Motivation

- Main problem with semantic networks and frames ... the lack of formal semantics!
- Disadvantage of simple inheritance networks ... concepts are atomic and do not have any structure
- Brachman's structural inheritance networks (1977)

Structural inheritance networks

- Concepts are defined/described using a small set of well-defined operators
- Distinction between conceptual and object-related knowledge
- Computation of subconcept relation and of instance relation
- Strict inheritance (of the entire structure of a concept): inherited properties cannot be overridden
**Systems and applications**

- **Systems:**
  - KL-ONE: First implementation of the ideas (1978)
  - then: NIKL, KL-TWO, KRYPTON, KANDOR, CLASSIC, BACK, KRIS, YAK, CRACK ...
  - later: FaCT, DLP, RACER 1998
  - currently: FaCT++, RACER, Pellet, HermiT, and many more ...

- **Applications:**
  - First, natural language understanding systems,
  - then configuration systems,
  - and information systems,
  - currently, it is one tool for the Semantic Web

- **Languages:** DAML+OIL, now OWL (Web Ontology Language)

---

**Description logics**

- Previously also known as KL-ONE-alike languages, frame-based languages, terminological logics, concept languages

**Description Logics (DL) allow us**

- to describe concepts using complex descriptions,
- to introduce the terminology of an application and to structure it (TBox),
- to introduce objects and relate them to the introduced terminology (ABox),
- and to reason about the terminology and the objects.

---

**Informal example**

<table>
<thead>
<tr>
<th>Male is:</th>
<th>the opposite of female</th>
</tr>
</thead>
<tbody>
<tr>
<td>A human is a kind of:</td>
<td>living entity</td>
</tr>
<tr>
<td>A woman is:</td>
<td>a human and a female</td>
</tr>
<tr>
<td>A man is:</td>
<td>a human and a male</td>
</tr>
<tr>
<td>A mother is:</td>
<td>a woman with at least one child that is a human</td>
</tr>
<tr>
<td>A father is:</td>
<td>a man with at least one child that is a human</td>
</tr>
<tr>
<td>A parent is:</td>
<td>a mother or a father</td>
</tr>
<tr>
<td>A grandmother is:</td>
<td>a woman, with at least one child that is a parent</td>
</tr>
<tr>
<td>A mother-wod is:</td>
<td>a mother with only male children</td>
</tr>
</tbody>
</table>

Elizabeth is a woman
Elizabeth has the child
Charles
Charles is a man
Charles is a grandmother a parent?
Diana is a mother-wod
Diana is a mother-wod
Diana has the child William
Is Elizabeth a mother-wod?

---

**2 Concepts and Roles**

- **Concept Forming Operators**
- **Role Forming Operators**
Atomic concepts and roles

- **Concept names:**
  - E.g., Grandmother, Male,... (in the following usually capitalized)
  - We will use symbols such as A, A1,... for concept names
  - Semantics: Monadic predicates A(·) or set-theoretically a subset of the universe $A \subseteq D$.

- **Role names:**
  - In our example, e.g., child. Often we will use names such as has-child or something similar (in the following usually lowercase).
  - Role names are disjoint from concept names
  - Symbolically: $t_1, ...$
  - Semantics: Binary relations $t(·, ·)$ or set-theoretically $t^I \subseteq D \times D$.

Boolean operators

- **Syntax:** let $C$ and $D$ be concept descriptions, then the following are also concept descriptions:
  - $C \land D$ (concept conjunction)
  - $C \lor D$ (concept disjunction)
  - $\neg C$ (concept negation)

- **Examples:**
  - Human $\sqcap$ Female
  - Father $\sqcup$ Mother
  - $\neg$ Female

- **FOL semantics:**
  - $C(x)$ and $D(x)$ as $C \land D(x)$, $C \lor D(x)$, $\neg C(x)$
  - **Set semantics:** $C^I \cap D^I$, $C^I \cup D^I$, $D \setminus C^I$

Concept and role description

- **From (atomic) concept and role names, complex concept and role descriptions can be created**
  - In our example, e.g., "Human and Female."
  - Symbolically: $C$ for concept descriptions and $r$ for role descriptions

Which particular constructs are available depends on the chosen description logic!

- **FOL semantics:** A concept description $C$ corresponds to a formula $C(x)$ with the free variable $x$.
  - Similarly with role descriptions $r$: they correspond to formulae $r(x, y)$ with free variables $x, y$.

