Description Logics

– an introduction into its basic ideas –

A. Heußner

WS 2003/2004

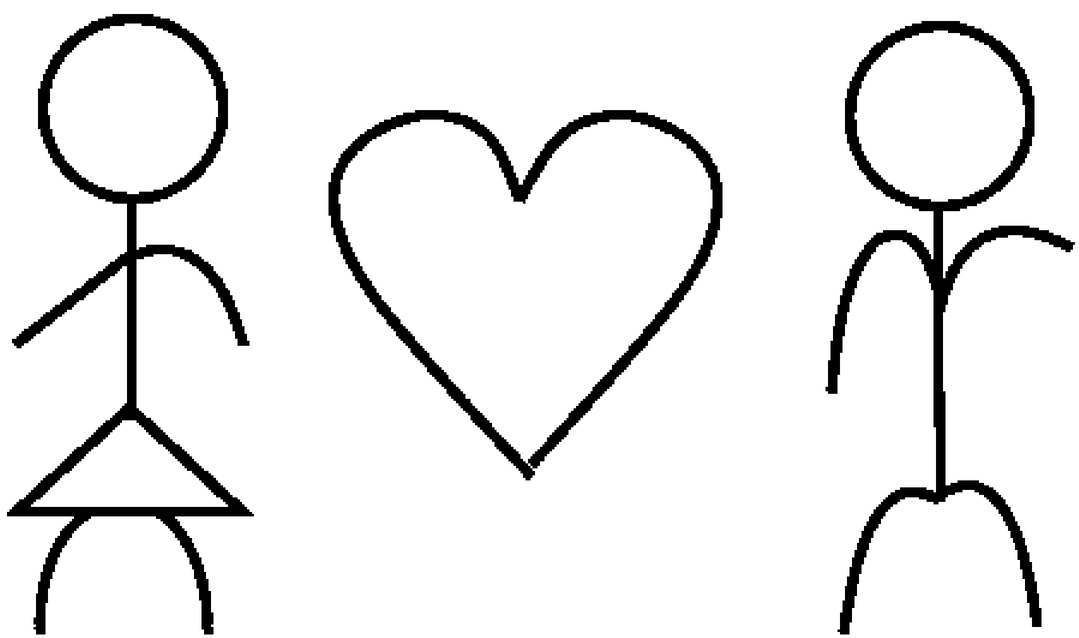
Preview:

- Basic Idea:
  from Network Based Structures to $\mathcal{DL}$
- $\mathcal{AL}$: Syntax / Semantics
- Enhancements of $\mathcal{AL}$
- Terminologies (TBox)
- Assertions (ABox)
- Inferences
- Rules and other Language Extensions
- from $\mathcal{SHIQ}$ to Semantic Web. . .

Knowledge Modeling: "simple" Example

Try to model a very simple sentence:

`Calypso loves Ulysses`



We “said” too much !!!

- ↯ Who said, Calypso is a woman and Ulysses a man?
- ↯ Why is Calypso wearing a skirt?
- ↯ What has a heart to do with love?

What we know is that:

- there is something called Calypso
- there is something called Ulysses
- between them there is some thing called Love going on

another “simple” Example:

`Grandparents love Children`

! relation between two sets
not just about two individuals (`Calypso`, `Ulysses`,. . .)

▷ in term of sets:
Grandparents is a subset of the set of things who love Children

? How do we construct, given the set of children, the set of all those who love them?

? What is meant by “those who love children”?
Do they love only one child or all children or a subset. . . ?

Def. think of concepts: $\exists$ `Love.Children` $\forall$ `Love.Children`

Definitions

- back to our example: $\texttt{Grandparent} \sqsubseteq \exists\, \texttt{Love.Children}$
- if we can express inclusion, we can also express equivalence, so we can write definitions:

$$\begin{aligned}\texttt{Grandparent} \;\doteq\; & \texttt{Human} \sqcap \\ & \exists\texttt{HasChildren}.\exists\texttt{HasChildren.Human} \sqcap \\ & \exists\texttt{Love.Children}\end{aligned}$$

