Cours 6

An Introduction to Description Logics

0. Some RDFS Limitations

G. Falquet

Limitations

- global domain/range
- no number restrictions
- no union
- no existential assertion/inference
- no negation

global domain/range

- The parents of a Human are Humans
  - parent domain Human
  - parent range Human
- The parents of a Human are Humans and the parents of a Cat are Cats
  - impossible on this vocabulary
  - must add properties
    - humanParent subproperty of parent ; domain Human ; range Human
    - catParent subproperty of parent ; domain Cat ; range Cat

no number restrictions

Impossible to express

- an individual has at most one address
- a house has at least one owner
- an individual has exactly two biological parents

no "exact" union

- Students are either bachelor or master of PhD students, and nothing else
- All we can do in RDFS is
  - `BacStd rdfs:subClassOf Student`
  - `MasterStd rdfs:subClassOf Student`
  - `PhDStd rdfs:subClassOf Student`
- Impossible to express "and nothing else"
- Impossible to express the disjointness of these classes (if it's the case)

no existential statement and inference

knowing that

- A car necessarily has an owner (not expressible)
- c is a car (:c a :Car)

the query

```
select ?x
where {?x :hasOwner ?y}
```

should answer {c}

even is no triple (`:c :hasOwner :o`) is present in the graph

no negation

- Impossible to state that something is false
- In knowledge bases we often consider that what is not expressed is not known, it can be true or false

```
:ChemicalProduct a rdfs:Class .
:ToxicProduct rdfs:subClassOf ChemicalProduct .
:p1 a :ToxicProduct .
:p2 a :ChemicalProduct
```

does not mean that :p2 is not toxic

An Introduction to Description Logics

1. Syntax and Semantics

G. Falquet

Contents

- Individuals, classes and properties
- Class constructors
- Ontologies
- Reasoning

The world according to DL

- There are **individuals**
- Individuals may be interconnected through **roles** property
- Individuals belong to **concepts** classes

Similar to RDF/S (resources, properties, classes)

But different

- individuals, classes, and roles are disjoint
  - a class may not be an instance of another class
  - an individual may not have instances

DL languages

a family of logical languages with different

- class constructors
- role constructors

aims

- formalize the knowledge representation languages of the 80-90's (CLASSIC, KL-ONE, semantic networks, ...)

- decidable
  - there is an algorithm to check the consistency of any set of formulas
  - ⇒ not as expressive as First order logic

$\mathcal{ALC}$ – concept constructors

classe propr

Start with a vocabulary of concept names (N_C), role names (N_R) and individual names (N_O)

Concept expressions may be

- the top concept (everything): $\top$
- the bottom (impossible) concept: $\bot$
- a concept name: A
- a conjunction of concepts: $C \sqcap D$
- a disjunction: $C \sqcup D$
- a complement: $\neg\ C$
- an existential restriction: $\exists\ R.C$
- a universal restriction: $\forall\ R.C$

Manchester notation

class name

Building

conjunction

Student **and** Employee

disjunction

Man **or** Woman

complement

not Student

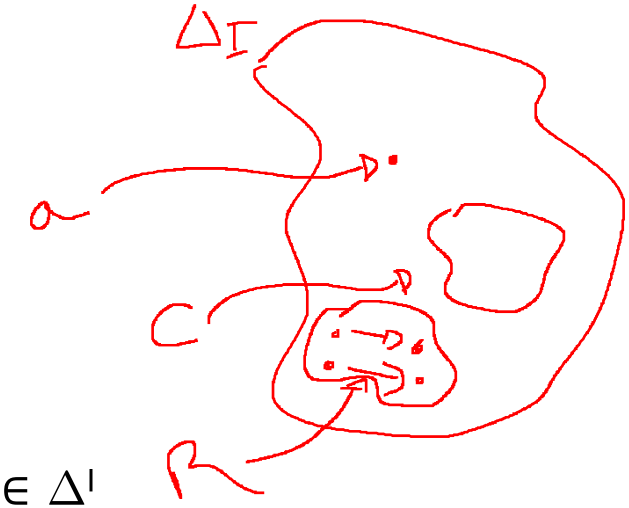
existential restriction

child **some** Student

universal restriction

child **only** Student

Semantics of $\mathcal{ALC}$



An interpretations I consists of

- a domain Δ^{I}
- an interpretation function I such that maps:

  a

  - every *individual name* a to an element $\mathrm{I}(a) \in \Delta^{\mathrm{I}}$

  C

  - every *concept* to a subset of Δ^{I}

  R

  - every *role* to a binary relation on Δ^{I} (a subset of $\Delta^{\mathrm{I}} \times \Delta^{\mathrm{I}}$)

The semantics of non atomic concepts and roles is (recursively) defined in terms of atomic concept and role interpretations.

