OWL and Description Logic Reasoning

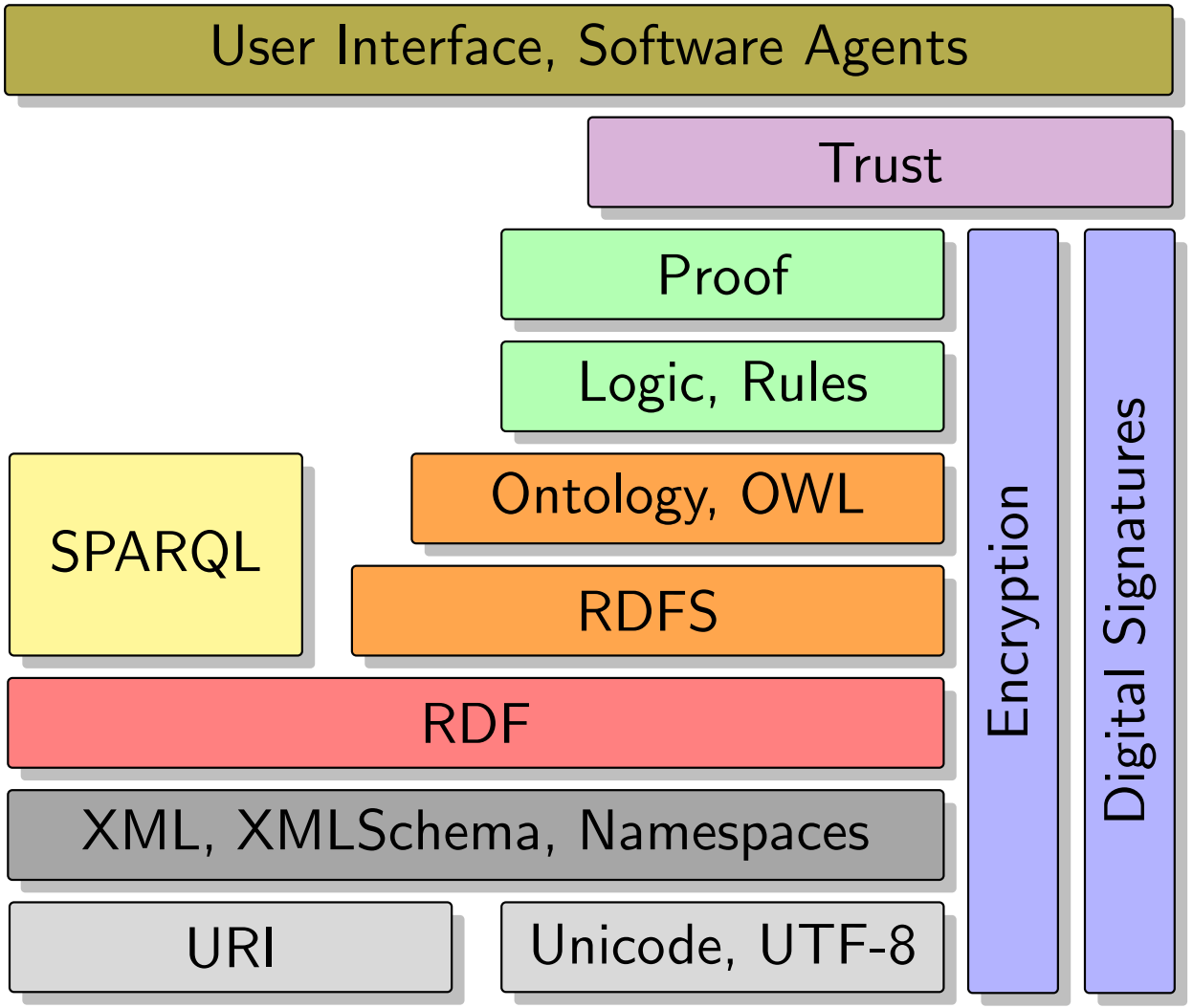
Camilo Thorne

Room 00.012
Institut für Maschinelle Sprachverarbeitung
Universität Stuttgart
+49 (0) 711 685-81369
camilo.thorne@ims.uni-stuttgart.de

Semantic Web, SS 2017
(based on slides by W. Kessler)

Institut für Maschinelle Sprachverarbeitung

The Semantic Web Stack [W3C, Tim Berners-Lee]



