Title: Description Logics

Name: Adila Krisnadhi, Pascal Hitzler

Affil./Addr.: Wright State University, Data Semantics Laboratory

465 Joshi Research Center, 3640 Colonel Glenn Highway, Dayton

OH 45435, USA

Phone: (937)775-5217

E-mail: krisnadhi@gmail.com, pascal.hitzler@wright.edu

Description Logics

Synonyms

Terminological systems. Terminological logics. Concept languages.

Glossary

KR: Knowledge Representation.

DLs: Description Logics; a family of logic-based KR languages for representing knowledge through assertions about concepts, individuals and relationships among them.

(Logic-based) Semantics: a way to interpret any statement in a language; logic-based semantics interprets such a statement using operations in mathematical logic.

Interpretation: a mathematical structure realizing the semantics of a language, typically consisting of an underlying set (domain of interest) and a mapping from the statements in the language to the set or mathematical operations on it.

Model: an interpretation that interprets logical statements in non-contradictory way.

Individual: an element of the domain of interest. Individual names are atomic expressions in a DL corresponding to such elements.

Concept: a logical expression in a DL corresponding to sets of individuals.

Class: synonymous with concept; usually used (instead of concept) in the context of ontology engineering literature.

Role: a logical expression in a DL corresponding to a binary relation between individuals.

Property: synonymous with role; usually used (instead of role) in the context of ontology engineering literature.

Axiom: a statement in a DL that asserts certain constraints that have to be satisfied by some concepts, roles and individuals.

TBox: a set of axioms constraining concepts.

ABox: a set of axioms constraining particular individuals.

RBox: a set of axioms constraining roles.

KB: knowledge base; a set of axioms of any kind; can be partitioned into (possibly empty) TBox, ABox, and RBox.

Reasoning: a process in which implicit knowledge/facts are inferred from explicit knowledge given through a set of axioms.

Subsumption: a reasoning problem that asks, given two concepts, if one concept is subsumed by the other, i.e., if every individual that belongs to the former also belongs to the latter.

Classification: a reasoning problem that, given a collection of concepts, corresponds to find the subsumption relationship between any two concepts; the answer of this problem is a subsumption hierarchy among the concepts.

Open-world Assumption: a meta-level semantical assumption in which a statement is considered false only when the KB forces it so. This means a lack of knowledge does not imply falsity.

Definition

Description logics is a family of formal, logic-based knowledge representation languages that are used to represent *terminological knowledge* of a domain of interest through *concepts* or classes, *roles* or properties or binary relationships, and *individuals*. Knowledge bases in description logics consist of logical axioms describing those concepts and roles – hence, the term *description* – as well as specific assertions regarding particular individuals.

Introduction

Description logics (DLs) [Baader et al 2017, 2007, Krötzsch et al 2012] refer to a family of knowledge representation (KR) languages typically employed for representing terminological knowledge of a domain interest. A terminology as represented in a DL knowledge base (KB) comprises a collection of terms in the form of atomic concepts, roles, and individuals, together with logical axioms that formally define the meaning (semantics) of the terms and constrain their interpretation. Given a DL KB, one can perform reasoning to infer implicit knowledge from the KB. The precise semantics of axioms in a KB allows reasoning to be done in an automated manner.

From a theoretical perspective, DLs can be seen as a fragment of first-order predicate logic, although unlike the latter, reasoning in DLs is typically decidable. From a practical perspective, DLs are prominent as the underlying formalism for the Web Ontology Language (OWL) [Hitzler et al 2010]. Further discussions on reasoning algorithms and OWL, however, are covered under separate titles.

Key Points

Contribution of this chapter covers the following key points.

- DLs is a family of KR languages, which are decidable fragments of first-order predicate logic.
- DLs trace their origin to network-based structures such as semantic networks and frame systems, which are prominent in the 1970s.
- DL syntax is given in the form of concepts, roles, and individuals. Concepts are also known as classes, while roles are also known as properties or binary relationships. DL syntax admits various constructs that can be used to form complex concept and role expressions. These constructs include Boolean constructs and several types of restrictions of roles with concepts. Using these concept and role expressions, DL syntax allow for KBs to be constructed as a set of axioms comprising those that describe general knowledge (intensional) and those that describe specific knowledge (extensional) about particular individuals.
- DL semantics is given in the form of interpretation, which itself consists of a non-empty set and a mapping from the syntactic elements to elements of, subsets of, or binary relations over the set.
- Standard reasoning tasks in DLs include concept and KB satisfiability, concept subsumption and classification, as well as instance checking and conjunctive query answering. Reasoning algorithms that solve these reasoning tasks include structural subsumption algorithms, tableau-based algorithms, completion-based algorithms, and Datalog-based algorithms. These algorithms are implemented in the form of various DL reasoners.
- DLs are prominent due to the fact that several DLs underpin the Web Ontology Language (OWL), which has been made a standard by W3C.
- Besides a close relationship with first-order predicate logic, DLs also possess a close relationship with modal logics, as well as with object-oriented and database modeling languages.

