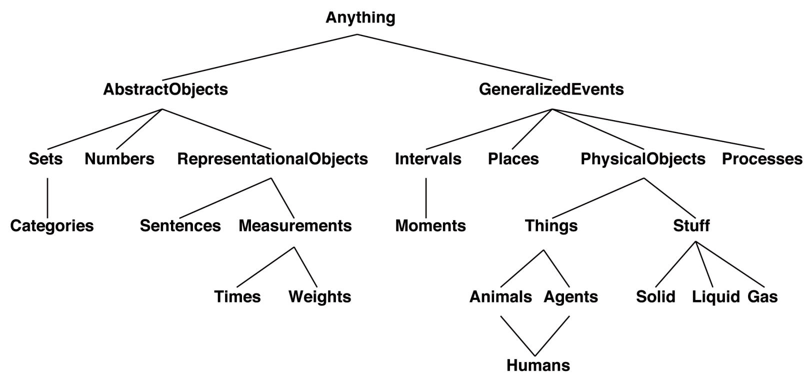
Knowledge Based Systems
Short Recapitulation

Jacek Malec
Dept. of Computer Science, Lund University, Sweden

March 1, 2012

Ontology



Anything

AbstractObjects GeneralizedEvents

Sets Numbers RepresentationalObjects Intervals Places PhysicalObjects Processes

Categories Sentences Measurements Moments Things Stuff

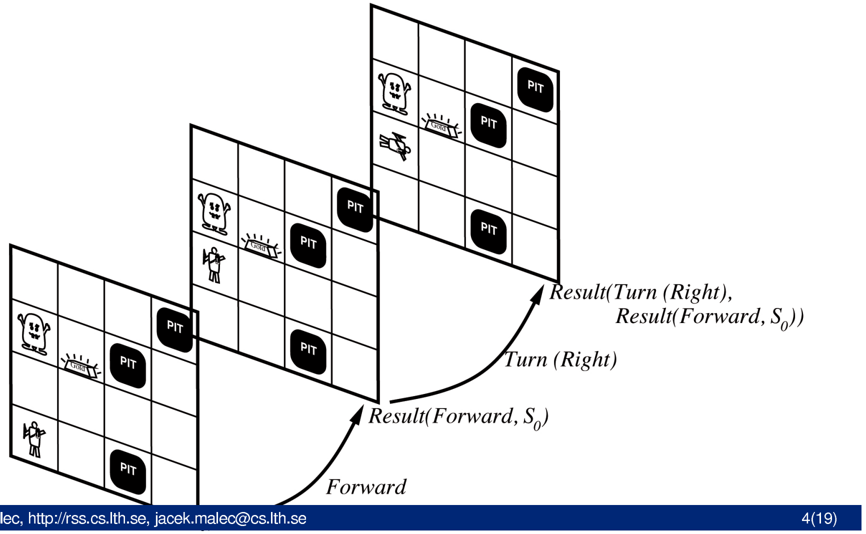
Times Weights Animals Agents Solid Liquid Gas

Humans

Problems with predicate logic:

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

Actions, situations



Situation calculus

Action descriptions:

- possibility axioms (when is an action possible)
- effect axioms (what's its effect, what changes)
- frame axioms (what remains the same)
  Important issue!

Quite often we need richer ontology.

Describing actions

"Effect" axiom—describe changes due to action

$$\forall s\ AtGold(s) \Rightarrow Holding(Gold, Result(Grab, s))$$

"Frame" axiom—describe *non-changes* due to action

$$\forall s\ HaveArrow(s) \Rightarrow HaveArrow(Result(Grab, s))$$

"Successor-state axioms" solve the representational frame problem

Successor-state axioms

Each axiom is "about" a *predicate* (not an action per se):

P true afterwards	$\Leftrightarrow$	[an action made P true
	$\vee$	P true already and no action made P false]

For holding the gold:

$$\forall a, s\ Holding(Gold, Result(a, s)) \Leftrightarrow$$

$$[(a = Grab \wedge AtGold(s)) \vee (Holding(Gold, s) \wedge a \neq Release)]$$

Making plans

Initial condition in KB:
$At(Agent, [1, 1], S_0)$
$At(Gold, [1, 2], S_0)$

Query: $Ask(KB, \exists s\ Holding(Gold, s))$
i.e., in what situation will I be holding the gold?

Answer: $\{s | Result(Grab, Result(Forward, S_0))\}$
i.e., go forward and then grab the gold

This assumes that the agent is interested in plans starting at S_0 and that S_0 is the only situation described in the KB

Making plans: A better way

Represent *plans* as action sequences $[a_1, a_2, \ldots, a_n]$

$PlanResult(p, s)$ is the result of executing p in s

Then the query $Ask(KB, \exists p\ Holding(Gold, PlanResult(p, S_0)))$ has the solution $\{p | [Forward, Grab]\}$

Definition of *PlanResult* in terms of *Result*:
$\forall s\ PlanResult([\,], s) = s$
$\forall a, p, s\ PlanResult([a|p], s) = PlanResult(p, Result(a, s))$

Planning systems are special-purpose reasoners designed to do this type of inference more efficiently than a general-purpose reasoner (later today)

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. Back in the game in 2000s.

Simple architecture:

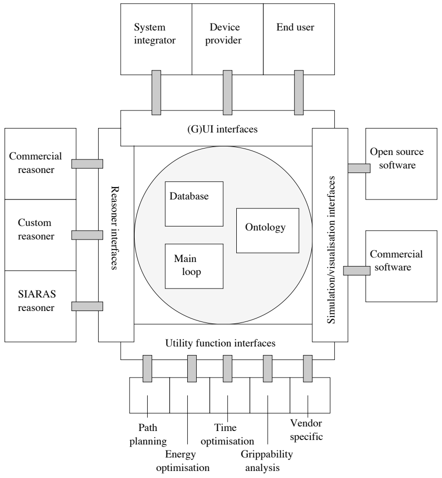
- Facts;
- Rules;
- Inference engine:
  - Matching;
  - Conflict resolution;
  - Rule application.

Blackboard systems

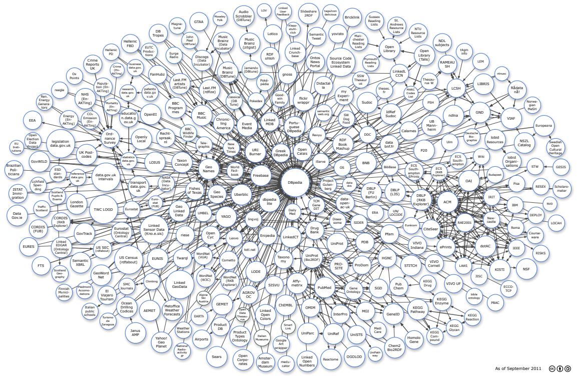
Architecture:

- 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 2a

Ready at last.

Task 3 asks you to investigate the library web world.