Semantics of concept expressions

$$
\begin{aligned}
\mathrm{I}(\bot) &= \emptyset \\
\mathrm{I}(\top) &= \Delta^{\mathrm{I}} \\
\mathrm{I}(C \sqcap D) &= \mathrm{I}(C) \cap \mathrm{I}(D) \\
\mathrm{I}(C \sqcup D) &= \mathrm{I}(C) \cup \mathrm{I}(D) \\
\mathrm{I}(\neg C) &= \Delta^{\mathrm{I}} - \mathrm{I}(C) \\
\mathrm{I}(\exists R.C) &= \{x \in \Delta^{\mathrm{I}} : \exists y\text{: } (x, y) \in \mathrm{I}(R) \text{ and } y \in \mathrm{I}(C)\} \\
\mathrm{I}(\forall R.C) &= \{x \in \Delta^{\mathrm{I}} : \forall y\text{: } (x, y) \in \mathrm{I}(R) \Rightarrow y \in \mathrm{I}(C)\}
\end{aligned}
$$

EZ

I(C)

x

y in C

Exercise

$x \xrightarrow{\text{hasparent}} y \in I(\top)$

Consider the vocabulary

concepts: `Human`, `Cat`; roles: `hasParent`

and the interpretation

- $\Delta^I = \{a, b, c, d, e, f, k, l, m\}$
- $I(\text{Human}) = \{a, b, c, d, e\}$
- $I(\text{Cat}) = \{k, l, m\}$
- $I(\text{hasParent}) = \{(a, b), (a, c), (c, d), (b, e), (b, f), (k, m)\}$

What are the formal interpretations (and informal meaning) of

- `Human ⊓ Cat` $\rightarrow \emptyset$
- `¬ Human` $\rightarrow \{f, k, m, l\}$
- `∃ hasParent.⊤` $\rightarrow \{a, c, b, k\}$
- `∃ hasParent.Cat` $\rightarrow \{k\}$

ici y is in cat

Exercise (cont.)

If

- $\Delta^{\mathrm{I}} = \{\mathrm{a, b, c, d, e, f}\}$
- $\mathrm{I}(\texttt{Human}) = \{\mathrm{a, b, c, d, e}\}$
- $\mathrm{I}(\texttt{Cat}) = \{\mathrm{k, l, m}\}$
- $\mathrm{I}(\texttt{hasParent}) = \{(\mathrm{a, b}), (\mathrm{a, c}), (\mathrm{c, d}), (\mathrm{b, e}), (\mathrm{b, f}), (\mathrm{k, m})\}$

what are the interpretations of

- $\forall$.hasParent.Cat → {k, d, f, e, l, m}

on veut les individus où tous leurs parents sont des chats
vu que (d) n a pas de parent, il est dans la classe

- $\exists$ hasParent.$\top$ $\sqcap$ $\forall$.hasParent.Cat → {k}
- $\exists$ hasParent.($\exists$ hasParent.Human) → {a, c, b}
- $\exists$ hasParent.($\forall$ hasParent.$\perp$) → {b, c, k}

More class constructors ($\mathcal{ALCOQ}$)

Number restrictions on properties

$$\geq \text{n R.C}$$

$$\{x \in \Delta^I : \#\{y | (x, y) \in I(R) \; and \; y \in I(C)\} \geq n\}$$

$$\leq \text{n R.C}$$

$$\{x \in \Delta^I : \#\{y | (x, y) \in I(R) \; and \; y \in I(C)\} \leq n\}$$

$$= \text{n R.C} \leftrightarrow \geq \text{n R.C and} \leq \text{n R.C}$$

Enumeration of individuals

$$\{\mathrm{i}_1, \mathrm{i}_2, \ldots, \mathrm{i}_\mathrm{n}\}$$

$$\{x \in \Delta: x = I(i_1) \; or \; x = I(i_2) \; or \; \ldots, or \; x = I(i_n)\}$$

DL Knowledge Base ~ RDFS

Vocabulary:

class names, property names, individual names

Terminological axioms (TBox):

provide class definitions and relationships between classes

Role axioms (RBox):

about roles

Assertional axioms (ABox):

about individuals

Terminological Axioms (TBox)