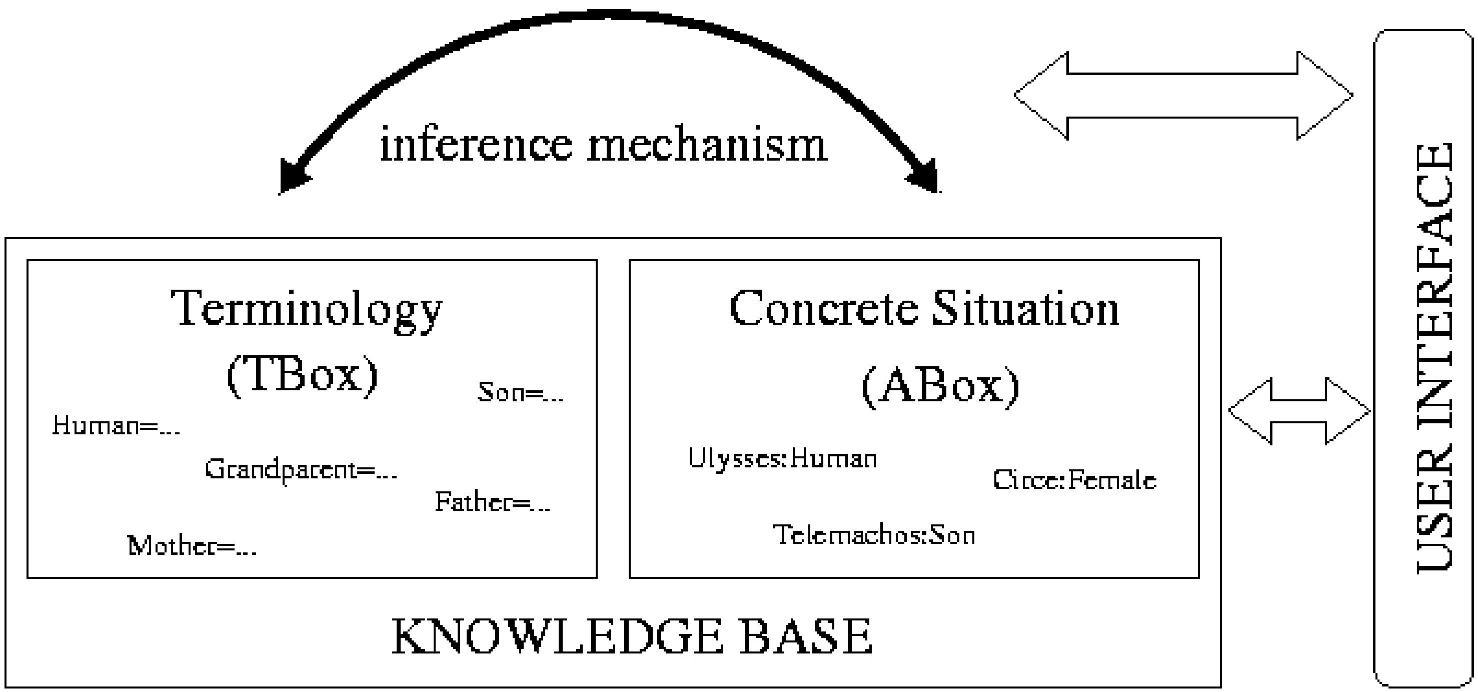
- a set of definitions forms a terminology (TBox)
- express facts (what is really going on) with assertions (ABox):

$$\texttt{Ulysses}:\texttt{Human} \qquad \texttt{Ulysses}:\neg\texttt{Grandparent}$$

$\mathcal{DL}$ - System

Description Logics ($\mathcal{DL}$) is the most recent name for a family of knowledge representation (KR) formalisms that represent the knowledge of an application domain (the world) by first defining the relevant concepts of the domain (its terminology), and then using these concepts to specify properties of objects and individuals occurring in the domain (the world description).

[Baader/Nutt]



$\mathcal{AL}$ – a simple $\mathcal{DL}$

- atomic concepts, roles (binary relations of concepts)

▷ add concept constructors ($\rightarrow$ different languages !)

- syntax:
  let A be an atomic concept, C, D concepts and R a role

$$
\begin{array}{lccl}
C, D \rightarrow & A & \mid & (\text{ atomic concept }) \\
 & \top & \mid & (\text{ universal concept }) \\
 & \bot & \mid & (\text{ bottom concept }) \\
 & \neg A & \mid & (\text{ atomic(!) negation }) \\
 & C \sqcap D & \mid & (\text{ intersection }) \\
 & \forall R.C & \mid & (\text{ value restriction }) \\
 & \exists R.\top & & (\text{ limited existential quantification })
\end{array}
$$

