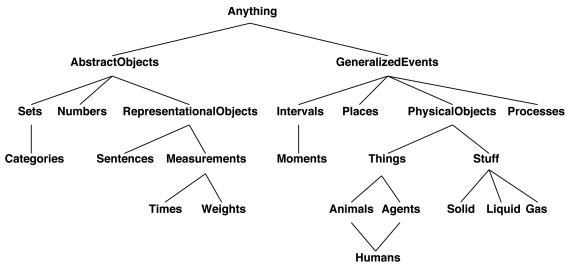
Knowledge Based Systems
Introduction

Jacek Malec
Dept. of Computer Science, Lund University, Sweden
February 4, 2013

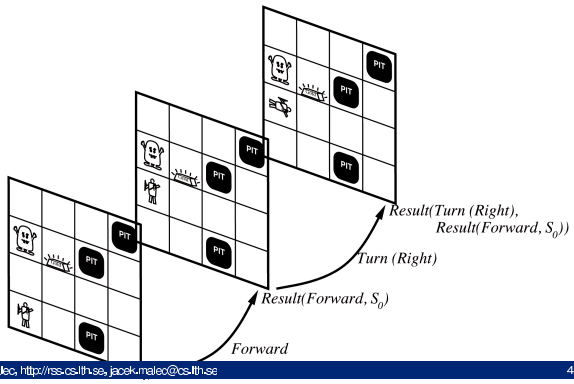
Ontology



Problems with predicate logic:

- static
- flat
- qualification/ramification/frame problem
- exceptions
- strength
- ...

Actions, situations



Representation of exceptions:

$\forall x(Bird(x) \wedge \neg Pinguin(x) \wedge \neg Ostrich(x) \rightarrow Flies(x))$
But if we know $Bird(Tweety)$ we can't say whether it flies!

Qualification Problem:

Unfortunately, many other birds don't fly either: caged birds, dead birds, birds with broken wing, bird with feathers covered by oil, ...
$\forall x(Bird(x) \wedge \neg Pinguin(x) \wedge \neg Ostrich(x) \wedge \ldots \rightarrow Flies(x))$

Other related problems: frame problem, ramification problem

Non-monotonic logic

$Th(\Delta)$

monotonic logic:
if $\Delta \subseteq \Delta'$ then $Th(\Delta) \subseteq Th(\Delta')$

non-monotonic logic:
if $\Delta \subseteq \Delta'$ and $Th(\Delta) \supset Th(\Delta')$

Closed World Assumption:

Things that cannot be proven true are probably false

leads to:

Negation as failure

Consider:

$$\Delta = \{A(Stockholm), A(Wroclaw), A(Copenhagen), A(Oslo), Conn(Oslo, Copenhagen), Conn(Stockholm, Oslo), \forall(x,y,z)(Conn(x,y) \wedge Conn(y,z) \rightarrow Conn(x,z))\}$$

Because it's not the case that $Conn(Wroclaw, Stockholm)$, CWA gives us immediately $\neg Conn(Wroclaw, Stockholm)$.
But let us add $Conn(Wroclaw, Copenhagen)$ to Δ.
$Conn(Wroclaw, Stockholm)$ can be proven now, so $\neg Conn(Wroclaw, Stockholm)$ should be removed as a consequence.
Non-monotonicity!

CWA is syntax-dependent:

If $\Delta = \{Single(John), Single(Mary), Clever(Kent)\}$
then CWA gives us:
$\{\neg Clever(John), \neg Clever(Mary), \neg Single(Kent)\}$.

CWA is syntax-dependent:

If $\Delta = \{Single(John), Single(Mary), Clever(Kent)\}$
then CWA gives us:
$\{\neg Clever(John), \neg Clever(Mary), \neg Single(Kent)\}$.

But if $\Delta = \{\neg Married(John), \neg Married(Mary), Clever(Kent)\}$
then CWA gives us:
$\{\neg Clever(John), \neg Clever(Mary), \neg Married(Kent)\}$.

