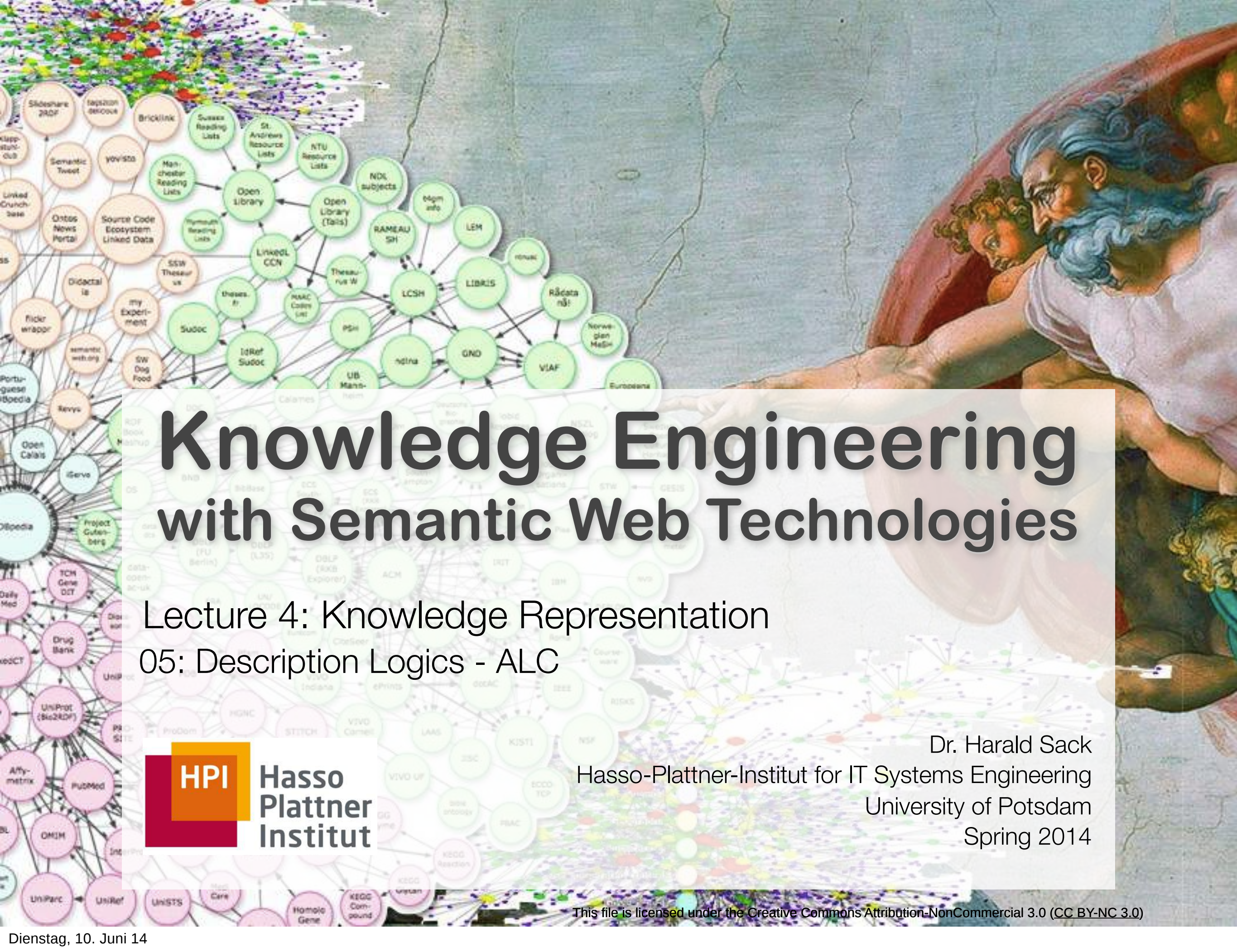




Knowledge Engineering with Semantic Web Technologies

Lecture 4: Knowledge Representation
05: Description Logics - ALC

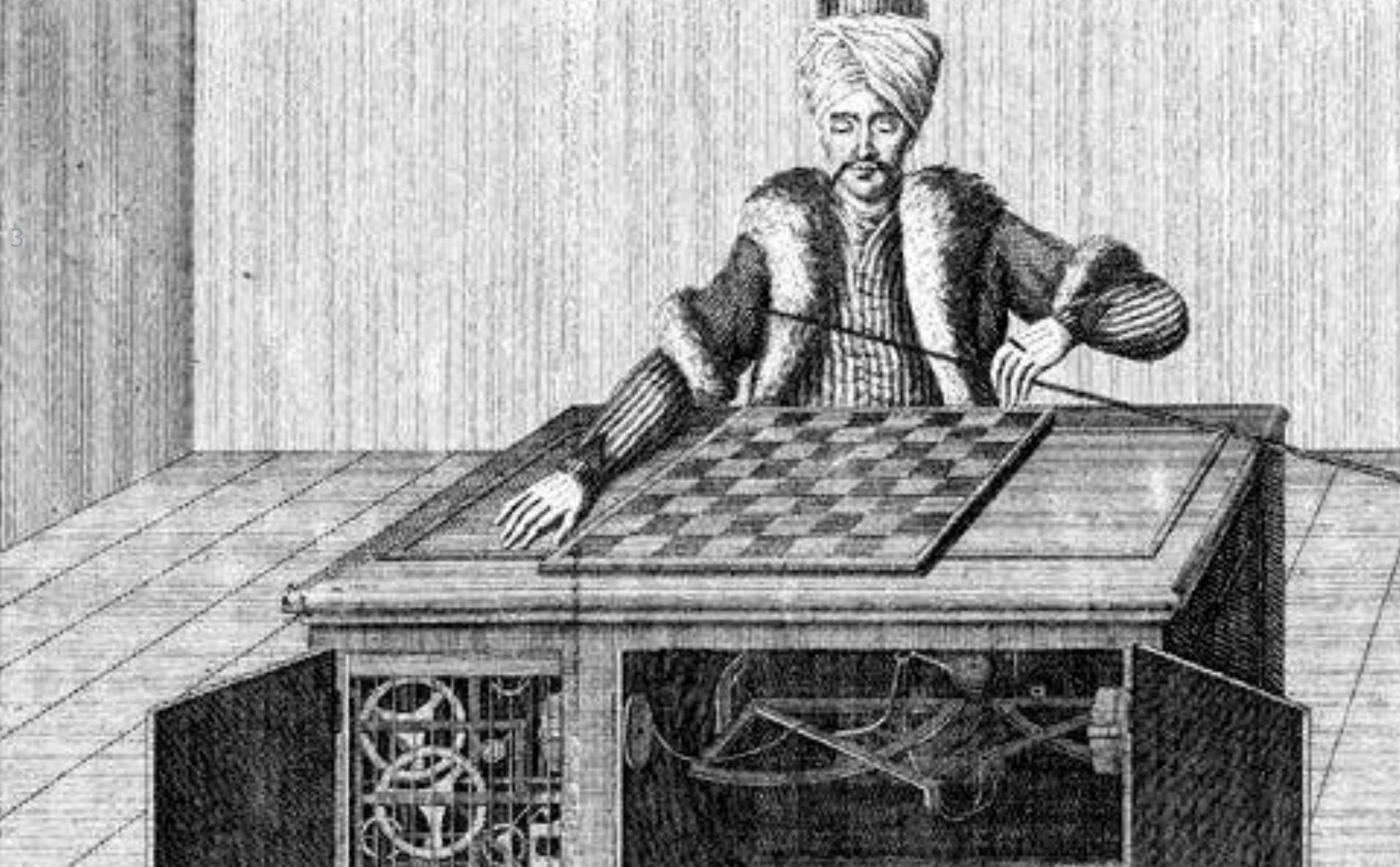


Dr. Harald Sack
Hasso-Plattner-Institut for IT Systems Engineering
University of Potsdam
Spring 2014



Lecture 4: Knowledge Representations I

Open HPI Course: Knowledge Engineering with Semantic Web Technologies



05 Description Logics - ALC

Open HPI Course: Knowledge Engineering with Semantic Web Technologies
Lecture 04: Knowledge Representations Pt.1

