Description Logics

Franz Baader

Theoretical Computer Science

TU Dresden

Germany

1. Motivation and introduction to Description Logics
2. Tableau-based reasoning procedures
3. Automata-based reasoning procedures
4. Complexity of reasoning in Description Logics
5. Reasoning in inexpressive Description Logics

Dresden

Description Logics

Franz Baader

Theoretical Computer Science

TU Dresden

Germany

Literature:

The Description Logic Handbook

edited by F. Baader, D. Calvanese, D. McGuinness, D. Nardi, P. Patel-Schneider

Cambridge University Press

Knowledge Representation

general goal

"develop formalisms for providing high-level descriptions of the world that can be effectively used to build intelligent applications"

[Brachman & Nardi, 2003]

- formalism: well-defined syntax and formal, unambiguous semantics
- high-level description: only relevant aspects represented, others left out
- intelligent applications: must be able to reason about the knowledge, and infer implicit knowledge from the explicitly represented knowledge
- effectively used: need for practical reasoning tools and efficient implementations

Dresden

Terminological knowledge

formalize the terminology of the application domain:

- define important notions (classes, relations, objects) of the domain
- state constraints on the way these notions can be interpreted
- deduce consequences of definitions and constraints:
  subclass relationships, instance relationships

Example: domain summer school

- classes (concepts) like Person, Lecturer, Course, Student, ...
- relations (roles) like teaches, attends, likes, ...
- objects (individuals) like Franz, Raj, ...
- constraints like: every course must have a student,
  courses are only taught by lecturers, ...

Description Logics

class of logic-based knowledge representation formalisms tailored towards representing terminological knowledge

Descended from semantic networks and frames via the system KL-ONE [Brachman&Schmolze 85]. Emphasis on well-defined basic inference procedures: subsumption and instance problem.

Phase 1:

- implementation of incomplete systems (Back, Classic, Loom)
- based on structural subsumption algorithms

Phase 2:

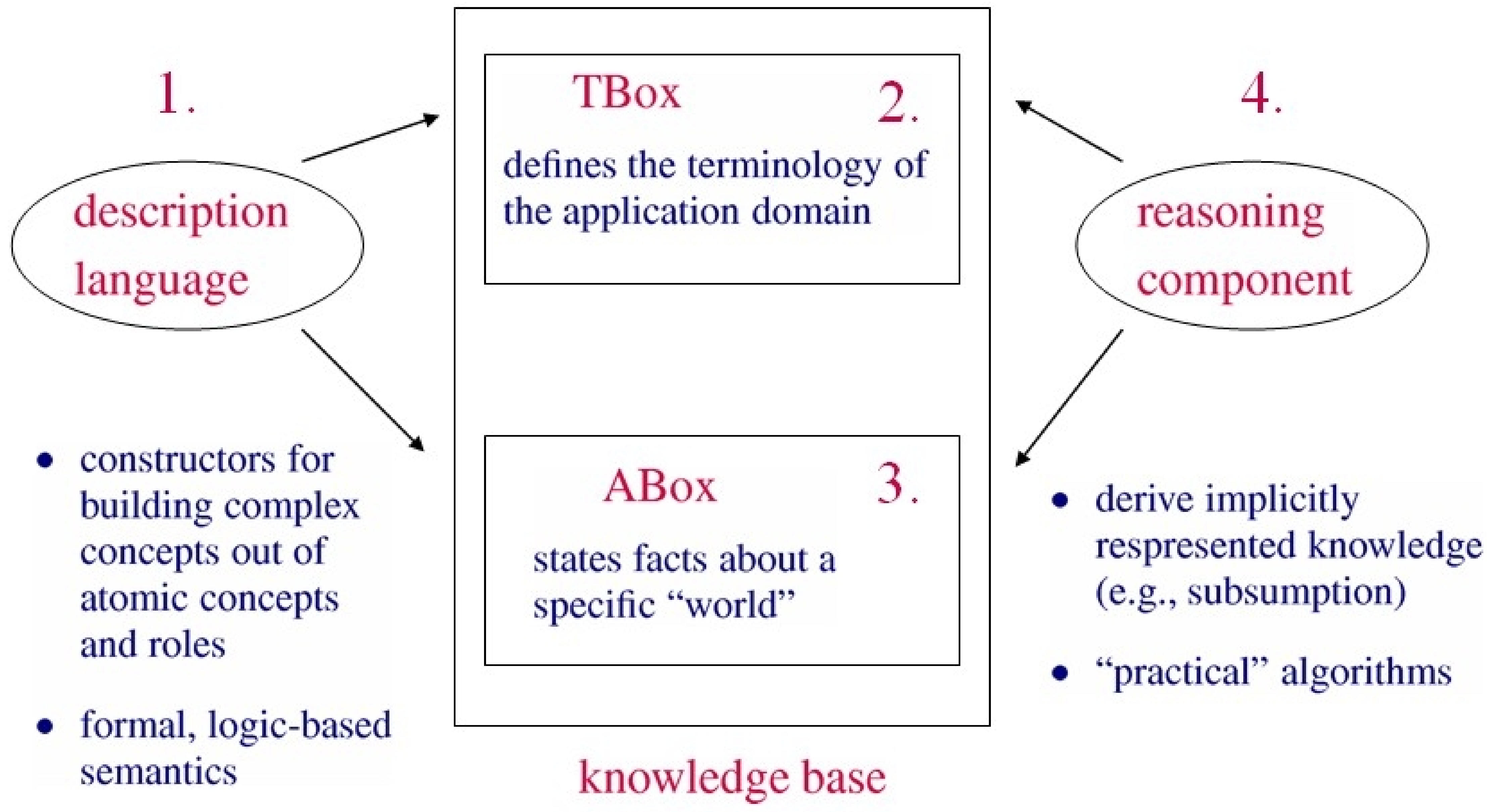
- development of tableau-based algorithms and complexity results
- first implementation of tableau-based systems (Kris, Crack)
- first formal investigation of optimization methods

