The Actual Weight of Lightweight Description Logics

Diego Calvanese

Faculty of Engineering
Free University of Bozen-Bolzano, Italy

unibz

38th International Workshop on Description Logics (DL 2025)
6 September 2025 – Opole, Poland

[Background images in this presentation were generated using ChatGPT (GPT-5, OpenAI, 2025).]

Outline

Outline


Prehistory
1970's

1970's: Class-based formalisms

Beginning of the 1970's

Many areas of CS independently proposed **class-based formalisms**:

- Semantic Networks (AI)
- Frame Systems (AI)
- Entity-Relationship Schemas (DB)
- Object-Orientation (SE)

By the end of the 1970's

The need for a formal account was evident:

- **William A. Woods – What's in a link?** [Woods, 1975, AAAI]: no clear semantics, reasoning not well understood.

Concept / Terminological Languages

Logic-based formalisms specifically designed to represent class-oriented structured knowledge:

Domain consists of objects organized into:

- Concepts: correspond to classes, denote sets of objects.
- Roles: correspond to (binary) relationships, denote binary relations on objects.

Knowledge asserted through TBox and ABox assertions, i.e., logical axioms.

In 1977

Brachman, Woods, and others developed a precursor of DL-based systems: KL-ONE [Woods and Brachman, 1977].


Ancient Times
1980's

1980's: Description Logics were born

Ron Brachman & Hector Levesque – The Tractability of Subsumption in Frame-Based Description Languages [Brachman and Levesque, 1984, AAAI]:

- Use **logic** to capture class-based formalisms.
- Be **decidable**!
- Use **complexity** to understand the intrinsic computational properties of the language.
- There is a **tradeoff** between expressivity and complexity.
- Focus on **effective/tractable** languages.

These points are still the focus of the research in Description Logics in the current days!

Bell Labs developed the system CLASSIC with tractable subsumption [Borgida *et al.*, 1989, SIGMOD], [Patel-Schneider *et al.*, 1991, SIGART].

unibz


Renaissance
1990’s

1990's: Focus on knowledge bases and expressive DLs

Various research groups worked on semantics, computational properties, and algorithms for **expressive DLs**.

- Tableaux algorithms for satisfiability / subsumption in $\mathcal{ALC}$ and sublanguages [Schmidt-Schauss and Smolka, 1991, AIJ], [Nebel, 1991], [Donini *et al.*, 1991, KR].
- Terminological cycles [Baader, 1990, AAAI] and transitive closure of roles [Baader, 1991, IJCAI].
- Description logics $\equiv$ Modal Logics of programs (with inverses, graded modalities, nominals, fixpoints) [Schild, 1991, IJCAI], [De Giacomo and Lenzerini, 1994a, AAAI], [C. *et al.*, 1995, DOOD].
- Concrete domains [Baader and Hanschke, 1991, IJCAI], [Lutz, 1999, IJCAI].
- Finite model reasoning [C., 1996, KR], [Lutz *et al.*, 2003, CADE].
- ...


Industrial
Revolution
2000's

2000's: Two important developments

On the applied side

Optimized fast tableaux for expressive DLs like $\mathcal{ALCQI}$ [Horrocks, 1998, KR], later $\mathcal{SHIQ}$ [Horrocks *et al.*, 2000, JIGPL].

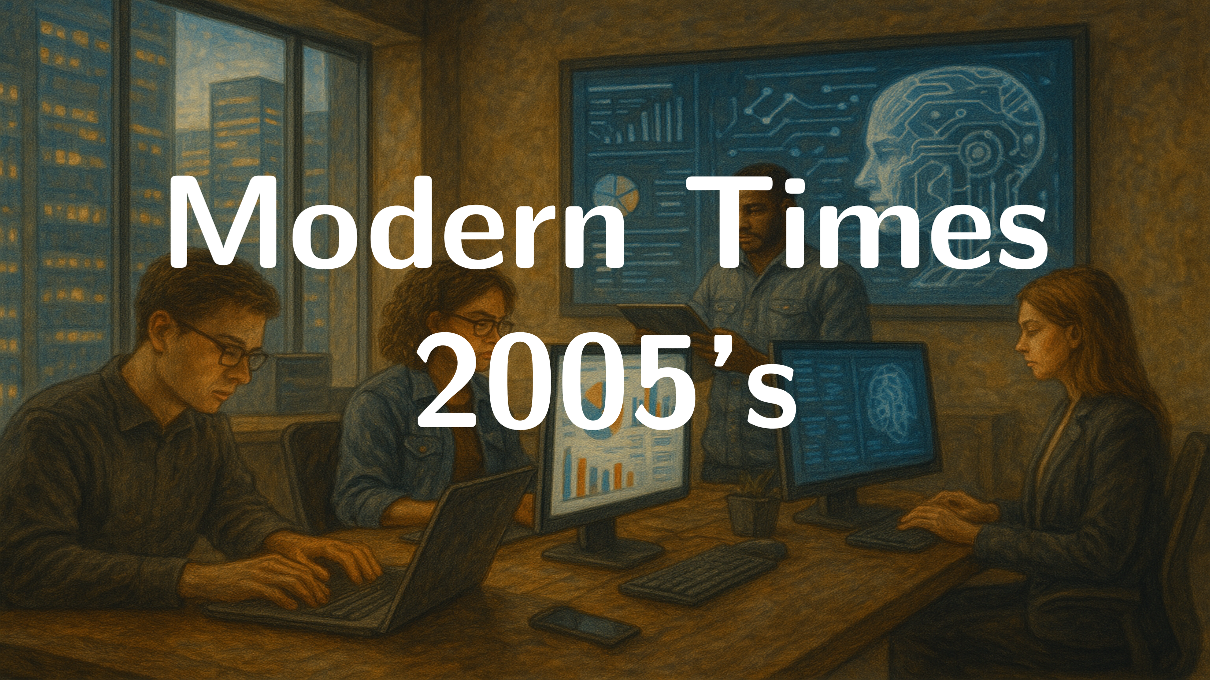
$\leadsto$ *Definition of* ***OWL W3C Standard****, based on scientific grounds!*

OWL 1 DL $\sim \mathcal{SHOIN}(\mathcal{D})$ OWL $\sim \mathcal{SROIQ}(\mathcal{D})$

On the theoretical side

Conjunctive query answering over DL KBs turns out to be decidable [C. *et al.*, 1998, PODS] and becomes a focus of research in DLs and knowledge representation and reasoning.

Modern Times 2005’s



2005's: Lightweight Description Logics

Scalability problems in handling **large ontologies** and **large amounts of data**.

$\rightsquigarrow$ **New kinds of DLs** were needed, with:

- tractable reasoning
- support for efficient query answering

Dresden: $\mathcal{EL}$ [Baader *et al.*, 2005, IJCAI; 2008, OWLED]

Ability to enforce the existence of tree-shaped structures in models (cf. $\mathcal{AL}$ of CLASSIC).

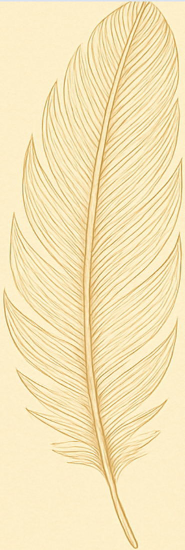
$\rightsquigarrow$ Captures SNOMED CT, a full-fledged medical ontology with 311,000 terms.

Rome + Bolzano: ***DL-Lite*** [C. *et al.*, 2005, AAAI; 2008, JAR]

Relies on dependency theory in Databases and query rewriting, thus scales with data.

$\rightsquigarrow$ Captures conceptual modeling formalisms (UML class diagrams, ER schemas).

Outline



unibz

Knowledge Bases – a.k.a. Ontologies

- We are dealing with description logics, hence the domain is modeled in terms of **concepts** (i.e., classes) and **roles** (i.e., binary relationships).
- Knowledge about the domain is represented in a KB.

A **KB** $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ consists of two components:

- A **TBox** $\mathcal{T}$ modeling intensional (i.e., schema-level) information in the form of universally quantified assertions (logical axioms).
- An **ABox** $\mathcal{A}$ modeling extensional (i.e., data-level) information, through assertions on individuals (facts) of the form:

$$A(c), \quad \text{e.g., } \mathsf{Actor}(\mathsf{Keanu}) \qquad \text{or} \qquad P(c_1, c_2), \quad \text{e.g., } \mathsf{manages}(\mathsf{Bill}, \mathsf{Carrie\text{-}Anne})$$

A KB is interpreted in **standard first-order semantics** (possibly adopting the unique name assumption – UNA).

unibz

Querying KBs – a.k.a. Ontology-mediated query answering

Query answering over KBs has been studied traditionally by considering as query language **conjunctive queries** (CQs) and their union (UCQs), under certain answer semantics.

