integration**
  - Assert inter-ontology relationships
  - Reasoner computes integrated class hierarchy/consistency

☞ Ontology **deployment**
  - Determine if set of facts are consistent w.r.t. ontology
  - Determine if individuals are instances of ontology classes

“*The Semantic Web needs a logic on top*” (Henry Thompson)
Why Decidable Reasoning?

Set of operators/axioms restricted so that reasoning is **decidable**

☞ Consistent with Semantic Web’s **layered architecture**
  - XML provides syntax transport layer
  - RDF provides basic relational language
  - RDFS provides basic ontological primitives
  - DAML+OIL provides (decidable) logical layer
  - Further layers (e.g., **rules**) will extend DAML+OIL
    ➔ Extensions will almost certainly be **undecidable**

☞ Facilitates provision of **reasoning services**
  - Known algorithms
  - Implemented systems
  - Evidence of **empirical tractability**
Challenges

- Increased expressive power
  - Datatypes
  - Nominals
  - Extensions to DAML+OIL

- Performance (even of existing SHIQ implementations)
  - Inverse roles and qualified number restrictions
  - Very large KBs
  - Reasoning with individuals

- Tools and Infrastructure
  - Support for large scale ontological engineering and deployment

- New reasoning tasks
  - Querying
  - Lcs/matching
  - Sanctioning
  - …
Increased Expressive Power: Datatypes

DAML+OIL extends $SHIQ$ with datatypes and nominals

**Datatypes**

- DAML+OIL has simple form of datatypes
  - Unary predicates plus disjoint abstract/datatype domains
- Theoretically not particularly challenging
  - Existing work on concrete domains [Baader & Hanschke, Lutz]
  - Algorithm already known for $SHOQ(D)$ [Horrocks & Sattler]
- May be practically challenging
  - All XMLS datatypes supported
- Already seeing some (limited) implementations
  - Cerebra system (Network Inference)
  - RACER system (Hamburg)
Increased Expressive Power: Nominals

**Nominals**

- DAML+OIL has **oneOf** constructor
  - Extensionally defined concepts, e.g., $\{Mary\}^I = \{Mary\}$
  - Equivalent to nominals in modal logic
- Theoretically **very** challenging
  - Resulting logic has known high complexity (NExpTime)
  - No known “practical” algorithm
  - Not obvious how to extend tableau techniques in this direction
    - Loss of tree model property
    - Spy-points: $\top \subseteq \exists R. \{S_{p_y}\}$
    - Finite domains: $\{S_{p_y}\} \subseteq nR^-$

- Relatively straightforward (in theory) without **inverse roles**
  - Algorithm for $\mathcal{SHOQ}(\mathcal{D})$ deals with nominals
  - Practical implementation still to be demonstrated
Increased Expressive Power: Extensions

- DAML+OIL not expressive enough for all applications

Extensions wish list includes:
- Feature chain (path) agreement, e.g., output of component of composite process equals input of subsequent process
- Complex roles/role inclusions, e.g., a city located in part of a country is located in that country
- Rules—proposal(s) already exist for “datalog/LP style rules”
- Temporal and spatial reasoning
- …

May be impossible/undesirable to resist such extensions

Extended language sure to be undecidable

How can extensions best be integrated with DAML+OIL?

How can reasoners be developed/adapted for extended languages
- Some existing work on language fusions and hybrid reasoners
Performance Problems I

Evidence of empirical tractability mostly w.r.t. $SHF$— problems can arise when systems extended to $SHIQ$

- Trace technique no longer works
  - Whole model must be kept in memory
  - More costly state saving/restoring when searching non-deterministic expansions
  - More complex flow of control during expansion/search

- E.g., $\exists S. \neg C \land \exists R. A$ w.r.t. $T = \{ A \subseteq (\forall R^-. \forall S.C) \cup (\forall R^-. \forall S.D) \}$
Important optimisations no longer (fully) work

- Problems with caching as cached models can affect parent
- E.g., consider \( \forall R^-. \neg C \) and \( C \sqcap \exists R. \forall R^-. \neg C \)

\[ L(w) = \{ \forall R^-. \neg C \} \]

\[ L(x) = \{ \forall R^-. \neg C \} \]

\[ L(w) = \{ C, \exists R. \forall R^-. \neg C, \neg C \} \]

Clash

- Interactions with blocking even more problematical
- Similar problems with model merging
Performance Problems III

- Qualified number restrictions can also cause problems
  - Even relatively small numbers can mean significant non-determinism

- Reasoning with very large KBs
  - Web ontologies can be expected to grow very large

- Reasoning with individuals (Abox)
  - Deployment of web ontologies will mean reasoning with (possibly very large numbers of) individuals
  - Unlikely that standard Abox techniques will be able to cope
Performance Solutions (Maybe)

- Excessive memory usage
  - Problem exacerbated by over-cautious double blocking condition (e.g., root node can never block)
  - Promising results from more precise blocking condition [Sattler & Horrocks]

- Qualified number restrictions
  - Problem exacerbated by naive expansion rules
  - Promising results from optimised expansion using Algebraic Methods [Haarslev & Möller]

- Caching and merging
  - Can still work in some situations (work in progress)

- Reasoning with very large KBs
  - RACER system shown to work with $\approx 100k$ concept KB [Haarslev & Möller]
  - But KB only exploited small part of DL language
Tools and Infrastructure

Tools and infrastructure required in order support use of DAML+OIL

☞ Ontology design and maintenance
  ● Several editors available, e.g., OilEd (Manchester), OntoEdit (Karlsruhe), Protégé (Stanford)
  ● Need integrated environments including modularity, versioning, visualisation, explanation, high-level languages, . . .

☞ Ontology Integration
  ● Some tools available, e.g., Chimera (Stanford)
  ● Need integrated environments . . .
  ● Can learn from DB integration work [Lenzerini, Calvanese et al]

☞ Reasoning engines
  ● Several DL systems available
  ● Need for improved usability

☞ . . .
New Reasoning Tasks

☞ Querying
  ● Retrieval (instances of a concept) and realisation (most specific class of instance) won’t be sufficient
  ● Minimum requirement will be conjunctive query style language [Tessaris & Horrocks]
  ● May also need to answer “what can I say about \( x \)?” style of query [Bechhofer & Horrocks]

☞ Explanation (e.g., to support ontology design) [McGuinness, Borgida et al]

☞ Least common subsumer and/or matching (e.g., to support ontology integration and “bottom up” design) [Baader, Küsters & Molitor]

☞ ...
Summary

Semantic Web may be killer app for KRR (and many other areas)

The good news:
☞ We made a big sale
☞ Huge opportunity for everyone working in the area

The bad news (maybe):
☞ Now we need to deliver
☞ Major challenges for everyone working in the area
☞ Must exploit, adapt and extend existing work

Customers not noted for their patience!