Phase 3:

- tableau-based algorithms for very expressive DLs
- highly optimized tableau-based systems (FaCT, Racer)
- relationship to modal logic and decidable fragments of FOL

Description logic system

structure



Dresden

The description language

prototypical DL $\mathcal{ALC}$

Set N_C of concept names and disjoint set N_R of role names.

$\mathcal{ALC}$-concept descriptions are defined by induction:

- If $A \in N_C$, then A is an $\mathcal{ALC}$-concept description.
- If C, D are $\mathcal{ALC}$-concept descriptions, and $r \in N_R$, then the following are $\mathcal{ALC}$-concept descriptions:
  - $C \sqcap D$ (conjunction)
  - $C \sqcup D$ (disjunction)
  - $\neg C$ (negation)
  - $\forall r.C$ (value restriction)
  - $\exists r.C$ (existential restriction)

Abbreviations:

- $\top := A \sqcup \neg A$ (top)
- $\bot := A \sqcap \neg A$ (bottom)
- $C \Rightarrow D := \neg C \sqcup D$ (implication)

The description language

examples of $\mathcal{ALC}$-concept descriptions

Person ⊓ Female

Person ⊓ ∃attends.Course

Person ⊓ ∀attends.(Course ⊓ ¬Easy)

Person ⊓ ∃teaches.(Course ⊓ ∀topic.DL)

Person ⊓ ∀teaches.(Course ⊓ ∃topic.(DL ⊔ NMR))

Dresden

The description language

semantics of $\mathcal{ALC}$-concept descriptions

An interpretation $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ consists of a non-empty domain $\Delta^\mathcal{I}$ and an interpretation function $\cdot^\mathcal{I}$:

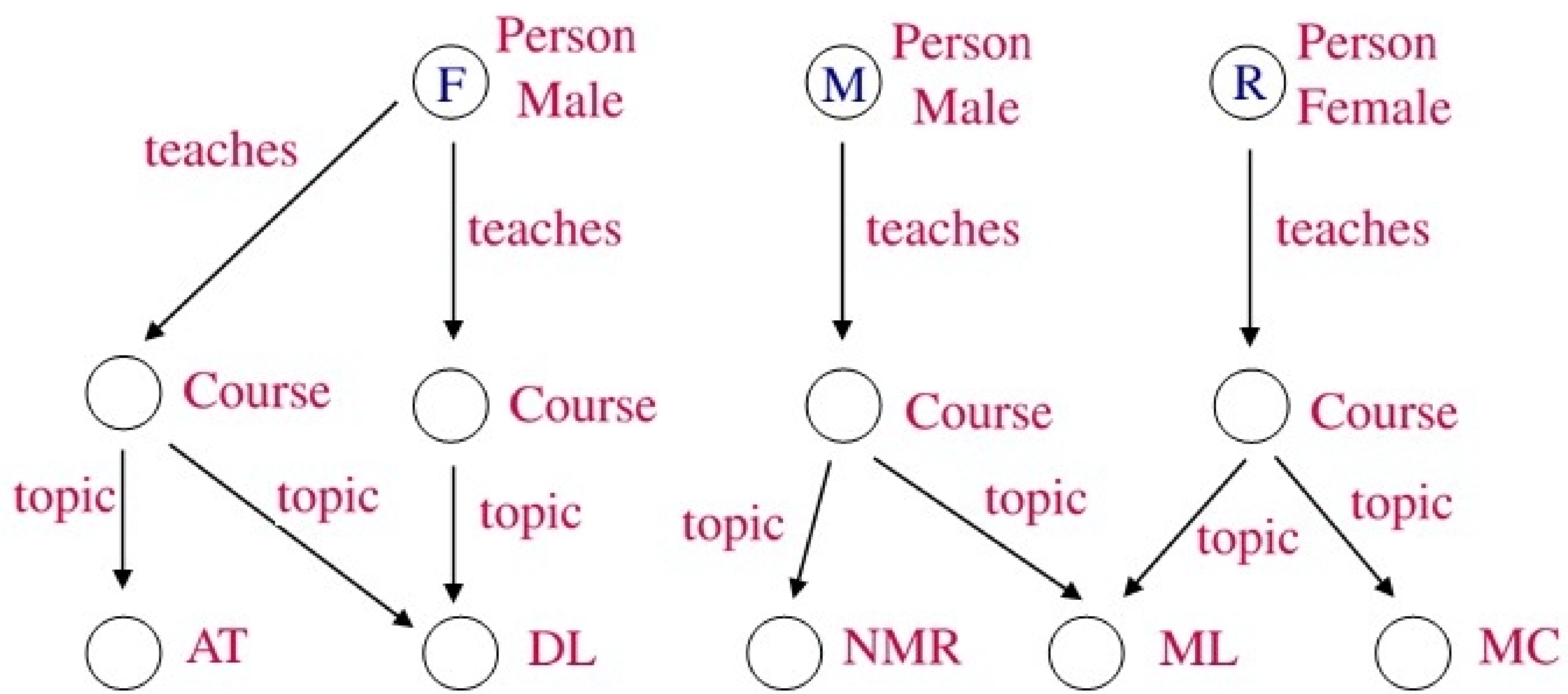
- $A^\mathcal{I} \subseteq \Delta^\mathcal{I}$ for all $A \in N_C$, concepts interpreted as sets
- $r^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$ for all $r \in N_R$. roles interpreted as binary relations

The interpretation function is extended to $\mathcal{ALC}$-concept descriptions as follows:

- $(C \sqcap D)^\mathcal{I} := C^\mathcal{I} \cap D^\mathcal{I}$
- $(C \sqcup D)^\mathcal{I} := C^\mathcal{I} \cup D^\mathcal{I}$
- $(\neg C)^\mathcal{I} := \Delta^\mathcal{I} \setminus C^\mathcal{I}$
- $(\forall r.C)^\mathcal{I} := \{d \in \Delta^\mathcal{I} \mid \text{for all } e \in \Delta^\mathcal{I} : (d,e) \in r^\mathcal{I} \text{ implies } e \in C^\mathcal{I}\}$
- $(\exists r.C)^\mathcal{I} := \{d \in \Delta^\mathcal{I} \mid \text{there is } e \in \Delta^\mathcal{I} : (d,e) \in r^\mathcal{I} \text{ and } e \in C^\mathcal{I}\}$