- Why not simply take FOL for Ontologies?
- FOL can do everything...
 - *you could also program everything with assembler instead of with higher programming languages*
- FOL has
 - high **expressivity**
 - too **bulky** for modelling
 - not appropriate to find consensus in modelling
 - proof theoretically very **complex** (semi-decidable)
- *FOL is of course not a Markup Language for the Web*
 - ▶ Look for an appropriate fragment of FOL....
 - ▶ *...and then make it a vocabulary for RDF(S)*

- DLs are Fragments of FOL
- In DL from *simple descriptions* more *complex descriptions* are created with the help of **Constructors**.
- DLs differ in the applied constructors (Expressivity)
- DLs have been developed from „semantic Networks“
- DLs are **decidable** (most times)
- DLs possess **sufficient expressivity** (most times)
- DLs are related to modal logics
- e.g.,
W3C Standard OWL 2 DL is based on description logics
 $\mathcal{SHROIQ}(\mathcal{D})$

Knowledge Base

TBox Terminological Knowledge

Knowledge about concepts of a domain
(classes, attributes, properties,..)

`Writer  $\equiv$  Person  $\sqcap$   $\exists$ author.Book`

ABox Assertional Knowledge

Knowledge about Individuals / Entities

`Writer(GeorgeOrwell)`

Inference Engine

Interface

- DLs are a **family** of logic-based formalisms applied for knowledge representation
- Special languages characterized by:
 - **Constructors** for complex concepts and roles from simpler concepts and roles
 - Set of **Axioms** to express Facts about concepts, roles and individuals
 - ***ALC*** (*Attribute Language with Complement*) is the smallest deductively complete DL
 - Conjunction, Disjunction, Negation are class constructors, denoted as $\sqcap$, $\sqcup$, $\neg$
 - Quantifiers restrict domain and range of roles

Attributive Language with Complements $\mathcal{ALC}$

8 Building Blocks:

- Classes
- Roles / Properties
- Individuals
- **Writer(GeorgeOrwell)**
Individual GeorgeOrwell is of class Writer
- **Book(NineteenEightyfour)**
Individual NineteenEightyfour is of class Book
- **author(NineteenEightyfour, GeorgeOrwell)**
The book NineteenEightyfour has the author GeorgeOrwell

- **Atomic Types**
 - Concept names $A, B, \dots$
 - Special concepts
 - $\top$ - Top (universal concept)
 - $\perp$ - Bottom concept
 - Role names $R, S, \dots$
- **Constructors**
 - Negation: $\neg C$
 - Conjunction: $C \sqcap D$
 - Disjunction: $C \sqcup D$
 - Existential quantifier: $\exists R. C$
 - Universal quantifier: $\forall R. C$

10

- **Class Inclusion**

- $\text{Novel} \sqsubseteq \text{Book}$

- *every novel is also a book*
- equals $(\forall x)(\text{Novel}(x) \rightarrow \text{Book}(x))$

- **Class Equivalence**

- $\text{Novel} \equiv \text{Prose}$

- *all Prose are exactly Novels*
- equals $(\forall x)(\text{Novel}(x) \leftrightarrow \text{Prose}(x))$

- Conjunction $\sqcap$
- Disjunction $\sqcup$
- Negation $\neg$

$\text{Novel} \sqsubseteq (\text{Book} \sqcap \text{Fiction})$
 $\sqcup (\text{Paperback} \sqcap \neg \text{Poetry})$

$(\forall x)(\text{Novel}(x) \rightarrow ((\text{Book}(x) \wedge \text{Fiction}(x))$
 $\vee (\text{Paperback}(x) \wedge \neg \text{Poetry}(x))))$

12

- **Strict Binding** of the Range of a Role to a Class

- $\text{Book} \sqsubseteq \forall \text{author} . \text{Writer}$

- *A Book must be authored by a Writer*

- $(\forall x)(\text{Book}(x) \rightarrow (\forall y)(\text{author}(x,y) \rightarrow \text{Writer}(y)))$

- **Open Binding** of the Range of a Role to a Class

- $\text{Book} \sqsubseteq \exists \text{author} . \text{Person}$

- *Every Book has at least one author (who is a person)*

- $(\forall x)(\text{Book}(x) \rightarrow (\exists y)(\text{author}(x,y) \wedge \text{Person}(y)))$

13

- Production rules for creating classes in $\mathcal{ALC}$:

(A is an atomic class, C and D are complex Classes and R a Role)