Conjunctive query over a KB

A **CQ $\boldsymbol{q}$ over a KB $\mathcal{K}$** is a first-order query of the form

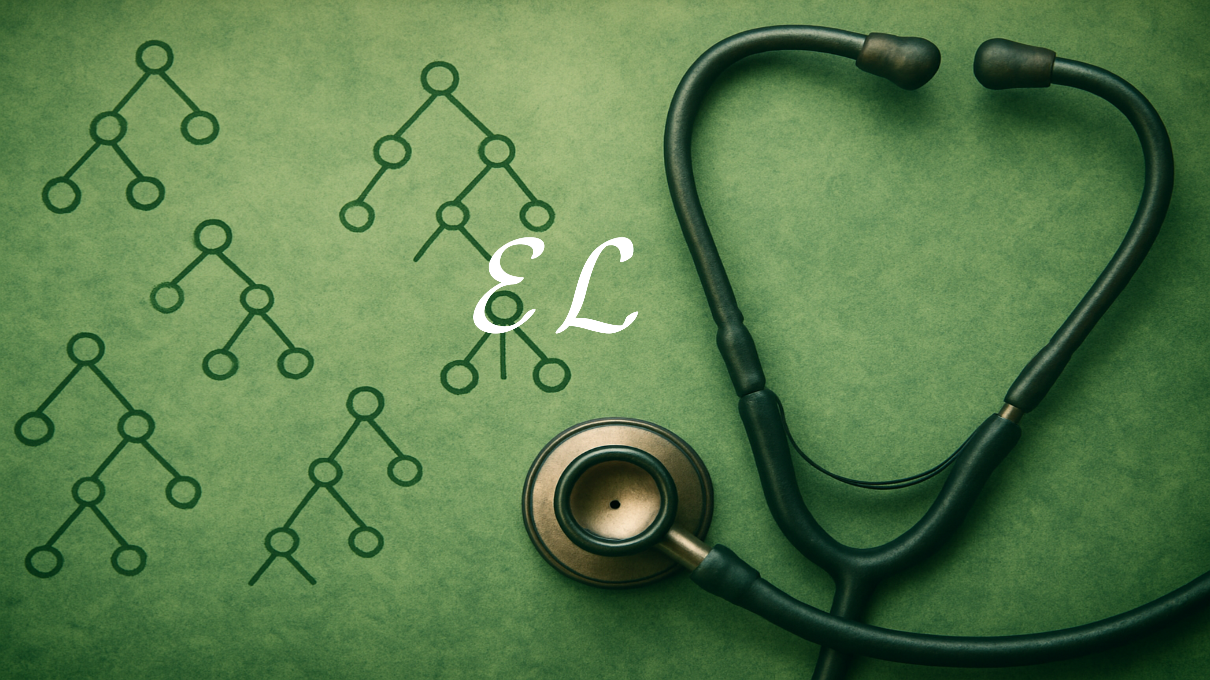
$$\boldsymbol{q}(\vec{x}) \;\leftarrow\; \exists\vec{y}.\, E_1(\vec{x},\vec{y}) \wedge \cdots \wedge E_n(\vec{x},\vec{y})$$

where each $E_i(\vec{x},\vec{y})$ is an atom that:

- has as predicate symbol a concept or role name of $\mathcal{K}$, and
- may use the answer variables $\vec{x}$, the existentially quantified variables $\vec{y}$, and constants.

Certain answer semantics

The **certain answers** $\mathsf{cert}(\boldsymbol{q},\mathcal{O})$ to a query $\boldsymbol{q}(\vec{x})$ over a KB $\mathcal{K}$ are the **tuples $\vec{c}$ of constants** that are logically implied to satisfy the query, i.e., **such that** $\mathcal{K} \models \boldsymbol{q}(\vec{c})$.


EL

The basic $\mathcal{EL}$

Essentially, $\mathcal{EL}$ is half of $\mathcal{ALC}$:

- It supports existential restrictions $\exists P.C$, but not universal ones.
- It supports conjunction $C_1 \sqcap C_2$, but not disjunction.
- Of course, no negation (in some variants $\bot$ is allowed).

$\mathcal{EL}$ concepts and TBoxes

$\mathcal{EL}$ concepts are defined by the grammar: (A denotes a concept name, P a role name)

$$C, C' \longrightarrow A \mid \top \mid C \sqcap C' \mid \exists P.C$$

An $\mathcal{EL}$ TBox consists of concept inclusions $C \sqsubseteq C'$.

Normal form for TBoxes: (A, A_i, B concept names or $\top$)

$$A \sqsubseteq B \qquad A_1 \sqcap A_2 \sqsubseteq B \qquad A \sqsubseteq \exists P.B \qquad \exists P.A \sqsubseteq B$$

Applications of $\mathcal{EL}$

In many applications, existential restrictions and conjunction seem to play a central role.

E.g., medical and Life Sciences KBs / ontologies rely on this kind of axioms.

Example $\mathcal{EL}$ TBox:

$$
\begin{array}{rcl}
\mathsf{ViralPneumonia} & \sqsubseteq & \exists \mathsf{CausativeAgent}.\mathsf{Virus} \\
\mathsf{ViralPneumonia} & \sqsubseteq & \mathsf{InfectiousPneumonia} \\
\mathsf{InfectiousPneumonia} & \sqsubseteq & \mathsf{Pneumonia} \sqcap \mathsf{InfectiousDisease} \\
\mathsf{Pneumonia} & \sqsubseteq & \exists \mathsf{AssociatedMorphology}.\mathsf{Inflammation} \\
\mathsf{Pneumonia} & \sqsubseteq & \exists \mathsf{FindingSite}.\mathsf{Lung}
\end{array}
$$

unibz

Some prominent ontologies in the $\mathcal{EL}$ Family

- SNOMED CT (*Systematized Nomenclature of Medicine – Clinical Terms*) [http://www.ihtsdo.org]
- Large fragments of the GALEN ontology (*Generalized Architecture for Languages, Encyclopedias and Nomenclatures in medicine*) [http://www.openclinical.org/prj_galen.html]
- The Gene Ontology, and ontologies for biology with the aim of "standardizing the representation of gene and gene product attributes across species and databases" [http://www.geneontology.org/]
- Many ontologies in the BioPortal repository [http://bioportal.bioontology.org] and the Open Biomedical Ontologies (OBO) Foundry [http://www.obofoundry.org]

DL-Lite

DL-Lite TBox

DL-Lite allows for **TBox** assertions of the following forms: (A denotes a concept name, P a role name)

DL Syntax	FOL Counterpart	Example	$\mathcal{L}$	Intuition	
$A_1 \sqsubseteq A_2$	$\forall x.A_1(x) \rightarrow A_2(x)$	Actor ⊑ Person	core	ISA on concepts	Inclusion assertions
$\exists P \sqsubseteq A$	$\forall x,y.P(x,y) \rightarrow A(x)$	∃playsIn ⊑ Actor	core	domain of role	Inclusion assertions
$\exists P^- \sqsubseteq A$	$\forall x,y.P(y,x) \rightarrow A(x)$	∃playsIn⁻ ⊑ Movie	core	range of role	Inclusion assertions
$A \sqsubseteq \exists P$	$\forall x.A(x) \rightarrow \exists y.P(x,y)$	Actor ⊑ ∃playsIn	core	mandatory participation	Inclusion assertions
$P_1 \sqsubseteq P_2$	$\forall x,y.P_1(x,y) \rightarrow P_2(x,y)$	hasFather ⊑ hasParent	$\mathcal{R}$	ISA on roles	Inclusion assertions
$P_1 \sqsubseteq P_2^-$	$\forall x,y.P_1(x,y) \rightarrow P_2(y,x)$	hasFather ⊑ hasChild⁻	$\mathcal{R}$	ISA on roles	Inclusion assertions
...	...	...	...	...	Inclusion assertions
$A_1 \sqsubseteq \neg A_2$	$\forall x.A_1(x) \rightarrow \neg A_2(x)$	Manager ⊑ ¬Actor	core	Disjointness assertions	
$A \sqsubseteq \neg\exists P$	$\forall x.A(x) \rightarrow \neg\exists y.P(x,y)$	Manager ⊑ ¬∃playsIn	core	Disjointness assertions	
$P_1 \sqsubseteq \neg P_2$	$\forall x,y.P_1(x,y) \rightarrow \neg P_2(x,y)$	hasParent ⊑ ¬hasSibling	$\mathcal{R}$	Disjointness assertions	
...	...	...	...	Disjointness assertions	
(**funct** P)	$\forall x,y,z.P(x,y) \land P(x,z) \rightarrow y = z$	(**funct** hasFather)	$\mathcal{F}$	Functionality assertions	
(**funct** P^-)	$\forall x,y,z.P(y,x) \land P(z,x) \rightarrow y = z$	(**funct** manages⁻)	$\mathcal{F}$	Functionality assertions	

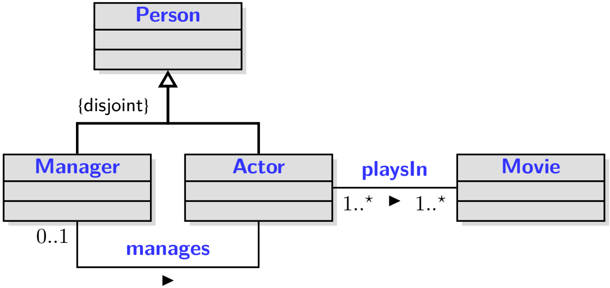
DL-Lite KB: Example and relationship with UML Class Diagrams

TBox $\mathcal{T}$:

$$\begin{array}{rcl}
\text{Manager} & \sqsubseteq & \text{Person} \\
\text{Actor} & \sqsubseteq & \text{Person} \\
\text{Manager} & \sqsubseteq & \neg\text{Actor} \\
\exists\text{playsIn} & \sqsubseteq & \text{Actor} \\
\exists\text{playsIn}^- & \sqsubseteq & \text{Movie} \\
\text{Actor} & \sqsubseteq & \exists\text{playsIn} \\
\text{Movie} & \sqsubseteq & \exists\text{playsIn}^- \\
\exists\text{manages} & \sqsubseteq & \text{Manager} \\
\exists\text{manages}^- & \sqsubseteq & \text{Actor} \\
 & (\textbf{funct}\ \text{manages}^-) &
\end{array}$$

ABox $\mathcal{A}$:

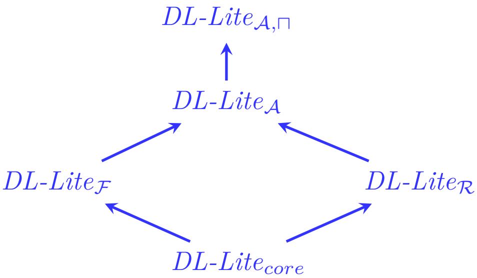
Actor(Keanu)
manages(Bill, Carrie-Anne)



The *DL-Lite* Family

The DLs of the *DL-Lite* Family share the property that:

answering UCQs is FO-rewritable.