Axioms of the form

$$C \sqsubseteq D$$
$$C \equiv D$$
$$C \text{ disjoint } D$$

Axiom satisfaction by an interpretation I (notation: $\mathrm{I} \vDash \mathrm{Axiom}$)

$$\mathrm{I} \vDash C \sqsubseteq D \text{ if and only if } I(C) \subseteq I(D)$$
$$\mathrm{I} \vDash C \equiv D \text{ if and only if } I(C) = I(D)$$
$$\mathrm{I} \vDash C \text{ disjoint } D \text{ if and only if } I(C) \cap I(D) \neq \emptyset$$

Exercises

Find an interpretation I of the vocabulary

- `Cat`, `Mammal`, `Human` (concept names),
- `hasParent` (role names),
- `felix`, `bob`, `alice` (individual names)

that has $\Delta^{\mathrm{I}}=\{a, b, c, d, e, f\}$ and satisfies the axioms

- Cat ⊑ Mammal
- Human ⊑ Mammal
- Human ⊑ ∀ hasParent.Human
- {bob, alice} ⊑ Human
- {felix} ⊑ Cat
- Cat ⊑ ∃ hasParent.⊤
- ⊤ ⊑ Mammal

Axioms on roles (RBox) (*ALCHOQ*)

$$P \sqsubseteq R$$

if a is linked to b through P

then a is linked to b through R

$$\mathrm{I} \models P \sqsubseteq R \text{ if and only if } \mathrm{I}(P) \subseteq \mathrm{I}(R)$$

Examples

- mother $\sqsubseteq$ parent
- primaryFunction $\sqsubseteq$ function

Axioms on roles: property chains

$$R_1 \circ R_2 \circ \ldots \circ R_n \sqsubseteq P$$

if a is linked to b through a property chain $R_1 \circ R_2 \circ \ldots \circ R_n$
then a is linked to b through P

$$\mathrm{I} \vDash R_1 \circ R_2 \circ \ldots \circ R_n \sqsubseteq P$$

if and only if

$$\forall y_0, y_1, \ldots y_n : (\mathrm{y}_0, \mathrm{y}_1) \in \mathrm{I}(R_1), \ldots, (y_{n\text{-}1}, y_n) \in \mathrm{I}(R_n) \rightarrow (y_0, y_n) \in \mathrm{I}(P)$$

Examples

- `parent o parent` $\sqsubseteq$ `grandParent`
- `parent o parent o child o child` $\sqsubseteq$ `cousinOrSiblingOrSelf`

More axioms on roles

functional(R)

$$(x, y) \in I(R) \text{ and } (x, z) \in I(R) \rightarrow y = z$$

inverse(R, S)

$$(x, y) \in I(R) \rightarrow (y, x) \in I(S)$$

symmetric(R)

$$(x, y) \in I(R) \rightarrow (y, x) \in I(R)$$

transitive(R)

$$(x, y) \in I(R) \; and \; (y, z) \in I(R) \rightarrow (x, z) \in I(R)$$

reflexive(R)

$$\forall x \in \Delta^I : (x, x) \in I(R)$$

Examples

functional(`biologicalMother`)

inverse(`mother, child`)

symmetric(`friend`)

transitive(`before`)

reflexive(`knows`)

Assertions on individuals (ABox)

Axioms asserting that

- a named individual belongs to a class

$$C(i)$$

$$I \vDash C(i) \text{ if and only if } I(i) \in I(C)$$

- a named individual is linked to another one through a role

$$R(i,j)$$

$$I \vDash R(i,j) \text{ if and only if } (I(i), I(j)) \in I(R)$$

Assertions on individuals (ABox)

Axioms asserting that

- two named individuals are different

$$i \text{ differentFrom } j$$

$I \vDash i$ differentFrom j if and only if $I(i) \neq I(j)$

- two named inidividuals are equal

$$i \text{ sameAs } j$$

$I \vDash i$ sameAs j if and only if $I(i) = I(j)$

Exercise

Find a minimal interpretation that satisfies

- `Man(bob)`
- `Woman(lisa)`
- `Human(sam)`
- Man $\sqcup$ Woman $\sqsubseteq$ Human
- `Man disjointFrom Woman`
- `hasSibling(bob, lisa)`
- `symmetric(hasSibling)`
- `father(lisa, max)`
- `father(lisa, mix)`
- `mix differentFrom sam`
- `functional(father)`