- **Set semantics:**
  - $C^I = \{d \in D : C(d) \text{ "is true in" } I\}$
  - $r^I = \{(d, e) \in D^2 : r(d, e) \text{ "is true in" } I\}$

Role restrictions

- **Motivation:**
  - Often we want to describe something by restricting the possible "fillers" of a role, e.g. Mother–wod.
  - Sometimes we want to say that there is at least a filler of a particular type, e.g. Grandmother

- **Idea:** Use quantifiers that range over the role-fillers
  - $\exists r.C$ has-child $\exists r.C$ has-child $\exists r.C$ has-child
  - $\forall r.C$ has-child $\forall r.C$ has-child $\forall r.C$ has-child

- **FOL semantics:**
  - $(\exists r.C)(x) = \exists y(r(x, y) \land C(y))$
  - $(\forall r.C)(x) = \forall y(r(x, y) \rightarrow C(y))$

- **Set semantics:**
  - $(\exists r.C)^I = \{d \in D : \text{ there ex. some } e \text{ s.t. } (d, e) \in r^I \land e \in C^I\}$
  - $(\forall r.C)^I = \{d \in D : \text{ for each } e \text{ with } (d, e) \in r^I, e \in C^I\}$
Cardinality restriction

Motivation:
Often we want to describe something by restricting the number of possible “fillers” of a role, e.g., a Mother with at least 3 children or at most 2 children.

Idea: We restrict the cardinality of the role filler sets:
- Mother ⊓≥3 has-child
- Mother ⊓≤2 has-child

FOL semantics:
\[
(\geq n \ r)(x) = \exists y_1 \ldots y_n (r(x, y_1) \land \cdots \land r(x, y_n) \land \\
\neg y_1 = y_2 \land \cdots \land y_{n-1} \neq y_n)
\]
\[
(\leq n \ r)(x) = \neg(\geq n+1 \ r)(x)
\]

Set semantics:
\[
(\geq n \ r)^I = \{ d \in D : \{ e \in D : r^I(d, e) \} \geq n \}
\]
\[
(\leq n \ r)^I = D \setminus (\geq n+1 \ r)^I
\]


Inverse roles

Motivation:
How can we describe the concept “children of rich parents”?

Idea: Define the “inverse” role for a given role (the converse relation)

FOL semantics:
\[
r^{-1}(y, x) = r(x, y)
\]

Set semantics:
\[
(r^{-1})^I = \{ (d, e) \in D^2 : (e, d) \in r^I \}
\]

Role composition

Motivation:
How can we define the role has-grandchild given the role has-child?

Idea: Compose roles (as one can compose binary relations)

FOL semantics:
\[
(r \circ s)(x, y) = \exists z(r(x, z) \land s(z, y))
\]

Set semantics:
\[
(r \circ s)^I = \{ (d, e) \in D^2 : \exists f \text{ s.t. } (d, f) \in r^I \land (f, e) \in s^I \}
\]

Role value maps

Motivation:
How do we express the concept “women who know all the friends of their children”

Idea: Relate role filler sets to each other

FOL semantics:
\[
(r \sqsubseteq s)(x) = \forall y (r(x, y) \rightarrow s(x, y))
\]

Set semantics: Let \( r^I(d) = \{ e : r^I(d, e) \} \).
\[
(r \sqsubseteq s)^I = \{ d \in D : r^I(d) \subseteq s^I(d) \}
\]

Note: Role value maps lead to undecidability of satisfiability testing of concept descriptions!
3 TBox and ABox

- Terminology Box
- Assertional Box
- Example

**Terminology box**

- In order to introduce new terms, we use two kinds of terminological axioms:
  - \( A \equiv C \)
  - \( A \sqsubseteq C \)

  where \( A \) is a concept name and \( C \) is a concept description.

- A terminology or TBox is a finite set of such axioms with the following additional restrictions:
  - no multiple definitions of the same symbol such as \( A \equiv C \), \( A \sqsubseteq D \)
  - no cyclic definitions (even not indirectly), such as \( A \equiv \forall r \cdot B \), \( B \equiv \exists s \cdot A \)

**TBoxes: semantics**

- TBoxes restrict the set of possible interpretations.
- **FOL semantics:**
  - \( A \equiv C \) corresponds to \( \forall x \left( A(x) \leftrightarrow C(x) \right) \)
  - \( A \sqsubseteq C \) corresponds to \( \forall x \left( A(x) \rightarrow C(x) \right) \)
- **Set semantics:**
  - \( A \equiv C \) corresponds to \( A^I = C^I \)
  - \( A \sqsubseteq C \) corresponds to \( A^I \subseteq C^I \)

  Non-empty interpretations which satisfy all terminological axioms are called models of the TBox.

**Assertional box**

- In order to state something about objects in the world, we use two forms of assertions:
  - \( a : C \)
  - \( (a, b) : r \)

  where \( a \) and \( b \) are individual names (e.g., ELIZABETH, PHILIP), \( C \) is a concept description, and \( r \) is a role description.

- An ABox is a finite set of assertions.
ABoxes: semantics

- **Individual names** are interpreted as elements of the universe under the unique-name-assumption, i.e., different names refer to different objects.
- **Assertions** express that an object is an instance of a concept or that two objects are related by a role.
- **FOL semantics:**
  - $a : C$ corresponds to $C(a)$
  - $(a, b) : r$ corresponds to $r(a, b)$
- **Set semantics:**
  - $a^T \in D$
  - $a : C$ corresponds to $a^T \in C^T$
  - $(a, b) : r$ corresponds to $(a^T, b^T) \in r^T$
- **Models** of an ABox and of ABox + TBox can be defined analogously to models of a TBox.