From [C. *et al.*, 2007b, JAR].

Note: $DL\text{-}Lite_{\mathcal{A}}$ and $DL\text{-}Lite_{\mathcal{A},\sqcap}$ combine functionality and role inclusions. To guarantee that CQ answering is FO-rewritable, we need to impose that no functional role can have a sub-role.

What makes $\mathcal{EL}$ and $\mathit{DL\text{-}Lite}$ lightweight?

Universal models

For interpretations $\mathcal{I}_1$ and $\mathcal{I}_2$, a homomorphism $\mathcal{I}_1 \rightsquigarrow \mathcal{I}_2$ is a mapping from the domain of $\mathcal{I}_1$ to that of $\mathcal{I}_2$ that preserves predicates and constants.

Universal model (a.k.a. canonical model) of a KB $\mathcal{K}$

Is a model $\mathcal{I}_\mathcal{K}$ s.t. for every model $\mathcal{I}$ of $\mathcal{K}$, there is a homomorphism from $\mathcal{I}_\mathcal{K}$ to $\mathcal{I}$.

- Answers to (U)CQs are preserved under homomorphisms, i.e.,

$$\text{if} \quad \vec{c} \in \text{ans}(\boldsymbol{q}, \mathcal{I}_1) \quad \text{and} \quad \mathcal{I}_1 \rightsquigarrow \mathcal{I}_2, \qquad \text{then} \qquad \vec{c} \in \text{ans}(\boldsymbol{q}, \mathcal{I}_2).$$

- Hence, if $\mathcal{K}$ admits a universal model $\mathcal{I}_\mathcal{K}$, we can "use it" to compute the certain answers:

$$\vec{c} \in \text{cert}(q, \mathcal{K}) \qquad \text{iff} \qquad \vec{c} \in \text{ans}(q, \mathcal{I}) \text{ for every model } \mathcal{I} \text{ of } \mathcal{K} \qquad \text{iff} \qquad \vec{c} \in \text{ans}(q, \mathcal{I}_\mathcal{K})$$

Horn DLs

A DL is **Horn**, if every satisfiable KB $\mathcal{K}$ has a universal model $\mathcal{I}_{\mathcal{K}}$.

Theorem ([Baader *et al.*, 2005, IJCAI; 2008, OWLED])

Every satisfiable $\mathcal{EL}$ KB $\mathcal{K}$ has a universal model. $\leadsto$ $\mathcal{EL}$ is Horn!

Theorem ([C. *et al.*, 2005, AAAI; 2008, JAR])

Every satisfiable *DL-Lite* KB $\mathcal{K}$ has a universal model. $\leadsto$ *DL-Lite* is Horn!

Can we directly exploit the universal model for reasoning / query answering?

Not necessarily so, since **the universal model might be infinite!**

Other features of making a DL lightweight

Being Horn alone does not ensure that the DL is lightweight and that reasoning is efficient.

Further factors that play a role:

- Limited interaction between different constructs.

  E.g., In $DL\text{-}Lite_{\mathcal{A}}$, where both role inclusions and functionality are allowed, we need to ensure that no functional role has a subrole.
  If we relax this restriction, TBox complexity jumps from NLogSpace / PTime to ExpTime, and data complexity from AC^0 to PTime /coNP.

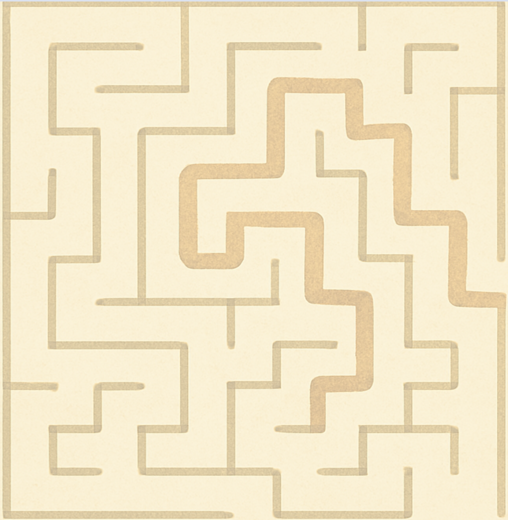
- Limited ability to propagate information along structures.

  E.g., $\mathcal{EL}$ does not allow for inverse roles, and with inverse roles complexity jumps to ExpTime.

- Do you have any additional suggestion, intuition, insight?

Outline

1. A Bit of History
2. Lighweight DLs
3. Reasoning
4. Extensions



Reasoning in $\mathcal{EL}$

Satisfiability in $\mathcal{EL}$

Very easy to see that:

- Every $\mathcal{EL}$ concept C is satisfiable: C induces a description tree, which represents of a model.
- Every $\mathcal{EL}$ concept C is satisfiable w.r.t. every $\mathcal{EL}$ TBox and w.r.t. every KB.

$\leadsto$ We concentrate on deciding subsumption.

Canonical model in $\mathcal{EL}$ – For plain concept subsumption

Let sub(C) be the set of subconcepts of C. We define:

$$\text{ex}(C) = \{C\} \cup \{D \mid \exists P.D \in \text{sub}(C)\}$$

Define the interpretation $\mathcal{I}_C = (\Delta^{\mathcal{I}_C}, \cdot^{\mathcal{I}_C})$ with:

$$\begin{array}{rcl} \Delta^{\mathcal{I}_C} & = & \{d_D \mid D \in \text{ex}(C)\} \\ A^{\mathcal{I}_C} & = & \{d_D \mid D = D' \sqcap A \;\; \text{(i.e., } A \text{ is a conjunct of } D\text{)}\} \\ P^{\mathcal{I}_C} & = & \{(d_D, d_{D'}) \mid D = D'' \sqcap \exists P.D' \;\; \text{(i.e., } \exists P.D' \text{ is a conjunct of } D\text{)}\} \end{array}$$

One can show that $\mathcal{I}_C$ is a **universal model** of C.

We can use this to check plain subsumption $C \sqsubseteq D$. Indeed: $C \sqsubseteq D$ iff $d_C \in D^{\mathcal{I}_C}$

$\leadsto$ PTime **algorithm**:
1. Build $\mathcal{I}_C$ in PTime (because $|\Delta^{\mathcal{I}_C}| \leq |C|$).
2. Test in PTime whether $d_C \in D^{\mathcal{I}_C}$.

Subsumption in $\mathcal{EL}$ w.r.t. TBoxes

We build again a **canonical model** $\mathcal{I}_\mathcal{T}$ of $\mathcal{T}$.

- Start from $\mathcal{I}_0$ that only has $d_A \in A^{\mathcal{I}_\mathcal{T}}$ for all A, and all roles are empty.
- Add objects to concepts and pairs of objects to roles to satisfy the inclusions.
- Always reuse d_A to satisfy $\exists P.A$ $\rightsquigarrow$ Only one witness $d_A \in A^{\mathcal{I}_\mathcal{T}}$ for each concept name A.

$\mathcal{I}_\mathcal{T}$ can be constructed in polytime. $\rightsquigarrow$ **Only polynomially many objects!**

We have that:

- $\mathcal{I}_\mathcal{T}$ is a model of $\mathcal{T}$ – Note: $\mathcal{I}_\mathcal{T}$ is not a universal model, but it suffices for checking subsumption.
- Indeed, for every pair A, B of concept names, $\mathcal{T} \models A \sqsubseteq B$ iff $d_A \in B^{\mathcal{I}_\mathcal{T}}$.

To decide subsumption of arbitrary concepts:

$$\mathcal{T} \models C \sqsubseteq D \quad \text{iff} \quad \mathcal{T} \cup \{A_C \sqsubseteq C,\ D \sqsubseteq A_D\} \models A_C \sqsubseteq A_D$$

Theorem

Subsumption (w.r.t. a TBox) in $\mathcal{EL}$ can be decided in $\textsc{PTime}$.

