




**OWL: A Description Logic Based
Ontology Language**

José Luis Ambite
Information Sciences Institute
University of Southern California




based on slides by
Ian Horrocks, University of Manchester
and others




University of Southern California 

Talk Outline


- Ontologies
- Introduction to Description Logics
- OWL Web Ontology Language
- Protégé OWL Demo
- Resources
- Summary




University of Southern California 

Ontology (in Computer Science)



- An ontology is an engineering artifact:
 - specific vocabulary used to describe a certain reality
 - explicit assumptions regarding the intended meaning of the vocabulary.
- Thus, an ontology describes a formal specification of a domain:
 - Shared understanding of a domain of interest
 - Formal and machine manipulable model of a domain of interest
- “An explicit specification of a conceptualisation” [Gruber93]




University of Southern California 

Description Logics


- A family of logic based Knowledge Representation formalisms
 - Descendants of [semantic networks](#)
 - Describe domain in terms of [concepts](#) (classes), [roles](#) (properties, relationships) and [individuals](#)
- Distinguished by:
 - [Formal semantics](#) (typically model theoretic)
 - [Decidable fragments of first-order logic \(FOL\)](#)
 - Closely related to Propositional Modal & Dynamic Logics
 - [Inference services](#):
 - Decision procedures for key problems: satisfiability, subsumption, classification, etc.
 - Implemented systems (highly optimized)





University of Southern California 

DL Basics


- **Concepts**: unary predicates/formulae with one free variable $p(x)$
 - E.g., Person, Doctor, HappyParent, (Doctor \sqcup Lawyer)
- **Roles**: binary predicates/formulae with two free variables $r(x,y)$
 - E.g., hasChild, loves, (hasBrother \circ hasDaughter)
- **Individuals**: constants
 - E.g., John, Mary, Italy
- **Concept/Role constructors** restricted so that:
 - Satisfiability/subsumption is decidable and, *if possible*, of low complexity
 - No need for explicit use of variables
 - Restricted form of \exists and \forall
 - Features such as counting can be succinctly expressed



University of Southern California 

The Description Logic Family

- Many description logics: depending on choice of concept/role constructors
- Smallest propositionally closed DL is *ALC*
 - Concepts constructed using boolean operators: \sqcap (and), \sqcup (or), \neg (complement)
 - plus restricted quantifiers \exists (some), \forall (all)
 - Only atomic roles
- Example: Person all of whose children are either Doctors or have a child who is a Doctor:
 $\text{Person} \sqcap \forall \text{hasChild} . (\text{Doctor} \sqcup \exists \text{hasChild} . \text{Doctor})$



University of Southern California **USC**

The DL Family (2)

- \mathcal{S} often used for \mathcal{ALC} extended with transitive roles (R_+)
- **Additional letters** indicate other extensions, e.g.:
 - \mathcal{H} for role hierarchy (e.g., hasDaughter \sqsubseteq hasChild)
 - \mathcal{O} for nominals/singleton classes (e.g., {Italy})
 - \mathcal{I} for inverse roles (e.g., isChildOf \equiv hasChild $^{-}$)
 - \mathcal{N} for number restrictions (e.g., ≥ 2 hasChild, ≤ 3 hasChild)
 - \mathcal{Q} for qualified number restrictions (e.g., ≥ 2 hasChild.Doctor)
 - \mathcal{F} for functional number restrictions (e.g., ≤ 1 hasMother)
- \mathcal{S} + role hierarchy (\mathcal{H}) + inverse (\mathcal{I}) + QNR (\mathcal{Q}) = \mathcal{SHIQ}
- \mathcal{SHIQ} is the basis for W3C's **OWL Web Ontology Language**
 - OWL DL \approx \mathcal{SHIQ} extended with nominals (i.e., \mathcal{SHOIQ})
 - OWL Lite \approx \mathcal{SHIQ} with only functional restrictions (i.e., \mathcal{SHIF})

ISI

University of Southern California **USC**

DL Semantics

Semantics given by standard FO model theory:

Interpretation function \mathcal{I} Interpretation domain $\Delta^{\mathcal{I}}$

Individuals $i^{\mathcal{I}} \in \Delta^{\mathcal{I}}$

John
Mary

Concepts $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$

Lawyer
Doctor

Roles $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$

hasChild
owns

(Lawyer \cap Doctor)

ISI

University of Southern California **USC**

DL Knowledge Base

- A **TBox** is a set of “schema” axioms (sentences), e.g.:


```
{ Doctor  $\sqsubseteq$  Person,
  HappyParent  $\equiv$  Person  $\cap$   $\forall$ hasChild.(Doctor  $\sqcup$   $\exists$ hasChild.Doctor) }
```
- An **ABox** is a set of “data” axioms (ground facts), e.g.:


```
{ John:HappyParent,
  John hasChild Mary }
```
- A **Knowledge Base (KB)** is just a TBox plus an ABox

ISI

University of Southern California **USC**

The Web Ontology Language OWL

- Semantic Web led to requirement for a “web ontology language”
- W3C set up Web-Ontology (**WebOnt**) Working Group
 - WebOnt developed **OWL** language
 - OWL based on earlier languages **OIL** and **DAML+OIL**
 - OWL now a W3C **recommendation** (i.e., a standard)
- OIL, DAML+OIL and OWL based on **Description Logics**
 - OWL effectively a “Web-friendly” syntax for **SHOIN**

ISI

University of Southern California **USC**

OWL RDF/XML Exchange Syntax

E.g., Person \cap \forall hasChild.(Doctor \sqcup \exists hasChild.Doctor):

```
<owl:Class>
  <owl:intersectionOf rdf:parseType=" collection">
    <owl:Class rdf:about="#Person"/>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasChild"/>
      <owl:allValuesFrom>
        <owl:unionOf rdf:parseType=" collection">
          <owl:Class rdf:about="#Doctor"/>
          <owl:Restriction>
            <owl:onProperty rdf:resource="#hasChild"/>
            <owl:someValuesFrom rdf:resource="#Doctor"/>
          </owl:Restriction>
        </owl:unionOf>
      </owl:allValuesFrom>
    </owl:Restriction>
  </owl:intersectionOf>
</owl:Class>
```

ISI

University of Southern California **USC**

OWL Abstract Syntax

E.g., Person \cap \forall hasChild.(Doctor \sqcup \exists hasChild.Doctor):

```
intersectionOf(
  restriction(hasChild allValuesFrom(
    unionOf(Doctor
      restriction(hasChild someValuesFrom(Doctor))))))
```

ISI

University of Southern California **USC**

Class/Concept Constructors

Constructor	DL Syntax	Example	FOL Syntax
intersectionOf	$C_1 \sqcap \dots \sqcap C_n$	Human \sqcap Male	$C_1(x) \wedge \dots \wedge C_n(x)$
unionOf	$C_1 \sqcup \dots \sqcup C_n$	Doctor \sqcup Lawyer	$C_1(x) \vee \dots \vee C_n(x)$
complementOf	$\neg C$	\neg Male	$\neg C(x)$
oneOf	$\{x_1\} \sqcup \dots \sqcup \{x_n\}$	$\{\text{John}\} \sqcup \{\text{Mary}\}$	$x = x_1 \vee \dots \vee x = x_n$
allValuesFrom	$\forall P.C$	$\forall \text{hasChild.Doctor}$	$\forall y.P(x, y) \rightarrow C(y)$
someValuesFrom	$\exists P.C$	$\exists \text{hasChild.Lawyer}$	$\exists y.P(x, y) \wedge C(y)$
maxCardinality	$\leq nP$	$\leq 1 \text{hasChild}$	$\exists^{\leq n} y.P(x, y)$
minCardinality	$\geq nP$	$\geq 2 \text{hasChild}$	$\exists^{\geq n} y.P(x, y)$