Historical Background

DLs owe their origin to research in *semantic networks* [Quillian 1967] and *frame systems* [Minsky 1981] in 1970s. Both were prominent examples of *network-based structures* in which the structure of the network captures sets of individuals and relationships between them. Such network structures were often motivated by some human-centric, cognitive intuitions, and thus were considered more appealing and practical than logic-based systems. Unfortunately, this advantage of such systems is also a weakness because the lack of precise semantic characterization of the ad-hoc structures and reasoning procedures cause identical-looking structures to behave differently in different situations.

The realization that such network structures could benefit from a more precise semantic characterization led to the work by Hayes [1979], Brachman [1977], and Brachman [1979]. The first recognized that the core features of frame systems can be given a first-order logic semantics, while the latter two studied a similar idea for semantic networks. This line of work culminated in the design of the Kl-One system [Brachman and Schmolze 1985], signaling the transition from semantic networks to a more well-founded *terminological* or *description logics*.

Work on Kl-One brought us the key notions that now constitute description logics such as the notions of concepts, roles, retrisctions of roles with concepts via, e.g., value restrictions, number restrictions, TBox, ABox, as well as inference problems of subsumption and classification. Following this work, theoretical research were done by examining theoretical properties of the terminological logics from KL-One and other early systems, mainly inspired by the work of Brachman and Levesque [1984]. The main lesson was that there is a trade-off between the expressivity of a DL language and the complexity of reasoning with it. After Kl-One, DL systems until early 1990s mainly employed so-called *structural subsumption algorithms*, which work by first normalizing the concept description and then performing the syntactic structure of the normalized descriptions recursively, for example, as implemented in K-Rep [Mays et al 1991], BACK [Peltason 1991], and Loom [MacGregor 1991]. These algorithms are usually efficient (polynomial), but only complete for inexpressive DLs; for more expressive DLs, they are not capable of detecting all existing subsumption relationships. The CLASSIC system attempted to cover more expressive DLs while maintaining completeness by carefully restricting the types of DL construct allowed in the language [Brachman et al 1991].

The development of *tableau-based algorithms* [Schmidt-Schauß and Smolka 1991, Hollunder et al 1990, Donini et al 1991a] for DLs signified the next phase of research in DLs. These algorithms do not just work for expressive DLs, but they are also complete for them. These algorithms do not compute concept subsumption directly, but rather, reduce it to knowledge base consistency problem, and then work by attempting to construct a model for the KB. Such algorithms terminate either when a canonical model of the KB is found or all attempts to construct such a model fail due to logical contradiction. Example of systems in this phase include KRIS [Baader and Hollunder 1991] and CRACK [Franconi 1998]. Theoretical developments also led to a comprehensive analysis of the complexity of reasoning in various DLs [Donini et al 1991a,b, 1992] and the recognition of the close relation between DLs and modal logics [Schild 1991].

The above developments paved the way for more modern DL systems that either employ highly optimized tableau-based decision procedures [Horrocks et al 1999, Horrocks and Sattler 1999] or translation to modal logics [De Giacomo and Lenzerini 1994, 1996]. The former led to implementations such as RACE [Haarslev and Möller 1999] and FaCT [Horrocks 1998], which behave well in practice on large KBs. Later developments also led to renewed focus on tractable DLs with *completion-based algorithms* [Baader et al 2005] and reasoning algorithms based on *translation into Datalog rules* [Krötzsch 2010]. In recent years, implementation of reasoning algorithms for DLs have achieved industrial-level maturity, signified by systems such as Pellet [Sirin et al 2007], Hermit [Glimm et al 2014], FaCT++ [Tsarkov and Horrocks 2006], RACER [Haarslev et al 2012], ELK [Kazakov et al 2014], and Konclude [Steigmiller et al 2014].

DL Syntax and Semantics

Knowledge is represented through DL KBs which consist of axioms composed from expressions, called *concepts*, *roles*, and the always-atomic *individual names*. Non-atomic concepts and roles are constructed from the atomic ones using various constructors admitted by particular DL in consideration. The semantics is realized via *interpretations*, each is a pair $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ where $\Delta^{\mathcal{I}}$, called the *domain*, is a non-empty (possibly infinite) set of individuals, and $\cdot^{\mathcal{I}}$, called the *interpretation function*, maps each atomic concept A to a set of individuals $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, each atomic role R to a binary relation $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, and each individual name a to an individual $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$. The semantics of non-atomic concepts and roles are then obtained by extending the mapping $\cdot^{\mathcal{I}}$ depending on which constructor is used to built them. Table 1 lists the syntax and semantics of some prominent DL constructors.

A DL KB consists of a set of axioms that can be categorized as *TBox*, *ABox* and *RBox* axioms. TBox and RBox axioms describe general knowledge about concepts and roles, respectively. On the other hand, ABox axioms describe specific knowledge in the form of membership of an individual in a concept and relationships between individuals through a role. Semantics of axioms are provided as criteria for which they are *satisfied* by an interpretation $\mathcal{I}$ as given in Table 2. Note that $\mathsf{Fun}(R)$ is equivalent to the concept inclusion $\top \sqsubseteq (\leq 1 R.C)$, $\mathsf{Tra}(R)$ is equivalent to the general role inclusion $R \circ R \sqsubseteq R$, and $\mathsf{Sym}(R)$ is equivalent to the role equivalence $R \equiv R^{-}$.

