COMBINING DESCRIPTION LOGICS, DESCRIPTION GRAPHS, AND RULES

B. Motik
Joint work with B. Cuenca Grau, I. Horrocks, U. Sattler

University of Oxford

University of Manchester

June 2, 2008
Ontologies and the Semantic Web

Key Idea

Make information on the Web machine-processable

- Develop a vocabulary of a domain—an ontology
- Annotate information by ontology terms

Benefits

- Implicit meaning in the data is made explicit
- Formal semantics of the ontology language can be used to explicate implicit information
DLs are KR formalisms with well-understood formal properties

- Underpin the Ontology Web Language (OWL)
- Basic DL is called $\mathcal{ALC}$
- $\mathcal{ALCF} = \mathcal{ALC} +$ functional roles
- $\mathcal{ALCIF} = \mathcal{ALCF} +$ inverse roles

KBs consist of concepts (= unary predicates), roles (= binary predicates), and individuals (= constants)

**$\mathcal{ALC}$: Syntax and Semantics**

**Interpretation of Roles and Concepts**

- $(\neg C)^I = x \notin C^I$
- $(C \sqcap D)^I = x \in C^I \land x \in D^I$
- $(C \sqcup D)^I = x \in C^I \lor x \in D^I$
- $(\exists R.C)^I = \exists y \in \Delta^I : \langle x, y \rangle \in R^I \land y \in C^I$
- $(\forall R.C)^I = \forall y \in \Delta^I : \langle x, y \rangle \in R^I \rightarrow y \in C^I$

**Interpretation of Axioms and Assertions**

- $I \models C \subseteq D$ iff $\forall x \in \Delta^I : x \in C^I \rightarrow x \in D^I$
- $I \models C(a)$ iff $a^I \in C^I$
- $I \models R(a, b)$ iff $\langle a^I, b^I \rangle \in R^I$
**RELEVANT REASONING PROBLEMS**

**CONCEPT SATISFIABILITY**
Check whether a model of $\mathcal{O}$ exists in which $C$ is not empty

**CONCEPT SUBSUMPTION**
Check whether $\mathcal{O} \models C \sqsubseteq D$

- UKCity $\sqsubseteq$ EUCity is a consequence of $\mathcal{O} = \{ \text{UKCity} \sqsubseteq \exists \text{cityLocation}.\text{UKRegion}, \text{UKRegion} \sqsubseteq \text{EURegion}, \exists \text{cityLocation}.\text{EURegion} \sqsubseteq \text{EUCity} \}$

**QUERY ANSWERING**
Check whether $\mathcal{O} \models q$ where $q$ is a conjunctive query
Structured Objects

What are structured objects?
Objects composed of other, possibly interrelated, objects

Examples
- the human body
- the benzene molecule
- an airplane
- ...

Why are structured objects important?
At the core of many ontologies (FMA, GALEN, SNOMED, ...)

B. Motik
Combining DLs, Description Graphs, and Rules
STRUCTURED OBJECTS

Ontological Representation of Structured Objects

Long-standing open problem

- Modeling not sufficiently precise ⇒ more expressive power needed
- Reasoning can be slow ⇒ constructed models unnecessarily large

Common solutions based on:

- Modeling patterns ⇒ fail to provide the required expressivity
- Language extensions (e.g., rules) ⇒ lead to undecidability

No suitable solution known ⇒ existing ontologies often inaccurate or wrong
## Motivation

### Structured Objects

**Ontological Representation of Structured Objects**

Long-standing open problem

- Modeling not sufficiently precise $\Rightarrow$ more expressive power needed
- Reasoning can be slow $\Rightarrow$ constructed models unnecessarily large

Common solutions based on:

- Modeling patterns $\Rightarrow$ fail to provide the required expressivity
- Language extensions (e.g., rules) $\Rightarrow$ lead to undecidability

No suitable solution known $\Rightarrow$ existing ontologies often inaccurate or wrong

---

### A Novel Solution

Based on an analysis of ontologies in practice:

- Provides the required expressive power
- Improves performance of reasoning

These benefits are not necessarily in conflict!
AN EXAMPLE: THE HUMAN HEART

Motivation

The human heart is a vital organ responsible for pumping blood throughout the body. Understanding its structure and function is crucial for various medical applications. This diagram illustrates the heart with its major components: the atria (left and right), the ventricles (left and right), and the valves (mitral, aortic, pulmonic, tricuspid). Each part plays a critical role in the heart's operation, ensuring efficient blood circulation.
AN EXAMPLE: THE HUMAN HEART

Motivation

B. Motik

Combining DLs, Description Graphs, and Rules
Model the heart in a DL TBox

- We want to represent the structure not of a particular heart, but of all hearts
- The structure should be a “template” that can be instantiated many times

**Informal Model**

```
LeftSide
   hasComponent
       AorticValve
       MitralValve
   hasConnection
       AorticValve
       MitralValve
   divisionOf
       LeftVentricle
```

**DL Ontology \( \mathcal{O} \)**

```
LeftSide ⊑ ∃hasComponent.AorticValve
LeftSide ⊑ ∃hasComponent.MitralValve
AorticValve ⊑ ∃hasConnection.LeftVentricle
MitralValve ⊑ ∃hasConnection.LeftVentricle
LeftVentricle ⊑ ∃divisionOf.LeftSide
```
Is this a Faithful Representation?

**DL Ontology \( \mathcal{O} \)**

\[
egin{align*}
\text{LeftSide} & \sqsubseteq \exists \text{hasComponent}. \text{AorticValve} \\
\text{LeftSide} & \sqsubseteq \exists \text{hasComponent}. \text{MitralValve} \\
\text{AorticValve} & \sqsubseteq \exists \text{hasConnection}. \text{LeftVentricle} \\
\text{MitralValve} & \sqsubseteq \exists \text{hasConnection}. \text{LeftVentricle} \\
\text{LeftVentricle} & \sqsubseteq \exists \text{divisionOf}. \text{LeftSide}
\end{align*}
\]
Is this a Faithful Representation?

**DL Ontology \( \mathcal{O} \)**

- \( \text{LeftSide} \sqsubseteq \exists \text{hasComponent} . \text{AorticValve} \)
- \( \text{LeftSide} \sqsubseteq \exists \text{hasComponent} . \text{MitralValve} \)
- \( \text{AorticValve} \sqsubseteq \exists \text{hasConnection} . \text{LeftVentricle} \)
- \( \text{MitralValve} \sqsubseteq \exists \text{hasConnection} . \text{LeftVentricle} \)
- \( \text{LeftVentricle} \sqsubseteq \exists \text{divisionOf} . \text{LeftSide} \)

**The Intended Model \( \mathcal{I} \)**

- \( o_1 : \text{LeftSide} \)
- \( o_2 : \text{AorticValve} \)
- \( o_3 : \text{MitralValve} \)
- \( o_4 : \text{LeftVentricle} \)

\( \mathcal{O} \) is satisfied in a model corresponding with our intuitions
Motivation

Is this a Faithful Representation?

**DL Ontology $\mathcal{O}$**

- $\text{LeftSide} \sqsubseteq \exists \text{hasComponent}.\text{AorticValve}$
- $\text{LeftSide} \sqsubseteq \exists \text{hasComponent}.\text{MitralValve}$
- $\text{AorticValve} \sqsubseteq \exists \text{hasConnection}.\text{LeftVentricle}$
- $\text{MitralValve} \sqsubseteq \exists \text{hasConnection}.\text{LeftVentricle}$
- $\text{LeftVentricle} \sqsubseteq \exists \text{divisionOf}.\text{LeftSide}$

**Unintended Model $\mathcal{J}_1$**

- $o_1: \text{LeftSide}$
- $o_2: \text{AorticValve}$
- $o_3: \text{MitralValve}$
- $o_5: \text{LeftVentricle}$
- $o_6: \text{LeftVentricle}$

$\mathcal{O}$ is also satisfied in a model not corresponding with our intuitions
**Motivation**

Is this a Faithful Representation?

**DL Ontology $\mathcal{O}$**

- $\text{LeftSide} \sqsubseteq \exists \text{hasComponent}. \text{AorticValve}$
- $\text{LeftSide} \sqsubseteq \exists \text{hasComponent}. \text{MitralValve}$
- $\text{AorticValve} \sqsubseteq \exists \text{hasConnection}. \text{LeftVentricle}$
- $\text{MitralValve} \sqsubseteq \exists \text{hasConnection}. \text{LeftVentricle}$
- $\text{LeftVentricle} \sqsubseteq \exists \text{divisionOf}. \text{LeftSide}$

$\mathcal{O}$ is also satisfied in an infinite model not corresponding with our intuitions.