- C is a concept (class); P is a role (property); x is an individual name
- XMLS **datatypes** as well as classes in $\forall P.C$ and $\exists P.C$
 - Restricted form of DL **concrete domains**

ISI

University of Southern California **USC**

Ontology Axioms

OWL Syntax	DL Syntax	Example
subClassOf	$C_1 \sqsubseteq C_2$	Human \sqsubseteq Animal \sqcap Biped
equivalentClass	$C_1 \equiv C_2$	Man \equiv Human \sqcap Male
subPropertyOf	$P_1 \sqsubseteq P_2$	hasDaughter \sqsubseteq hasChild
equivalentProperty	$P_1 \equiv P_2$	cost \equiv price
transitiveProperty	$P^+ \sqsubseteq P$	ancestor ⁺ \sqsubseteq ancestor

OWL Syntax	DL Syntax	Example
type	$a : C$	John : Happy-Father
property	$\langle a, b \rangle : R$	$\langle \text{John}, \text{Mary} \rangle : \text{has-child}$

- OWL ontology equivalent to DL KB (Tbox + Abox)

ISI

University of Southern California **USC**

Why Description Logic?

- OWL exploits results of 20+ years of DL research
 - Well defined (model theoretic) **semantics**

ISI

University of Southern California **USC**

Why Description Logic?

- OWL exploits results of 15+ years of DL research
 - Well defined (model theoretic) **semantics**
 - Formal properties** well understood (complexity, decidability)

[Garey & Johnson. Computers and Intractability: A Guide to the Theory of NP-Completeness. Freeman, 1979.]

ISI

University of Southern California **USC**

Why Description Logic?

- OWL exploits results of 15+ years of DL research
 - Well defined (model theoretic) **semantics**
 - Formal properties** well understood (complexity, decidability)
 - Known **reasoning algorithms**

```

forall
  1. if  $(C_1 \sqcap C_2) \in \mathcal{C}(x)$ ,  $x$  is not indirectly blocked, and
  2.  $(C_1, C_2) \notin \mathcal{C}(x)$ 
  then  $\mathcal{C}(x) \leftarrow \mathcal{C}(x) \cup \{(C_1, C_2)\}$ 
endforall
forall
  1. if  $(C_1 \sqcup C_2) \in \mathcal{C}(x)$ ,  $x$  is not indirectly blocked, and
  2.  $(C_1, C_2) \in \mathcal{C}(x)$ 
  then  $\mathcal{C}(x) \leftarrow \mathcal{C}(x) \setminus \{(C_1, C_2)\}$ 
endforall
forall
  1.  $\exists r.C \in \mathcal{C}(x)$ ,  $x$  is not blocked, and
  2.  $x$  has no safe  $r$ -neighbour  $x_2$  with  $r.C \in \mathcal{C}(x_2)$ 
  then create a new node  $x_2$  and an edge  $(x, x_2)$ 
  with  $\mathcal{C}(x_2) = \{C\} \cup \mathcal{C}(x)$ 
endforall
forall
  1.  $\forall r.C \in \mathcal{C}(x)$ ,  $x$  is not indirectly blocked, and
  2. there is an  $r$ -neighbour  $x_2$  of  $x$  with  $r.C \notin \mathcal{C}(x_2)$ 
  then  $\mathcal{C}(x) \leftarrow \mathcal{C}(x) \cup \{C\}$ 
endforall
forall
  1.  $\forall r.C \in \mathcal{C}(x)$ ,  $x$  is not indirectly blocked, and
  2. there is some role  $r$  with  $\text{Trans}(r)$  and  $r.C \notin \mathcal{C}(x)$ 
  then  $\mathcal{C}(x) \leftarrow \mathcal{C}(x) \cup \{r.C\}$ 
endforall
forall
  1.  $\forall r.C \in \mathcal{C}(x)$ ,  $x$  is not indirectly blocked, and
  2. there is an  $r$ -neighbour  $x_2$  of  $x$  with  $(r.C) \in \mathcal{C}(x_2) = \emptyset$ 
  then  $\mathcal{C}(x) \leftarrow \mathcal{C}(x) \cup \{r.C\}$  for some  $r.C \in \mathcal{C}(x)$ 
endforall
forall
  1.  $\exists r.C \in \mathcal{C}(x)$ ,  $x$  is not blocked, and
  2. there are not  $n$  safe  $r$ -neighbours  $x_1, \dots, x_n$  of  $x$ 
  with  $r.C \in \mathcal{C}(x_i)$  and  $x_i \neq x_j$  for  $1 \leq i < j \leq n$ 