Institut für Maschinelle Sprachverarbeitung

Outline

1. Introduction to Description Logics
2. Summary
3. References

Institut für Maschinelle Sprachverarbeitung

Outline

1 Introduction to Description Logics

2 Summary

3 References

Institut für Maschinelle Sprachverarbeitung

Description Logic and OWL

- OWL DL is based on **Description Logics** (DLs)
- DL is a family of knowledge representation languages that correspond to a decidable fragment of first-order predicate logic
- This means, we can use everything we know about first-order predicate logic and apply it to ontologies described with OWL
- First-order logic is well-established and there are several existing inference algorithms (resolution, tableaux algorithms) with well-understood theoretical properties
- This section assumes a basic knowledge of first order predicate logic (FOL)

Institut für Maschinelle Sprachverarbeitung

Basic vocabulary:

- Atomic concepts: $Pizza$ (...)
- Individuals: $mypizza$, (...)
- Properties (roles, relations): $hasTopping$ (...)

The sets of atomic concepts, properties and individuals are disjoint

The basic Description Logic $\mathcal{ALC}$, is built by recursively closing the set of concepts under concept constructors

Institut für Maschinelle Sprachverarbeitung

Intersection, Union, Negation

- Intersection (conjunction):
  - $VegetarianPizza \sqcap SpicyPizza$
  - A pizza that is vegetarian **and** spicy

Institut für Maschinelle Sprachverarbeitung

Intersection, Union, Negation

- Intersection (conjunction):
  - $VegetarianPizza \sqcap SpicyPizza$
  - A pizza that is vegetarian **and** spicy
- Union (disjunction):
  - $VegetarianPizza \sqcup SpicyPizza$
  - A pizza that is vegetarian **or** spicy

Institut für Maschinelle Sprachverarbeitung

Intersection, Union, Negation

- Intersection (conjunction):
  - $VegetarianPizza \sqcap SpicyPizza$
  - A pizza that is vegetarian **and** spicy
- Union (disjunction):
  - $VegetarianPizza \sqcup SpicyPizza$
  - A pizza that is vegetarian **or** spicy
- Negation (complement):
  - $\neg VegetarianPizza$
  - A pizza that is not a vegetarian pizza

Institut für Maschinelle Sprachverarbeitung

Intersection, Union, Negation

- Intersection (conjunction):
  - $VegetarianPizza \sqcap SpicyPizza$
  - A pizza that is vegetarian **and** spicy
  - FOL: $VegetarianPizza(x) \wedge SpicyPizza(x)$
- Union (disjunction):
  - $VegetarianPizza \sqcup SpicyPizza$
  - A pizza that is vegetarian **or** spicy
  - FOL: $VegetarianPizza(x) \vee SpicyPizza(x)$
- Negation (complement):
  - $\neg VegetarianPizza$
  - A pizza that is not a vegetarian pizza
  - FOL: $\neg VegetarianPizza(x)$

Institut für Maschinelle Sprachverarbeitung

Quiz: Basics of DLs

Which DL statements match the definition of "landlocked political entity"?

A sovereign state or country entirely enclosed by land

A) $(Country \sqcap State) \sqcap EnclosedByLand$

B) $(Country \sqcup State) \sqcap EnclosedByLand$

C) $(Country \sqcup State) \sqcup EnclosedByLand$

D) $(Country \sqcap State) \sqcup EnclosedByLand$

Institut für Maschinelle Sprachverarbeitung

Universal Quantifier

- Universal Quantifier:
  - For a cheese pizza: $\forall hasTopping.CheeseTopping$
  - Every cheese pizza has only cheese toppings.
  - FOL: $(\forall y)(hasTopping(x, y) \rightarrow CheeseTopping(y))$

- `owl:allValuesFrom` restriction:

```
[
   rdf:type owl:Restriction ;
   owl:onProperty ex:hasTopping ;
   owl:allValuesFrom ex:CheeseTopping
] .
```

Institut für Maschinelle Sprachverarbeitung

Existential Quantifier

- Existential Quantifier:
  - For a fish pizza: $\exists hasTopping.FishTopping$
  - Every fish pizza has at least one fish topping.
  - FOL: $(\exists y)(hasTopping(x, y) \wedge FishTopping(y))$

- `owl:someValuesFrom` restriction:

```
[
   rdf:type owl:Restriction ;
   owl:onProperty ex:hasTopping ;
   owl:someValuesFrom ex:FishTopping
] .
```

Institut für Maschinelle Sprachverarbeitung

Quiz: Restrictions in DLs

The property $hasBorder$ links a country and what it borders:

$$hasBorder(germany, northSea) \qquad Ocean(northSea)$$
$$hasBorder(germany, france) \qquad Country(france)$$

Which DL statements match the definition of "enclosed by land"?