Example of the canonical model construction (1/2) [Credits to Magdalena & Meghyn]

TBox $\mathcal{T}$:

$$
\begin{array}{rcl}
\text{PenneArrabbiata} & \sqsubseteq & \exists\text{hasIngred}.\text{Penne} \\
\text{Penne} & \sqsubseteq & \text{Pasta} \\
\text{PenneArrabbiata} & \sqsubseteq & \exists\text{hasIngred}.\text{ArrabSauce} \\
\text{ArrabSauce} & \sqsubseteq & \exists\text{hasIngred}.\text{Peperoncino} \\
\text{Peperoncino} & \sqsubseteq & \text{Spicy} \\
\text{PizzaCalabrese} & \sqsubseteq & \exists\text{hasIngred}.\text{Nduia} \\
\text{Nduia} & \sqsubseteq & \text{Spicy}
\end{array}
$$

We saturate the TBox, so that it additionally contains:

$$
\begin{array}{rcl}
\text{PenneArrabbiata} & \sqsubseteq & \exists\text{hasIngred}.(\text{Penne} \sqcap \text{Pasta}) \\
\text{ArrabSauce} & \sqsubseteq & \exists\text{hasIngred}.(\text{Peperoncino} \sqcap \text{Spicy}) \\
\text{PizzaCalabrese} & \sqsubseteq & \exists\text{hasIngred}.(\text{Nduia} \sqcap \text{Spicy})
\end{array}
$$

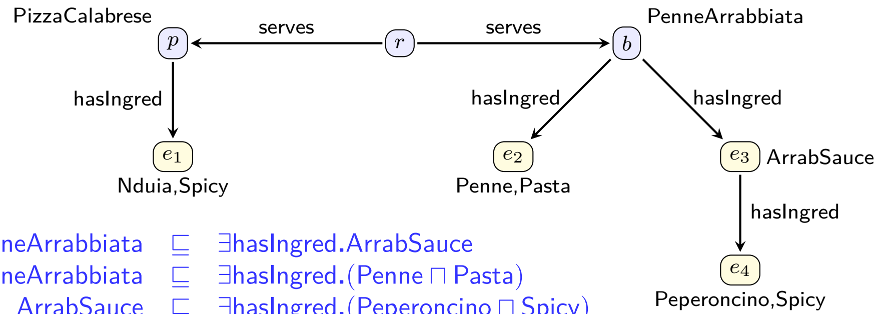
We consider also an ABox $\mathcal{A}$:

$$
\text{PizzaCalabrese}(p) \qquad \text{serves}(r,p) \qquad \text{serves}(r,b) \qquad \text{PenneArrabbiata}(b)
$$

Example of the canonical model construction (2/2) [Credits to Magdalena & Meghyn]

The canonical model $\mathcal{I}_\mathcal{K}$ For a KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ contains the ABox $\mathcal{A}$ and is **closed under inclusions**.

$\mathsf{PizzaCalabrese}(p) \qquad \mathsf{serves}(r,p) \qquad \mathsf{serves}(r,b) \qquad \mathsf{PenneArrabbiata}(b)$



$$
\begin{array}{rcl}
\mathsf{PenneArrabbiata} & \sqsubseteq & \exists \mathsf{hasIngred}.\mathsf{ArrabSauce} \\
\mathsf{PenneArrabbiata} & \sqsubseteq & \exists \mathsf{hasIngred}.(\mathsf{Penne} \sqcap \mathsf{Pasta}) \\
\mathsf{ArrabSauce} & \sqsubseteq & \exists \mathsf{hasIngred}.(\mathsf{Peperoncino} \sqcap \mathsf{Spicy}) \\
\mathsf{PizzaCalabrese} & \sqsubseteq & \exists \mathsf{hasIngred}.(\mathsf{Nduia} \sqcap \mathsf{Spicy})
\end{array}
$$

Notice that the **anonymous objects** witnessing existential concepts form **trees**.

Complexity of $\mathcal{EL}$

Theorem

Concept subsumption w.r.t. $\mathcal{EL}$ TBoxes is PTime-complete.

- Essentially, $\mathcal{EL}$ contains propositional Horn logic:

$$v_1 \wedge \cdots \wedge v_n \rightarrow v \quad \leadsto \quad A_{v_1} \sqcap \cdots \sqcap A_{v_n} \sqsubseteq A_v$$

- Reduction from entailment of a variable from propositional Horn theories is immediate.
- PTime-hard even with no roles / existential concepts.

Data complexity of $\mathcal{EL}$

Theorem

Instance checking in $\mathcal{EL}$ is PTIME-hard, even for a fixed TBox.

Straightforward reduction from Boolean circuit evaluation:

- The circuit is given as an ABox:
  - concept names AndGate and OrGate for the gates
  - role names leftInput and rightInput between gates.
- The input is asserted using a concept name True.
- The fixed TBox contains:

$$
\begin{aligned}
\mathsf{OrGate} \sqcap \exists \mathsf{leftInput}.\mathsf{True} &\sqsubseteq \mathsf{True}\\
\mathsf{OrGate} \sqcap \exists \mathsf{rightInput}.\mathsf{True} &\sqsubseteq \mathsf{True}\\
\mathsf{AndGate} \sqcap \exists \mathsf{leftInput}.\mathsf{True} \sqcap \exists \mathsf{rightInput}.\mathsf{True} &\sqsubseteq \mathsf{True}
\end{aligned}
$$

This means that the **data complexity** of $\mathcal{EL}$ is PTIME-complete.

unibz

Adding $\bot$ to $\mathcal{EL}$ $\leadsto$ $\mathcal{EL}^\bot$

With $\bot$, one can express in $\mathcal{EL}$ concept disjointness and create unsatisfiable concepts and roles:

$$A_1 \sqcap A_2 \sqsubseteq \bot \qquad A_3 \sqsubseteq \bot \qquad \exists P.\top \sqsubseteq \bot$$

Concept satisfiability is not trivial anymore, but can be decided in PTime.

- Simply build the canonical model of C, and return "*unsat*" iff some element must satisfy $\bot$.

In $\mathcal{EL}^\bot$, satisfiability and subsumption are inter-reducible:

- C is satisfiable w.r.t. $\langle \mathcal{T}, \mathcal{A} \rangle$ iff $\langle \mathcal{T}, \mathcal{A} \rangle \not\models C \sqsubseteq \bot$
- $\langle \mathcal{T}, \mathcal{A} \rangle \models C \sqsubseteq D$ iff $C \sqcap A_{\neg D}$ is unsatisfiable w.r.t. $\langle \mathcal{T} \cup \{A_{\neg D} \sqcap D \sqsubseteq \bot\}, \mathcal{A} \rangle$, where $A_{\neg D}$ is a fresh concept name.

Additional polynomial extensions of $\mathcal{EL}$

- Nominals $\{a\}$
- Range restrictions $\top \sqsubseteq \forall P.C$ (also written $\exists P^-.\top \sqsubseteq C$)
  (domain restrictions $\exists P.\top \sqsubseteq C$ are already expressible in $\mathcal{EL}$)
- Complex role inclusions $P_1 \circ \cdots \circ P_n \sqsubseteq P$.
- Concrete domains $p(f_1, \ldots, f_k)$, where the f_is are feature names and p is a k-ary domain predicate.

We can still adapt the canonical model construction to accommodate these features, and reasoning is still feasible in PTime.

The resulting DL is called $\mathcal{EL}^{++}$
[Baader *et al.*, 2008, OWLED]

$\mathcal{EL}^{++}$ is at the basis of the OQL 2 EL profile of OWL 2, standardized by the W3C [Motik *et al.*, 2012, W3C].

Practical Reasoning in $\mathcal{EL}$

A different approach to reasoning in $\mathcal{EL}$

In practice, $\mathcal{EL}$ (and Horn DL) reasoners adopt a different approach:

Consequence Driven Reasoning.

- Make all consequences of the provided information explicit.
- Done by adding implied axioms to the TBox until saturation.
- This is achieved by defining a suitable set of inference rules.
- Mostly used for classification: computing all subsumptions between concept names:
  - no ABox, only TBox saturation
  - sound and complete for atomic entailment, that is:

$$\mathcal{T} \models A \sqsubseteq B \qquad \text{iff} \qquad A \sqsubseteq B \text{ is in the saturated TBox}$$

- The approach can also be extended to instance checking.

Classification for $\mathcal{ELH}$: the ELK reasoner [Kazakov *et al.*, 2014, JAR]

IR1 $\dfrac{}{A \sqsubseteq A}$ **IR2** $\dfrac{}{A \sqsubseteq \top}$

CR1 $\dfrac{A \sqsubseteq B \quad B \sqsubseteq C \in \mathcal{O}}{A \sqsubseteq C}$

CR2 $\dfrac{A \sqsubseteq B \quad A \sqsubseteq C \quad B \sqcap C \sqsubseteq D \in \mathcal{O}}{A \sqsubseteq D}$

CR3 $\dfrac{A \sqsubseteq B \quad B \sqsubseteq \exists r.C \in \mathcal{O}}{A \sqsubseteq \exists r.C}$

CR4 $\dfrac{A \sqsubseteq \exists r.B \quad r \sqsubseteq s \in \mathcal{O}}{A \sqsubseteq \exists s.B}$

CR5 $\dfrac{A \sqsubseteq \exists r.B \quad B \sqsubseteq C \quad \exists r.C \sqsubseteq D \in \mathcal{O}}{A \sqsubseteq D}$

Reasoning in $DL\text{-}Lite$

Reasoning in *DL-Lite*

TBox reasoning is rather straightforward.

- A *DL-Lite* TBox is always satisfiable (possibly in a model where all atomic concepts are empty).
- Concept and role subsumption amounts to computing reachability along the concept / role hierarchy. $\rightsquigarrow$ NLogSpace, or PTime for Horn-variants.

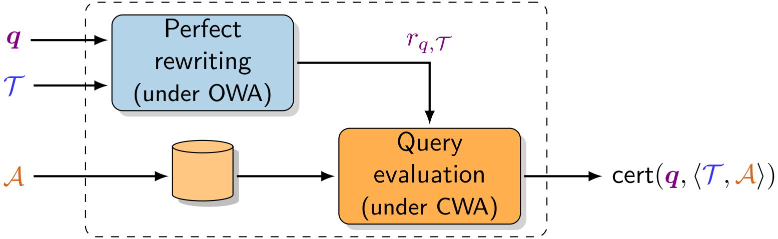
Query answering

- It is the main inference task that is of interest.
- Other TBox+ABox reasoning tasks get reduced to query answering.
- In general we cannot adopt an approach based on deriving all consequences:
  - They might necessarily be infinite – Not all variants of *DL-Lite* have the finite model property.
  - The amount of information to derive might depend on the query and cannot be determined a priory,

$\rightsquigarrow$ **Query answering by query rewriting**

unibz

Query answering by query rewriting



To deal with data efficiently, we separate the contribution of $\mathcal{A}$ from the contribution of $\boldsymbol{q}$ and $\mathcal{T}$.

