IsisWorld: an open source commonsense simulator for AI researchers

Dustin A. Smith, Bo Morgan

MIT Media Lab

AAAI-2010: Metacognition Workshop

Question

In 15 seconds...

1. **Think of all the things could fit in a .5m^3 box.**

Question

In 15 seconds...

1. **Think of all the things could fit in a .5m^3 box.**

2. **Think of all the things in your fridge.**

Question

Now try this...

1. **Think of all English words that match: -----n-**

Question

Now try this...

1. **Think of all English words that match: -----n-**

2. **Think of all English words that match: ----ing**

Question

Now try this...

1. **Think of all English words that match: -----n-**

2. **Think of all English words that match: ----ing**

Which is larger? (Tversky & Kahneman, 1983)

Question

Now try this...

1. **Think of all English words that match: -----n-**

2. **Think of all English words that match: ----ing**

Which is larger? (Tversky & Kahneman, 1983)

Thinking abstractly is more difficult. Be concrete!

Seymour Papert

You can't think about thinking, without thinking about thinking about **something**.

Seymour Papert **on Metareasoning... ?**

You can't think about thinking **about thinking**, without thinking about thinking **about thinking** about **something**!

Seymour Papert on Metareasoning... ?

You can't think about thinking **about thinking**, without thinking about thinking **about thinking** about **something**!

Metareasoning is hard to think about! We could benefit from a concrete, shared problem domain!

Metareasoning

Demonstrating a metareasoning algorithm requires three components:

1. **a set of concrete problem domains**
2. **a reasoner that solves problems in (1)**
3. **a metareasoner that solves problems in (2)**

Other problem domains...

- **Turing Test** *(Turing, 1950)*
- **Chess** *(Shannon, 1950)*
- **Compression of Wikipedia Text** *(Hutter 2006)*
- **RoboCup** *(1997)*

Turing Test

- **"All or nothing".** Doesn't give insight into the problem's solution. *It's just too hard.*

- **Irreproducible:** Tests cannot be easily replicated.

- **Effective?** Sufficient but not necessary condition for intelligence. Linked too much to "human intelligence"?

Really good chess

- Claude Shannon thought that chess' enormous state space would lead to general purpose problem solvers.

- **AI: Already Implemented!** What did we learn?

that solutions to canonical problems will always "overfit" the problem domain, and not generalize to other classes of problems.

... unless the problem itself is itself *generated* from a *space* of problems!

Hutter Prize

- Hutter prize, 50000€, for compressing human knowledge (expressed as Wikipedia text).

Program	Compression Options	Compressed size enwik8	Compressed size enwik9	Decompresser size (zip)	**Total size** enwik9+prog	Time (ns/byte) Comp	Time (ns/byte) Decomp	Mem	Alg	Note
durilca'kingsize	-m13000 -o40 -t2	16,209,167	127,377,411	407,477 xd	127,784,888	1398	1797	13000	PPM	31
paq8hp12any	-8	16,230,028	132,045,026	330,700 x	132,375,726	56993		1850	CM	22
drt|lpaq9m	9	17,964,751	143,943,759	110,579 x	144,054,338	2107	2151	1542	CM	26
paq8px_v60_turbo	-8	17,733,057	146,272,609	53,846 s	146,326,455	143846		1643	CM	26
zpaq 1.03	cmax_enwik9.cfg	18,238,435	149,376,058	14,317 xd	149,390,375	11961		2002	CM	32
xwrt 3.2	-l14 -b255 -m96 -s -e40000 -f200	18.679.742	151.171.364	52.569 s	151.223.933	2537	2328	1691	CM	

- **Reductionist design**: Reduces knowledge to a *coding theory* problem, along lines of Algorithmic Information Theory.

- Turing Tar Pit: How does this reduction help us solve the problem?

intelligence → 11010001011110011001111010101001111001100111101010 10

RoboCup

- Physical, Spatial and basic Planning task: But what about linguistics and commonsense reasoning?

- Culturally biased: Many Americans don't know much about soccer.

Let us not wait until low-level perceptual-motor problems have been solved to study reflective, social, linguistic and rich commonsense problems!

Qualities of a good task

- **Easily understandable**: *i.e.* a researcher studying language acquisition can have gross understanding of how his model affects planning and learning.

- **Tests only relevant components** of intelligence, abstracting away "irrelevant" parts.

- **Continual/Scalable benchmarks** that extend from easy to difficult.

- **Modular whenever possible**, recognizing the merits of specialization (and the limitations of humans).