The following simple examples provide better intuition on how DLs are used to represent knowledge. The axiom $Man \equiv Human \sqcap Male$ asserts that a man is

Table 1. Common DL concept and role constructors where C, D are (possibly non-atomic) concepts, R, S are (possibly non-atomic) roles, a is an individual name, n is a nonnegative integer, and for a set M, $|M|$ is the cardinality of M.

Name	Syntax	Semantics based on an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$
top and concept	$\top$, $\bot$	$\Delta^{\mathcal{I}}$, $\emptyset$
nominal	$\{a\}$	$\{a^{\mathcal{I}}\}$
concept intersection & union	$C \sqcap D$, $C \sqcup D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$, $C^{\mathcal{I}} \cup D^{\mathcal{I}}$
concept complement	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
value/universal restriction	$\forall R.C$	$\{x \mid \forall y \colon \langle x, y\rangle \in R^{\mathcal{I}}$ implies $y \in C^{\mathcal{I}}\}$
existential restriction	$\exists R.C$	$\{x \mid \exists y \colon \langle x, y\rangle \in R^{\mathcal{I}}$ and $y \in C^{\mathcal{I}}\}$
number (at-least) restriction	$\geq nR.C$	$\{x \mid \lvert\{y \mid \langle x, y\rangle \in R^{\mathcal{I}}$ and $y \in C^{\mathcal{I}}\}\rvert \geq n\}$
number (at-most) restriction	$\leq nR.C$	$\{x \mid \lvert\{y \mid \langle x, y\rangle \in R^{\mathcal{I}}$ and $y \in C^{\mathcal{I}}\}\rvert \leq n\}$
universal role	U	$\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
inverse role	R^{-}	$\{\langle x, y\rangle \mid \langle y, x\rangle \in R^{\mathcal{I}}\}$
role intersection	$R \sqcap S$	$R^{\mathcal{I}} \cap S^{\mathcal{I}}$

precisely a male human. The axiom $Father \equiv Man \sqcap \exists hasChild.Human$ asserts that a father is precisely a man who has a human child. Meanwhile, $Woman \sqsubseteq Human \sqcap \neg Man$ asserts that a woman is a human that is not a man, and $Mother \equiv Woman \sqcap \exists hasChild.Human$ states that a mother is a woman who has a human child. To say that having parent is an inverse relationship of having a child, one can state $hasParent \sqsubseteq hasChild^{-}$. Thus, one can also define a child as a human who has either a father or a mother using axiom $Child \sqsubseteq Human \sqcap \exists hasParent.(Father \sqcup Mother)$. Furthermore, the axioms $Son \equiv Child \sqcap Male$ and $Daughter \equiv Child \sqcap \neg Son$ define the concepts son and daughter. If one asserts that a father-with-many-sons must have at least three sons, then this axiom can be used: $FatherWithManySons \sqsubseteq Father \sqcap \geq 3 hasChild.Son$. Meanwhile, the axiom $Father \sqcap \forall hasChild.\neg Son \sqsubseteq FatherWithoutSons$ asserts that a father who has no son is a father-without-sons.

Table 2. Axioms in DLs. A is a concept name, C, D are concepts, $R_{(i)}, R, S$ are roles.

Name	Syntax	Satisfaction Criteria
concept definition (TBox)	$A \equiv C$	$A^{\mathcal{I}} = C^{\mathcal{I}}$
concept inclusion (TBox)	$C \sqsubseteq D$	$C^{\mathcal{I}} \sqsubseteq D^{\mathcal{I}}$
concept assertion (ABox)	$C(a)$	$a^{\mathcal{I}} \in C^{\mathcal{I}}$
role assertion (ABox)	$R(a, b)$	$\langle a^{\mathcal{I}}, b^{\mathcal{I}} \rangle \in R^{\mathcal{I}}$
negative role assertion	$\neg R(a, b)$	$\langle a^{\mathcal{I}}, b^{\mathcal{I}} \rangle \notin R^{\mathcal{I}}$
role equivalence (RBox)	$R \equiv S$	$R^{\mathcal{I}} = S^{\mathcal{I}}$
role hierarchy (RBox)	$R \sqsubseteq S$	$R^{\mathcal{I}} \subseteq S^{\mathcal{I}}$
general role inclusion (RBox)	$R_1 \circ \cdots \circ R_k \sqsubseteq S$	$R_1^{\mathcal{I}} \circ \cdots \circ R_k^{\mathcal{I}} \subseteq S^{\mathcal{I}}$ where '$\circ$' is binary composition of relation
role functionality, transitivity (RBox)	$\mathsf{Fun}(R)$, $\mathsf{Tra}(R)$,	$R^{\mathcal{I}}$ is functional, transitive,
role symmetry, asymmetry (RBox)	$\mathsf{Sym}(R)$, $\mathsf{Asy}(R)$,	$R^{\mathcal{I}}$ is symmetric, asymmetric,
role reflexivity, irreflexivity (RBox)	$\mathsf{Ref}(R)$, $\mathsf{Irr}(R)$	$R^{\mathcal{I}}$ is reflexive, irreflexive

To assert specific statements for particular individuals, one can use ABox assertions. For instance, $FatherWithoutSons(\texttt{bill})$ asserts that Bill is a father-without-sons. To say that Bill has a child, called Chelsea, one can assert $hasChild(\texttt{bill}, \texttt{chelsea})$.