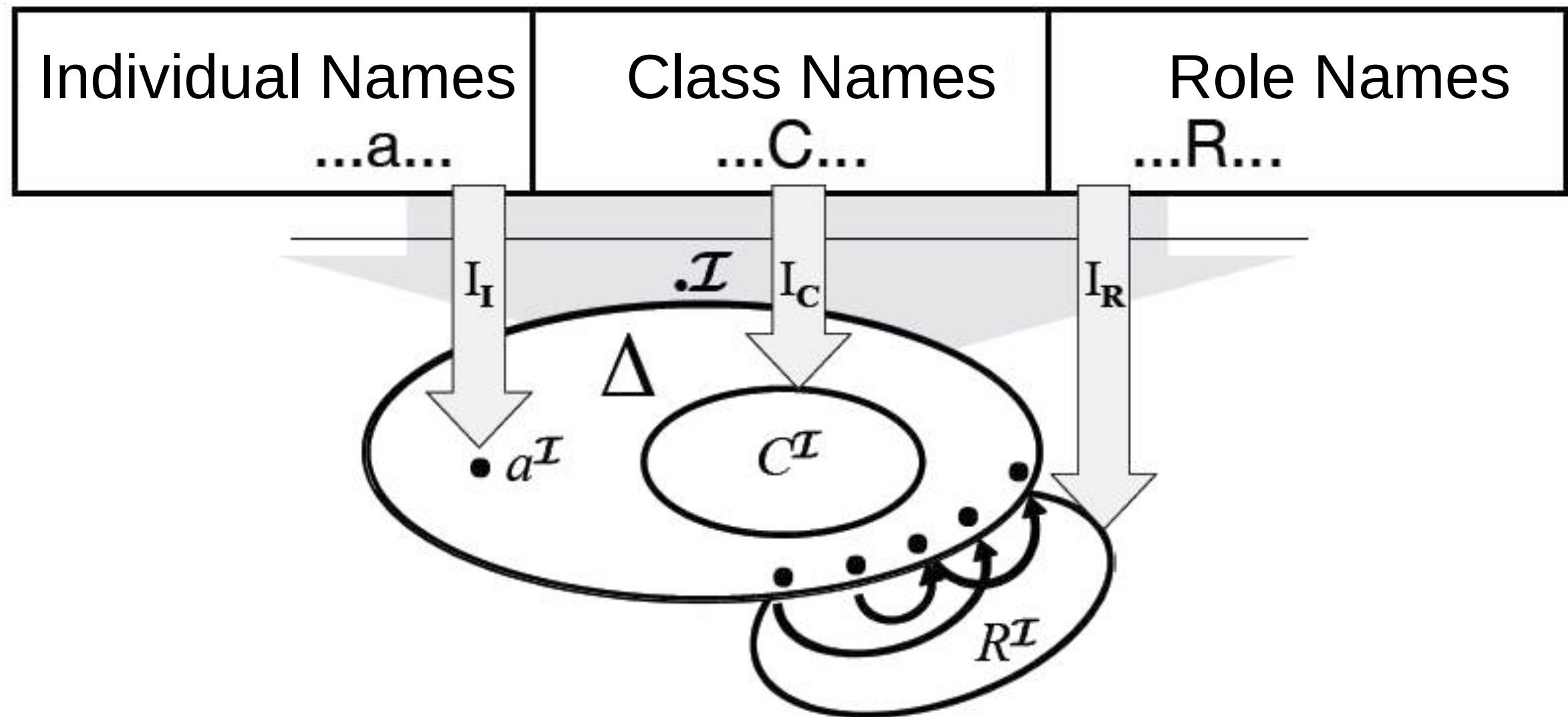
$$\bullet C, D ::= A \mid \top \mid \perp \mid \neg C \mid C \sqcap D \mid C \sqcup D \mid \exists R.C \mid \forall R.C$$

- An $\mathcal{ALC}$ **TBox** contains assertions of the form $C \sqsubseteq D$ and $C \equiv D$, where C, D are complex classes.
- An $\mathcal{ALC}$ **ABox** contains assertions of the form $C(a)$ and $R(a, b)$, where C is a complex Class, R a Role and a, b Individuals.
- An $\mathcal{ALC}$ -**Knowledge Base** contains an ABox and a TBox.