Example

of an interpretation



$$(\ \text{Person} \sqcap \exists \text{teaches}.(\text{Course} \sqcap \forall \text{topic}.\text{DL})\ )^{\mathcal{I}} = \{F\}$$

$$(\ \text{Person} \sqcap \forall \text{teaches}.(\text{Course} \sqcap \exists \text{topic}.(\text{DL} \sqcup \text{NMR}))\ )^{\mathcal{I}} = \{F, M\}$$

Relationship with First-order Logic

$\mathcal{ALC}$ can be seen as a fragment of first-order logic:

- Concept names are unary predicates, and role names are binary predicates.
- Concept descriptions C yield formulae with one free variable $\tau_x(C)$:
  - $\tau_x(A) := A(x)$ for $A \in N_C$
  - $\tau_x(C \sqcap D) := \tau_x(C) \wedge \tau_x(D)$
  - $\tau_x(C \sqcup D) := \tau_x(C) \vee \tau_x(D)$
  - $\tau_x(\neg C) := \neg \tau_x(C)$
  - $\tau_x(\forall r.C) := \forall y.(r(x,y) \rightarrow \tau_y(C))$
  - $\tau_x(\exists r.C) := \exists y.(r(x,y) \wedge \tau_y(C))$

  y variable different from x

C and $\tau_x(C)$ have the same semantics:

$$C^{\mathcal{I}} = \{d \in \Delta^{\mathcal{I}} \mid \mathcal{I} \models \tau_x(C)[x \leftarrow d]\}$$

Relationship with First-order Logic

$\mathcal{ALC}$ can be seen as a fragment of first-order logic:

- Concept names are unary predicates, and role names are binary predicates.
- Concept descriptions C yield formulae with one free variable $\tau_x(C)$:

These formulae belong to known decidable subclasses of first-order logic:

– two-variable fragment

– guarded fragment

$$
\begin{aligned}
\tau_x(\forall r.(A \sqcap \exists r.B)) &= \forall y.(r(x,y) \rightarrow \tau_y(A \sqcap \exists r.B)) \\
&= \forall y.(r(x,y) \rightarrow (A(y) \wedge \exists z.(r(y,z) \wedge B(z))))
\end{aligned}
$$

Relationship with Modal Logic

multimodal K:

several pairs of boxes and diamonds

$\mathcal{ALC}$ is a syntactic variant of the basic modal logic K:

- Concept names are propositional variables, and role names are names for transition relations.
- Concept descriptions C yield modal formulae $\theta(C)$:
  - $\theta(A) := a$ for $A \in N_C$
  - $\theta(C \sqcap D) := \theta(C) \wedge \theta(D)$
  - $\theta(C \sqcup D) := \theta(C) \vee \theta(D)$
  - $\theta(\neg C) := \neg\theta(C)$
  - $\theta(\forall r.C) := \Box_r\theta(C)$
  - $\theta(\exists r.C) := \Diamond_r\theta(C)$

C and $\theta(C)$ have the same semantics: $C^{\mathcal{I}}$ is the set of worlds that make $\theta(C)$ true in the Kripke structure described by $\mathcal{I}$.

Dresden

Additional constructors

$\mathcal{ALC}$ is only an example of a description logic.

DL researchers have introduced and investigated many additional constructors.

Example

Number restrictions: $(\geq n\, r.C)$, $(\leq n\, r.C)$ with semantics

$$(\geq n\, r.C)^{\mathcal{I}} := \{d \in \Delta^{\mathcal{I}} \mid \mathrm{card}(\{e \mid (d,e) \in r^{\mathcal{I}} \wedge e \in C^{\mathcal{I}}\}) \geq n\}$$

$$(\leq n\, r.C)^{\mathcal{I}} := \{d \in \Delta^{\mathcal{I}} \mid \mathrm{card}(\{e \mid (d,e) \in r^{\mathcal{I}} \wedge e \in C^{\mathcal{I}}\}) \leq n\}$$

Persons that attend at most 3 courses, of which at least 2 have the topic DL:

$$\mathrm{Person} \sqcap (\leq 3\, \mathrm{attends.Course}) \sqcap (\geq 2\, \mathrm{attends.}(\mathrm{Course} \sqcap \exists \mathrm{topic.DL}))$$

Additional constructors

In addition to concept constructors, one can also introduce role constructors.