Reasoning in DLs

Based on the semantics of axioms, there are a few basic notions which form the core of reasoning in DLs: satisfiability/consistency checking, subsumption checking and instance checking. First, any set of axioms (including TBoxes, ABoxes, RBoxes, and KBs in general) is *satisfiable* or *consistent* if it has a *model*. A model of a set of axioms is an interpretation $\mathcal{I}$ that *satisfies* all of its axioms. Satisfaction criteria for each type of axiom can be found in Table 2. *KB satisfiability/consistency problem* is thus a reasoning problem of deciding whether a given KB is consistent. One may also be interested in *concept satisfiability* which is deciding whether for a given concept C and a KB, there is a model $\mathcal{I}$ of the KB such that $C^{\mathcal{I}} \neq \emptyset$.

Subsumption checking is a problem of deciding whether a concept C is subsumed by a concept D w.r.t. a KB. This holds when $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for every model $\mathcal{I}$ of the KB. Note that the subsumption relationship includes not only the ones explicitly stated in the KB (through concept inclusions), but also the ones that can be *inferred* from it. Computing subsumption among all concept names occurring in a KB is called a *classification* which allows one to construct the so-called "is-a" hierarchy if a concept name A is subsumed by a concept name B, then A is below B in the hierarchy (i.e., every (individual in) A is a (individual in) B). The "is-a" hierarchy has also been a key feature in semantic networks and frames.

Instance checking is the problem of deciding whether an individual (name) a belongs to a concept C w.r.t. a KB. This holds when $a^{\mathcal{I}} \in C^{\mathcal{I}}$ in every model $\mathcal{I}$ of the KB. This problem places greater emphasis to ABox axioms due to the involvement of explicit individuals. Instance checking can be generalized to *conjunctive query entailment*: given a KB and a set of expressions $\{C_1(x_1), \ldots, C_m(x_m), R_1(y_1, z_1), \ldots, R_n(y_n, z_n)\}$ where $C_1, \ldots, C_m$ are concepts, $R_1, \ldots, R_n$ are roles, and $x_1, \ldots, x_m, y_1, \ldots, y_n, z_1, \ldots, z_n$ are (not necessarily distinct) variables, find a substitution of those variables with individual names occurring in the KB such that the resulting ABox axioms are satisfied by all model of the KB. This problem is closely related to conjunctive query answering which is important considering many real-life situation in which a huge amount of data (which can be seen as an ABox) is augmented with schematic knowledge (in the form of TBox or RBox). Note also the resemblance with notions of query answering from the study of databases, altough unlike modeling in databases which are characterized with closed-world assumption and finiteness of the domain, DLs are distinguished with open-world assumption and possible non-finiteness of the domain.

Some Notable DLs

- $\mathcal{ALC}$ [Baader and Nutt 2007]: simplest Boolean-closed DL that admits top and bottom concept; concept intersection, union, and complement; value and existential restrictions; TBox axioms; and ABox axioms. Reasoning is ExpTime-complete.
- $\mathcal{FL}_0$: simple DL that admits top concept, concept intersection, value restriction, TBox axioms and ABox axioms. It is notable since reasoning is ExpTime-complete, but polynomial if done with an empty KB [Donini 2007, Baader et al 2005].
- $\mathcal{EL}$: simple DL that allows top concept, concept intersection, existential restriction, TBox axioms and ABox axioms. It is notable since reasoning is polynomial. In fact, $\mathcal{ROEL}$ (also known as $\mathcal{EL}^{++}$ [Baader et al 2005]), obtained by adding bottom concept, role hierarchy (thus role equivalence), and general role inclusion to $\mathcal{EL}$ is still polynomial. $\mathcal{EL}^{++}$ is adopted for the OWL 2 EL profile of OWL 2 DL standard [Motik et al 2012]. Its sublanguages also found extensive use in biomedical applications.
- $\mathcal{SHIF}$: an expressive DL obtained from $\mathcal{ALC}$ by adding role transitivity, role hierarchy, inverse roles, functionality (concepts of the form $\leq 1R$ and axioms of the form $\mathsf{Fun}(R)$), role symmetry. This DL underlies the OWL 1 Lite standard and has an ExpTime-complete reasoning [Horrocks and Patel-Schneider 2004, Patel-Schneider et al 2004].
- $\mathcal{SHOIN}$: a very expressive DL, obtained from $\mathcal{SHIF}$ by adding unqualified number restrictions and nominals. It underlies the OWL 1 DL standard and has an NExpTime-complete reasoning [Horrocks and Patel-Schneider 2004, Patel-Schneider et al 2004].
- $\mathcal{SROIQ}$: very expressive DL, obtained from $\mathcal{SHOIN}$ by adding general role inclusion, qualified number restriction, role asymmetry, role reflexivity and role ir-

reflexivity. It underlies OWL 2 DL and has an N2ExpTime-complete reasoning [Horrocks et al 2006, Kazakov 2008, W3C OWL Working Group 2012].