**Unintended Model $\mathcal{I}_2$**

- $o_1: \text{LeftSide}$
- $o_2: \text{AorticValve}$
- $o_3: \text{LeftVentricle}$
- $o_4: \text{LeftSide}$
Is this a Faithful Representation?

**A Tree Model Property**

- DL ontology $\mathcal{O}$ has a model $\Rightarrow$ it has a “tree-shaped” one
- Key to ensuring decidability
IS THIS A FAITHFUL REPRESENTATION?

A TREE MODEL PROPERTY

- DL ontology $\mathcal{O}$ has a model $\implies$ it has a “tree-shaped” one
- Key to ensuring decidability

PROBLEMS

- Unintended tree models cannot be ruled out in DLs
- Underconstrained representation
- Cannot draw inferences that rely on having only intended models
- Unintended models can be big and expensive to construct $\implies$ performance problems
A NovaL Solution: Description Graphs

Basic Intuition

“Draw” the intended structure as a graph $G = (V, E, \lambda, M)$

- $V$: set of nodes
- $E$: set of edges
- $\lambda$: labels nodes with concepts and edges with roles
- $M$: set of main concepts

Formal Representation

```
1:LeftSide
  \overset{\text{hasComponent}}{\rightarrow}
2:AorticValve
\quad \overset{\text{hasConnection}}{\rightarrow}
3:MitralValve
\quad \overset{\text{divisionOf}}{\rightarrow}
4:LeftVentricle
```

$V = \{1, 2, 3, 4\}$

$\lambda\langle 1 \rangle = \text{LeftSide}$

$\lambda\langle 1, 2 \rangle = \text{hasComponent}$

$E = \{\langle 1, 2 \rangle, \langle 1, 3 \rangle, \langle 2, 4 \rangle, \langle 3, 4 \rangle, \langle 4, 1 \rangle\}$

$M = \{\text{LeftSide}\}$
Description Graphs: Semantics

**Interpretation** $\mathcal{I}$

Graph with $\ell$ vertices $\Rightarrow$ $\ell$-ary predicate

**Key Property:** for each $1 \leq i \leq \ell$,
\[
\forall x_1, \ldots, x_\ell, y_1, \ldots, y_\ell \in \Delta^\mathcal{I} : \langle x_1, \ldots, x_\ell \rangle \in G^\mathcal{I} \land \\
\langle y_1, \ldots, y_\ell \rangle \in G^\mathcal{I} \land x_i = y_i \rightarrow \bigwedge_{1 \leq j \leq \ell} x_j = y_j
\]

**Disjointness Property:**
\[
\forall x_1, \ldots, x_\ell, y_1, \ldots, y_\ell \in \Delta^\mathcal{I} : \langle x_1, \ldots, x_\ell \rangle \in G^\mathcal{I} \land \\
\langle y_1, \ldots, y_\ell \rangle \in G^\mathcal{I} \rightarrow \bigwedge_{1 \leq i < j \leq n} x_i \neq y_j
\]

**Start Property:** for each atomic concept $A \in M$,
\[
\forall x \in \Delta^\mathcal{I} : x \in A^\mathcal{I} \rightarrow \exists x_1, \ldots, x_\ell \in \Delta^\mathcal{I} : \\
\langle x_1, \ldots, x_\ell \rangle \in G^\mathcal{I} \land \bigvee_{k \in V_A} x = x_k
\]

**Layout Property:**
\[
\forall x_1, \ldots, x_\ell \in \Delta^\mathcal{I} : \langle x_1, \ldots, x_\ell \rangle \in G^\mathcal{I} \rightarrow \\
\bigwedge_{i \in V, B \in \lambda \langle i \rangle} x_i \in B^\mathcal{I} \land \bigwedge_{\langle i, j \rangle \in E, R \in \lambda \langle i, j \rangle} \langle x_i, x_j \rangle \in R^\mathcal{I}
\]
MODULARIZATION OF THE KNOWLEDGE BASE

(a) Anatomy of the Hand

Legend:

Legend:

(b) Model of the Hand ($G_{hand}$)

Legend:

Legend:

Legend:

(c) Model of a Finger ($G_{finger}$)

Legend:

Legend:

Legend:

Legend:

(d) Model of the Thumb ($G_{thumb}$)

Legend:

Legend:

Legend:

Legend:

Legend:

Legend:

(e) Model of the Index Finger ($G_{index_finger}$)
Syntax and Semantics

Syntax

Graph Specialization

- \( G_1 \prec G_2 \) with \( V_1 \subseteq V_2 \)
- \( G_2 \) more specific than \( G_1 \)
- \( G_{\text{finger}} \prec G_{\text{thumb}} \)

Graph Alignment

- \( G_{\text{hand}}[3,4] \leftrightarrow G_{\text{thumb}}[1,3] \)
- \( G_1[u_1,\ldots,u_n] \leftrightarrow G_2[w_1,\ldots,w_n] \)
- \( G_1 \) and \( G_2 \) connected at the specified vertices

Semantics

Graph Specialization

\( \mathcal{I} \models G_1 \prec G_2 \) if

\[ \forall x_1, \ldots, x_{\ell_2} \in \Delta^I : \langle x_1, \ldots, x_{\ell_1}, \ldots, x_{\ell_2} \rangle \in G_2^I \rightarrow \langle x_1, \ldots, x_{\ell_1} \rangle \in G_1^I \]

Graph Alignment

\( \mathcal{I} \models G_1[u_1,\ldots,u_n] \leftrightarrow G_2[w_1,\ldots,w_n] \) if, for each \( 1 \leq i \leq n \),

\[ \forall x_1, \ldots, x_{\ell_1}, y_1, \ldots, x_{\ell_2} \in \Delta^I : \langle x_1, \ldots, x_{\ell_1} \rangle \in G_1^I \land \langle y_1, \ldots, y_{\ell_2} \rangle \in G_2^I \]

\[ \land \ x_{u_i} = y_{w_i} \rightarrow \bigwedge_{1 \leq j \leq n} x_{u_j} = y_{w_j} \]
A graph-extended knowledge base consists of

- a DL TBox $\mathcal{T}$
- a graph box (GBox) $\mathcal{G}$ composed of:
  - a finite set of description graphs $\mathcal{G}_G$
  - a finite set of graph specializations $\mathcal{G}_S$
  - a finite set of graph alignments $\mathcal{G}_A$
- a set of rules (i.e., function-free implications) $\mathcal{P}$
- an ABox $\mathcal{A}$
## Computational Properties

### Undecidability Results

<table>
<thead>
<tr>
<th>DLs + Rules</th>
<th>Known from SWRL/CARIN</th>
</tr>
</thead>
</table>

| Description Graphs + Rules | Checking the satisfiability of $\mathcal{K} = (\emptyset, \mathcal{P}, \mathcal{G}, \mathcal{A})$ is undecidable for $\mathcal{P}$ a Horn program and $\mathcal{G} = (\mathcal{G}_G, \emptyset, \emptyset)$. |

| Description Graphs + DLs | Checking the satisfiability of $\mathcal{K} = (\mathcal{T}, \emptyset, \mathcal{G}, \mathcal{A})$ is undecidable for $\mathcal{T}$ in $\mathcal{ALCF}$ and $\mathcal{G} = (\mathcal{G}_G, \emptyset, \emptyset)$. |
**Undecidability: DLs + Graphs**

Reduction from the DOMINO problem

**Knowledge Base $\mathcal{K}$**

\[
\begin{align*}
\mathbf{T} & \subseteq 1H & \mathbf{T} & \subseteq 1V \\
\lambda_1 &= \begin{cases} 1 \mapsto \{A_1\}, 2 \mapsto \{A_2\}, \\ 3 \mapsto \{A_3\}, 4 \mapsto \{A_4\} \end{cases} \\
\lambda_2 &= \begin{cases} 1 \mapsto \{A_2\}, 2 \mapsto \{A_1\}, \\ 3 \mapsto \{A_4\}, 4 \mapsto \{A_3\} \end{cases} \\
\lambda_3 &= \begin{cases} 1 \mapsto \{A_3\}, 2 \mapsto \{A_4\}, \\ 3 \mapsto \{A_1\}, 4 \mapsto \{A_2\} \end{cases} \\
\lambda_4 &= \begin{cases} 1 \mapsto \{A_4\}, 2 \mapsto \{A_3\}, \\ 3 \mapsto \{A_2\}, 4 \mapsto \{A_1\} \end{cases} \\
M_i &= \{A_i\}
\end{align*}
\]

