UNIVERSITY OF ABERDEEN

Knowledge-Based Systems

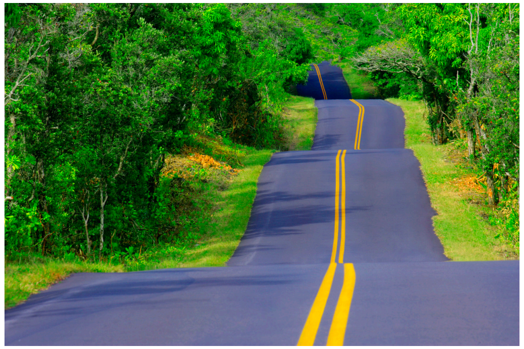
A Formal Introduction to Description Logic

Jeff Z. Pan

http://homepages.abdn.ac.uk/jeff.z.pan/pages/

UNIVERSITY OF ABERDEEN

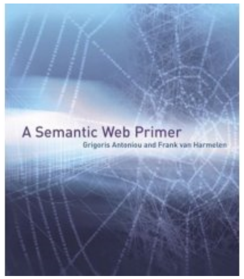
Roadmap



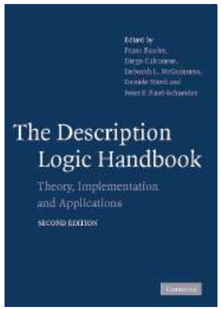
- Foundation
  - KR, ontology and rule; set theory
- Knowledge capture
- Knowledge representation
  - Ontology: Semantic Web standards RDF and OWL, Description Logics
  - Rule: Jess
- Knowledge reasoning
  - Ontology: formal semantics, tableaux algorithm
  - Rule: forward chaining, backward chaining
- Knowledge reuse and evaluation
- Meeting the real world
  - Jess and Java, Uncertainty, Invited talk

Lecture Outline

- The Story So Far
- Motivation
- Introduction to DL
- An Exercise on DL
- Practical



[Chapter 4]



[Sections: 2.2.1.1, 2.2.1.2,]

Database and RDF

- An RDF statement is a data unit with global and linkable ids for data and schema

Student ID	Name	take-course
p001	John	cs3019
p002	Tom	cs3023

  - [csd:p001 rdf:type csd:Student .]
  - [csd:p002 rdf:type csd:Student .]
  - [csd:p001 csd:name “John” .]
  - [csd:p002 csd:name “Tom” .]
  - [csd:p001 csd:take-course csd:cs3019 .]
  - [csd:p002 csd:take-course csd:cs3023 .]

Schema in a Database System

- A database system includes some schema constraints, such as the foreign key constraint

Student ID	Name	take-course
p001	John	~~cs3015~~
p002	Tom	cs3025

Course ID	Title	coordinator
cs3017	AIS	AS
cs3025	KBS	JP

UNIVERSITY OF ABERDEEN

OWL Profiles and Reasoning

- Current version of OWL is OWL 2
- OWL 2 has three tractable (with low computational complexity) profiles
  - OWL 2 EL: schema reasoning, instance reasoning and query answering
  - OWL 2 QL: query answering
  - OWL 2 RL: the “rule fragment” of OWL 2
- Unions of any of the two profiles are no longer tractable

UNIVERSITY OF ABERDEEN

Schema in a Knowledge Based System

1) Allow schema constraints, such as DisjointClasses (UndgStudent MastStudent)

UndgStudent ID	Name	take-course
csd:p001	John	csd:cs3014
~~csd:p002~~	Tom	csd:cs3025

MastStudent ID	Name	take-course
csd:p008	Yuan	csd:cs5010
~~csd:p002~~	Tom	csd:cs5017

Schema in a Knowledge Based System

2) Allow some reasoning based on axioms (open world assumption), such as SubClassOf (MastStudent Student)