$r_{q,\mathcal{T}}$ is a new query over $\mathcal{T}$, called the **perfect rewriting** of $\boldsymbol{q}$ w.r.t. $\mathcal{T}$

FO-rewritability of conjunctive query answering

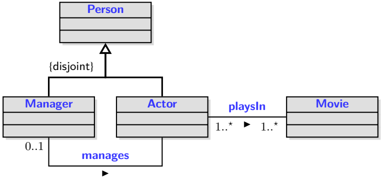
In $\mathit{DL\text{-}Lite}$, the perfect rewriting of a UCQ is always a **first-order query** (in fact a union of CQs)! Thus, **answering UCQs in $\mathit{DL\text{-}Lite}$ is in AC^0 in data complexity** (i.e., w.r.t. the ABox only).

unibz

Perfect rewriting: Example

TBox $\mathcal{T}$:

Manager	$\sqsubseteq$	Person	Actor	$\sqsubseteq$	$\exists$playsIn
Actor	$\sqsubseteq$	Person	Movie	$\sqsubseteq$	$\exists$playsIn$^-$
Manager	$\sqsubseteq$	$\neg$Actor	$\exists$manages	$\sqsubseteq$	Manager
$\exists$playsIn	$\sqsubseteq$	Actor	$\exists$manages$^-$	$\sqsubseteq$	Actor
$\exists$playsIn$^-$	$\sqsubseteq$	Movie	(**funct** manages$^-$)		



Query: $q(x) \leftarrow \exists y.\, \text{playsIn}(x, y) \wedge \text{Movie}(y)$

ABox $\mathcal{A}$: Actor(Keanu), manages(Bill, Carrie-Anne)

Perfect rewriting $r_{q,\mathcal{T}}$:
(Algorithm **PerfectRef**)

$q(x) \leftarrow \exists y.\, \text{playsIn}(x, y) \wedge \text{Movie}(y)$

⇊ Use $\exists$playsIn$^-$ $\sqsubseteq$ Movie to rewrite the atom Movie(y).

$q(x) \leftarrow \exists y.\, \text{playsIn}(x, y) \wedge \text{playsIn}(\_, y)$

⇊ Unify the two atoms playsIn(x, y) and playsIn($\_, y$).

$q(x) \leftarrow \exists y.\text{playsIn}(x, y)$

⇊ Use Actor $\sqsubseteq$ $\exists$playsIn to rewrite the atom playsIn(x, y).

$q(x) \leftarrow \text{Actor}(x)$

⇊ Use $\exists$manages$^-$ $\sqsubseteq$ Actor to rewrite the atom Actor(x).

$q(x) \leftarrow \exists y.\text{manages}(y, x)$

⇊ ...

The evaluation of $r_{q,\mathcal{T}}$ over $\mathcal{A}$ returns the set {Keanu, Carrie-Anne} $= \text{cert}(\boldsymbol{q}, \langle \mathcal{T}, \mathcal{A} \rangle)$.

Complexity of query answering in *DL-Lite*

Ontology satisfiability and all classical DL reasoning tasks are:

- PTIME in the size of the TBox.
- In AC^0 the size of the ABox, i.e., in data complexity.

In fact, reasoning can be done by constructing suitable FOL/SQL queries and evaluating them over the ABox (FOL-rewritability).

Query answering for UCQs / SPARQL queries is:

- PTIME in the size of the TBox.
- In AC^0 the size of the ABox, i.e., in data complexity.
- NP-complete in the size of the query, i.e., in combined complexity.

This is precisely the complexity of evaluating CQs in plain relational DBs.

Practical Reasoning in $DL\text{-}Lite$

The *Ontop* system [C. *et al.*, 2017, Semantic Web J.], [Xiao *et al.*, 2020, ISWC]

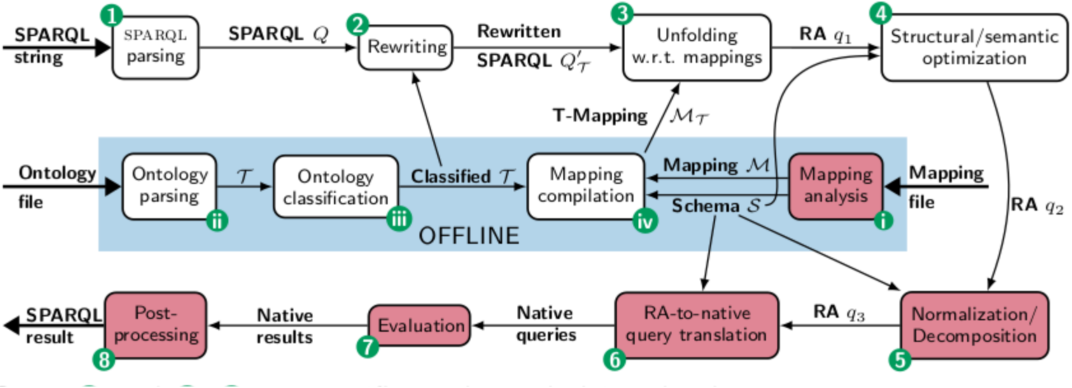
ontop

https://ontop-vkg.org/

- State-of-the-art VKG system.
- Addresses the key challenges in query answering of scalability and performance.
- Compliant with all relevant Semantic Web standards:
  RDF, RDFS, OWL 2 QL, R2RML, SPARQL, and GeoSPARQL.
- Supports all major relational DBMSs:
  Oracle, DB2, MS SQL Server, Postgres, MySQL, Teiid, Dremio, Denodo, etc.
- Open-source and released under Apache 2 license.

unibz

Query answering in *Ontop*



unibz

Outline

Beyond $\mathcal{EL}$ – Adding inverse roles

Existential restrictions with inverses behave like universal restrictions:

$$\exists P^{-}.A \sqsubseteq B \qquad \text{is equivalent to} \qquad A \sqsubseteq \forall P.B$$

- Enables alternation.
- Types of connected objects influence each other.
- Generates exponentially many types.

$\leadsto$ The canonical model construction would fail with inverses.

Theorem

Concept subsumption in $\mathcal{ELI}$ is $\textsc{ExpTime}$-complete.

ExpTime-hard extensions of $\mathcal{EL}$

Reasoning (w.r.t. arbitrary TBoxes) becomes ExpTime-hard for the following extensions of $\mathcal{EL}$:

- $\mathcal{ELU}^\bot$, which extends $\mathcal{EL}^\bot$ with disjunction.
  - We can reduce concept satisfiability w.r.t. to a TBox in $\mathcal{ALC}$ to TBox satisfiability in $\mathcal{ELU}^\bot$.
- $\mathcal{ELU}$, which extends $\mathcal{AL}$ with disjunction.
  - We can reduce concept satisfiability w.r.t. to a TBox in $\mathcal{ELU}^\bot$ to concept subsumption w.r.t. to a TBox in $\mathcal{ELU}$.
- $\mathcal{EL}^\forall$, which extends $\mathcal{EL}$ with universal restrictions $\forall P.C$.
  - We can reduce concept subsumption w.r.t. to a TBox in $\mathcal{ELU}$ to the same problem in $\mathcal{EL}^\forall$

There is no known extension of $\mathcal{EL}$ for which reasoning is between PTime and ExpTime.

unibz

Beyond *DL-Lite*: The border of FO-rewritability

[C. *et al.*, 2006, KR; 2013, AIJ]

	Left-hand side of inclusions	Right-hand side of inclusions	functionalities	Role inclusions	Data complexity of CQ answering
0	*DL-Lite*		$\surd^*$	$\surd^*$	in AC^0
1	$A \mid \exists P.A$	A	–	–	NLogSpace-hard
2	A	$A \mid \forall P.A$	–	–	NLogSpace-hard
3	A	$A \mid \exists P.A$	$\surd$	–	NLogSpace-hard
4	$A \mid \exists P.A \mid A_1 \sqcap A_2$	A	–	–	PTime-hard
5	$A \mid A_1 \sqcap A_2$	$A \mid \forall P.A$	–	–	PTime-hard
6	$A \mid A_1 \sqcap A_2$	$A \mid \exists P.A$	$\surd$	–	PTime-hard
7	$A \mid \exists P.A \mid \exists P^-.A$	$A \mid \exists P$	–	–	PTime-hard
8	$A \mid \exists P \mid \exists P^-$	$A \mid \exists P \mid \exists P^-$	$\surd$	$\surd$	PTime-hard
9	$A \mid \neg A$	A	–	–	coNP-**hard**
10	A	$A \mid A_1 \sqcup A_2$	–	–	coNP-**hard**
11	$A \mid \forall P.A$	A	–	–	coNP-**hard**

* With the “proviso” that functional roles cannot have subroles.

Extensions of *DL-Lite*

Additional constructs and language extensions have been considered, such as:

- Different forms of constraints:
  - identification assertions [C. *et al.*, 2008a, KR]
  - denial assertions [Lembo *et al.*, 2015, JWebSem]
  - epistemic constraints [C. *et al.*, 2007a, IJCAI], [Console and Lenzerini, 2020, AAAI]

  These constraints do not affect query answering, provided the ontology is satisfiable.
  However, the presence of **constraints may cause an ontology to become unsatisfiable**.

