G53KRR handout on description logic ALC

OWL Web Ontology Language is a W3C standard. It extends most description logics and has slightly different terminology (based on RDF rather than description logic semantics. A subset of OWL, OWL DL, is based on description logic).

Reading:

The Description Logic Handbook. Franz Baader, Diego Calvanese, Deborah L.McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. Cambridge University Press, 2003. ISBN 0-521-78176-0.

A good on-line course: `http://www.inf.unibz.it/~franconi/dl/course/`

Basic idea description logic talks about relationships between *concepts* (noun phrases). There are many different description logics, ALC below is one of them. Most extend ALC with things like counting, transitive roles, inverse roles. Some are subsets of ALC, such as EL which only contains $\exists$, $\sqcup$ and $\sqsubseteq$. EL is mostly what Snomed CT is written in.

Precise definition of the syntax of ALC :

Logical symbols (apart from brackets etc.):

- concept-forming operators: $\forall$, $\exists$, $\sqcup$, $\sqcap$, $\neg$
- connectives: $\sqsubseteq$, $\doteq$

Non-logical symbols:

- Atomic concepts: *Person*, *Thing*,.... Correspond to unary predicates in FOL.
- Roles: *age*, *employer*, *child*, *arm*, ... Correspond to binary predicates in FOL.
- Constants: $john, mary, roomA7, \ldots$. Correspond to constants (0-ary functional symbols) in FOL.

Concepts:

- atomic concept is a concept
- if r is a role and C is a concept, then $\forall r.C$ is a concept (e.g. $\forall.child.Girl$ describes someone all of whose children are girls)
- if r is a role and C is a concept, then $\exists r.C$ is a concept (e.g. $\exists.child.Girl$ describes someone who has a daughter)
- if C is a concept, then $\neg C$ (not C) is a concept
- if C_1 and C_2 are concepts then $C_1 \sqcap C_2$ (C_1 and C_2) is a concept
- if C_1 and C_2 are concepts then $C_1 \sqcup C_2$ (C_1 or C_2) is a concept

Sentences:

- if C_1 and C_2 are concepts then $C_1 \sqsubseteq C_2$ is a sentence (all C_1s are C_2s, C_1 is *subsumed* by C_2)
- if C_1 and C_2 are concepts then $C_1 \doteq C_2$ is a sentence (C_1 is equivalent to C_2)
- if a is a constant and C a concept then $C(a)$ is a sentence (the individual denoted by a satisfies the description expressed by C)
- if a, b are constants and r a role then $r(a, b)$ is a sentence (the individuals denoted by a and b are connected by the role r)

A description logic knowledge base is a set of description logic sentences.

TBox and ABox A description logic knowledge base is usually split into terminological part or TBox which describes general relationships between concepts, e.g. $\mathit{Surgeon} \sqsubseteq \mathit{Doctor}$, and assertions about individuals or ABox (e.g. $\mathit{Doctor}(\mathit{mary})$).

Interpretations for description logic same as for FOL: a set of individuals D and an interpretation mapping I such that

- for a constant a, $I(a) \in D$
- for an atomic concept A, $I(A) \subseteq D$
- for a role r, $I(r) \subseteq D \times D$
- $I(\forall r.C]) = \{x \in D : \text{for any } y, \text{ if } (x, y) \in I(r), \text{ then } y \in I(C)\}$. Same as
$$\forall y(R(x, y) \supset B(y))$$
- $I(\exists r.C]) = \{x \in D : \text{there is a } y \text{ such that } (x, y) \in I(r) \text{ and } y \in I(C)\}$. Same as
$$\exists y(R(x, y) \wedge C(y))$$
- $I(\neg C) = D \setminus I(C)$
- $I(C_1 \sqcap C_2]) = I(C_1) \cap I(C_2)$. Same as
$$C_1(x) \wedge C_2(x)$$
- $I(C_1 \sqcup C_2]) = I(C_1) \cup I(C_2)$. Same as
$$C_1(x) \vee C_2(x)$$

Finally, for sentences:

- $(D, I) \models C(a)$ iff $I(a) \in I(C)$. Same as $C(a)$
- $(D, I) \models r(a, b)$ iff $(I(a), I(b)) \in I(r)$. Same as $R(a, b)$
- $(D, I) \models C_1 \sqsubseteq C_2$ iff $I(C_1) \subseteq I(C_2)$. Same as
$$\forall x(C_1(x) \supset C_2(x))$$
- $(D, I) \models C_1 \doteq C_2$ iff $I(C_1) = I(C_2)$. Same as
$$\forall x(C_1(x) \equiv C_2(x))$$

Reasoning Entailment is defined exactly like in FOL: a set of sentences Γ entails a sentence ϕ (in symbols $\Gamma \models \phi$) if and only if ϕ is true in every interpretation where all of the sentences in Γ are true.

Since ALC is a fragment of first order logic, reasoning in it is more efficient (it is decidable whether a sentence is satisfiable, or whether a finite set of sentences entails another sentence). This holds for many other description logics, although not all of them.