Student ID	Name	take-course
csd:p001	John	csd:cs3015
csd:p002	Tom	csd:cs3025

MastStudent ID	Name	take-course
csd:p008	Yuan	csd:cs5010
csd:p002	Tom	csd:cs5017

thus all the students include csd:p001, csd:p002, and csd:p008

Lecture Outline

- The Story So Far
- Motivation
- Introduction to DL
- Some Further Discussions on DL
- Practical

UNIVERSITY OF ABERDEEN

Motivations

- Well defined and simplified syntax
- Clear formal semantics
- Good balance between expressive power and decidability

[used as the underpinnings of the Semantic Web standard **ontology language**]

Designing and Building Business Ontologies

An Intensive 4-DAY SEMINAR with Workshops and Demonstrations, Semantically Enabling the Enterprise led by Dave McComb and Simon Robe

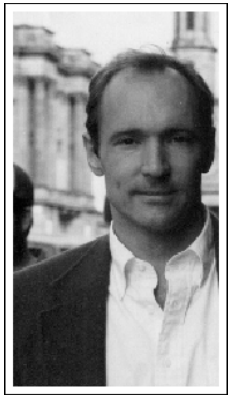
Tuition Fee: $2,450

Seminar Objectives

Participants will:

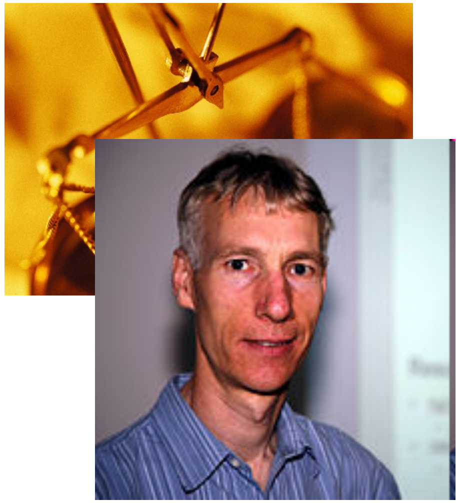
- Gain an understanding of what an ontology is and what it can be used for.
- Understand how representing information in an ontology goes beyond a conceptual model or a simple taxonomy
- Understand the difference between frame based/ declarative classes and description logic based/ derivable classes.
- Understand the difference between open world and closed world models.
- Understand the basic principles for designing Ontologies for corporate applications.

Tim’s Dream of the Web



- Allow people to share knowledge
  - To work together better
- Allow programs to handle the shared knowledge and data
  - In a meaningful way
  - To help people analysis and manage activities

At roughly the same time …

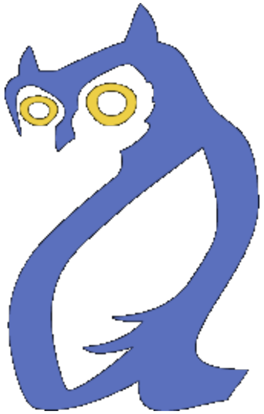


- Breakthrough around 2000-2007
  - Highly optimisation algorithms for ontology reasoners
  - By Prof. Ian Horrocks (Oxford U.) and colleagues
    - FaCT reasoner (Ian Horrocks)
    - FaCT-DG reasoner (Jeff Pan and Ian Horrocks)
    - FaCT++ reasoner (Dmitry Tsarkov and Ian Horrocks)
- Some recent breakthrough (2008-2016)
  - TrOWL: Approximate reasoning (Jeff Pan)
  - ...

Lecture Outline

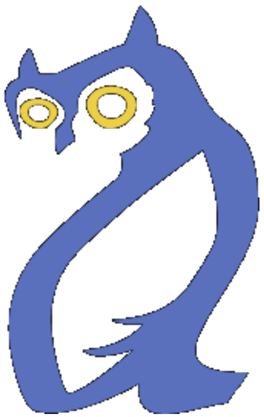
- The Story So Far
- Motivation
- Introduction to DL
  - The big picture
- Some Further Discussions on DL
- Practical

Syntax of OWL



- Abstract syntax
  - Rather lengthy
  - But easier for people who are not familiar with logical symbols e.g. ∀, ∃ …
- DL syntax
  - Compact
  - Widely used in scientific papers and reports