Relationships with Other Formalisms

As a logic-based formalism, DLs are related to many other formalisms [Sattler et al 2007]. Many of these correspondences were in fact exploited to derive complexity results and reasoning algorithms.

Notably, most DLs are a decidable fragment of first-order predicate logic (FOL) with equality. In fact, many DLs are expressible in either $\mathcal{L}^k$ (FOL over unary and binary predicates with at most k variables) or $\mathcal{C}^k$ (like $\mathcal{L}^k$, but allows counting quantifiers). In the translation, concepts are translated into FOL formulas with one free variable in which concept names correspond to unary predicate, role names to binary predicate and individual names to constants. This relationship also extends to the rule (i.e., Horn) fragment of FOL. Development of Description Logic Programs [Grosof et al 2003] which is roughly an intersection between DLs and binary Datalog rules has resulted in the OWL 2 RL profile of OWL 2 [Motik et al 2012]. Many other formalisms have also been proposed to realize integration between rule languages and DLs [Krisnadhi et al 2011].

DLs are also strongly related to modal logics [Baader and Lutz 2007]. For example, $\mathcal{ALC}$ can be seen as a notational variant of multi-modal logic $\mathbf{K_m}$ in which concept names correspond to propositional letters, while value and existential restrictions correspond to the modal operators $\Box$ and $\Diamond$. Other close relationships with modal logic families have also been noted, e.g., with propositional dynamic logics, hybrid logics and guarded fragments.

Relationships with object-oriented and database modeling languages have also been observed. For example, Entity Relationship (ER) models can be translated into

DL KBs which enables formally checking for inconsistency. Parts of Unified Modeling Language (UML) specification can also be translated into DL KBs. On the other hand, the need for more expressive query languages over relational database has led to the development of DL-Lite, a family of very simple DLs which can be used to perform very efficient queries. Due to this reason, DL-Lite [Calvanese et al 2007] has been adopted to underlie the OWL 2 QL profile of OWL 2 [Motik et al 2012].

Key Applications

The first category of key applications is not so much application of DLs to a particular domain or problem, but rather, applications built to as practical results of theoretical research in DLs. Belonging to this category are all DL reasoners built to implement reasoning algorithms for DLs. A number of examples of DL reasoners have been mentioned in an earlier section on Historical Background, and the reader can refer to that section to find them.

The second category corresponds to applications of DLs in a particular domain. Such applications are typically built in conjunction with an existing DL reasoner. Arguably, the most prominent application of DLs in this category is the use of DLs as ontology languages, particularly the Web Ontology Language (OWL) 2 [W3C OWL Working Group 2012] and its predecessor OWL 1 [McGuinness and van Harmelen 2004]. As a standardized ontology language, OWL (and thus DL) becomes de facto standard both for several ontologies in particular domains and various applications in which ontology is a key component. For example, OWL is used by ontologies in life science, including the widely used Gene Ontology (GO), Microarray Gene Expression Data (MGED), the US National Cancer Institute (NCI) Thesaurus, the GALEN ontology, and SNOMED ontology. The United Nations Food and Agriculture Organization (FAO) employed OWL to develop a number of ontologies in agriculture and fisheries domains. In geoscience, NASA developed the Semantic Web for Earth and Environmental Terminology (SWEET) ontologies. There are many more examples in industry and other domains where OWL is used as the de facto ontology language. The reader is welcome to consult the cross-referenced chapter on OWL to find more examples of such applications.

Another type of applications closely related to the use of DLs as ontology languages is the use of DLs as a layer of conceptualization on top of relational databases and semi-structured data models, such as Web data, spreadsheets, etc. The idea is that a DL can provide the user with language constructs that are closer to the way the user conceptualizes the domain of discourse compared to the way the data are actually modeled in the lower level, hence eases the data access. Applications in this area fall into the so-called ontology-based data access (OBDA), which among others, motivated the development of DL-Lite and the OWL 2 QL Profile mentioned in the preceding section.

Future Directions

From a theoretical perspective, many fundamental questions relevant to DLs as KR languages have been answered or investigated. Consequently, study of DLs has been subtly shifting toward topics that are more applied in nature, particularly concerning the use of DLs for data access and integration, non-standard inferences in DLs, extension of DLs beyond classical logics such as with probabilistic, fuzzy, temporal, or spatial formalisms. This trend has been spotted in recent years and it is reasonable to assume that it will continue.