---

Example TBox

<table>
<thead>
<tr>
<th>Concept</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male</td>
<td>$\neg$Female</td>
</tr>
<tr>
<td>Human</td>
<td>$\sqsubseteq$ Living_entity</td>
</tr>
<tr>
<td>Woman</td>
<td>$\sqsubseteq$ Human $\sqcap$ Female</td>
</tr>
<tr>
<td>Man</td>
<td>$\sqsubseteq$ Human $\sqcap$ Male</td>
</tr>
<tr>
<td>Mother</td>
<td>$\sqsubseteq$ Woman $\sqcap$ has-child.Human</td>
</tr>
<tr>
<td>Father</td>
<td>$\sqsubseteq$ Man $\sqcap$ has-child.Human</td>
</tr>
<tr>
<td>Parent</td>
<td>$\sqsubseteq$ $\sqcap$ Father $\sqcap$ Mother</td>
</tr>
<tr>
<td>Grandmother</td>
<td>$\sqsubseteq$ Woman $\sqcap$ has-child.Parent</td>
</tr>
<tr>
<td>Mother-without-daughter</td>
<td>$\sqsubseteq$ Mother $\sqcap$ has-child.Male</td>
</tr>
<tr>
<td>Mother-with-many-children</td>
<td>$\sqsubseteq$ Mother $\sqcap$ $\geq$ 3 has-child</td>
</tr>
</tbody>
</table>

---

Example ABox

<table>
<thead>
<tr>
<th>Name</th>
<th>Gender</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHARLES</td>
<td>Man</td>
</tr>
<tr>
<td>DIANA</td>
<td>Woman</td>
</tr>
<tr>
<td>EDWARD</td>
<td>Man</td>
</tr>
<tr>
<td>ELIZABETH</td>
<td>Woman</td>
</tr>
<tr>
<td>ANDREW</td>
<td>Man</td>
</tr>
<tr>
<td>DIANA</td>
<td>Mother-without-daughter</td>
</tr>
<tr>
<td>ELIZABETH</td>
<td>has-child</td>
</tr>
<tr>
<td>EDWARD</td>
<td>has-child</td>
</tr>
<tr>
<td>ANDREW</td>
<td>has-child</td>
</tr>
<tr>
<td>WILLIAM</td>
<td>has-child</td>
</tr>
<tr>
<td>WILLIAM</td>
<td>has-child</td>
</tr>
</tbody>
</table>

---

4 Reasoning Services
Some reasoning services

- Does a description \( C \) make sense at all, i.e., is it satisfiable? A concept description \( C \) is satisfiable, if there exists an interpretation \( \mathcal{I} \) such that \( C^\mathcal{I} \neq \emptyset \).
- Is one concept a specialization of another one, is it subsumed? \( C \) is subsumed by \( D \) (in symbols \( C \sqsubseteq D \)) if we have for all interpretations \( C^\mathcal{I} \subseteq D^\mathcal{I} \).
- Is \( a \) an instance of a concept \( C \)? \( a \) is an instance of \( C \) if for all interpretations, we have \( a^\mathcal{I} \in C^\mathcal{I} \).
- Note: These questions can be posed with or without a TBox that restricts the possible interpretations.

Outlook

- Can we reduce the reasoning services to perhaps just one problem?
- What could be reasoning algorithms?
- What can we say about complexity and decidability?
- What has all that to do with modal logics?
- How can one build efficient systems?

Literature I

## Summary: Role descriptions

<table>
<thead>
<tr>
<th>Abstract</th>
<th>Concrete</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t$</td>
<td>$t$</td>
<td>$t^f$</td>
</tr>
<tr>
<td>$f$</td>
<td>$f$</td>
<td>$f^f$ (functional role)</td>
</tr>
<tr>
<td>$r \sqcap s$</td>
<td>$(r \land s)$</td>
<td>$r^f \sqcap s^f$</td>
</tr>
<tr>
<td>$r \sqcup s$</td>
<td>$(r \lor s)$</td>
<td>$r^f \sqcup s^f$</td>
</tr>
<tr>
<td>$\neg r$</td>
<td>$(\neg r)$</td>
<td>$(\neg r)^f$</td>
</tr>
<tr>
<td>$r^{-1}$</td>
<td>$(\text{inverse } r)$</td>
<td>${ (d, d') : (d', d) \in r^f }$</td>
</tr>
<tr>
<td>$r</td>
<td>_C$</td>
<td>$(\text{restr } C)$</td>
</tr>
<tr>
<td>$r^* (\text{trans } r)$</td>
<td>$(r^*)^f$</td>
<td>$r^f \circ s^f$</td>
</tr>
<tr>
<td>$r \circ s$</td>
<td>$(\text{compose } r, s)$</td>
<td>$r^f \circ s^f$</td>
</tr>
<tr>
<td>$1$</td>
<td>self</td>
<td>${ (d, d) : d \in D }$</td>
</tr>
</tbody>
</table>