Ian Horrocks  
Information Management Group  
University of Manchester, UK

Ulrike Sattler  
Teaching and Research Area for Theoretical Computer Science  
RWTH Aachen, Germany
• **History and Basics:** Syntax, Semantics, ABoxes, Tboxes, Inference Problems and their interrelationship, and Relationship with other (logical) formalisms

• **Applications of DLs:** ER-diagrams with i.com demo, ontologies, etc. including system demonstration

• **Reasoning Procedures:** simple tableaux and why they work

• **Reasoning Procedures II:** more complex tableaux, non-standard inference problems

• **Complexity issues**

• **Implementing/Optimising DL systems**
Description Logics

- family of logic-based knowledge representation formalisms well-suited for the representation of and reasoning about
  - terminological knowledge
  - configurations
  - ontologies
  - database schemata
    - schema design, evolution, and query optimisation
    - source integration in heterogeneous databases/data warehouses
    - conceptual modelling of multidimensional aggregation
  - ...

- descendents of semantics networks, frame-based systems, and KL-ONE

- aka terminological KR systems, concept languages, etc.
Architecture of a Standard DL System

Knowledge Base

Terminology
Father = Man \land \exists \text{ has_child. } \top 
\text{ Human } = \text{ Mammal } \land \text{ Biped } 

Concrete Situation
John: \text{ Human } \land \text{ Father } 
John \text{ has_child } Bill 

Description Logic
A Description Logic - mainly characterised by a set of constructors that allow to build complex concepts and roles from atomic ones,

concepts correspond to classes / are interpreted as sets of objects,
roles correspond to relations / are interpreted as binary relations on objects,

Example: Happy Father in the DL $\mathcal{ALC}$

\[
\text{Man} \sqcap (\exists \text{has-child}. \text{Blue}) \sqcap \\
(\exists \text{has-child}. \text{Green}) \sqcap \\
(\forall \text{has-child}. \text{Happy} \sqcap \text{Rich})
\]
Introduction to DL: Syntax and Semantics of \( \mathcal{ALC} \)

Semantics given by means of an interpretation \( \mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I}) \):

<table>
<thead>
<tr>
<th>Constructor</th>
<th>Syntax</th>
<th>Example</th>
<th>Semantics</th>
</tr>
</thead>
<tbody>
<tr>
<td>atomic concept</td>
<td>( A )</td>
<td>Human</td>
<td>( A^\mathcal{I} \subseteq \Delta^\mathcal{I} )</td>
</tr>
<tr>
<td>atomic role</td>
<td>( R )</td>
<td>likes</td>
<td>( R^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I} )</td>
</tr>
</tbody>
</table>

For \( C, D \) concepts and \( R \) a role name

<table>
<thead>
<tr>
<th>Constructor</th>
<th>Syntax</th>
<th>Example</th>
<th>Semantics</th>
</tr>
</thead>
<tbody>
<tr>
<td>conjunction</td>
<td>( C \sqcap D )</td>
<td>Human \sqcap Male</td>
<td>( C^\mathcal{I} \cap D^\mathcal{I} )</td>
</tr>
<tr>
<td>disjunction</td>
<td>( C \sqcup D )</td>
<td>Nice \sqcup Rich</td>
<td>( C^\mathcal{I} \cup D^\mathcal{I} )</td>
</tr>
<tr>
<td>negation</td>
<td>( \neg C )</td>
<td>\neg Meat</td>
<td>( \Delta^\mathcal{I} \setminus C^\mathcal{I} )</td>
</tr>
<tr>
<td>exists restrict.</td>
<td>( \exists R.C )</td>
<td>\exists has-child.Human</td>
<td>( { x \mid \exists y. \langle x, y \rangle \in R^\mathcal{I} \land y \in C^\mathcal{I} } )</td>
</tr>
<tr>
<td>value restrict.</td>
<td>( \forall R.C )</td>
<td>\forall has-child.Blond</td>
<td>( { x \mid \forall y. \langle x, y \rangle \in R^\mathcal{I} \implies y \in C^\mathcal{I} } )</td>
</tr>
<tr>
<td>Constructor</td>
<td>Syntax</td>
<td>Example</td>
<td>Semantics</td>
</tr>
<tr>
<td>------------------------</td>
<td>----------------</td>
<td>--------------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>number restriction</td>
<td>$(\geq n \ R)$</td>
<td>$(\geq 7 \ \text{has-child})$</td>
<td>${x \mid</td>
</tr>
<tr>
<td>$\sim \mathcal{ALCN}$</td>
<td>$(\leq n \ R)$</td>
<td>$(\leq 1 \ \text{has-mother})$</td>
<td>${x \mid</td>
</tr>
<tr>
<td>inverse role</td>
<td>$R^-$</td>
<td>has-child$^-$</td>
<td>${\langle x, y \rangle \mid \langle y, x \rangle \in R^I}$</td>
</tr>
<tr>
<td>trans. role</td>
<td>$R^*$</td>
<td>has-child$^*$</td>
<td>$(R^I)^*$</td>
</tr>
<tr>
<td>concrete domain</td>
<td>$u_1, \ldots, u_n.P$</td>
<td>h-father-age, age. $&gt;$</td>
<td>${x \mid \langle u_1^T(x), \ldots, u_n^T(x)\rangle \in P}$</td>
</tr>
<tr>
<td>etc.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Many other different DLs/DL constructors have been investigated.
For terminological knowledge: TBox contains