Cross-references

- Automated Reasoning — 110188
- Reasoning — 115
- Web Ontology Language (OWL) — 113
- RIF: The Rule Interchange Format — 118
- Rule-Based Systems — 100878

Acknowledgements

- Both authors acknowledge the support of the National Science Foundation (NSF) project "TROn – Tractable Reasoning with Ontologies" under the award 1017225 III: Small.
- The first author acknowledges the support of Fulbright Indonesia Presidential Scholarship PhD Grant. 2010–2013

References

Baader F, Hollunder B (1991) KRIS: knowledge representation and inference system. SIGART Bulletin 2(3):8–14, doi: 10.1145/122296.122298, URL `http://doi.acm.org/10.1145/122296.122298`

Baader F, Lutz C (2007) Description logic. In: Blackburn P, van Benthem J, Wolter F (eds) Handbook of Modal Logic, Studies in Logic and Practical Reasoning, vol 3, Elsevier, chap 13, pp 757–819

Baader F, Nutt W (2007) Basic description logics. In: Baader F, Calvanese D, McGuinness DL, Nardi D, Patel-Schneider PF (eds) The Description Logic Handbook: Theory, Implementation, and Applications, 2nd edn, Cambridge University Press, chap 2, pp 47–104

Baader F, Brandt S, Lutz C (2005) Pushing the EL envelope. In: Kaelbling LP, Saffiotti A (eds) IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, UK, July 30-August 5, 2005, Professional Book Center, pp 364–369, URL `http://www.ijcai.org/papers/0372.pdf`

Baader F, Calvanese D, McGuinness DL, Nardi D, Patel-Schneider PF (eds) (2007) The Description Logic Handbook: Theory, Implementation, and Applications, 2nd edn. Cambridge University Press

Baader F, Horrocks I, Lutz C, Sattler U (2017) An Introduction to Description Logics. Cambridge University Press

Brachman RJ (1977) A structural paradigm for representing knowledge. PhD thesis, Harvard University, Cambridge, MA

Brachman RJ (1979) On the epistemological status of semantic networks. In: Findler NV (ed) Associative Networks, Academic Press, pp 3–50, republished in [Brachman and Levesque 1985]

Brachman RJ, Levesque HJ (1984) The tractability of subsumption in frame-based description languages. In: Brachman RJ (ed) Proceedings of the National Conference on Artificial Intelligence. Austin, TX, August 6-10, 1984., AAAI Press, pp 34–37, URL http://www.aaai.org/Library/AAAI/1984/aaai84-036.php

Brachman RJ, Levesque HJ (eds) (1985) Readings in Knowledge Representation. Morgan Kauffmann, Los Altos

Brachman RJ, Schmolze JG (1985) An overview of the KL-ONE knowledge representation system. Cognitive Science 9(2):171–216, doi: 10.1207/s15516709cog0902_1, URL http://dx.doi.org/10.1207/s15516709cog0902_1

Brachman RJ, McGuinness DL, Patel-Schneider PF, Resnick LA, Borgida A (1991) Living with CLASSIC: When and how to use a KL-ONE-like language. In: Sowa JF (ed) Principles of Semantic Networks, Morgan Kauffmann, Los Altos, pp 401–456

Calvanese D, De Giacomo G, Lembo D, Lenzerini M, Rosati R (2007) Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family. Journal of Automated Reasoning 39(3):385–429, doi: 10.1007/s10817-007-9078-x, URL http://dx.doi.org/10.1007/s10817-007-9078-x

De Giacomo G, Lenzerini M (1994) Boosting the correspondence between description logics and propositional dynamic logics. In: Hayes-Roth B, Korf RE (eds) Proceedings of the 12th National Conference on Artificial Intelligence, Seattle, WA, USA, July 31 - August 4, 1994, Volume 1., AAAI Press / The MIT Press, pp 205–212, URL http://www.aaai.org/Library/AAAI/1994/aaai94-032.php

De Giacomo G, Lenzerini M (1996) Tbox and abox reasoning in expressive description logics. In: Aiello LC, Doyle J, Shapiro SC (eds) Proceedings of the Fifth International Conference on Principles of Knowledge Representation and Reasoning (KR'96), Cambridge, Massachusetts, USA, November 5-8, 1996., Morgan Kaufmann, pp 316–327

Donini FM (2007) Complexity of reasoning. In: Baader F, Calvanese D, McGuinness DL, Nardi D, Patel-Schneider PF (eds) The Description Logic Handbook: Theory, Implementation, and Applications, 2nd edn, Cambridge University Press, chap 3, pp 105–148

Donini FM, Lenzerini M, Nardi D, Nutt W (1991a) The complexity of concept languages. In: Allen JF, Fikes R, Sandewall E (eds) Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR'91). Cambridge, MA, USA, April 22-25, 1991., Morgan Kaufmann, pp 151–162

Donini FM, Lenzerini M, Nardi D, Nutt W (1991b) Tractable concept languages. In: Mylopoulos J, Reiter R (eds) Proceedings of the 12th International Joint Conference on Artificial Intelligence. Sydney, Australia, August 24-30, 1991, Morgan Kaufmann, pp 458–465

Donini FM, Lenzerini M, Nardi D, Hollunder B, Nutt W, Marchetti-Spaccamela A (1992) The complexity of existential quantification in concept languages. Artif Intell 53(2-3):309–327, doi: 10.1016/0004-3702(92)90076-A, URL `http://dx.doi.org/10.1016/0004-3702(92)90076-A`