- we define a model-theoretic semantic for $\mathcal{ALC}$
(i.e. *Entailment will be defined via Interpretations*)
- an **Interpretation** $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ contains
 - a set $\Delta^{\mathcal{I}}$ (**Domain**) of Individuals and
 - an **interpretation function** $\cdot^{\mathcal{I}}$ that maps
 - Individual names a
to domain elements $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$
 - Class names C
to a set of domain elements $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
 - Role names R
to a set of pairs of domain elements $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$

$\mathcal{ALC}$ - Semantic (Interpretation)

15



- Extension (of Interpretation $\mathbf{I}$) for complex classes:

- $\top^{\mathbf{I}} = \Delta^{\mathbf{I}}$ and $\perp^{\mathbf{I}} = \emptyset$
- $(C \sqcup D)^{\mathbf{I}} = C^{\mathbf{I}} \cup D^{\mathbf{I}}$ and
 $(C \sqcap D)^{\mathbf{I}} = C^{\mathbf{I}} \cap D^{\mathbf{I}}$
- $(\neg C)^{\mathbf{I}} = \Delta^{\mathbf{I}} \setminus C^{\mathbf{I}}$
- $\forall R. C = \{a \in \Delta^{\mathbf{I}} \mid (\forall b \in \Delta^{\mathbf{I}}) ((a, b) \in R^{\mathbf{I}} \rightarrow b \in C^{\mathbf{I}})\}$
- $\exists R. C = \{a \in \Delta^{\mathbf{I}} \mid (\exists b \in \Delta^{\mathbf{I}}) ((a, b) \in R^{\mathbf{I}} \wedge b \in C^{\mathbf{I}})\}$

- ...and Axioms:

- $C(a)$ holds, iff $a^I \in C^I$
- $R(a, b)$ holds, iff $(a^I, b^I) \in R^I$
- $C \sqsubseteq D$ holds, iff $C^I \subseteq D^I$
- $C \equiv D$ holds, iff $C^I = D^I$

ALC - Knowledgebase

18

- **Terminological Knowledge (TBox)**

Axioms that describe the Structure of the modelled domain (conceptual Schema):

- $\text{Writer} \sqsubseteq \exists \text{author} . \text{Book}$
- $\text{ScienceFictionNovel} \equiv \text{Novel} \sqcap \exists \text{literaryGenre} . \text{ScienceFiction}$

- **Assertional Knowledge (ABox)**

Axioms that describe specific situations (data):

- $\text{Writer}(\text{GeorgeOrwell})$
- $\text{author}(\text{NineteenEightyfour}, \text{GeorgeOrwell})$

Description Logics

19

Operator/Constructor	Syntax	Language			
Conjunction	$A \sqcap B$	FL	S*		
Value Restriction	$\forall R.C$				
Existential Quantifier	$\exists R$				
Top	$\top$	AL*		S*	
Bottom	$\perp$				
Negation	$\neg A$				
Disjunction	$A \sqcup B$				
Existential Restriction	$\exists R.C$				
Number Restriction	$(\leq nR) (\geq nR)$				
Set of Inviduals	$\{a_1, \dots, a_2\}$				
Role Hierarchy	$R \sqsubseteq S$	H			
inverse Role	R^{-1}	I			
Qualified Number Restriction	$(\leq nR.C) (\geq nR.C)$	Q			

- $\mathcal{ALC}$: Attribute Language with Complement
- $\mathcal{S}$: $\mathcal{ALC}$ + Transitivity of Roles
- $\mathcal{H}$: Role Hierarchies
- $\mathcal{O}$: Nominals
- $\mathcal{I}$: Inverse Roles
- $\mathcal{N}$: Number restrictions $\leq n R$ etc.
- $\mathcal{Q}$: Qualified number restrictions $\leq n R.C$ etc.
- $(\mathcal{D})$: Datatypes
- $\mathcal{F}$: Functional Roles
- $\mathcal{R}$: Role Constructors
- OWL 2 DL is $\mathcal{SHRO1Q(D)}$



06 DL Inference and Reasoning

Open HPI Course: Knowledge Engineering with Semantic Web Technologies
Lecture 04: Knowledge Representations Pt.1