# What are Description Logics?

Description logics (DLs) are syntactic variants of first-order logic that are specifically designed for the conceptual representation of an application domain in terms of concepts and relationships between concepts [1].

Expressions in DLs are constructed from *atomic concepts* (unary predicates), *atomic roles* (binary predicates) and *individuals* (constants). Complex expressions can be built inductively from these atomic elements using concept constructors. Formally a concept represents a set of individuals and a role a binary relation between individuals [2].

Formally every DL ontology consists of a set of axioms that are based on finite sets of concepts, roles and individuals. Axioms in a DL ontology are divided into the TBox, the RBox and the ABox. A TBox is used to define concepts and relationships between concepts (that is the terminology or taxonomy) and an ABox is used to assert knowledge regarding the domain of interest (i.e. that an individual is a member of a concept). Depending on the expressivity of the DL used, an ontology may include an RBox. An RBox is used to define relations between roles as well as properties of roles [2].

A feature of DLs is that they have decidable reasoning procedures for standard reasoning tasks. This means these reasoning procedures will give an answer, unlike undecidable reasoning procedures which may not terminate and thus may not give an answer. A fundamental goal of DL research is to preserve decidability to the point that decidability is considered to be a precondition for claiming that a formalism is a DL. Standard DL reasoning algorithms are sound and complete and, even though the worst-case computational complexity of these algorithms is ExpTime and worse, in practical applications they are well-behaved [3].

Standard reasoning procedures for DLs are the following [2].

* **Satisfiability checking** checks that every axiom in an ontology can be instantiated. Axioms that cannot be instantiated indicates that modelling errors exist within the ontology.
* **Consistency checking** checks whether there are axioms that contradict each other, which again is indicative of modelling errors.
* **Subsumption checking** checks whether an axiom subsumes another axiom, which is used for classifying axioms into a parent-child taxonomy.

Various DLs exist with different levels of expressivity and computational complexity. The most widely supported DL is SROIQ(D) which forms the mathematical basis of the W3C OWL 2 standard [4]. In subsequent posts I will provide some intuitive understanding of OWL 2 and explain some of its uses.

# Works Cited

|  |  |
| --- | --- |
| [1] | D. Berardi, D. Calvanese and G. De Giacomo, “Reasoning on UML class diagrams,” *Artificial Intelligence,* vol. 168, no. 1-2, p. 70–118, 2005. |
| [2] | F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi and P. F. Patel-Schneider, The Description Logic Handbook: Theory, Implementation and Applications, Cambridge University Press, 2007. |
| [3] | F. Baader, “What’s new in Description Logics,” *Informatik-Spektrum,* vol. 34, no. 5, p. 434–442, 2011. |
| [4] | W3C, “OWL 2 Web Ontology Language - Document Overview (Second Edition),” W3C, 11 December 2012. [Online]. Available: https://www.w3.org/TR/owl2-overview/. [Accessed 9 September 2017]. |