Something that has no border with an ocean.

A) $\neg\forall hasBorder.Ocean$

B) $\neg\exists hasBorder.Ocean$

C) $\forall hasBorder.\neg Ocean$

D) $\exists hasBorder.\neg Ocean$

Institut für Maschinelle Sprachverarbeitung

Subclass and Equivalence

- Subclass relation:
  - $VegetarianPizza \sqsubseteq Pizza$
  - Every vegetarian pizza is a pizza
    (but not all pizzas are vegetarian)
  - FOL: $(\forall x)(VegetarianPizza(x) \rightarrow Pizza(x))$
  - FOL: $(\forall x)(Pizza(x) \vee \neg VegetarianPizza(x))$

Institut für Maschinelle Sprachverarbeitung

Subclass and Equivalence

- Subclass relation:
  - $VegetarianPizza \sqsubseteq Pizza$
  - Every vegetarian pizza is a pizza
    (but not all pizzas are vegetarian)
  - FOL: $(\forall x)(VegetarianPizza(x) \rightarrow Pizza(x))$
  - FOL: $(\forall x)(Pizza(x) \vee \neg VegetarianPizza(x))$
- Equivalence relation:
  - $VegetarianPizza \equiv PizzaVegetarian$
  - VegetarianPizza and PizzaVegetarian are exactly the same thing, every A is a B and every B is a
  - FOL: $(\forall x)(VegetarianPizza(x) \leftrightarrow PizzaVegetarian(x))$
  - FOL: $(\forall x)((VegetarianPizza(x) \wedge PizzaVegetarian(x)) \vee (\neg VegetarianPizza(x) \wedge \neg PizzaVegetarian(x)))$

Institut für Maschinelle Sprachverarbeitung

Subclass vs. Equivalence

- Be careful with $\sqsubseteq$ and $\equiv$.
- A yummy pizza is equivalent to a vegetarian and spicy pizza:

$$YummyPizza \equiv VegetarianPizza \sqcap SpicyPizza$$

- Some of the pizzas that are vegetarian and spicy are yummy:

$$YummyPizza \sqsubseteq VegetarianPizza \sqcap SpicyPizza$$

- $\equiv$ means that the classes are **exactly** the same, it gives a definition of the class. Every pizza that is vegetarian and spicy is a yummy pizza.
- $\sqsubseteq$ means that one class is a **subset** of the other. No pizza is yummy that is not vegetarian and spicy, but there could be a pizza that is vegetarian and spicy and not yummy.

Institut für Maschinelle Sprachverarbeitung

Quiz: The Good Friday Problem

We know that fish pizza is not a vegetarian pizza

Is a fish pizza a meaty pizza, if meaty pizza is defined with

A) Equivalent class:

$$MeatyPizza \equiv \neg VegetarianPizza$$

B) Subclass:

$$MeatyPizza \sqsubseteq \neg VegetarianPizza$$

Institut für Maschinelle Sprachverarbeitung

Quiz: DLs

Which of the DL statements best describes the content of the following sentence:

A wizard is defined as being a human who has to wear at least one magic ring, i.e., every human who wears at least one magic ring is a wizard and every wizard is a human who wears at least one magic ring.

A) $Wizard \sqsubseteq Human \sqcap \exists wearsRing.MagicRing$

B) $Wizard \equiv Human \sqcap \exists wearsRing.MagicRing$

C) $Wizard \sqsubseteq Human \sqcup \forall wearsRing.MagicRing$

D) $Wizard \equiv Human \sqcap \forall wearsRing.MagicRing$

Institut für Maschinelle Sprachverarbeitung

OWL and DLs

- In these slides we have only introduced $\mathcal{ALC}$, the basic Description Logic
- In addition to $\mathcal{ALC}$ some operators from other logics are needed to fully represent OWL DL
- Logics are indicated by letters which describe which language constructs are allowed in the fragment
- Specifically, OWL 1 DL is $\mathcal{SHOIN}(\mathcal{D})$, OWL 2 DL is $\mathcal{SROIQ}(\mathcal{D})$ (you need not know this)
- OWL Full is not a Description Logic

Institut für Maschinelle Sprachverarbeitung

Equivalence of OWL and DLs (1)