An Introduction to Description Logics

2. Reasoning Tasks

G. Falquet

Reasoning Tasks

- Consistency
- Subsumption
- Open world
- Unique name
- Instance checking

Consider the axioms

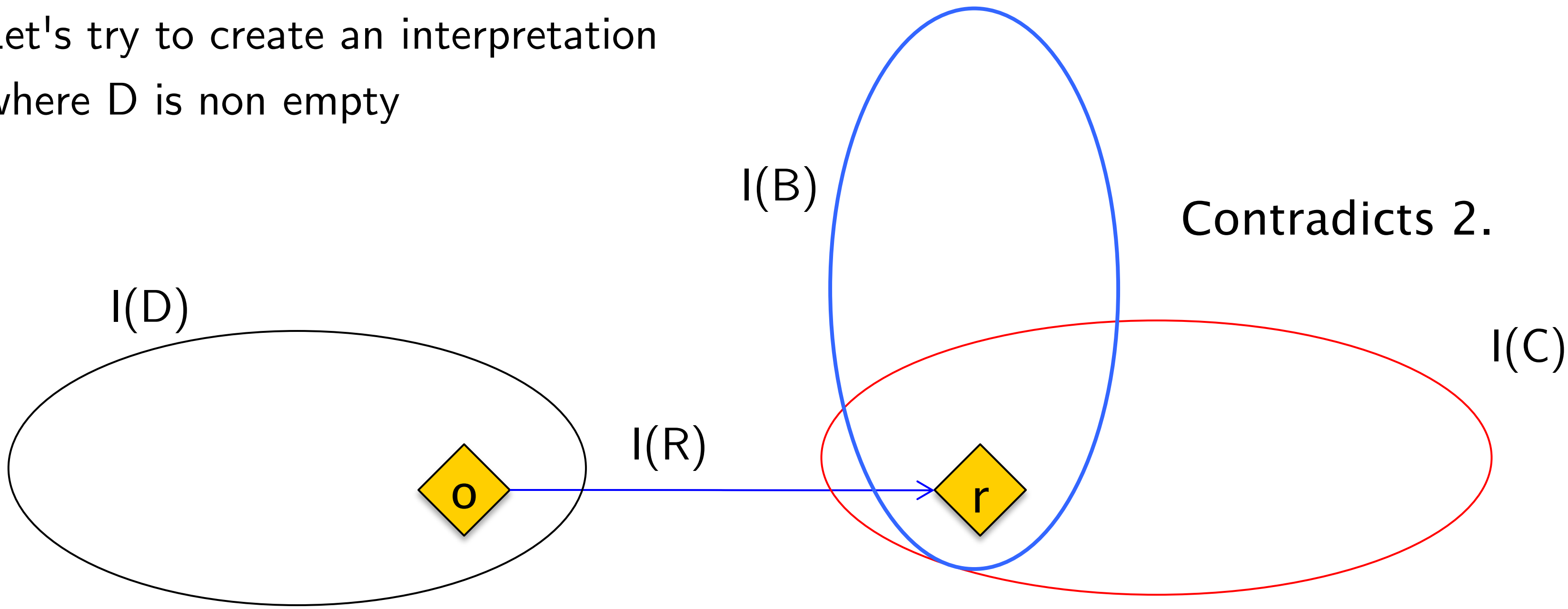
1. A $\sqsubseteq$ ($\forall$ R . B)
2. C disjoint B
3. D $\sqsubseteq$ (($\exists$ R . C) $\sqcap$ A)

Let's try to create an interpretation where D is non empty

Consider the axioms

1. A $\sqsubseteq$ $\boxed{(\forall\ R\ .\ B)}$
2. C disjoint B
3. D $\sqsubseteq$ ($\boxed{(\exists\ R\ .\ C)}$ $\sqcap$ $\boxed{A}$)

Let's try to create an interpretation where D is non empty



Consistency

- a knowledge base is consistent if there is an interpretation such that all the axioms are satisfied

- a concept C is consistent if we can populate the ontology so as to
  - satisfy all the axioms
  - have at least one object in C