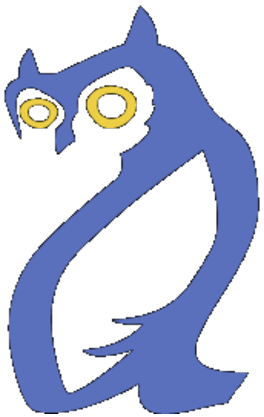
Symbols in DL



- Non-logical symbols
  - Predicates
    - Classes (unary predicates)
    - Properties (binary predicates)
  - Constants (0-ary function symbols)
    - Objects/Individuals
- Logical symbols
  - Punctuation: (,), .
  - Connectives: $\neg$, $\sqcap$, $\sqcup$, $\forall$, $\exists$, $\leq$, $\geq$
  - No variable symbols

UNIVERSITY OF ABERDEEN

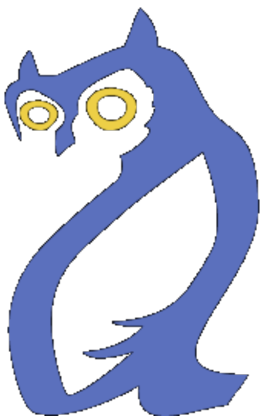
Descriptions



- Class descriptions
  - C ⊓ D: intersectionOf (C D)
  - C ⊔ D: unionOf (C D)
  - ¬C: complementOf (C)
  - {o1,…,on}: oneOf (o1, …,on)
  - **Restriction**
- Example class descriptions
  - Burgundy ⊓ Whitewine: intersectionOf (Burgundy Whitewine)
  - Male ⊔ Female: unionOf (Male Female)
  - ¬Frenchwine: complementOf (Frenchwine)
  - {White Rose Red}: oneOf (White Rose Red)

UNIVERSITY OF ABERDEEN

OWL DL: Class Axioms



- Class axioms
  - $A \sqsubseteq C$: Class (A partial C)
  - $A \equiv C$: Class (A complete C)
  - **$C \sqsubseteq D$: SubClassOf** (C D)
  - $C \equiv D$: EquivalantClasses (C D)
  - $C \sqsubseteq \neg D$: DisjointClasses (C D)
- Example class axioms
  - $\text{Lecturer} \sqsubseteq \text{Staff}$: Class (Lecturer partial Staff)
  - $\exists \text{hasS.PhDS} \sqsubseteq \text{Academic} \sqcap \text{FTStaff}$: SubClassOf (restriction (hasS someValuesFrom (PhDS)) intersectionOf (Academic FTStaff))
  - $\text{Lecturer} \equiv \text{AsistentProf}$: EquivalantClasses (Lecturer AsistentProf)
  - $\text{Male} \sqsubseteq \neg \text{Female}$: DisjointClasses (Male Female)

Semantics and Interpretations

[Source: blogspot.com]