Franconi E (1998) CRACK. In: Franconi E, Giacomo GD, MacGregor RM, Nutt W, Welty CA (eds) Proceedings of the 1998 International Workshop on Description Logics (DL'98), IRST, Povo - Trento, Italy, June 6-8, 1998, CEUR-WS.org, CEUR Workshop Proceedings, vol 11, URL `http://ceur-ws.org/Vol-11/CRACK.ps`

Glimm B, Horrocks I, Motik B, Stoilos G, Wang Z (2014) HermiT: An OWL 2 reasoner. Journal of Automated Reasoning 53(3):245–269, doi: 10.1007/s10817-014-9305-1, URL `http://dx.doi.org/10.1007/s10817-014-9305-1`

Grosof BN, Horrocks I, Volz R, Decker S (2003) Description logic programs: combining logic programs with description logic. In: Hencsey G, White B, Chen YR, Kovács L, Lawrence S (eds) Proceedings of the Twelfth International World Wide Web Conference, WWW 2003, Budapest, Hungary, May 20-24, 2003, ACM, pp 48–57, doi: 10.1145/775152.775160, URL `http://doi.acm.org/10.1145/775152.775160`

Haarslev V, Möller R (1999) RACE system description. In: Lambrix P, Borgida A, Lenzerini M, Möller R, Patel-Schneider PF (eds) Proceedings of the 1999 International Workshop on Description Logics (DL'99), Linköping, Sweden, July 30 - August 1, 1999, CEUR-WS.org, CEUR Workshop Proceedings, vol 22, URL http://ceur-ws.org/Vol-22/haarslev.ps

Haarslev V, Hidde K, Möller R, Wessel M (2012) The RacerPro knowledge representation and reasoning system. Semantic Web 3(3):267–277, doi: 10.3233/SW-2011-0032, URL http://dx.doi.org/10.3233/SW-2011-0032

Hayes PJ (1979) The logic of frames. In: Metzing D (ed) Frame Conceptions and Text Understanding, Walter de Gruyter and Co., pp 46–61, republished in [Brachman and Levesque 1985]

Hitzler P, Krötzsch M, Rudolph S (2010) Foundations of Semantic Web Technologies. Chapman and Hall/CRC Press, URL http://www.semantic-web-book.org/

Hollunder B, Nutt W, Schmidt-Schauß M (1990) Subsumption algorithms for concept description languages. In: ECAI, pp 348–353

Horrocks I (1998) Using an expressive description logic: Fact or fiction? In: Cohn AG, Schubert LK, Shapiro SC (eds) Proceedings of the Sixth International Conference on Principles of Knowledge Representation and Reasoning (KR'98), Trento, Italy, June 2-5, 1998., Morgan Kaufmann, pp 636–649

Horrocks I, Patel-Schneider PF (2004) Reducing OWL entailment to description logic satisfiability. Journal of Web Semantics 1(4):345–357, doi: 10.1016/j.websem.2004.06.003, URL http://dx.doi.org/10.1016/j.websem.2004.06.003

Horrocks I, Sattler U (1999) A description logic with transitive and inverse roles and role hierarchies. Journal of Logic and Computation 9(3):385–410, doi: 10.1093/logcom/9.3.385, URL http://dx.doi.org/10.1093/logcom/9.3.385

Horrocks I, Sattler U, Tobies S (1999) Practical reasoning for expressive description logics. In: Ganzinger H, McAllester DA, Voronkov A (eds) Logic Programming and Automated Reasoning, 6th International Conference, LPAR'99, Tbilisi, Georgia, September 6-10, 1999, Proceedings, Springer, Lecture Notes in Computer Science, vol 1705, pp 161–180, doi: 10.1007/3-540-48242-3_11, URL http://dx.doi.org/10.1007/3-540-48242-3_11

Horrocks I, Kutz O, Sattler U (2006) The even more irresistible SROIQ. In: Doherty P, Mylopoulos J, Welty CA (eds) Proceedings, Tenth International Conference on Principles of Knowledge Representation and Reasoning, Lake District of the United Kingdom, June 2-5, 2006, AAAI Press, pp 57–67, URL http://www.aaai.org/Library/KR/2006/kr06-009.php

Kazakov Y (2008) RIQ and SROIQ are harder than SHOIQ. In: Brewka G, Lang J (eds) Principles of Knowledge Representation and Reasoning: Proceedings of the Eleventh International Conference, KR 2008, Sydney, Australia, September 16-19, 2008, AAAI Press, pp 274–284, URL http://www.aaai.org/Library/KR/2008/kr08-027.php

Kazakov Y, Krötzsch M, Simancik F (2014) The incredible ELK - from polynomial procedures to efficient reasoning with EL ontologies. J Autom Reasoning 53(1):1–61, doi: 10.1007/s10817-013-9296-3, URL http://dx.doi.org/10.1007/s10817-013-9296-3