- n-ary relationships (as opposed to binary roles only) [C. *et al.*, 2006, KR; 2013, AIJ]

- Attributes and datatypes [Savkovic and C., 2012, ECAI], [Artale *et al.*, 2012, ECAI]:
  - Attributes are used to relate abstract objects to values of datatypes (such as integers, reals, strings).
  - The presence of datatypes has an impact on query answering, and restrictions need to be imposed to ensure FO-rewritability.

Extensions of $DL\text{-}Lite$ (cont'd)

Several works have studied also the **combined complexity** of satisfiability, in addition to data complexity and FO-rewritability for query answering.

Again, various language extensions have been considered.

- Number restrictions, role constructs, different types of concept inclusions, UNA yes/no
  [Artale *et al.*, 2009, JAIR]
  Inference exploits a translation into FOL over unary predicates only.

unibz

Extensions of *DL-Lite* (cont'd)

Several works have studied also th
complexity and FO-rewritability fo

Again, various language extensions

- Number restrictions, role cons
  [Artale *et al.*, 2009, JAIR]
  Inference exploits a translatio

Languages	UNA	Complexity		
		Combined complexity	Data complexity	
		Satisfiability	Instance checking	Query answering
$DL\text{-}Lite_{core}^{[\ \|\mathcal{H}]}$	yes/no	NLogSpace $\geq$ [A]	in AC^0	in AC^0
$DL\text{-}Lite_{horn}^{[\ \|\mathcal{H}]}$		P $\leq$ [Th.8.2] $\geq$ [A]	in AC^0	in AC^0 $\leq$ [C]
$DL\text{-}Lite_{krom}^{[\ \|\mathcal{H}]}$		NLogSpace $\leq$ [Th.8.2]	in AC^0	coNP $\geq$ [B]
$DL\text{-}Lite_{bool}^{[\ \|\mathcal{H}]}$		NP $\leq$ [Th.8.2] $\geq$ [A]	in AC^0 $\leq$ [Th.8.3]	coNP
$DL\text{-}Lite_{core}^{[\mathcal{F}\|\mathcal{N}\|(\mathcal{HF})\|(\mathcal{HN})]}$	yes	NLogSpace	in AC^0	in AC^0
$DL\text{-}Lite_{horn}^{[\mathcal{F}\|\mathcal{N}\|(\mathcal{HF})\|(\mathcal{HN})]}$		P $\leq$ [Th.5.8, 5.13]	in AC^0	in AC^0 $\leq$ [Th.7.1]
$DL\text{-}Lite_{krom}^{[\mathcal{F}\|\mathcal{N}\|(\mathcal{HF})\|(\mathcal{HN})]}$		NLogSpace $\leq$ [Th.5.7,5.13]	in AC^0	coNP
$DL\text{-}Lite_{bool}^{[\mathcal{F}\|\mathcal{N}\|(\mathcal{HF})\|(\mathcal{HN})]}$		NP $\leq$ [Th.5.6, 5.13]	in AC^0 $\leq$ [Cor.6.2]	coNP
$DL\text{-}Lite_{core/horn}^{[\mathcal{F}\|(\mathcal{HF})]}$	no	P $\leq$ [Cor.8.8] $\geq$ [Th.8.7]	P $\geq$ [Th.8.7]	P
$DL\text{-}Lite_{krom}^{[\mathcal{F}\|(\mathcal{HF})]}$		P $\leq$ [Cor.8.8]	P	coNP
$DL\text{-}Lite_{bool}^{[\mathcal{F}\|(\mathcal{HF})]}$		NP	P $\leq$ [Cor.8.8]	coNP
$DL\text{-}Lite_{core/horn}^{[\mathcal{N}\|(\mathcal{HN})]}$		NP $\geq$ [Th.8.4]	coNP $\geq$ [Th.8.4]	coNP
$DL\text{-}Lite_{krom/bool}^{[\mathcal{N}\|(\mathcal{HN})]}$		NP $\leq$ [Th.8.5]	coNP	coNP
$DL\text{-}Lite_{core/horn}^{\mathcal{HF}}$	yes/no	ExpTime $\geq$ [Th.5.10]	P $\geq$ [Th.6.7]	P $\leq$ [D]
$DL\text{-}Lite_{krom/bool}^{\mathcal{HF}}$		ExpTime	coNP $\geq$ [Th.6.5]	coNP
$DL\text{-}Lite_{core/horn}^{\mathcal{HN}}$		ExpTime	coNP $\geq$ [Th.6.6]	coNP
$DL\text{-}Lite_{krom/bool}^{\mathcal{HN}}$		ExpTime $\leq$ [F]	coNP	coNP $\leq$ [E]

Extensions of *DL-Lite* (cont'd)

Several works have studied also the **combined complexity** of satisfiability, in addition to data complexity and FO-rewritability for query answering.

Again, various language extensions have been considered.

- Number restrictions, role constructs, different types of concept inclusions, UNA yes/no [Artale *et al.*, 2009, JAIR]
  Inference exploits a translation into FOL over unary predicates only.

- Complex role inclusion axioms [Kontchakov *et al.*, 2019, DL]:
  The complexity of query answering ranges from FO-rewritable to undecidable.

Extensions of *DL-Lite* (cont'd)

Several works have studied als
complexity and FO-rewritabili

Again, various language exten

- Number restrictions, role
  [Artale *et al.*, 2009, JAIR]
  Inference exploits a trans

Table 1. Combined complexity of satisfiability checking (all bounds are tight). For the languages $DL\text{-}Lite^{r}_{c}$ in the grey cells, the complexity is the same as for $DL\text{-}Lite^{r}_{r}$ as concept inclusions in $\boldsymbol{c}$ can be expressed by means of role inclusions in $\boldsymbol{r}$.

CIs \ RIs	Bool	guarded Bool	Horn	Krom	core
Bool	NEXPTIME (Th. 3)	EXPTIME (Th. 3)	NP (Th. 6)	NP (Th. 4)	NP [1]
Horn			P (Th. 6)	NP (Th. 4)	P [10]
Krom			P (Th. 6)	NL (Th. 4)	NL [1]
core					NL [9]

- Complex role inclusion axioms [Kontchakov *et al.*, 2019, DL]:
  The complexity of query answering ranges from FO-rewritable to undecidable.

unibz

Extensions of *DL-Lite* (cont'd)

Several works have studied also the **combined complexity** of satisfiability, in addition to data complexity and FO-rewritability for query answering.

Again, various language extensions have been considered.

- Number restrictions, role constructs, different types of concept inclusions, UNA yes/no [Artale *et al.*, 2009, JAIR]
  Inference exploits a translation into FOL over unary predicates only.

- Complex role inclusion axioms [Kontchakov *et al.*, 2019, DL]:
  The complexity of query answering ranges from FO-rewritable to undecidable.

- Temporal extensions [Artale *et al.*, 2007, TIME], [Artale *et al.*, 2013, IJCAI], [Artale *et al.*, 2014, TOCL]
  - The most expressive variants (with temporalized concepts and roles) are typically undecidable.
  - By carefully restricting the temporal operators and the form of inclusions, reasoning and query answering become decidable.

unibz

Reasoning in (lightweight) DLs beyond satisfiability and query answering

- Meta-modeling and meta-reasoning [Lenzerini *et al.*, 2016b, KR, IJCAI], [Lenzerini *et al.*, 2021, AIJ]
- Bag semantics, aggregation operators, and counting [Nikolaou *et al.*, 2019, AIJ], [Bienvenu *et al.*, 2020, IJCAI], [C. *et al.*, 2020, IJCAI]
- Explanation and provenance [Borgida *et al.*, 2008, ODBASE], [C. *et al.*, 2013b, JAIR], [Bourgaux and Ozaki, 2019, AAAI], [C. *et al.*, 2019, IJCAI]
- Inconsistency tolerant reasoning [Lembo and Ruzzi, 2007], [Lembo *et al.*, 2015, JWebSem], [Bienvenu and Bourgaux, 2016, RW], [Bienvenu *et al.*, 2019, JAIR], [Baader *et al.*, 2023, JELIA]
- KB and knowledge graph embeddings [Lacerda *et al.*, 2024, TGDK], [Bourgaux *et al.*, 2024, KR]
- . . . and many others, e.g., finite model reasoning [Rosati, 2008, ESWC], view-based query answering [C. *et al.*, 2008b, KR], query inseparability [Konev *et al.*, 2011, AAAI], [Botoeva *et al.*, 2014, KR], unification [Baader and Gil, 2024, IJCAR], mining KBs [Guimarães *et al.*, 2023, JAIR], . . .

Typical statement that we find in many papers:

We look at problem XYZ, . . . focusing on DLs of the $\mathcal{EL}$ and/or $\mathit{DL\text{-}Lite}$ families.

Semantic Web

- **OWL 2**, the second version of the Web Ontology Language, was developed by the W3C shortly after the introduction of $\mathcal{EL}$ and $\textit{DL-Lite}$.
- OWL 2 comes with three **profiles**[1], i.e., sub-languages tailored to specific needs.

> "The **OWL 2 EL profile** is designed as a subset of OWL 2 that is particularly suitable for applications employing ontologies that define very large numbers of classes and/or properties, captures the expressive power used by many such ontologies, and for which ontology consistency, class expression subsumption, and instance checking can be decided in polynomial time. For example, OWL 2 EL provides class constructors that are sufficient to express the very large biomedical ontology SNOMED CT."