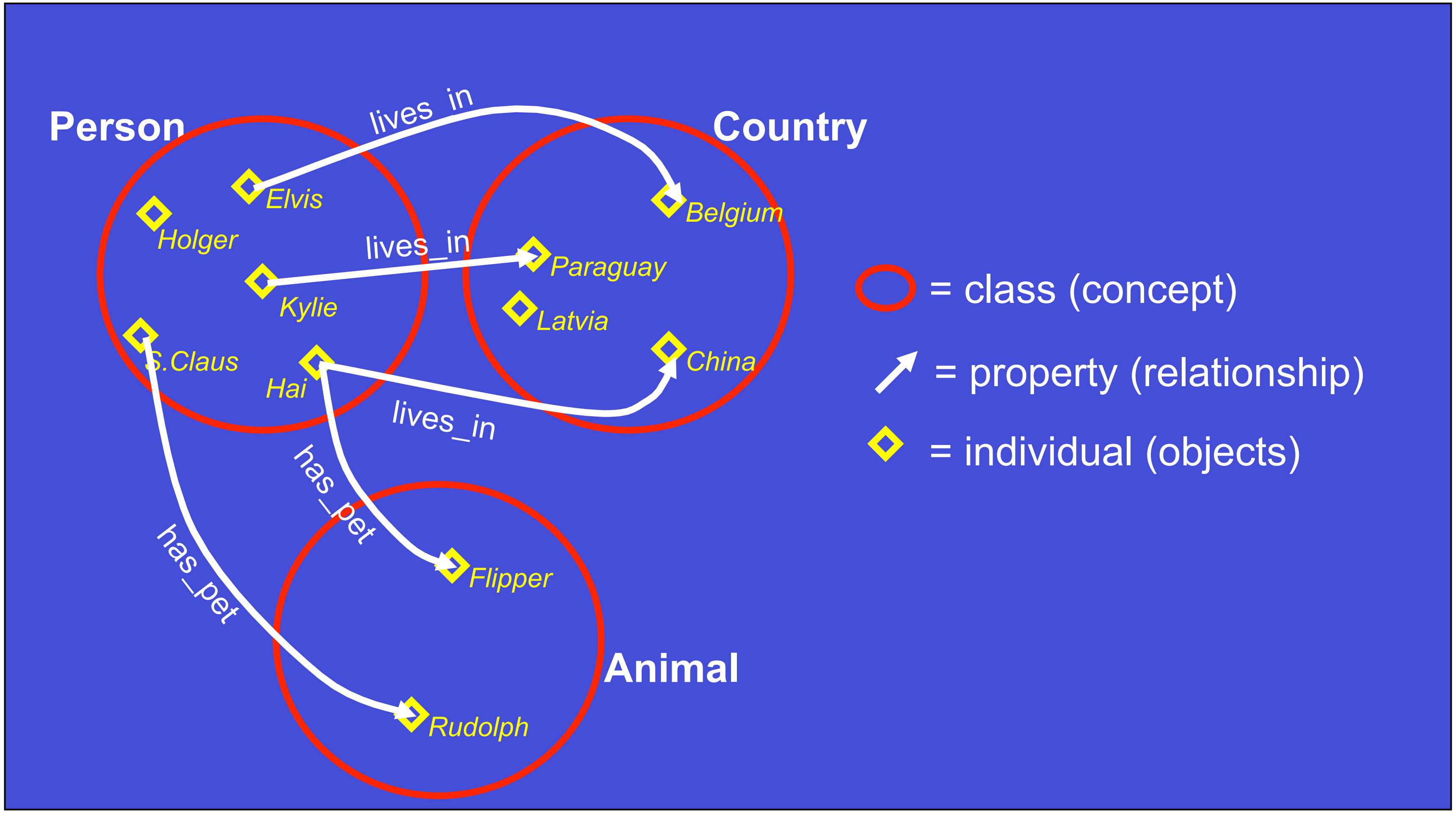
DL Interpretations

- An interpretation **I** is written as $(\Delta^I, \bullet^I)$
  - Δ^I is the non-empty **domain** (similar to universal set)
  - $\bullet^I$ is the **interpretation function**
    - all individuals are members of the domain: $o^I \in \Delta^I$
    - all classes are subsets of the domain $A^I \subseteq \Delta^I$
      - e.g., $Employee^I = \{E1, E2, E3, E4\}$
    - all properties are subsets $R^I \subseteq \Delta^I \times \Delta^I$

      e.g., Works-for^I= {<E1,P1>, <E2,P1>, <E2,P2>, <E3,P1>, <E3,P2>, <E4,P2>}

- Domain is a mathematical representation of the world
- Interpretation function allows us to consider all possible assignment of class and property memberships
  - all possible databases for the given schema

Example: DL Interpretations



[Picture Credit: Protégé Team]