- **Classes:** $(\mathcal{ALC})$

Subclass	$A \sqsubseteq B$	`rdfs:subClassOf`
Class equivalence	$A \equiv B$	`owl:equivalentClass`
Class disjointness	$A \sqcap B \sqsubseteq \bot$	`owl:disjointWith`
Class intersection	$A \sqcap B$	`owl:intersectionOf`
Class union	$A \sqcup B$	`owl:unionOf`
Class negation	$\neg A$	`owl:complementOf`

- **Thing and Nothing:** $(\mathcal{ALC})$

Thing	$\top \equiv A \sqcup \neg A$	`owl:Thing`
Nothing	$\bot \equiv A \sqcap \neg A$	`owl:Nothing`

Institut für Maschinelle Sprachverarbeitung

Equivalence of OWL and DLs (2)

- **Instances:** ($\mathcal{O}$)

Instances equivalence	$\{d\} \equiv \{e\}$	`owl:sameIndividualAs`
Instances difference	$\{d\} \sqsubseteq \neg\{e\}$	`owl:differentFrom`

- **Quantifiers:** ($\mathcal{ALC}$ and $\mathcal{O}$ for specific value)

Existential quantifier	$\exists r.A$	`owl:someValuesFrom`
Universal quantifier	$\forall r.A$	`owl:allValuesFrom`
Specific value	$\exists r.\{d\}$	`owl:hasValue`

- **Domain and Range:** ($\mathcal{ALC}$)

Domain	$\exists r.\top \sqsubseteq A$	`rdfs:domain`
Range	$\top \sqsubseteq \forall r.A$	`rdfs:range`

Institut für Maschinelle Sprachverarbeitung

Equivalence of OWL and DLs (3)

- Data types can be modeled ($\mathcal{D}$)
- Cardinality restrictions ($\mathcal{N}$) and qualified cardinality restrictions ($\mathcal{Q}$) can be modeled
- Property equivalence, disjointness, and subproperties can be modeled ($\mathcal{H}$ for subproperty)
- Characteristics of properties like inverse property, (in)transitive, (a)symmetric, (inverse) functional can be modeled ($\mathcal{S}$ for transitive and $\mathcal{I}$ for inverse)
- Property chains and disjointness can be modeled ($\mathcal{R}$)

Institut für Maschinelle Sprachverarbeitung

TBoxes and ABoxes

In Description Logic ontologies $\mathcal{O}$ it is common to distinguish between terminological (conceptual) and logical relations, and instance-level information or facts, mirroring RDF and RDFS levels in RDF(S)

- A **terminological box** (TBox) $\mathcal{T}$ is the set of all concept inclusion (or equivalece) statements $A \sqsubseteq B$ in $\mathcal{O}$
- A **assertional box** (ABox) $\mathcal{A}$ is the set of all statements $A(d)$, $r(d, e)$ in $\mathcal{O}$ involving only individuals

TBox and Abox statements as usually called **assertions**

Institut für Maschinelle Sprachverarbeitung

Set-theoretical Interpretations

Formal semantics is defined much like RDFS in terms of set-theoretical models

An **interpretation** is a pair $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ where $\Delta^{\mathcal{I}}$ is a domain of individuals and $\cdot^{\mathcal{I}}$ recursively maps individuals to domain elements, concepts to sets over $\Delta^{\mathcal{I}}$ and roles to binary relations over $\Delta^{\mathcal{I}}$; it is said to be a **model** of $\mathcal{O}$ if it makes true all of $\mathcal{O}$'s statements

Institut für Maschinelle Sprachverarbeitung

Formal Semantics of DLs (summary)