Example

Inverse roles: if r is a role, then r^{-1} denotes its inverse

$$(r^{-1})^{\mathcal{I}} := \{(e,d) \mid (d,e) \in r^{\mathcal{I}}\}$$

Inverse roles can be used like role names in value and existential restrictions.

Teacher of a boring course:

$$\text{Person} \sqcap \exists \text{teaches}.(\text{Course} \sqcap \forall \text{attends}^{-1}.(\text{Bored} \sqcup \text{Sleeping}))$$

Terminologies

introduce names for complex descriptions

A concept defintion is of the form $A \equiv C$ where

- A is a concept name;
- C is a concept description.

A TBox is a finite set of concept definitions that

- does not contain multiple definitions;
- does not contain cyclic definitions.

$$\begin{array}{lcl} A & \equiv & C \\ A & \equiv & D \end{array} \quad \text{for } C \neq D$$

$$\begin{array}{lcl} A & \equiv & B \sqcap \forall r.P \\ B & \equiv & P \sqcap \forall r.C \\ C & \equiv & \exists r.A \end{array}$$

Defined concept occurs on left-hand side of a definition

Primitive concept does not occur on left-hand side of a definition

Dresden

Terminologies

semantics and example

An interpretation $\mathcal{I}$ is a model of a TBox $\mathcal{T}$ if it satisfies all its concept definitions:

$$A^{\mathcal{I}} = C^{\mathcal{I}} \text{ for all } A \equiv C \in \mathcal{T}$$

$$
\begin{aligned}
\text{Woman} &\equiv \text{Person} \sqcap \text{Female} \\
\text{Man} &\equiv \text{Person} \sqcap \neg\text{Female} \\
\text{Course} &\equiv \exists\text{topic}.\top \\
\text{Lecturer} &\equiv \text{Person} \sqcap \exists\text{teaches}.\text{Course} \\
\text{Student} &\equiv \text{Person} \sqcap \exists\text{attends}.\text{Course} \\
\text{BusyLecturer} &\equiv \text{Lecturer} \sqcap (\geq 3\,\text{teaches}.\text{Course}) \\
\text{BadLecturer} &\equiv \text{Lecturer} \sqcap \forall\text{teaches}.(\forall\text{attends}^{-1}.(\text{Bored} \sqcup \text{Sleeping}))
\end{aligned}
$$

Terminologies

beyond concept definitions

Modern DL systems allow their users to state more general constraints for the interpretation of concepts.

A general concept inclusion axiom (GCI) is of the form

$C \sqsubseteq D$ where C, D may be complex concept descriptions.

general TBox

An interpretation $\mathcal{I}$ is a model of a set of GCIs $\mathcal{T}$ if it satisfies all its concept inclusions:

$$C^{\mathcal{I}} \subseteq D^{\mathcal{I}} \text{ for all } C \sqsubseteq D \in \mathcal{T}$$

$\text{Course} \sqcap \forall \text{attends}^{-1}.\text{Sleeping} \sqsubseteq \text{Boring}$

$\text{Lecturer} \sqcap \text{Student} \sqsubseteq \bot$

Dresden

ABox assertions

state properties of individuals

An assertion is of the form

$C(a)$ (concept assertion) or $r(a, b)$ (role assertion)

where C is a concept description, r is a role, and a, b are individual names from a set N_I of such names.

An ABox is a finite set of assertions.

An interpretation $\mathcal{I}$ is a model of an ABox $\mathcal{A}$ if it satisfies all its assertions:

$$a^{\mathcal{I}} \in C^{\mathcal{I}} \quad \text{for all } C(a) \in \mathcal{A}$$
$$(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}} \quad \text{for all } r(a, b) \in \mathcal{A}$$

$\mathcal{I}$ assigns elements of $\Delta^{\mathcal{I}}$ to individual names

Lecturer(FRANZ),	teaches(FRANZ, C1),
Course(C1),	topic(C1, T1),
DL(T1)	

Dresden

Reasoning

makes implicitly represented knowledge explicit, provided as service by the DL system, e.g.:

Subsumption: Is C a subconcept of D?