Interpretations of Class Descriptions

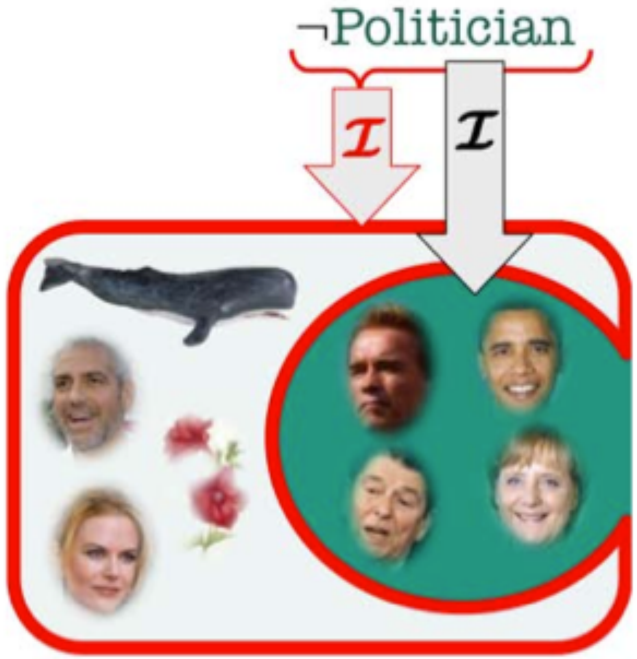
Given an interpretation, we can compute the semantic counterparts of class descriptions

$$(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$$
$$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$$
$$(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$$

Interpretations of Class Descriptions

Given an interpretation, we can compute the semantic counterparts of class descriptions

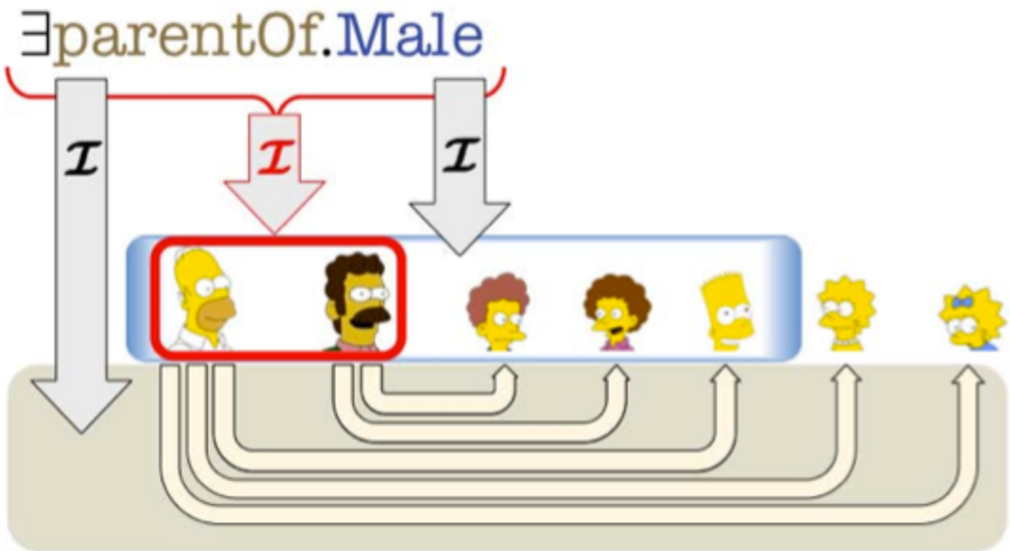
$$(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$$
$$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$$
$$(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$$



Interpretations of Class Descriptions

Given an interpretation, we can compute the semantic counterparts of class descriptions

$$\exists r.C = \{\ x \mid \exists y.\ (x,y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\ \}$$
$$\forall r.C = \{\ x \mid \forall y.\ (x,y) \in r^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\ \}$$