Syntax	Semantics w.r.t. interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$		
d	$d^{\mathcal{I}}$	$\in$	$\Delta^{\mathcal{I}}$
A	$A^{\mathcal{I}}$	$\subseteq$	$\Delta^{\mathcal{I}}$
$\{d_1, \ldots, d_n\}$	$(\{d_1, \ldots, d_n\})^{\mathcal{I}}$	$=$	$\{d_1^{\mathcal{I}}, \ldots, d_n^{\mathcal{I}}\}$
$\neg A$	$(\neg A)^{\mathcal{I}}$	$=$	$\Delta^{\mathcal{I}} \setminus A^{\mathcal{I}}$
$\exists r.A$	$(\exists r.A)^{\mathcal{I}}$	$=$	$\{d \mid \text{exists } e \text{ s.t. } (d,e) \in r^{\mathcal{I}} \text{ and } e \in A^{\mathcal{I}}\}$
$\forall r.A$	$(\forall r.A)^{\mathcal{I}}$	$=$	$\{d \mid \text{for all } e, \text{ if } (d,e) \in r^{\mathcal{I}}, \text{ then } e \in A^{\mathcal{I}}\}$
$\exists_{\leq k} r.A$	$(\exists_{\leq k} r.A)^{\mathcal{I}}$	$=$	$\{d \mid \text{exist at most } k \ e \in A \text{ s.t. } (d,e) \in r^{\mathcal{I}}\}$
$\exists_{\geq k} r.A$	$(\exists_{\geq k} r.A)^{\mathcal{I}}$	$=$	$\{d \mid \text{exist at least } k \ e \in A \text{ s.t. } (d,e) \in r^{\mathcal{I}}\}$
$A \sqcap B$	$(A \sqcap B)^{\mathcal{I}}$	$=$	$A^{\mathcal{I}} \cap B^{\mathcal{I}}$
$A \sqcup B$	$(A \sqcup B)^{\mathcal{I}}$	$=$	$A^{\mathcal{I}} \cup B^{\mathcal{I}}$
r	$r^{\mathcal{I}}$	$\subseteq$	$\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
r^-	$(r^-)^{\mathcal{I}}$	$=$	$\{(d,e) \mid (d,e) \in r^{\mathcal{I}}\}$
$A \sqsubseteq B$	$\mathcal{I} \models A \sqsubseteq B$	iff	$A^{\mathcal{I}} \subseteq B^{\mathcal{I}}$
$r \sqsubseteq s$	$\mathcal{I} \models r \sqsubseteq s$	iff	$r^{\mathcal{I}} \subseteq s^{\mathcal{I}}$
$A(d)$	$\mathcal{I} \models \phi(d)$	iff	$d^{\mathcal{I}} \in A^{\mathcal{I}}$
$r(d,e)$	$\mathcal{I} \models r(d,e)$	iff	$(d,e)^{\mathcal{I}} \in r^{\mathcal{I}}$
$\mathcal{T}$	$\mathcal{I} \models \mathcal{T}$	iff	for all $\alpha \in \mathcal{T}, \mathcal{I} \models \alpha$
$\mathcal{A}$	$\mathcal{I} \models \mathcal{A}$	iff	for all $\alpha \in \mathcal{A}, \mathcal{I} \models \alpha$
$(\mathcal{T}, \mathcal{A})$	$\mathcal{I} \models (\mathcal{T}, \mathcal{A})$	iff	$\mathcal{I} \models \mathcal{T}$ and $\mathcal{I} \models \mathcal{A}$

Rem: Other concept constructors (for e.g., $\top$, $\bot$ or $\exists r.\{d_1, \ldots, d_n\}$) can be derived from the ones in this table!

Institut für Maschinelle Sprachverarbeitung

Interpretation – Example

The following interpretation $\mathcal{I}_b$ is a model of the following ontology $\mathcal{O}_b$, i.e., $\mathcal{I}_b \models \mathcal{O}_b$

$$\begin{aligned}
\mathsf{ScienceFiction} &\sqsubseteq \mathsf{Genre} \\
\mathsf{ScienceFiction} &\sqsubseteq \exists \mathsf{hasReader}.\top \\
\mathsf{ElizabethanTeatre} &\sqsubseteq \mathsf{Genre} \\
\mathsf{ScienceFiction} \sqcap \mathsf{ElizabethanTeatre} &\sqsubseteq \bot \\
\mathsf{ElizabethanTeatre}(\mathsf{the_tempest}) & \\
\mathsf{ScienceFiction}(\mathsf{dune}), \mathsf{hasReader}(\mathsf{dune}, \mathsf{joe}) &
\end{aligned}$$

Institut für Maschinelle Sprachverarbeitung

Interpretation – Example

The following interpretation $\mathcal{I}_b$ is a model of the following ontology $\mathcal{O}_b$, i.e., $\mathcal{I}_b \models \mathcal{O}_b$

$$\begin{aligned}
\mathsf{ScienceFiction} &\sqsubseteq \mathsf{Genre} \\
\mathsf{ScienceFiction} &\sqsubseteq \exists \mathsf{hasReader}.\top \\
\mathsf{ElizabethanTeatre} &\sqsubseteq \mathsf{Genre} \\
\mathsf{ScienceFiction} \sqcap \mathsf{ElizabethanTeatre} &\sqsubseteq \bot \\
\mathsf{ElizabethanTeatre}(\mathsf{the_tempest}) & \\
\mathsf{ScienceFiction}(\mathsf{dune}), \mathsf{hasReader}(\mathsf{dune}, \mathsf{joe}) &
\end{aligned}$$