i.e. there is an interpretation I such that

1. $I \models \text{TBox}$
2. $I \not\models C \sqsubseteq \bot$

Example : TBox vs. Concept Consistency

TBox $\boldsymbol{T}$ =

$W \sqsubseteq \{w\}$

$W \sqsubseteq \exists r.\top$

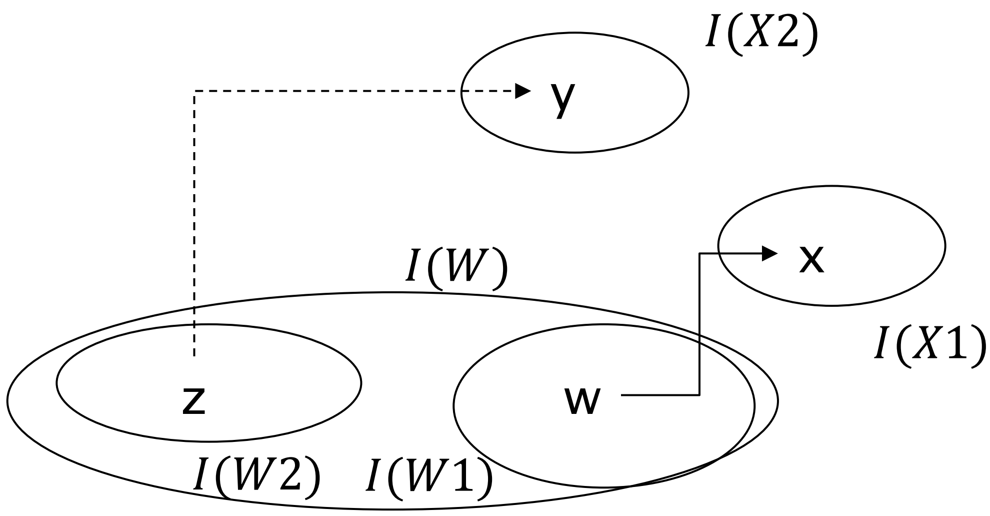
$W \sqsubseteq (\forall r.X1) \sqcup (\forall r.X2)$

$W1 \sqsubseteq W \sqcap (\forall r.X1)$

$W2 \sqsubseteq W \sqcap (\forall r.X2)$

$W1\ disjoint\ W2$

$X1\ disjoint\ X2$



$\boldsymbol{T}$ is consistent but in every model I of **T**,
if $I(W1)$ is non-empty then $I(W2)$ is empty, and vice versa.

Reasoning tasks: subsumption

Given a TBox $\boldsymbol{T}$, C subsumes D if

for every model I of $\boldsymbol{T}$, $I(D) \subseteq I(C)$

or equivalently

$\boldsymbol{T} \cup \{D \sqcap \neg C\}$ is inconsistent

Reasoning task:

input: a Tbox T, two classes C, D

output: true iff C subsumes D for T

Reasoning tasks: Instance checking

1. check if $C(o)$ is a consequence of the axioms and asserted facts

   amounts to check if C subsumes the concept $\{o\}$

2. find all the individuals that belong to C

   similar to query answering in (deductive) databases

Example

Find facts about individuals belonging to classes.

1. Parent $\equiv$ $\exists$ hasChild . Person
2. hasChild(Bob, Alice)
3. Woman(Alice)
4. Woman $\sqsubseteq$ Person

consequence

Parent(Bob)

Open World Semantics

What is not explicitly asserted is unknown (maybe true maybe false). Leads to counter intuitive results:

1. $GoParent \equiv \forall\, hasChild\,.\, Girl$
2. hasChild(Bob, Alice)
3. Girl(Alice)

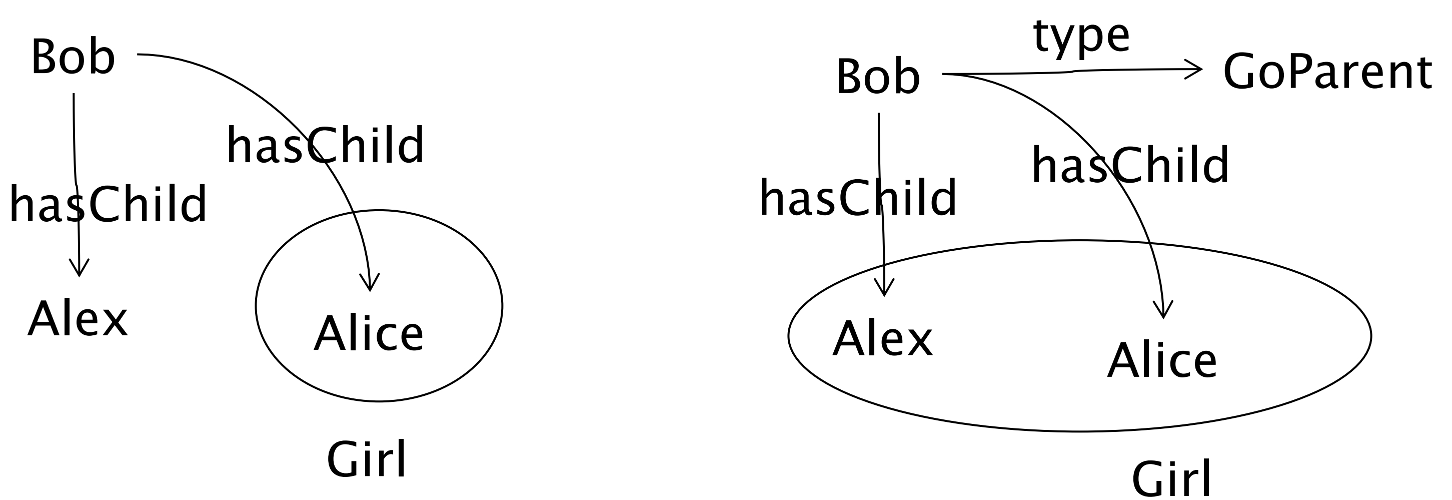
can we infer GoParent(Bob) ?