**Concept definitions**

\[ A \doteq C \quad (A \text{ a concept name, } C \text{ a complex concept}) \]

Father \doteq \text{Man} \cap \exists \text{has-child.Human}

Human \doteq \text{Mammal} \cap \forall \text{has-child}^- \text{.Human}

\sim \text{ introduce macros/names for concepts, can be (a)cyclic}

**Axioms**

\[ C_1 \subseteq C_2 \quad (C_i \text{ complex concepts}) \]

\[ \exists \text{favourite.Brewery} \subseteq \exists \text{drinks.Beer} \]

\sim \text{ restrict your models}

**An interpretation } \mathcal{I} \text{ satisfies}

a concept definition \[ A \doteq C \quad \text{iff} \quad A^\mathcal{I} = C^\mathcal{I} \]

an axiom \[ C_1 \subseteq C_2 \quad \text{iff} \quad C_1^\mathcal{I} \subseteq C_2^\mathcal{I} \]

a TBox \[ \mathcal{T} \quad \text{iff} \quad \mathcal{I} \text{ satisfies all definitions and axioms in } \mathcal{T} \]

\sim \mathcal{I} \text{ is a model of } \mathcal{T}
For assertional knowledge: ABox contains

Concept assertions \( a : C \) (\( a \) an individual name, \( C \) a complex concept)
   John : Man \( \sqcap \) \( \forall \) has-child.(Male \( \sqcap \) Happy)

Role assertions \( \langle a_1, a_2 \rangle : R \) (\( a_i \) individual names, \( R \) a role)
   \( \langle \text{John, Bill} \rangle : \text{has-child} \)

An interpretation \( \mathcal{I} \) satisfies

a concept assertion \( a : C \) iff \( a^{\mathcal{I}} \in C^{\mathcal{I}} \)

a role assertion \( \langle a_1, a_2 \rangle : R \) iff \( \langle a_1^{\mathcal{I}}, a_2^{\mathcal{I}} \rangle \in R^{\mathcal{I}} \)

an ABox \( \mathcal{A} \) iff \( \mathcal{I} \) satisfies all assertions in \( \mathcal{A} \)
\[ \sim \] \( \mathcal{I} \) is a model of \( \mathcal{A} \)
Introduction to DL: Basic Inference Problems

Subsumption: \( C \sqsubseteq D \) 

w.r.t. TBox \( \mathcal{T} \): \( C \sqsubseteq_{\mathcal{T}} D \)

Is \( C^I \subseteq D^I \) in all interpretations \( I \)?

Is \( C^I \subseteq D^I \) in all models \( I \) of \( \mathcal{T} \)?

\( \leadsto \) structure your knowledge, compute taxonomy

Consistency: Is \( C \) consistent w.r.t. \( \mathcal{T} \)? Is there a model \( I \) of \( \mathcal{T} \) with \( C^I \neq \emptyset \)?

of ABox \( \mathcal{A} \): Is \( \mathcal{A} \) consistent? Is there a model of \( \mathcal{A} \)?

of KB \((\mathcal{T}, \mathcal{A})\): Is \((\mathcal{T}, \mathcal{A})\) consistent? Is there a model of both \( \mathcal{T} \) and \( \mathcal{A} \)?

Inference Problems are closely related:

\[ C \sqsubseteq_{\mathcal{T}} D \text{ iff } C \cap \neg D \text{ is inconsistent w.r.t. } \mathcal{T}, \]

(no model of \( I \) has an instance of \( C \cap \neg D \))

\[ C \text{ is consistent w.r.t. } \mathcal{T} \text{ iff not } C \sqsubseteq_{\mathcal{T}} A \cap \neg A \]

\( \leadsto \) Decision Procedures for consistency (w.r.t. TBoxes) suffice
For most DLs, the basic inference problems are **decidable**, with complexities between $\mathbf{P}$ and $\mathbf{ExpTime}$.