> "The **OWL 2 QL profile** is designed so that sound and complete query answering is in AC^0 with respect to the size of the data, while providing many of the main features necessary to express conceptual models such as UML class diagrams and ER diagrams. [...] It is designed so that data stored in a standard relational database system can be queried through an ontology via a simple rewriting mechanism. [...] $\textit{DL-Lite}_{\mathcal{R}}$ provides the logical underpinning for OWL 2 QL."

- OWL 2 QL/ $\textit{DL-Lite}_{\mathcal{R}}$ is also an extension of the RDF schema language RDFS.

[1] http://www.w3.org/TR/owl2-profiles/

unibz

Conclusions

$\mathcal{EL}$ and $DL\text{-}Lite$ are still the subject of many investigations.

Research:

data privacy, ontology-mediated querying, meta-modeling, abstraction, data quality, virtual knowledge graphs, explainable AI, KB embeddings, ...

Exploitation:

- **ELK Reasoner**, University of Ulm.
- **OBDA Systems**, start-up of Sapienza University of Rome, 2017.
  https://www.obdasystems.com/
- ONTOPIC spin-off of the Free University of Bozen-Bolzano, 2019.
  https://ontopic.ai/

unibz


Thank you!

References I

[Artale *et al.*, 2007] Alessandro Artale, Roman Kontchakov, Carsten Lutz, Frank Wolter, and Michael Zakharyaschev.
Temporalising tractable description logics.
In *Proc. of the 14th Int. Symp. on Temporal Representation and Reasoning (TIME)*, pages 11–22, 2007.

[Artale *et al.*, 2009] Alessandro Artale, Diego C., Roman Kontchakov, and Michael Zakharyaschev.
The *DL-Lite* family and relations.
J. of Artificial Intelligence Research, 36:1–69, 2009.

[Artale *et al.*, 2012] Alessandro Artale, Roman Kontchakov, and Vladislav Ryzhikov.
DL-Lite with attributes and datatypes.
In *Proc. of the 20th Eur. Conf. on Artificial Intelligence (ECAI)*, volume 242 of *Frontiers in Artificial Intelligence and Applications*, pages 61–66. IOS Press, 2012.

[Artale *et al.*, 2013] Alessandro Artale, Roman Kontchakov, Frank Wolter, and Michael Zakharyaschev.
Temporal description logic for ontology-based data access.
In *Proc. of the 23rd Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 711–717. AAAI Press, 2013.

[Artale *et al.*, 2014] Alessandro Artale, Roman Kontchakov, Vladislav Ryzhikov, and Michael Zakharyaschev.
A cookbook for temporal conceptual data modelling with description logics.
ACM Trans. on Computational Logic, 15(3):25:1–25:50, 2014.

References II

[Baader and Gil, 2024] Franz Baader and Oliver Fernández Gil.
Unification in the description logic $\text{ELH}_{\mathcal{R}+}$ without the top concept modulo cycle-restricted ontologies.
In *Proc. of the 12th Int. Joint Conf. on Automated Reasoning (IJCAR)*, volume 14740 of *Lecture Notes in Computer Science*, pages 279–297. Springer, 2024.

[Baader and Hanschke, 1991] Franz Baader and Philipp Hanschke.
A schema for integrating concrete domains into concept languages.
In *Proc. of the 12th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 452–457, 1991.

[Baader *et al.*, 2005] Franz Baader, Sebastian Brandt, and Carsten Lutz.
Pushing the $\mathcal{EL}$ envelope.
In *Proc. of the 19th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 364–369, 2005.

[Baader *et al.*, 2008] Franz Baader, Sebastian Brandt, and Carsten Lutz.
Pushing the $\mathcal{EL}$ envelope further.
In Kendall Clark and Peter F. Patel-Schneider, editors, *Proc. of the 4th Int. Workshop on OWL: Experiences and Directions (OWLED)*, 2008.

References III

[Baader *et al.*, 2023] Franz Baader, Patrick Koopmann, and Francesco Kriegel.
Optimal repairs in the description logic EL revisited.
In *Proc. of the 18th Eur. Conf. on Logics in Artificial Intelligence (JELIA)*, volume 14281 of *Lecture Notes in Computer Science*, pages 11–34. Springer, 2023.

[Baader, 1990] Franz Baader.
Terminological cycles in KL-ONE-based knowledge representation languages.
In *Proc. of the 8th Nat. Conf. on Artificial Intelligence (AAAI)*, pages 621–626, Boston (Ma, USA), 1990.

[Baader, 1991] Franz Baader.
Augmenting concept languages by transitive closure of roles: An alternative to terminological cycles.
In *Proc. of the 12th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, 1991.

[Bienvenu and Bourgaux, 2016] Meghyn Bienvenu and Camille Bourgaux.
Inconsistency-tolerant querying of description logic knowledge bases.
In Jeff Z. Pan, Diego C., Thomas Eiter, Ian Horrocks, Michael Kifer, Fangzhen Lin, and Yuting Zhao, editors, *Reasoning Web: Logical Foundation of Knowledge Graph Construction and Query Answering – 12th Int. Summer School Tutorial Lectures (RW)*, volume 9885 of *Lecture Notes in Computer Science*, pages 156–202. Springer, 2016.

References IV

[Bienvenu *et al.*, 2019] Meghyn Bienvenu, Camille Bourgaux, and François Goasdoué.
Computing and explaining query answers over inconsistent DL-Lite knowledge bases.
J. of Artificial Intelligence Research, 64:563–644, 2019.

[Bienvenu *et al.*, 2020] Meghyn Bienvenu, Quentin Manière, and Michaël Thomazo.
Answering counting queries over DL-Lite ontologies.
In Christian Bessiere, editor, *Proc. of the 29th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 1608–1614. IJCAI Org., 2020.

[Borgida *et al.*, 1989] Alexander Borgida, Ronald J. Brachman, Deborah L. McGuinness, and Lori Alperin Resnick.
CLASSIC: A structural data model for objects.
In *Proc. of the 10th ACM Int. Conf. on Management of Data (SIGMOD)*, pages 59–67, 1989.

[Borgida *et al.*, 2008] Alexander Borgida, Diego C., and Mariano Rodriguez-Muro.
Explanation in the *DL-Lite* family of description logics.
In *Proc. of the 7th Int. Conf. on Ontologies, DataBases, and Applications of Semantics (ODBASE)*, volume 5332 of *Lecture Notes in Computer Science*, pages 1440–1457. Springer, 2008.

References V

[Botoeva *et al.*, 2014] Elena Botoeva, Roman Kontchakov, Vladislav Ryzhikov, Frank Wolter, and Michael Zakharyaschev.
Query inseparability for description logic knowledge bases.
In *Proc. of the 14th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR)*, pages 238–247. AAAI Press, 2014.

[Bourgaux and Ozaki, 2019] Camille Bourgaux and Ana Ozaki.
Querying attributed DL-Lite ontologies using provenance semirings.
In *Proc. of the 33rd AAAI Conf. on Artificial Intelligence (AAAI)*, pages 2719–2726. AAAI Press, 2019.

[Bourgaux *et al.*, 2024] Camille Bourgaux, Ricardo Guimarães, Raoul Koudijs, Victor Lacerda, and Ana Ozaki.
Knowledge base embeddings: Semantics and theoretical properties.
In *Proc. of the 21st Int. Conf. on Principles of Knowledge Representation and Reasoning (KR)*, 2024.

[Brachman and Levesque, 1984] Ronald J. Brachman and Hector J. Levesque.
The tractability of subsumption in frame-based description languages.
In *Proc. of the 4th Nat. Conf. on Artificial Intelligence (AAAI)*, pages 34–37, 1984.

References VI

[Brachman and Levesque, 1985] Ronald J. Brachman and Hector J. Levesque, editors.
Readings in Knowledge Representation.
Morgan Kaufmann, 1985.

[C. *et al.*, 1995] Diego C., Giuseppe De Giacomo, and Maurizio Lenzerini.
Structured objects: Modeling and reasoning.
In *Proc. of the 4th Int. Conf. on Deductive and Object-Oriented Databases (DOOD)*, volume 1013 of *Lecture Notes in Computer Science*, pages 229–246. Springer, 1995.

[C. *et al.*, 1998] Diego C., Giuseppe De Giacomo, and Maurizio Lenzerini.
On the decidability of query containment under constraints.
In *Proc. of the 17th ACM Symp. on Principles of Database Systems (PODS)*, pages 149–158, 1998.

[C. *et al.*, 2005] Diego C., Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati.
DL-Lite: Tractable description logics for ontologies.
In *Proc. of the 20th Nat. Conf. on Artificial Intelligence (AAAI)*, pages 602–607, 2005.

[C. *et al.*, 2006] Diego C., Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati.
Data complexity of query answering in description logics.
In *Proc. of the 10th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR)*, pages 260–270, 2006.

References VII

[C. *et al.*, 2007a] Diego C., Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati.
EQL-Lite: Effective first-order query processing in description logics.
In *Proc. of the 20th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 274–279, 2007.

[C. *et al.*, 2007b] Diego C., Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati.
Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family.
J. of Automated Reasoning, 39(3):385–429, 2007.

[C. *et al.*, 2008a] Diego C., Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati.
Path-based identification constraints in description logics.
In *Proc. of the 11th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR)*, pages 231–241, 2008.