$C \sqsubseteq_{\mathcal{T}} D$ iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for all models $\mathcal{I}$ of the TBox $\mathcal{T}$.

Satisfiability: Is the concept C non-contradictory?

C is satisfiable w.r.t. $\mathcal{T}$ iff $C^{\mathcal{I}} \neq \emptyset$ for some model $\mathcal{I}$ of $\mathcal{T}$.

Consistency: Is the ABox $\mathcal{A}$ non-contradictory?

$\mathcal{A}$ is consistent w.r.t. $\mathcal{T}$ iff it has a model that is also a model of $\mathcal{T}$.

Instantiation: Is e an instance of C?

$\mathcal{A} \models_{\mathcal{T}} C(e)$ iff $e^{\mathcal{I}} \in C^{\mathcal{I}}$ for all models $\mathcal{I}$ of $\mathcal{T}$ and $\mathcal{A}$.

polynomial reductions

in presence of negation

Dresden

Reductions

between inference problems

Subsumption to satisfiability:

$$C \sqsubseteq_{\mathcal{T}} D \text{ iff } C \sqcap \neg D \text{ is unsatisfiable w.r.t. } \mathcal{T}$$

Satisfiability to subsumption:

$$C \text{ is satisfiable w.r.t. } \mathcal{T} \text{ iff not } C \sqsubseteq_{\mathcal{T}} \bot$$

Satisfiability to consistency:

$$C \text{ is satisfiable w.r.t. } \mathcal{T} \text{ iff } \{C(a)\} \text{ is consistent w.r.t. } \mathcal{T}$$

Instance to consistency:

a is an instance of C w.r.t. $\mathcal{T}$ and $\mathcal{A}$ iff $\mathcal{A} \cup \{\neg C(a)\}$ is inconsistent w.r.t. $\mathcal{T}$

Consistency to instance :

$$\mathcal{A} \text{ is consistent w.r.t. } \mathcal{T} \text{ iff } a \text{ is not an instance of } \bot \text{ w.r.t. } \mathcal{T} \text{ and } \mathcal{A}$$

Reduction

getting rid of the TBox

Expansion of concepts:

For a given TBox $\mathcal{T}$ and concept description C, the expansion $C^{\mathcal{T}}$ of C w.r.t. $\mathcal{T}$ is obtained from C by

- replacing defined concepts by their definitions
- until no more defined concepts occur.

$$\mathcal{T} \quad \boxed{\begin{array}{rcl} \text{Woman} & \equiv & \text{Person} \sqcap \text{Female} \\ \text{Course} & \equiv & \exists \text{topic}.\top \\ \text{Lecturer} & \equiv & \text{Person} \sqcap \exists \text{teaches}.\text{Course} \end{array}}$$

Woman $\sqcap$ Lecturer expands to

$$\text{Person} \sqcap \text{Female} \sqcap \text{Person} \sqcap \exists \text{teaches}.(\exists \text{topic}.\top)$$

Dresden

Reduction

getting rid of the TBox

Since TBoxes are acyclic, expansion always terminates,

but the expanded concept may be exponential in the size of $\mathcal{T}$.

$$\begin{array}{rcl} A_0 & \equiv & \forall r.A_1 \sqcap \forall s.A_1 \\ A_1 & \equiv & \forall r.A_2 \sqcap \forall s.A_2 \\ & \vdots & \\ A_{n-1} & \equiv & \forall r.A_n \sqcap \forall s.A_n \end{array}$$

The size of $\mathcal{T}$ is linear in n,

but the expansion $A_0^{\mathcal{T}}$ contains A_n 2^n times.