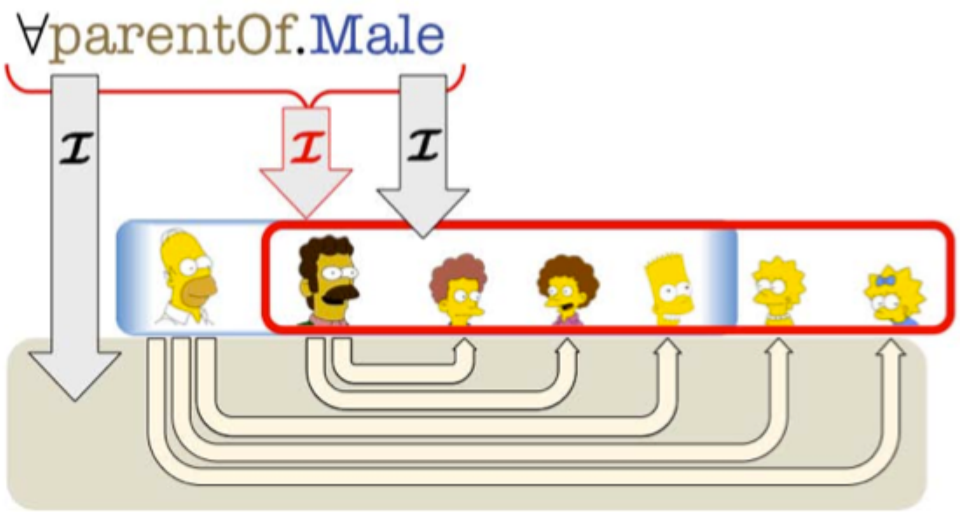


Interpretations of Class Descriptions

Given an interpretation, we can compute the semantic counterparts of class descriptions

$$\exists r.C = \{\ x \mid \exists y.\ (x,y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\ \}$$

$$\forall r.C = \{\ x \mid \forall y.\ (x,y) \in r^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\ \}$$



Lecture Outline

- The Story So Far
- Motivation
- Introduction to DL Syntax
- Some Further Discussions on DL
- Practical

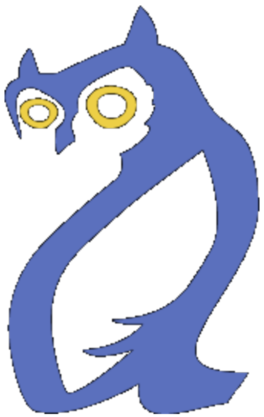
Interpretations of Descriptions

Abstract Syntax	DL Syntax	Semantics
Class(A) Class(owl:Thing) Class(owl:Nothing)	A $\top$ $\bot$	$\mathsf{A}^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ $\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$ $\bot^{\mathcal{I}} = \emptyset$
intersectionOf($C_1, C_2, \ldots$) unionOf($C_1, C_2, \ldots$) complementOf(C) oneOf($o_1, o_2, \ldots$)	$C_1 \sqcap C_2$ $C_1 \sqcup C_2$ $\neg C$ $\{o_1\} \sqcup \{o_2\}$	$(C_1 \sqcap C_2)^{\mathcal{I}} = C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}}$ $(C_1 \sqcup C_2)^{\mathcal{I}} = C_1^{\mathcal{I}} \cup C_2^{\mathcal{I}}$ $(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$ $(\{o_1\} \sqcup \{o_2\})^{\mathcal{I}} = \{o_1^{\mathcal{I}}, o_2^{\mathcal{I}}\}$
restriction(R someValuesFrom(C)) restriction(R allValuesFrom(C)) restriction(R hasValue(o)) restriction(R minCardinality(m)) restriction(R maxCardinality(m))	$\exists R.C$ $\forall R.C$ $\exists R.\{o\}$ $\geqslant mR$ $\leqslant mR$	$(\exists R.C)^{\mathcal{I}} = \{x \mid \exists y.\langle x,y\rangle \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$ $(\forall R.C)^{\mathcal{I}} = \{x \mid \forall y.\langle x,y\rangle \in R^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\}$ $(\exists R.\{o\})^{\mathcal{I}} = \{x \mid \langle x, \mathsf{o}^{\mathcal{I}}\rangle \in R^{\mathcal{I}}\}$ $(\geqslant mR)^{\mathcal{I}} = \{x \mid \sharp\{y.\langle x,y\rangle \in R^{\mathcal{I}}\} \geq m\}$ $(\leqslant mR)^{\mathcal{I}} = \{x \mid \sharp\{y.\langle x,y\rangle \in R^{\mathcal{I}}\} \leq m\}$
restriction(T someValuesFrom(u)) restriction(T allValuesFrom(u)) restriction(T hasValue(w)) restriction(T minCardinality(m)) restriction(T maxCardinality(m))	$\exists T.u$ $\forall T.u$ $\exists T.\{w\}$ $\geqslant mT$ $\leqslant mT$	$(\exists T.u)^{\mathcal{I}} = \{x \mid \exists t.\langle x,t\rangle \in T^{\mathcal{I}} \wedge t \in u^{\mathbf{D}}\}$ $(\forall T.u)^{\mathcal{I}} = \{x \mid \exists t.\langle x,t\rangle \in T^{\mathcal{I}} \rightarrow t \in u^{\mathbf{D}}\}$ $(\exists T.\{w\})^{\mathcal{I}} = \{x \mid \langle x, w^{\mathbf{D}}\rangle \in T^{\mathcal{I}}\}$ $(\geqslant mT)^{\mathcal{I}} = \{x \mid \sharp\{t \mid \langle x,t\rangle \in T^{\mathcal{I}}\} \geq m\}$ $(\leqslant mT)^{\mathcal{I}} = \{x \mid \sharp\{t \mid \langle x,t\rangle \in T^{\mathcal{I}}\} \leq m\}$
ObjectProperty(S) ObjectProperty(S' inverseOf(S)) DatatypeProperty(T)	S S^{-} T	$S^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ $(S^{-})^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ $T^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta_{\mathbf{D}}$