```

ISI

University of Southern California **USC**

Why Description Logic?


- OWL exploits results of 15+ years of DL research
 - Well defined (model theoretic) **semantics**
 - Formal properties** well understood (complexity, decidability)
 - Known **reasoning algorithms**
 - Implemented systems** (highly optimised)

ISI

University of Southern California **USC**

Why Description Logic?

- Foundational research was **crucial** to design of OWL
 - Informed Working Group decisions at every stage, e.g.:
 - “Why not extend the language with feature x, which is clearly harmless?”



- “Adding x would lead to undecidability - see proof in [...].”

ISI

University of Southern California **USC**

Why Decidable Reasoning?

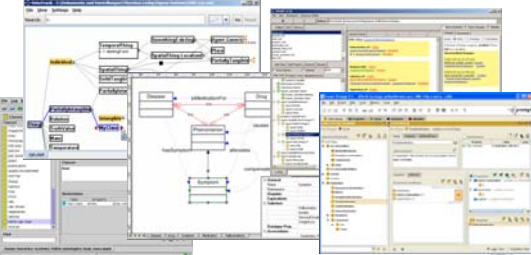
- OWL's expressive power **restricted** so reasoning is decidable
- Design was motivated by:
 - **Layered architecture** of Semantic Web languages
 - RDF(S) provides basic **relational language** and simple ontological primitives (or this is what RDF *should* be)
 - OWL provides powerful but still decidable **ontology language**
 - Further layers (e.g. RIF) will extend OWL (and may be undecidable)
 - W3C requirement for “**implementation experience**”
 - Evidence that language can be / is being used “in practice”
 - Should be several implemented systems
 - Users expectations of (automated reasoning) systems
 - Should exhibit **correct, consistent** and **predictable** behaviour
 - And should be quick about it!

ISI

University of Southern California **USC**

Tools and Infrastructure

- **Editors/environments**
 - Protégé, Oiled, Swoop, Construct, Ontotrack, ...



ISI

University of Southern California **USC**

Tools and Infrastructure

- **Editors/environments**
 - Protégé, Oiled, Swoop, Construct, Ontotrack, ...
- **Reasoning systems**
 - Cerebra, FaCT++, Kaon2, Pellet, Racer, ...



ISI

Protégé OWL demo

USC

University of Southern California **USC**

Resources

- Description Logic Handbook, Cambridge University Press
 - <http://books.cambridge.org/0521781760.htm>
- Description Logic: <http://dl.kr.org/>
 - complexity: <http://www.cs.man.ac.uk/~ezolin/logic/complexity.html>
- Web Ontology Language (OWL): <http://www.w3.org/2004/OWL/>
- Reasoners:
 - FaCT++ (open source): <http://owl.man.ac.uk/factplusplus/>
 - Racer (comercial): <http://www.racer-systems.com/>
 - Pellet: <http://www.mindswap.org/2003/pelle/>
 - (Loom and Powerloom: <http://www.isi.edu/isd/LOOM/>)
- Ontology Editors:
 - Protégé: <http://protege.stanford.edu/plugins/owl/>
- More slides on description logics and OWL:
 - <http://www.cs.man.ac.uk/~horrocks/Slides/>

ISI

Summary

- DLs are a family of **logic based** KR formalisms
 - Describe domain in terms of **concepts, roles** and **individuals**
 - DLs have **many applications**
- Basis of **web ontology language (OWL)**