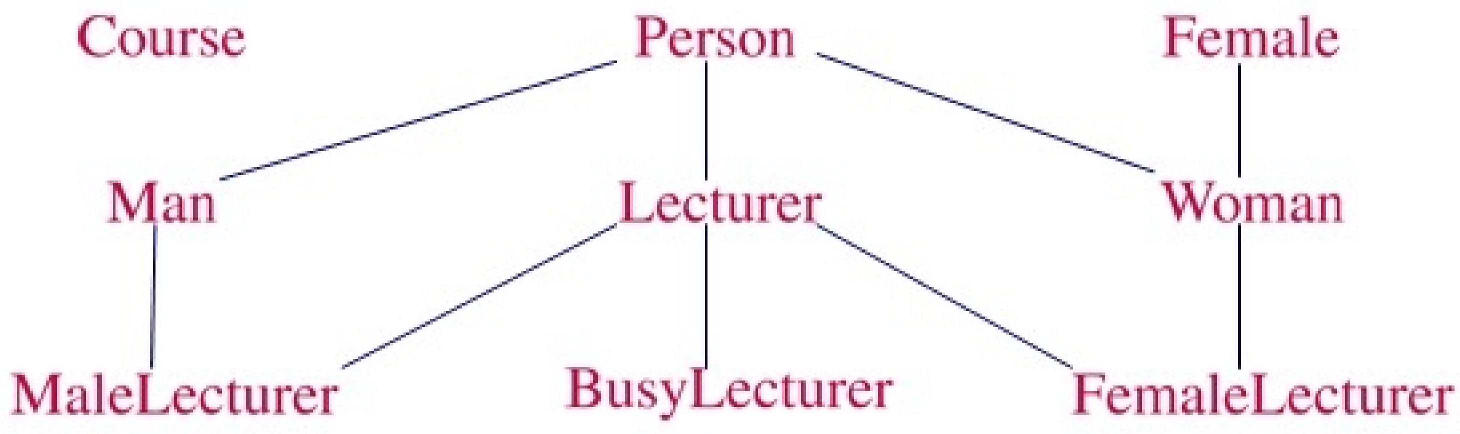
Reductions:

- C is satisfiable w.r.t. $\mathcal{T}$ iff $C^{\mathcal{T}}$ is satisfiable w.r.t. the empty TBox $\emptyset$.
- $C \sqsubseteq_{\mathcal{T}} D$ iff $C^{\mathcal{T}} \sqsubseteq_{\emptyset} D^{\mathcal{T}}$.
- Consistency and the instance problem can be treated similarly.

Classification

Computing the subsumption hierarchy of all concept names occurring in the TBox.

$$
\begin{aligned}
\text{Man} &\equiv \text{Person} \sqcap \neg\text{Female}\\
\text{Woman} &\equiv \text{Person} \sqcap \text{Female}\\
\text{MaleLecturer} &\equiv \text{Man} \sqcap \exists\text{teaches.Course}\\
\text{FemaleLecturer} &\equiv \text{Woman} \sqcap \exists\text{teaches.Course}\\
\text{Lecturer} &\equiv \text{FemaleLecturer} \sqcup \text{MaleLecturer}\\
\text{BusyLecturer} &\equiv \text{Lecturer} \sqcap (\geq 3\,\text{teaches.Course})
\end{aligned}
$$



Realization

Computing the most specific concept names in the TBox to which an ABox individual belongs.

$$
\begin{aligned}
\text{Man} &\equiv \text{Person} \sqcap \neg\text{Female} \\
\text{Woman} &\equiv \text{Person} \sqcap \text{Female} \\
\text{MaleLecturer} &\equiv \text{Man} \sqcap \exists\text{teaches.Course} \\
\text{FemaleLecturer} &\equiv \text{Woman} \sqcap \exists\text{teaches.Course} \\
\text{Lecturer} &\equiv \text{FemaleLecturer} \sqcup \text{MaleLecturer} \\
\text{BusyLecturer} &\equiv \text{Lecturer} \sqcap (\geq 3\,\text{teaches.Course})
\end{aligned}
$$

Man(FRANZ), teaches(FRANZ, C1), Course(C1)

FRANZ is an instance of Man, Lecturer, <u>MaleLecturer</u>.

most specific

Dresden