- semantics:
  let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an interpretation

constructor	syntax	semantics (set-theoretic)
atomic concept	A	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
atomic role	R	$R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
universal concept	$\top$	$\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$
bottom concept	$\bot$	$\bot^{\mathcal{I}} = \emptyset$
atomic negation	$\neg A$	$(\neg A)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus A^{\mathcal{I}}$
intersection	$C \sqcap D$	$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$
value restr.	$\forall R.C$	$\{a \in \Delta^{\mathcal{I}} \mid \forall b.(a,b) \in R^{\mathcal{I}} \rightarrow b \in C^{\mathcal{I}}\}$
lim. ex. quant.	$\exists R.\top$	$\{a \in \Delta^{\mathcal{I}} \mid \exists b.(a,b) \in R^{\mathcal{I}}\}$
union $(\mathcal{U})$	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
full $\exists$ $(\mathcal{E})$	$\exists R.C$	$\{a \in \Delta^{\mathcal{I}} \mid \exists b.(a,b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\}$
number restr. $(\mathcal{N})$	$(\leq nR)$	$\{a \in \Delta^{\mathcal{I}} \mid \, \lvert\{b \mid (a,b) \in R^{\mathcal{I}}\}\rvert \leq n\}$
negation $(\mathcal{C})$	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$

- family of $\mathcal{AL}$-languages:

$$\mathcal{AL}[\mathcal{U}][\mathcal{E}][\mathcal{N}][\mathcal{C}]$$

- remember: $C \sqcup D \equiv \neg(\neg C \sqcap \neg D)$ and $\exists\, R.C \equiv \neg\forall\, R.\neg C$

▷ therefore: $\mathcal{AL} * \mathcal{N} * \equiv \mathcal{ALUE} *$

. . .

connection to FOL

concept names A	$\Leftrightarrow$	unary predicates P_A
concept roles R	$\Leftrightarrow$	binary predicates P_R
concepts	$\Leftrightarrow$	formulae with one free variable

$$
\begin{aligned}
\phi^x(A) &= P_A(x) \\
\phi^x(\neg C) &= \neg\phi^x(C) \\
\phi^x(C \sqcup D) &= \phi^x(C) \vee \phi^x(D) \\
\phi^x(C \sqcap D) &= \phi^x(C) \wedge \phi^x(D) \\
\phi^x(\exists R.C) &= \exists y.P_R(x,y) \wedge \phi^y(C) \\
\phi^x(\forall R.C) &= \forall y.P_r(x,y) \rightarrow \phi^y(C)
\end{aligned}
$$

ϕ^y is symmetric with x and y exchanged

! two variables suffices (no "=", no constants, no fct. symbols)

! not all $\mathcal{DL}$ are purely first order (transitive closure, etc.)

! $\mathcal{ALC}$ is decidable fragment of FOL with 2 variables ($\mathcal{L}$)

Terminologies – TBox

- concept definitions:
  $A \doteq C$ (A is individual name, C is cmplx. concept)
  $\leadsto$ introduce macros/names for concepts
  $\leadsto$ can be (a)cyclic (need fixpoint semantics)

- axioms: $C \sqsubseteq D$ (C, D cmplx. concept)
  $\leadsto$ restrict models

Def. an interpretation $\mathcal{I}$ satisfies. . .

a concept definition	$A \doteq C$	iff	$A^{\mathcal{I}} = C^{\mathcal{I}}$
an axiom	$C \sqsubseteq D$	iff	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
a TBox	$\mathcal{T}$	iff	$\mathcal{I}$ satisfies all definitions and axioms in $\mathcal{T}$
		$\longrightarrow$	$\mathcal{I}$ is a model of $\mathcal{T}$

Assertions – ABox

- concept assertions: $a : C$ (a is individual name)
- role assertion: $< a_1, a_2 >: R$

Def. an interpretation $\mathcal{I}$ satisfies (with respect to $\mathcal{T}$)...

a concept assertion	$a : C$	iff	$a^{\mathcal{I}} \in C^{\mathcal{I}}$
a role assertion $< a_1, a_2 >:$	R	iff	$< a^{\mathcal{I}}, b^{\mathcal{I}} > \in R^{\mathcal{I}}$
a ABox	$\mathcal{A}$	iff	$\mathcal{I}$ satisfies all assertions in $\mathcal{A}$
		$\longrightarrow$	$\mathcal{I}$ is a model of $\mathcal{A}$

!!! Open-World-Assumption

▷ model of $\mathcal{A}$ and $\mathcal{T}$ is an abstraction of a concrete world

Tasks of Inference

in TBox:

- satisfiability:
  a concept C is satisfiable with respect to $\mathcal{T}$ iff there exists a model $\mathcal{I}$ of $\mathcal{T}$ such that $C^{\mathcal{I}} \neq \emptyset$

- subsumption: $C \sqsubseteq_{\mathcal{T}} D$ (or $\mathcal{T} \models C \sqsubseteq D$) iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$

- equivalence: $C \doteq_{\mathcal{T}} D$ iff $C^{\mathcal{I}} = D^{\mathcal{I}}$ (for every $\mathcal{I}$ of $\mathcal{T}$)