$$\begin{aligned}
\Delta^{\mathcal{I}} &= \{\mathsf{dune}, \mathsf{joe}, \mathsf{the_tempest}\} \\
\mathsf{dune}^{\mathcal{I}} &= \mathsf{dune} \\
\mathsf{the_tempest}^{\mathcal{I}} &= \mathsf{the_tempest} \\
\mathsf{joe}^{\mathcal{I}} &= \mathsf{joe} \\
\mathsf{ScienceFiction}^{\mathcal{I}} &= \{\mathsf{dune}\} \\
\mathsf{ElizabethanTeatre}^{\mathcal{I}} &= \{\mathsf{the_tempest}\} \\
\mathsf{Genre}^{\mathcal{I}} &= \{\mathsf{dune}, \mathsf{the_tempest}\} \\
\mathsf{hasReader}^{\mathcal{I}} &= \{(\mathsf{dune}, \mathsf{joe})\}
\end{aligned}$$

Institut für Maschinelle Sprachverarbeitung

Inference Problems as (Un)satisfiability Problems

An ontology $\mathcal{O}$ is said to entail an assertion or statement α much in the same way as RDFS; it can also be inconsistent, as we have negation

Let $\mathcal{O}$ be an ontology and α an assertion:

- $\mathcal{O}$ is **consistent** iff it exists a model $\mathcal{I}$ such that $\mathcal{I} \models \mathcal{O}$
- $\mathcal{O}$ **entails** α iff for all $\mathcal{I}$, if $\mathcal{I} \models \mathcal{O}$ then $\mathcal{I} \models \alpha$

Ontology consistency and entailment can be reduced to each modulo negation:

consistency $\triangleright$ non-entailment: $\mathcal{O} \cup$ is satisfiable $\iff \mathcal{O} \not\models \bot$

entailment $\triangleright$ inconsistency: $\mathcal{O} \models \alpha \iff \mathcal{O} \cup \neg\alpha$ is inconsistent

Institut für Maschinelle Sprachverarbeitung

Logic and OWL

- In Slide 23 $\mathcal{O}_b$ is consistent ($\mathcal{I}_b$ is a model of $\mathcal{O}_b$, hence a model exists)
- In Slide 23 $\mathcal{O}_b$ entails
  - Genre(the_tempest): The Tempest belongs to a literary genre (it can be shown that every model of $\mathcal{O}_b$ is a model of Genre(the_tempest))
  - Genre(dune): Dune belongs to a literary genre (idem)

OWL DL Ontology editing and management tools such a Protégé implement algorithms that deal with these reasoning tasks, called **reasoners**

Institut für Maschinelle Sprachverarbeitung

Logic and OWL

- In Slide 23 $\mathcal{O}_b$ is consistent ($\mathcal{I}_b$ is a model of $\mathcal{O}_b$, hence a model exists)
- In Slide 23 $\mathcal{O}_b$ entails
  - Genre(the_tempest): The Tempest belongs to a literary genre (it can be shown that every model of $\mathcal{O}_b$ is a model of Genre(the_tempest))
  - Genre(dune): Dune belongs to a literary genre (idem)

OWL DL Ontology editing and management tools such a Protégé implement algorithms that deal with these reasoning tasks, called **reasoners**

They support also SPARQL queries as querying can be seen as a form of entailment (and thus reduces to the cases seen in Slide 24)

Institut für Maschinelle Sprachverarbeitung

Outline

1 Introduction to Description Logics

2 Summary

3 References

Institut für Maschinelle Sprachverarbeitung

Summary: Reasoning

- OWL can be expressed in **Description Logics** (DLs)
- OWL uses the **Open World Assumption** (OWA)
- OWL formal semantics is based on set-theoretical interpretations, just as RDFS; we have the same notion of entailment
- With the addition of negation, we can consider ontology consistency
- Based on first-order logic

Institut für Maschinelle Sprachverarbeitung

Outline

1 Introduction to Description Logics

2 Summary

3 References

Institut für Maschinelle Sprachverarbeitung

Suggested Reading

- Pascal Hitzler, Markus Krötzsch, Sebastian Rudolph and York Sure. Semantic Web. Grundlagen. Springer textbook, 2008. (Chapter 6)
- Pascal Hitzler, Markus Krötzsch and Sebastian Rudolph. Foundations of Semantic Web Technologies. Chapman & Hall/CRC, 2009. (Chapter 5)

Institut für Maschinelle Sprachverarbeitung