**A Model of $\mathcal{K}$**

- $V_g \in G_i \cap V$
**Computational Properties**

**AN ATTEMPT TO REGAIN DECIDABILITY**

**ACYCLICITY**

Syntactic restriction on $\mathcal{G}$ that prevents cyclic implications of the existence of graph instances

- Limits the number of objects whose existence is implied by the graphs
- Objects in many domains are naturally bounded

**DOES NOT DO THE TRICK**

Checking the satisfiability of $\mathcal{K} = (\mathcal{T}, \emptyset, \mathcal{G}, \mathcal{A})$ is undecidable for $\mathcal{T}$ in $\mathcal{ALCIF}$ and $\mathcal{G} = (\mathcal{G}_G, \emptyset, \emptyset)$ an acyclic GBox.
**Additional Restriction: Role Separation**

**Weak Separation**

The roles occurring in $\mathcal{P}$ are disjoint with the roles occurring in $\mathcal{T}$

- Prevents the application of the rules in $\mathcal{P}$ to the part of the model constructed by $\mathcal{T}$
- “Neutralizes” the undecidability due to rules

**Strong Separation**

The roles occurring in $\mathcal{P}$ and $\mathcal{G}$ are disjoint with the roles occurring in $\mathcal{T}$

- Separates the parts of the model constructed by $\mathcal{T}$ and $\mathcal{G}$
- “Neutralizes” additionally the problems with inverse roles
WEAK SEPARATION
Checking the satisfiability of a weakly separated acyclic KB $\mathcal{K} = (\mathcal{T}, \mathcal{P}, \mathcal{G}, \mathcal{A})$ is decidable for $\mathcal{T}$ in $SHOQ^+$. 

STRONG SEPARATION
Checking the satisfiability of a strongly separated acyclic KB $\mathcal{K} = (\mathcal{T}, \mathcal{P}, \mathcal{G}, \mathcal{A})$ is decidable for $\mathcal{T}$ in $SHOIQ^+$. 
Decidability

Canonical models composed of:

- A (possibly infinite) tree backbone
  ⇒ Generated by DL axioms
- Arbitrarily connected, yet finite and “isolated” collections of graph instances
  ⇒ Generated by description graphs

A tableau algorithm with the following termination argument:

- Roles in a graph and the backbone interact in a limited way or not at all (role separation)
- As in DLs, an algorithm can eventually “block” the expansion of the tree backbone
- Graphs clusters are bounded (acyclicity, key, and disjointness)
**NExpTime-Hardness**

Applies to $\mathcal{K} = (\emptyset, \mathcal{P}, \mathcal{G}, \mathcal{A})$ where $\mathcal{G} = (\mathcal{G}_G, \emptyset, \emptyset)$ is an acyclic GBox and each rule in $\mathcal{P}$ contains only atomic concepts and roles and at most four variables.

- Reduction from the bounded DOMINO problem

---

**A Model of $\mathcal{K}$**

GBox $\mathcal{G}$

![Diagram of GBox $\mathcal{G}$]
**Weak Separation**

Checking the satisfiability of a weakly separated acyclic KB $\mathcal{K} = (\mathcal{T}, \mathcal{P}, \mathcal{G}, \mathcal{A})$ with $\mathcal{T}$ in $SHOQ^+$ is $\text{NEXPTIME}$-hard.

**Strong Separation**

Checking the satisfiability of a strongly separated acyclic KB $\mathcal{K} = (\mathcal{T}, \mathcal{P}, \mathcal{G}, \mathcal{A})$ with $\mathcal{T}$ in $SHIQ^+$ is $\text{NEXPTIME}$-hard.
Future Work

- Complexity of the strongly separated acyclic case with $\mathcal{T}$ in $\text{SHOIQ}^+$ is open
- Optimizations of the reasoning algorithm
- Tool support in an ontology editor
- New applications (e.g., molecule recognition)