- disjointness: $C \sqcap_{\mathcal{T}} D \neq \emptyset$ iff $C^{\mathcal{I}} \cap D^{\mathcal{I}} = \emptyset$

in ABox:

- consistency of ABox $\mathcal{A}$ with $\mathcal{T}$:
  there exists interpretation $\mathcal{I}$ which is a model of both $\mathcal{A}$ and $\mathcal{T}$

- instantiation of assertion α by $\mathcal{A}$:
  $\mathcal{A} \models \alpha$ iff every interpret. $\mathcal{I}$ which satisfies $\mathcal{A}$ also satisfies α

- retrieval / realization:
  find mostspecific concept for given individuals / find individual to given concepts

use (logical) Deduction:

usage: make implicit knowledge explicit !

Prop. all above tasks can be reduced to subsumption / satisfiability

- ▷ all above tasks decidable in $\mathcal{DL}$ (without proof ;-)
- ▷ need only one reasoning algorithm
- ▷ use 'standard' reasoning algorithms
  (highly improved tableaux-algorithms like [Fact], [Racer],. . .)

Rules

- extend KnowledgeBase $(\mathcal{T},\mathcal{A})$ with rules:
  $C \Rightarrow D$ iff every instance individual of C is instance of D

- add trigger rules to Knowledge Base

▷ extend semantics. . .

! attention: $C \Rightarrow D$ not equals $\neg D \Rightarrow C$

Language Extensions

- use usual binary relation operators as role constructors ($R \sqcap S$, $\neg R$, $R \circ S$, R^+ (closure), R^- (inverse),. . .)
- expressive number restriction:
  - ▷ qualified restriction: $\leq 2\,\texttt{hasChild.Male}$
  - ▷ number variables:
    $\texttt{Person} \sqcap \leq \alpha\,\texttt{hasChild.Male} \geq \alpha\,\texttt{hasChild.Female}$
- set-/ "one-of"-relation:
  set of individual names $\{a_1, ..., a_n\}$ with $\{a_1...\}^{\mathcal{I}} = \{a_1^{\mathcal{I}}...\}$
  - ▷ introduce individual names (nominals) also in $\mathcal{DL}$
  - ▷ can describe finite sets: $\{a_1\}$ is singleton, $\{a_1\} \sqcup \{a_2\} ... \sqcup \{a_n\}$

↯ possible loss of properties like finite-models, decidability,. . .

$\mathcal{DL}$ and Ontologies

- use $\mathcal{DL}$ to build up ontologies from "bottom-up"

▷ instead of manually create a hierarchy and then assign properties to the concepts:

  ▷ first assign to each concept a logic definition
  ▷ which is then used to infer a classification

▷ advantages:

  ▷ hierarchy of ontology evolves with added concepts
  ▷ algorithm for classification is proved to be correct

from $\mathcal{DL}$ to Semantic Web

- web ontology languages OIL and OWL are based on $\mathcal{DL}$:

$$\begin{array}{llcl} & \mathcal{S} & = & \mathcal{ALC} \text{ with transitive roles} \\ \hookrightarrow & \mathcal{SH} & = & \mathcal{S} \text{ with role } \mathcal{H}\text{ierarchies} \\ \hookrightarrow & \mathcal{SHI} & = & \mathcal{SH} \text{ with } \mathcal{I}\text{nverse roles } (R^-) \\ \hookrightarrow & \mathcal{SHIQ} & = & \mathcal{SHI} \text{ with number restriction} \end{array}$$

▷ semantic foundation of ontology via semantics of $\mathcal{SHIQ}$

References

[Areces] Carlos Areces: *"Ontologies and Description Logics"*, presentation from May 2003, (link to PDF-version)

[Baader/Nutt] Franz Baader, Werner Nutt: *"Basic Description Logics"* in Baader(ed.) et.al. "Description Logics Handbook", Cambridge 2002

[Brachman/Nardi] Danielle Nardi, Ronald J. Brachman: *"An Introduction to Description Logics"* in Baader(ed.) et.al. "Description Logics Handbook", Cambridge 2002

[Description Logics] Description Logics Website at http://dl.kr.org/

[weblinks tested: 07.12.2003]

Thanks to. . .

. . . Carlos Areces
whose eidetic introduction this presentation tried to track
. . . the $\mathcal{DL}$-community at Manchester, especially Ulrike Sattler, from whom I "lent" some foils
(to be honest, this presentation is a melange of the ones above)

this presentation was designed and made with the help of vim, (pdf)latex, P^4, xfig, and gv under OpenBSD; many thanks to the "community" for the possibility to use FREE Software