Krisnadhi A, Maier F, Hitzler P (2011) OWL and rules. In: Polleres A, d'Amato C, Arenas M, Handschuh S, Kroner P, Ossowski S, Patel-Schneider PF (eds) Reasoning Web. Semantic Technologies for the Web of Data - 7th International Summer School 2011, Galway, Ireland, August 23-27, 2011, Tutorial Lectures, Springer, Lecture Notes in Computer Science, vol 6848, pp 382–415, doi: 10.1007/978-3-642-23032-5_7, URL http://dx.doi.org/10.1007/978-3-642-23032-5_7

Krötzsch M (2010) Efficient inferencing for OWL EL. In: Janhunen T, Niemelä I (eds) Logics in Artificial Intelligence - 12th European Conference, JELIA 2010, Helsinki, Finland, September 13-15, 2010. Proceedings, Springer, Lecture Notes in Computer Science, vol 6341, pp 234–246, doi: 10.1007/978-3-642-15675-5_21, URL http://dx.doi.org/10.1007/978-3-642-15675-5_21

Krötzsch M, Simancik F, Horrocks I (2012) A description logic primer. CoRR abs/1201.4089, URL http://arxiv.org/abs/1201.4089

MacGregor RM (1991) Inside the LOOM description classifier. SIGART Bulletin 2(3):88–92, doi: 10.1145/122296.122309, URL http://doi.acm.org/10.1145/122296.122309

Mays E, Dionne R, Weida RA (1991) K-Rep system overview. SIGART Bulletin 2(3):93–97, doi: 10.1145/122296.122310, URL http://doi.acm.org/10.1145/122296.122310

McGuinness DL, van Harmelen F (eds) (2004) OWL Web Ontology Language Overview. W3C Recommendation 10 February 2004, URL https://www.w3.org/TR/owl-features/

Minsky M (1981) A framework for representing knowledge. In: Haugeland J (ed) Mind Design, The MIT Press, a longer version appears in *The Psychology of Computer Vision* (1975), republished in [Brachman and Levesque 1985]

Motik B, Fokoue A, Horrocks I, Wu Z, Lutz C, Grau BC (eds) (2012) OWL 2 Web Ontology Language Profiles (Second Edition). W3C Recommendation 11 December 2012, URL http://www.w3.org/TR/owl2-profiles/

Patel-Schneider PF, Hayes P, Horrocks I (eds) (2004) OWL Web Ontology Language Semantics and Abstract Syntax. W3C Recommendation 10 February 2004, URL https://www.w3.org/TR/owl-semantics/

Peltason C (1991) The BACK system - an overview. SIGART Bulletin 2(3):114–119, doi: 10.1145/122296.122314, URL http://doi.acm.org/10.1145/122296.122314

Quillian M (1967) Word concepts: A theory and simulation of some basic capabilities. Behavioral Science 12:410–430, republished in [Brachman and Levesque 1985]

Sattler U, Calvanese D, Molitor R (2007) Relationships with other formalisms. In: Baader F, Calvanese D, McGuinness DL, Nardi D, Patel-Schneider PF (eds) The Description Logic Handbook: Theory, Implementation, and Applications, 2nd edn, Cambridge University Press, chap 4, pp 149–192

Schild K (1991) A correspondence theory for terminological logics: Preliminary report. In: Mylopoulos J, Reiter R (eds) Proceedings of the 12th International Joint Conference on Artificial Intelligence. Sydney, Australia, August 24-30, 1991, Morgan Kaufmann, pp 466–471

Schmidt-Schauß M, Smolka G (1991) Attributive concept descriptions with complements. Artificial Intelligence 48(1):1–26

Sirin E, Parsia B, Grau BC, Kalyanpur A, Katz Y (2007) Pellet: A practical OWL-DL reasoner. Journal of Web Semantics 5(2):51–53, doi: 10.1016/j.websem.2007.03.004, URL http://dx.doi.org/10.1016/j.websem.2007.03.004

Steigmiller A, Liebig T, Glimm B (2014) Konclude: System description. Journal of Web Semantics 27:78–85, doi: 10.1016/j.websem.2014.06.003, URL http://dx.doi.org/10.1016/j.websem.2014.06.003

Tsarkov D, Horrocks I (2006) FaCT++ description logic reasoner: System description. In: Furbach U, Shankar N (eds) Automated Reasoning, Third International Joint Conference, IJCAR 2006, Seattle, WA, USA, August 17-20, 2006, Proceedings, Springer, Lecture Notes in Computer Science, vol 4130, pp 292–297, doi: 10.1007/11814771_26, URL http://dx.doi.org/10.1007/11814771_26

W3C OWL Working Group (2012) OWL 2 Web Ontology Language Document Overview (Second Edition). W3C Recommendation 11 December 2012, URL https://www.w3.org/TR/owl2-overview/

Recommended Reading

- [Baader et al 2007] is the standard text for DLs; covers almost all major results in DLs, written in semi-textbook style; requires some basics in mathematical logic.
- [Krötzsch et al 2012] is a text intended as a very first reading on DLs without requiring formal logic background.
- [Hitzler et al 2010] is a introductory level textbook in semantic web technologies which also covers significant amount of DLs material, especially in the context of their application in the Semantic Web.
- [Baader et al 2017] is a principled and thorough textbook formally introducing description logics, including mathematical proofs of properties and algorithm correctness.