No, (Bob may have other children who are not girls)

Open World Semantics

Some models of

1. GoParent $\equiv \forall$ hasChild . Girl
2. hasChild(Bob, Alice)
3. Girl(Alice)



closing the world

1. GoParent $\equiv$ $\forall$ hasChild . Girl
2. hasChild(Bob, Alice)
3. Girl(Alice)
4. ParentOf1 $\sqsubseteq$ hasChild $=_1$ Thing
5. ParentOf1(Bob)

now we can infer Bob a GoParent

No Unique Name Assumption (UNA)

1. BusyParent $\equiv$ hasChild $\geq_2$ Person
2. hasChild (Cindy, Bob)
3. hasChild (Cindy, John)

consequence: BusyParent (Cindy) ?

no, because *Bob* and *John* may be the same person

yes if we add the axiom

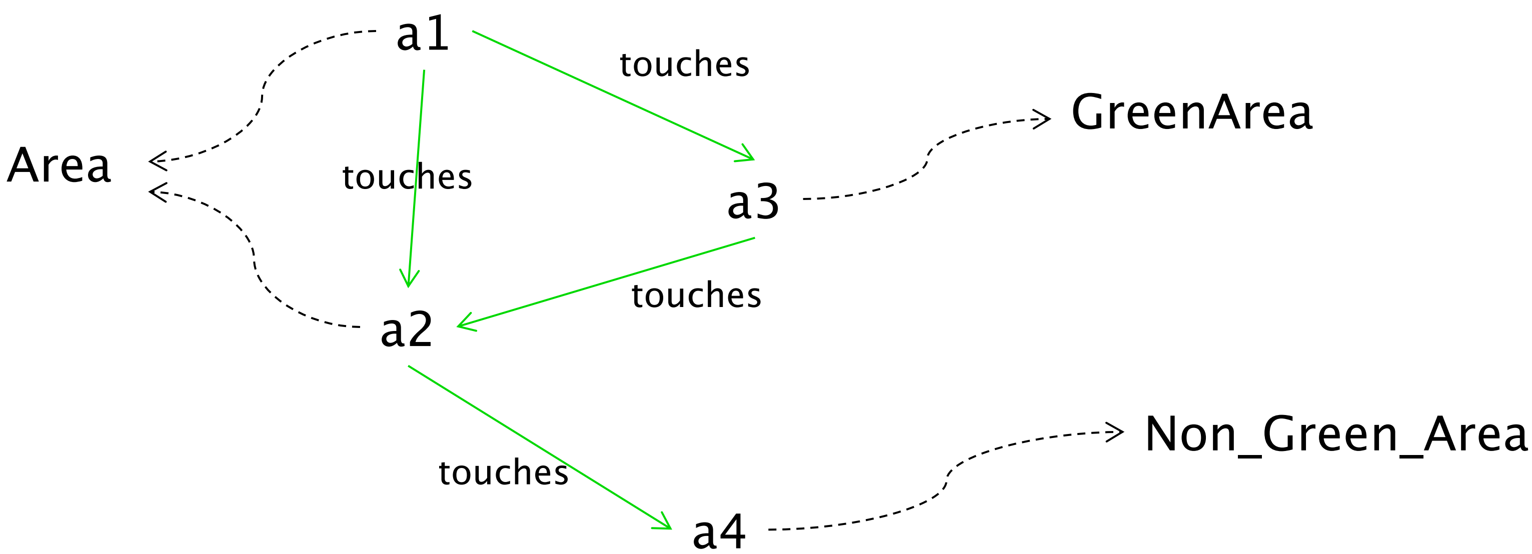
Bob $\neq$ John

Sophisticated “open world” reasoning

Terminological Axioms (TBox)