DL Axioms

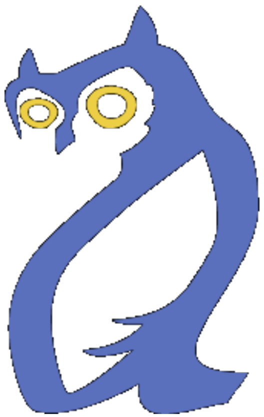
Abstract Syntax	DL Syntax
Class(A partial $C_1 \ \ldots \ C_n$)	$A \sqsubseteq C_1 \sqcap \ldots \sqcap C_n$
Class(A complete $C_1 \ \ldots \ C_n$)	$A \equiv C_1 \sqcap \ldots \sqcap C_n$
EnumeratedClass(A $o_1 \ \ldots \ o_n$)	$A \equiv \{o_1\} \sqcup \ldots \sqcup \{o_n\}$
SubClassOf(C_1, C_2)	$C_1 \sqsubseteq C_2$
EquivalentClasses($C_1 \ \ldots \ C_n$)	$C_1 \equiv \ldots \equiv C_n$
DisjointClasses($C_1 \ \ldots \ C_n$)	$C_i \sqsubseteq \neg C_j$, $(1 \leq i < j \leq n)$
SubPropertyOf(R_1, R_2)	$R_1 \sqsubseteq R_2$
EquivalentProperties($R_1 \ \ldots \ R_n$)	$R_1 \equiv \ldots \equiv R_n$
ObjectProperty(R super(R_1) ... super(R_n)	$R \sqsubseteq R_i$
domain(C_1) ... domain(C_k)	$\geqslant 1 R \sqsubseteq C_i$
range(C_1) ... range(C_h)	$\top \sqsubseteq \forall R.C_i$
[Symmetric]	$R \equiv R^-$
[Functional]	Func(R)
[InverseFunctional]	Func(R^-)
[Transitive])	Trans(R)
AnnotationProperty(R)	
Individual(o type(C_1) ... type(C_n)	$o : C_i, 1 \leq i \leq n$
value(R_1, o_1) ... value(R_n, o_n)	$\langle o, o_i \rangle : R_i, 1 \leq i \leq n$
SameIndividual($o_1 \ \ldots \ o_n$)	$o_1 = \ldots = o_n$
DifferentIndividuals($o_1 \ \ldots \ o_n$)	$o_i \neq o_j, 1 \leq i < j \leq n$