**Why is decidability important?** Why does semi-decidability not suffice?

If subsumption (and hence consistency) is undecidable, and

- subsumption is semi-decidable, then consistency is **not** semi-decidable
- consistency is semi-decidable, then subsumption is **not** semi-decidable

- Quest for a “highly expressive” DL with decidable/“practicable” inference problems

  where **expressiveness** depends on the application

  **practicability** changed over the time
Introduction to DL: History

Complexity of Inferences provided by DL systems over the time

Investigation of Complexity of Inference Problems/Algorithms starts

Undecidable

KL-ONE
NIKL

ExpTime

Loom

PSpace

Fact, DLP, Race

NP

Crack, Kris

PTime

Classic (AT&T)

late '80s          early '90s          mid '90s          late '90s
In the last 5 years, DL-based systems were built that

✔ can handle DLs far more expressive than $\mathcal{ALC}$ (close relatives of converse-DPDL)

- Number restrictions: “people having at most 2 children”
- Complex roles: inverse (“has-child” — “child-of”),
  transitive closure (“offspring” — “has-child”),
  role inclusion (“has-daughter” — “has-child”), etc.

✔ implement provably sound and complete inference algorithms
  (for ExpTime-complete problems)

✔ can handle large knowledge bases
  (e.g., Galen medical terminology ontology: 2,740 concepts, 413 roles, 1,214 axioms)

✔ are highly optimised versions of tableau-based algorithms

✔ perform (surprisingly well) on benchmarks for modal logic reasoners
  (Tableaux’98, Tableaux’99)
Most DLs are decidable fragments of FOL: Introduce

a unary predicate $\phi_A$ for a concept name $A$
a binary relation $\rho_R$ for a role name $R$

Translate complex concepts $C, D$ as follows:

\[
\begin{align*}
t_x(A) &= \phi_A(x), & t_y(A) &= \phi_A(y), \\
t_x(C \sqcap D) &= t_x(C) \land t_x(D), & t_y(C \sqcap D) &= t_y(C) \land t_y(D), \\
t_x(C \sqcup D) &= t_x(C) \lor t_x(D), & t_y(C \sqcup D) &= t_y(C) \lor t_y(D), \\
t_x(\exists R.C) &= \exists y.\rho_R(x, y) \land t_y(C), & t_y(\exists R.C) &= \exists x.\rho_R(y, x) \land t_x(C), \\
t_x(\forall R.C) &= \forall y.\rho_R(x, y) \Rightarrow t_y(C), & t_y(\forall R.C) &= \forall x.\rho_R(y, x) \Rightarrow t_x(C).
\end{align*}
\]

A TBox $\mathcal{T} = \{C_i \sqsubseteq D_i\}$ is translated as

\[
\Phi_{\mathcal{T}} = \forall x. \bigwedge_{1 \leq i \leq n} t_x(C_i) \Rightarrow t_x(D_i)
\]
\( C \) is consistent iff its translation \( t_x(C) \) is satisfiable,

\( C \) is consistent w.r.t. \( T \) iff its translation \( t_x(C) \land \Phi_T \) is satisfiable,

\( C \sqsubseteq D \) iff \( t_x(C) \Rightarrow t_x(D) \) is valid

\( C \sqsubseteq_T D \) iff \( \Phi_t \Rightarrow \forall x.(t_x(C) \Rightarrow t_x(D)) \) is valid.

\( \rightsquigarrow \mathcal{ALC} \) is a fragment of FOL with 2 variables (L2), known to be decidable

\( \rightsquigarrow \mathcal{ALC} \) with inverse roles and Boolean operators on roles is a fragment of L2

\( \rightsquigarrow \) further adding number restrictions yields a fragment of C2

(L2 with “counting quantifiers”), known to be decidable

✦ in contrast to most DLs, adding transitive roles/transitive closure operator to L2 leads to **undecidability**

✦ many DLs (like many modal logics) are fragments of the **Guarded Fragment**

✦ most DLs are less complex than L2:

L2 is NExpTime-complete, most DLs are in ExpTime
DLs and Modal Logics are closely related:

\[ ALC \sqsubseteq \text{multi-modal } K: \]

\[
\begin{align*}
C \sqcap D & \sqsubseteq C \land D, & C \sqcup D & \sqsubseteq C \lor D \\
\neg C & \sqsubseteq \neg C, & \\
\exists R.C & \sqsubseteq \langle R \rangle C, & \forall R.C & \sqsubseteq [R]C
\end{align*}
\]

transitive roles \(\sqsubseteq\) transitive frames (e.g., in K4)

regular expressions on roles \(\sqsubseteq\) regular expressions on programs (e.g., in PDL)

inverse roles \(\sqsubseteq\) converse programs (e.g., in C-PDL)

number restrictions \(\sqsubseteq\) deterministic programs (e.g., in D-PDL)

\(\nvdash\) no TBoxes available in modal logics

\(\leadsto\) “internalise” axioms using a universal role \(u\): \(C \vDash D \sqsubseteq [u](C \leftrightarrow D)\)

\(\nleq\) no ABox available in modal logics \(\leadsto\) use nominals