1. $\text{Green\_Area} \sqsubseteq \text{Area}$
2. $\text{Non\_Green\_Area} \equiv \text{Area} \sqcap (\neg\ \text{Green\_Area})$

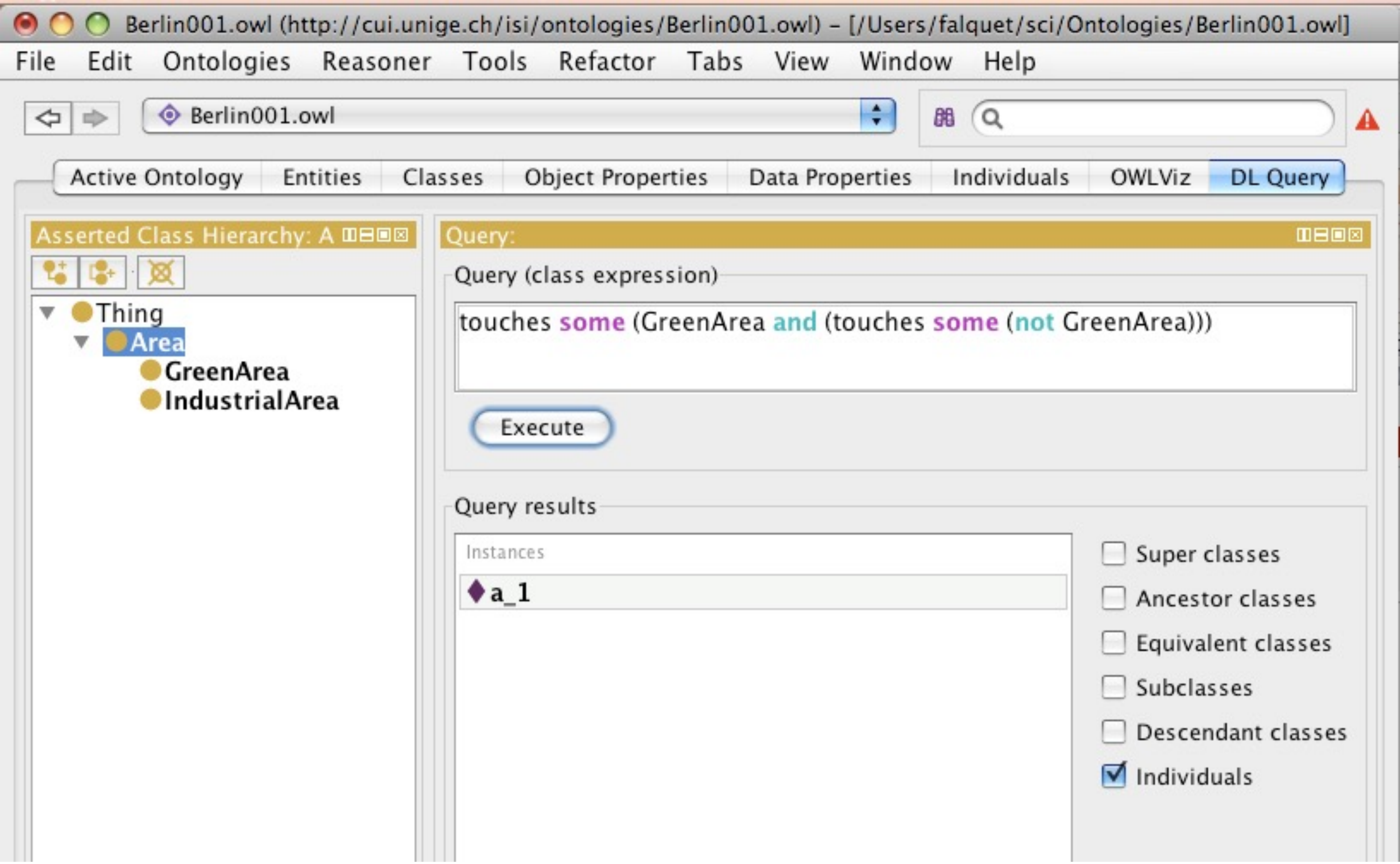
ABox



Q: Does a1 touch some Green Area that touches some non Green Area?