[C. *et al.*, 2008b] Diego C., Giuseppe De Giacomo, Maurizio Lenzerini, and Riccardo Rosati.
View-based query answering over description logic ontologies.
In *Proc. of the 11th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR)*, pages 242–251, 2008.

[C. *et al.*, 2013a] Diego C., Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati.
Data complexity of query answering in description logics.
Artificial Intelligence, 195:335–360, 2013.

References VIII

[C. *et al.*, 2013b] Diego C., Magdalena Ortiz, Mantas Simkus, and Giorgio Stefanoni.
Reasoning about explanations for negative query answers in *DL-Lite*.
J. of Artificial Intelligence Research, 48:635–669, 2013.

[C. *et al.*, 2017] Diego C., Benjamin Cogrel, Sarah Komla-Ebri, Roman Kontchakov, Davide Lanti, Martin Rezk, Mariano Rodriguez-Muro, and Guohui Xiao.
Ontop: Answering SPARQL queries over relational databases.
Semantic Web J., 8(3):471–487, 2017.

[C. *et al.*, 2019] Diego C., Davide Lanti, Ana Ozaki, Rafael Peñaloza, and Guohui Xiao.
Enriching ontology-based data access with provenance.
In *Proc. of the 28th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 1616–1623. IJCAI Org., 2019.

[C. *et al.*, 2020] Diego C., Julien Corman, Davide Lanti, and Simon Razniewski.
Counting query answers over a DL-Lite knowledge base.
In Christian Bessiere, editor, *Proc. of the 29th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 1658–1666. IJCAI Org., 2020.

References IX

[C., 1996] Diego C.
Finite model reasoning in description logics.
In *Proc. of the 5th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR)*, pages 292–303, 1996.

[Console and Lenzerini, 2020] Marco Console and Maurizio Lenzerini.
Epistemic integrity constraints for ontology-based data management.
In *Proc. of the 34th AAAI Conf. on Artificial Intelligence (AAAI)*, pages 2790–2797, 2020.

[De Giacomo and Lenzerini, 1994a] Giuseppe De Giacomo and Maurizio Lenzerini.
Boosting the correspondence between description logics and propositional dynamic logics.
In *Proc. of the 12th Nat. Conf. on Artificial Intelligence (AAAI)*, pages 205–212, 1994.

[De Giacomo and Lenzerini, 1994b] Giuseppe De Giacomo and Maurizio Lenzerini.
Concept language with number restrictions and fixpoints, and its relationship with μ-calculus.
In *Proc. of the 11th Eur. Conf. on Artificial Intelligence (ECAI)*, pages 411–415, 1994.

[Donini *et al.*, 1991] Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, and Werner Nutt.
The complexity of concept languages.
In *Proc. of the 2nd Int. Conf. on Principles of Knowledge Representation and Reasoning (KR)*, pages 151–162, 1991.

References X

[Guimarães *et al.*, 2023] Ricardo Guimarães, Ana Ozaki, Cosimo Persia, and Baris Sertkaya.
Mining EL⊥ bases with adaptable role depth.
J. of Artificial Intelligence Research, 76:883–924, 2023.

[Horrocks *et al.*, 2000] Ian Horrocks, Ulrike Sattler, and Stephan Tobies.
Practical reasoning for very expressive description logics.
J. of the Interest Group in Pure and Applied Logic, 8(3):239–264, 2000.

[Horrocks, 1998] Ian Horrocks.
Using an expressive description logic: FaCT or fiction?
In *Proc. of the 6th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR)*, pages 636–647, 1998.

[Kazakov *et al.*, 2014] Yevgeny Kazakov, Markus Krötzsch, and Frantisek Simancik.
The incredible ELK - From polynomial procedures to efficient reasoning with EL ontologies.
J. of Automated Reasoning, 53(1):1–61, 2014.

[Konev *et al.*, 2011] Boris Konev, Roman Kontchakov, Michel Ludwig, Thomas Schneider, Frank Wolter, and Michael Zakharyaschev.
Conjunctive query inseparability of OWL 2 QL TBoxes.
In *Proc. of the 25th AAAI Conf. on Artificial Intelligence (AAAI)*, pages 221–226, 2011.

References XI

[Kontchakov *et al.*, 2019] Roman Kontchakov, Vladislav Ryzhikov, Frank Wolter, and Michael Zakharyaschev.
A note on DL-Lite with Boolean role inclusions.
In *Proc. of the 32nd Int. Workshop on Description Logics (DL)*, volume 2373 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2019.

[Lacerda *et al.*, 2024] Victor Lacerda, Ana Ozaki, and Ricardo Guimarães.
Strong faithfulness for ELH ontology embeddings.
Transactions on Graph Data and Knowledge, 2(3):2:1–2:29, 2024.

[Lembo and Ruzzi, 2007] Domenico Lembo and Marco Ruzzi.
Consistent query answering over description logic ontologies.
In *Proc. of the 1st Int. Conf. on Web Reasoning and Rule Systems (RR)*, 2007.

[Lembo *et al.*, 2015] Domenico Lembo, Maurizio Lenzerini, Riccardo Rosati, Marco Ruzzi, and Domenico Fabio Savo.
Inconsistency-tolerant query answering in ontology-based data access.
J. of Web Semantics, 33:3–29, 2015.

[Lenzerini *et al.*, 2016a] Maurizio Lenzerini, Lorenzo Lepore, and Antonella Poggi.
Answering metaqueries over Hi(OWL 2 QL) ontologies.
In *Proc. of the 25th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 1174–1180, 2016.

References XII

[Lenzerini *et al.*, 2016b] Maurizio Lenzerini, Lorenzo Lepore, and Antonella Poggi.
A higher-order semantics for metaquerying in OWL 2 QL.
In *Proc. of the 15th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR)*, pages 577–580. AAAI Press, 2016.

[Lenzerini *et al.*, 2021] Maurizio Lenzerini, Lorenzo Lepore, and Antonella Poggi.
Metamodeling and metaquerying in OWL 2 QL.
Artificial Intelligence, 292:103432, 2021.

[Lutz *et al.*, 2003] Carsten Lutz, Ulrike Sattler, and Lidia Tendera.
The complexity of finite model reasoning in description logics.
In *Proc. of the 19th Int. Conf. on Automated Deduction (CADE)*, pages 60–74, 2003.

[Lutz, 1999] Carsten Lutz.
Reasoning with concrete domains.
In *Proc. of the 16th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 90–95, 1999.

[Motik *et al.*, 2012] Boris Motik, Bernardo Cuenca Grau, Ian Horrocks, Zhe Wu, Achille Fokoue, and Carsten Lutz.
OWL 2 Web Ontology Language profiles (second edition).
W3C Recommendation, World Wide Web Consortium, December 2012.

References XIII

[Nebel, 1991] Bernhard Nebel.
Terminological cycles: Semantics and computational properties.
In John F. Sowa, editor, *Principles of Semantic Networks*, pages 331–361. Morgan Kaufmann, 1991.

[Nikolaou *et al.*, 2019] Charalampos Nikolaou, Egor V. Kostylev, George Konstantinidis, Mark Kaminski, Bernardo Cuenca Grau, and Ian Horrocks.
Foundations of ontology-based data access under bag semantics.
Artificial Intelligence, 274:91–132, 2019.

[Patel-Schneider *et al.*, 1991] Peter F. Patel-Schneider, Deborah L. McGuinness, Ronald J. Brachman, Lori Alperin Resnick, and Alexander Borgida.
The CLASSIC knowledge representation system: Guiding principles and implementation rational.
SIGART Bulletin, 2(3):108–113, 1991.

[Rosati, 2008] Riccardo Rosati.
Finite model reasoning in *DL-Lite*.
In *Proc. of the 5th European Semantic Web Conf. (ESWC)*, volume 5021 of *Lecture Notes in Computer Science*, pages 215–229. Springer, 2008.

References XIV

[Savkovic and C., 2012] Ognjen Savkovic and Diego C.
Introducing datatypes in *DL-Lite*.
In *Proc. of the 20th Eur. Conf. on Artificial Intelligence (ECAI)*, volume 242 of *Frontiers in Artificial Intelligence and Applications*, pages 720–725. IOS Press, 2012.

[Schild, 1991] Klaus Schild.
A correspondence theory for terminological logics: Preliminary report.
In *Proc. of the 12th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 466–471, 1991.

[Schmidt-Schauss and Smolka, 1991] Manfred Schmidt-Schauss and Gert Smolka.
Attributive concept descriptions with complements.
Artificial Intelligence, 48(1):1–26, 1991.

[Woods and Brachman, 1977] William A. Woods and Ronald J. Brachman, editors.
Research in Natural Language Understanding.
Quarterly Progress Report No. 1, BBN Report No. 3742. Bolt, Beranek and Newman Inc., Cambridge, Mass., 1977.

References XV

[Woods, 1975] William A. Woods.

What's in a link: Foundations for semantic networks.

In D. G. Bobrow and A. M. Collins, editors, *Representation and Understanding: Studies in Cognitive Science*, pages 35–82. Academic Press, 1975.

Republished in [Brachman and Levesque, 1985].

[Xiao *et al.*, 2020] Guohui Xiao, Davide Lanti, Roman Kontchakov, Sarah Komla-Ebri, Elem Güzel-Kalayci, Linfang Ding, Julien Corman, Benjamin Cogrel, Diego C., and Elena Botoeva.

The virtual knowledge graph system Ontop.

In *Proc. of the 19th Int. Semantic Web Conf. (ISWC)*, volume 12507 of *Lecture Notes in Computer Science*, pages 259–277. Springer, 2020.