# Handout: DL Basics

## Syntax and semantics

Basic building blocks:

- concept names (unary predicates, classes)	A, B
– role names (binary predicates, properties)	r,s
- individuals names (constants)	$a, b, c, \dots$
Complex concepts (built using constructors: see below)	C, D
TBox (ontology) = set of terminological axioms	${\mathcal T}$
ABox (dataset) = set of ABox assertions $(A(a), r(a, b))$	${\mathcal A}$
Knowledge base $(KB) = TBox + ABox$	${\cal K}$

Name	Syntax	Semantics	
Top concept	Т	$\Delta^{\mathcal{I}}$	Concepts
Bottom concept	$\perp$	Ø	
Negation	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$	
Conjunction	$C_1 \sqcap C_2$	$C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}}$	
Disjunction	$C_1 \sqcup C_2$	$C_1^{\mathcal{I}} \cup C_2^{\mathcal{I}}$	
Existential restriction	$\exists r.C$	$\{d_1 \mid \text{there exists } (d_1, d_2) \in r^{\mathcal{I}} \text{ with } d_2 \in C^{\mathcal{I}}\}$	
Universal restriction	$\forall r.C$	$\{d_1 \mid d_2 \in C^{\mathcal{I}} \text{ for all } (d_1, d_2) \in r^{\mathcal{I}}\}\$	
(Qualified) number	$\geqslant m r.C$	$\{d_1 \mid m \le  \{d_2 \mid (d_1, d_2) \in r^{\mathcal{I}} \text{ and } d_2 \in C^{\mathcal{I}}\} \}$	
restrictions	$\leqslant m r.C$	$\{d_1 \mid m \ge  \{d_2 \mid (d_1, d_2) \in$	$r^{\mathcal{I}} $ and $d_2 \in C^{\mathcal{I}} \}   \}$
Inverse	$r^{-}$	$\{(d_2, d_1) \mid (d_1, d_2) \in r^{\mathcal{I}}\}$	Roles
Concept inclusion	$C\sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$	TBox Axioms
Role inclusion	$r \sqsubseteq s$	$r^{\mathcal{I}} \subseteq s^{\mathcal{I}}$	
Functionality axiom	funct(r)	$r^{\mathcal{I}}$ is a functional relation (if $(d, e) \in r^{\mathcal{I}}$	
		and $(d, f) \in r^{\mathcal{I}}$ , then	e = f)
Transitivity axiom	trans(r)	$r^{\mathcal{I}}$ is a transitive relation (if $(d, e) \in r^{\mathcal{I}}$	
		and $(e, f) \in r^{\mathcal{I}}$ , then	$(d,f)\in r^{\mathcal{I}})$
Concept assertion	C(a)	$a^{\mathcal{I}} \in A^{\mathcal{I}}$	ABox Assertions
Role assertion	r(a,b)	$(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$	

Note: if the considered logic allows inverse roles, then we can use inverse roles in place of role names when building complex concepts and inclusions

Note: we use  $C\equiv D$  as an abbreviation for the pair of axioms  $C\sqsubseteq D$  and  $D\sqsubseteq C$ 

### Models, Satisfiability, Entailment

#### Models:

- $\mathcal{I}$  is a model of a TBox  $\mathcal{T}$  if  $\mathcal{I}$  satisfies every statement in  $\mathcal{T}$
- $\mathcal{I}$  is a model of an ABox  $\mathcal{A}$  if  $\mathcal{I}$  satisfies every assertion in  $\mathcal{A}$
- $\mathcal{I}$  is a model of a KB  $(\mathcal{T}, \mathcal{A})$  if  $\mathcal{I}$  is a model of  $\mathcal{T}$  and  $\mathcal{A}$

#### Satisfiability:

- A concept C is satisfiable w.r.t. TBox  $\mathcal{T}$  if there exists a model  $\mathcal{I}$  of  $\mathcal{T}$  such that  $C^{\mathcal{I}} \neq \emptyset$
- A KB  $(\mathcal{T}, \mathcal{A})$  is satisfiable if  $(\mathcal{T}, \mathcal{A})$  has at least one model

#### Entailment:

- A TBox  $\mathcal{T}$  entails an inclusion  $\alpha$  (written  $\mathcal{T} \models \alpha$ ) if every model of  $\mathcal{T}$  satisfies  $\alpha$
- A KB  $(\mathcal{T}, \mathcal{A})$  entails an ABox assertion  $\alpha$  (written  $(\mathcal{T}, \mathcal{A}) \models \alpha$ ) if every model of  $(\mathcal{T}, \mathcal{A})$  satisfies  $\alpha$

#### **Reasoning tasks**

Concept satisfiability: is C satisfiable w.r.t.  $\mathcal{T}$ ?

KB satisfiability: is  $\mathcal{K} = (\mathcal{T}, \mathcal{A})$  satisfiable?

Subsumption: does  $\mathcal{T} \models C \sqsubseteq D$ ?

Classification: find all concept names A,B such that  $\mathcal{T} \models A \sqsubseteq B$ 

Instance checking: does  $(\mathcal{T}, \mathcal{A}) \models C(b)$ ?