Effective representation of knowledge:

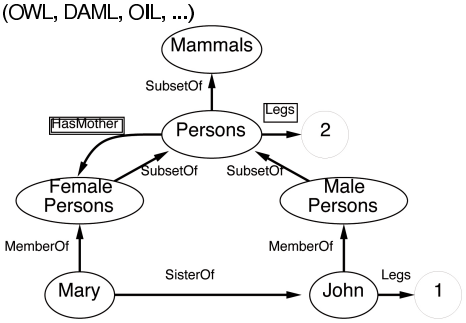
- storing
- retrieval
- modification

What should be represented?
- use logic
- not necessarily FOL

How should it be represented?
- whatever method you find suitable

Semantic networks:

Precursor of Description Logics and Semantic Web Languages (OWL, DAML, OIL, ...)



Reasoning in SN:

- inheritance via SubsetOf (SubClass) and MemberOf (isA) links
- intersection paths
- special meaning associated with some links (like cardinality constraints, etc.)
- classification, consistency, subsumption

May be effective given some restrictions on the logic (DL variants).
Rule-based reasoning on top (RIF and co.)

Knowledge-Based Systems

A generic term, might denote anything that involves encoded knowledge.
Or might mean a system where the knowledge component is *explicit* and manipulable.

Paradigms throughout history of AI:

- Logic-based systems;
- Rule-based systems (expert systems);
- Blackboard systems;
- Semantic web systems.

Rule-based systems

Or *expert systems*.
Promised much. Delivered (too) little. Caused "AI Winter" in the 90s.
Simple architecture:

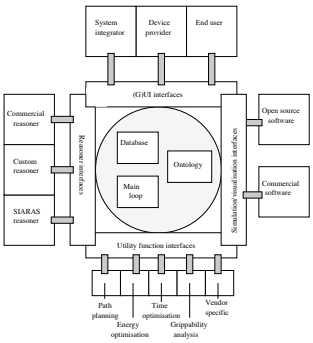
- Facts;
- Rules;
- Inference engine:
  - Matching;
  - Conflict resolution;
  - Rule application.

Blackboard systems

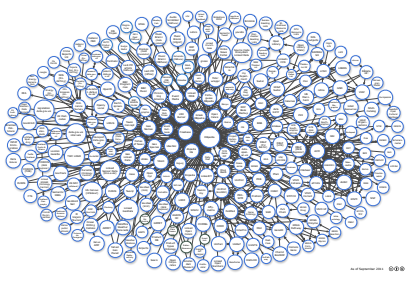
Architecture:

- Knowledge base (blackboard);
- Knowledge sources (expert problem solvers);
- Controller (agenda maintainer).

A blackboard system



Semantic Web



Linking Open Data cloud diagram, by Richard Cyganiak and Anja Jentzsch. `http://lod-cloud.net/`

Semantic Web

Lots of hype. Lots of acronyms. But some are important!

- URI – Uniform Resource Identifier
- RDF – Resource Description Framework
- RIF – Rule Interchange Format
- SPARQL – SPARQL Protocol and RDF Query Language
- OWL – Web Ontology Language

Open World Assumption!

SPARQL

W3C Recommendation
Queries:

- SELECT (returns a table)
- CONSTRUCT (returns RDF)
- ASK (returns a boolean)
- DESCRIBE (freedom)

Example:

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?name ?email
WHERE {
  ?person a foaf:Person.
  ?person foaf:name ?name.
  ?person foaf:mbox ?email.
}
```


A SPARQL query

What are all the country capitals in Africa?

```
PREFIX abc: <http://example.com/exampleOntology#>
SELECT ?capital ?country
WHERE {
  ?x abc:cityname ?capital ;
     abc:isCapitalOf ?y .
  ?y abc:countryname ?country ;
     abc:isInContinent abc:Africa .
}
```


A real SPARQL query

```
http://wiki.dbpedia.org/OnlineAccess
http://asimov.ludat.lth.se
```

```
select ?s where {
  ?s a rosetta:Camera.
}

select distinct ?s ?v where {
?s a rosetta:Camera.
?s ?p ?n.
?n caex-xml:hasName "FocusRange".
?n caex-xml:hasValue ?v.
}
```


Assignment 2

Sorry: not ready yet. Will be announced late tomorrow.