A: Yes

- a2 is either green or non green (axioms 1 and 2)
- if it is green a1 satisfies the condition (using a3, a2)
- if it is non green a1 satisfies the condition (using a2, a4)


Berlin001.owl (http://cui.unige.ch/isi/ontologies/Berlin001.owl) – [/Users/falquet/sci/Ontologies/Berlin001.owl]
File Edit Ontologies Reasoner Tools Refactor Tabs View Window Help
Berlin001.owl
Active Ontology
Entities
Classes
Object Properties
Data Properties
Individuals
OWLViz
DL Query
Asserted Class Hierarchy: A
Thing
Area
GreenArea
IndustrialArea
Query:
Query (class expression)
touches some (GreenArea and (touches some (not GreenArea)))
Execute
Query results
Instances
a_1
Super classes
Ancestor classes
Equivalent classes
Subclasses
Descendant classes
Individuals

Reasoning Services for DL Ontologies

- In most description logics consistency and subsumption can be computed (with sophisticated tableau algorithms), with different time and space complexities

- Consequences
  - the consistency of an ontology can be checked
  - it is possible to compute the class subsumption hierarchy
  - it is possible to find the closest concept corresponding to a query

- There are description logics for which consistency and subsumption can be computed in polynomical time or better
  - OWL-RL, OWL-QL

Everything about DL

- at http://dl.kr.org/
- and http://www.cs.man.ac.uk/~ezolin/dl/

Complexity of reasoning in Description Logics

Note: the information here is (always) incomplete and updated often

Base description logic: $\mathcal{A}$ttributive $\mathcal{L}$anguage with $\mathcal{C}$omplements

$\mathcal{ALC} ::= \bot \mid \top \mid A \mid \neg C \mid C \sqcap D \mid C \sqcup D \mid \exists R.C \mid \forall R.C$

Concept constructors:

- [ ] $\mathcal{F}$ – functionality[2]: $(\leq 1\ R)$
- [x] $\mathcal{N}$ – (unqualified) number restrictions: $(\geq n\ R)$, $(\leq n\ R)$
- [x] $\mathcal{Q}$ – qualified number restrictions: $(\geq n\ R.C)$, $(\leq n\ R.C)$
- [x] $\mathcal{O}$ – nominals: $\{a\}$ or $\{a_1, \ldots, a_n\}$ ("one-of")
- [ ] μ – least fixpoint operator: $\mu X.C$

Forbid complex roles[5] in number restrictions[6]

Role constructors: trans reg

- [x] $\mathcal{I}$ – role inverse: R^-
- [ ] $\cap$ – role intersection[3]: $R \cap S$
- [ ] $\cup$ – role union: $R \cup S$
- [ ] $\neg$ – role complement: $\neg R$ full
- [ ] $\circ$ – role chain (composition): $R \circ S$
- [ ] $*$ – reflexive-transitive closure[4]: R^*
- [ ] id – concept identity: $id(C)$

TBox (concept axioms) is *internalizable* in extensions of $\mathcal{ALCIO}$, see [82, Lemma 4.12], [61, p.3]

- (•) empty TBox
- ( ) acyclic TBox ($A \equiv C$, A is a concept name; no cycles)
- ( ) general TBox ($C \sqsubseteq D$, for arbitrary concepts C and D)

RBox (role axioms): OWL-Lite OWL-DL OWL 1.1

- [x] $\mathcal{S}$ – role transitivity: $Tr(R)$
- [x] $\mathcal{H}$ – role hierarchy: $R \sqsubseteq S$
- [ ] $\mathcal{R}$ – complex role inclusions: $R \circ S \sqsubseteq R$, $R \circ S \sqsubseteq S$
- [ ] s – some additional features (click to see them)

Reset — You have selected a Description Logic: $\mathcal{SHOIQ}$

Complexity[7] of reasoning problems[8]

Concept satisfiability	**NExpTime-complete**	• Hardness of even $\mathcal{ALCFIO}$ is proved in [82, Corollary 4.13]. • A different proof of the NExpTime-hardness for $\mathcal{ALCFIO}$ is given in [61] (even with 1 nominal, and inverse roles not used in number restrictions). • Upper bound for $\mathcal{SHOIQ}$ is proved in [12, Corollary 6.31] with numbers coded in unary (for binary coding, the upper bound remains an open problem for all logics in between $\mathcal{ALCNIO}$ and $\mathcal{SHOIQ}$). • A tableaux algorithm for $\mathcal{SHOIQ}$ is presented in [51]. • **Important:** in number restrictions, only *simple* roles (i.e. which are neither transitive nor have a transitive subroles) are allowed; otherwise we gain undecidability even in $\mathcal{SHN}$; see [54]. • **Remark:** recently [55] it was observed that, in many cases, one can use transitive roles in number restrictions –