OWL DL: Object Property Axioms



- Property axioms
  - $R1 \sqsubseteq R2$: ObjectProperty (R1 super(R2))
  - $R1 \equiv R2^-$: ObjectProperty **(**R1 inverseOf (R2)**)**
  - $\exists R.\top \sqsubseteq C$: ObjectProperty(R domain(C))
  - $\exists R^-\top \sqsubseteq C$: ObjectProperty(R range(C))
  - Trans(R): ObjectProperty(R Transitive)
  - $R \equiv R^-$: ObjectProperty(R Symmetric)
  - Func(R): ObjectProperty(R Functional)
  - Func(R^-): ObjectProperty(R InverseFunctional)

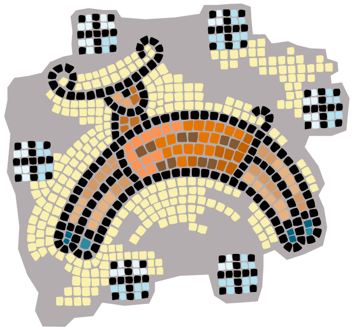
OWL DL: Individual Axioms



- Individual axioms
  - a:C - Individual (a type C)
  - <a,b>: R - Individual (a value (R b))
  - a = b - SameIndividual (a b)
  - a ≠ b - DifferentIndividuals (a b)
- Example individual axioms
  - CS4021:Course - Individual (CS4021 type Cource)
  - <jpan,CS4021>: teach - Individual (jpan value (teach CS4021))
  - w3c:jpan = csd:jpan - SameIndividual (w3c:jpan csd:jpan)
  - w3c:jsmith ≠ csd:jsmith - DifferentIndividuals (w3c:jsmith csd:jsmith)

UNIVERSITY OF ABERDEEN

DL Exercise



Q: Given this interpretation,

Δ^I = {Tom, Jane, Mike, Jack, Ian}

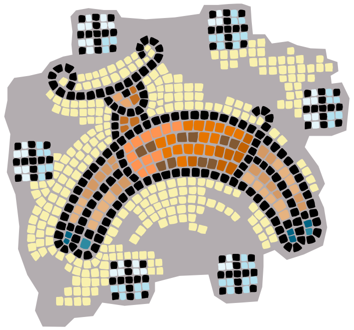
Student^I = {Tom, Jane, Mike}

Researcher^I = {Jane, Ian}

write down the interpretations of the following class descriptions

1. Student $\sqcup$ Researcher = **{Tom, Jane, Mike, Ian}**
2. $\neg$Student = **{Jack, Ian}**

DL Syntax Exercise



Q: Write down the following OWL axioms in DL syntax

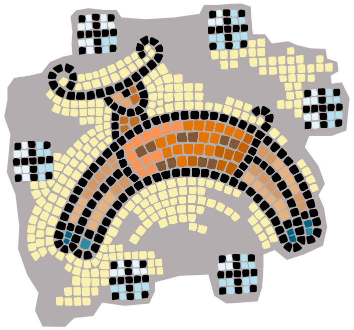
1. Class (Cat partial Animal)

   Cat $\sqsubseteq$ Animal

2. Class (Dog partial Animal)

   Dog $\sqsubseteq$ Animal

DL Exercise



Q: Write down the following OWL axioms in DL syntax

1. ObjectProperty (eats domain (Animal) range (LivingThing))

2. ObjectProperty (owns domain (Person) range (intersectionOf (LivingThing complementOf (Person))))

Practical

- More exercises with Protégé
  - Building the Animal ontology
- More exercises on DL
  - interpretations
  - reasoner implementation

Summary

- DL syntax
  - compact
  - widely used in DL literature
- DL semantics defined in terms of interpretations

"Billy, I'm not going to argue the semantics of biting. Whether or not you penetrated skin, I'm calling your mother."

UNIVERSITY OF ABERDEEN

Acknowledgement

- Ian Horrocks (Oxford University, UK)

- Sebastian Rudolph (KIT, Germany)