IsisWorld

- a generated 3D virtual environment
- about human commonsense tasks
- directly maps to real world problems
- open source and multiplatform

IsisWorld

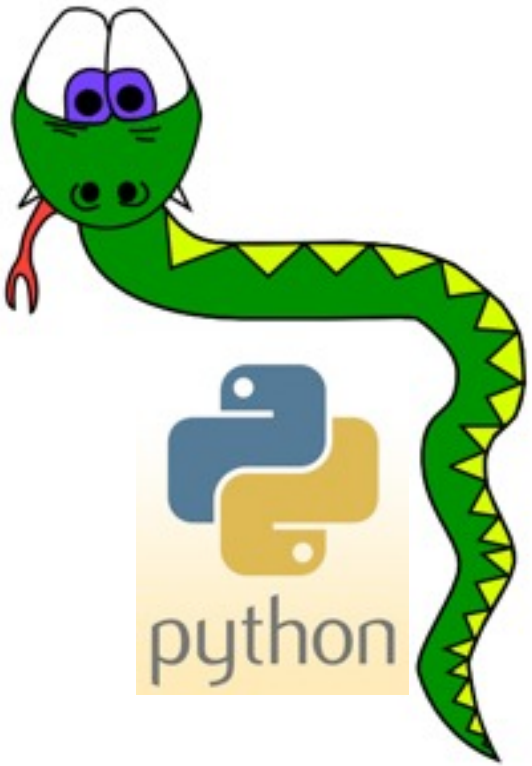
- a generated 3D virtual environment
- about human commonsense tasks
- directly maps to real world problems
- open source and multiplatform

IsisWorld

- a generated 3D virtual environment
- about human commonsense tasks
- directly maps to real world problems
- open source and multiplatform

IsisWorld

- a generated 3D virtual environment
- about human commonsense tasks
- directly maps to real world problems
- open source and multiplatform



Generate a problem space

Initialization file (e.g., /scenarios/pick_up_toast.py)

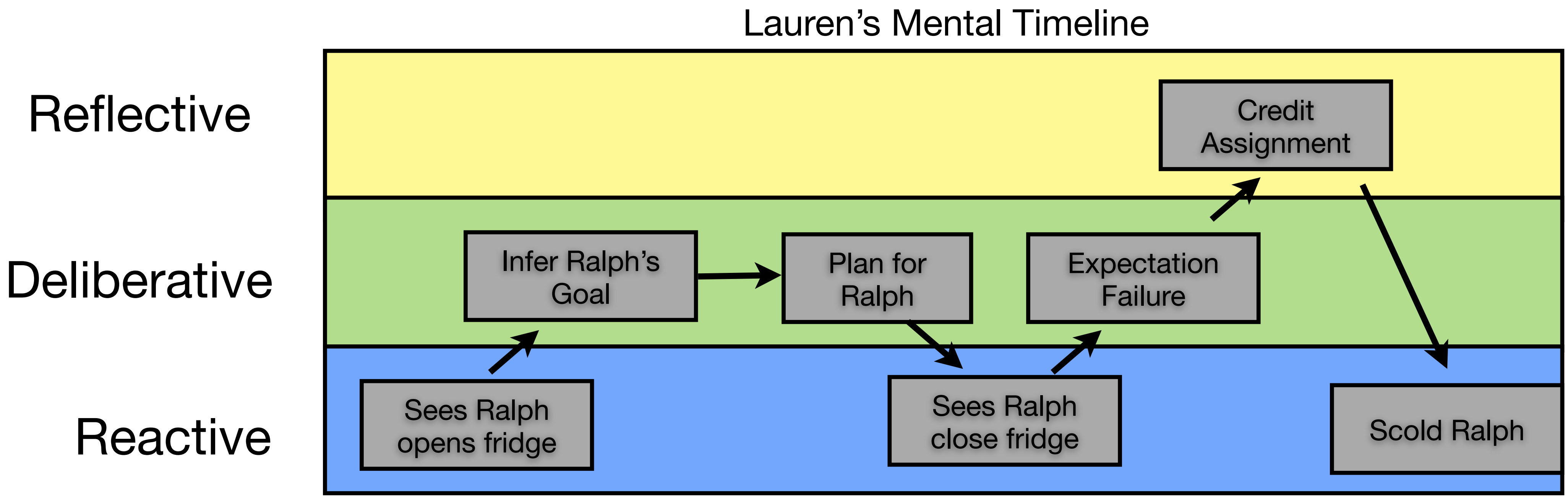
```
def environment():
    k = kitchen()
    put_in_world(k)

    f = fridge()
    put_in(f, k)

    ralph = IsisAgent(“Ralph)
```

From specification and defaults, **generates** an environment.

Example of Metareasoning



- **Ralph** and **Lauren** are in the kitchen.
- **Ralph** opens the fridge.
- **Lauren** infers **Ralph**'s intention.
- **Laruen** expects **Ralph** to close the fridge.


IsisWorld v0.4
20.0 fps
enter text and hit return


IsisWorld v0.4
20.0 fps
spect enter text and hit return


IsisWorld v0.4
19.2 fps


IsisWorld v0.4
15.1 fps
close that door

Metareasoning Problems

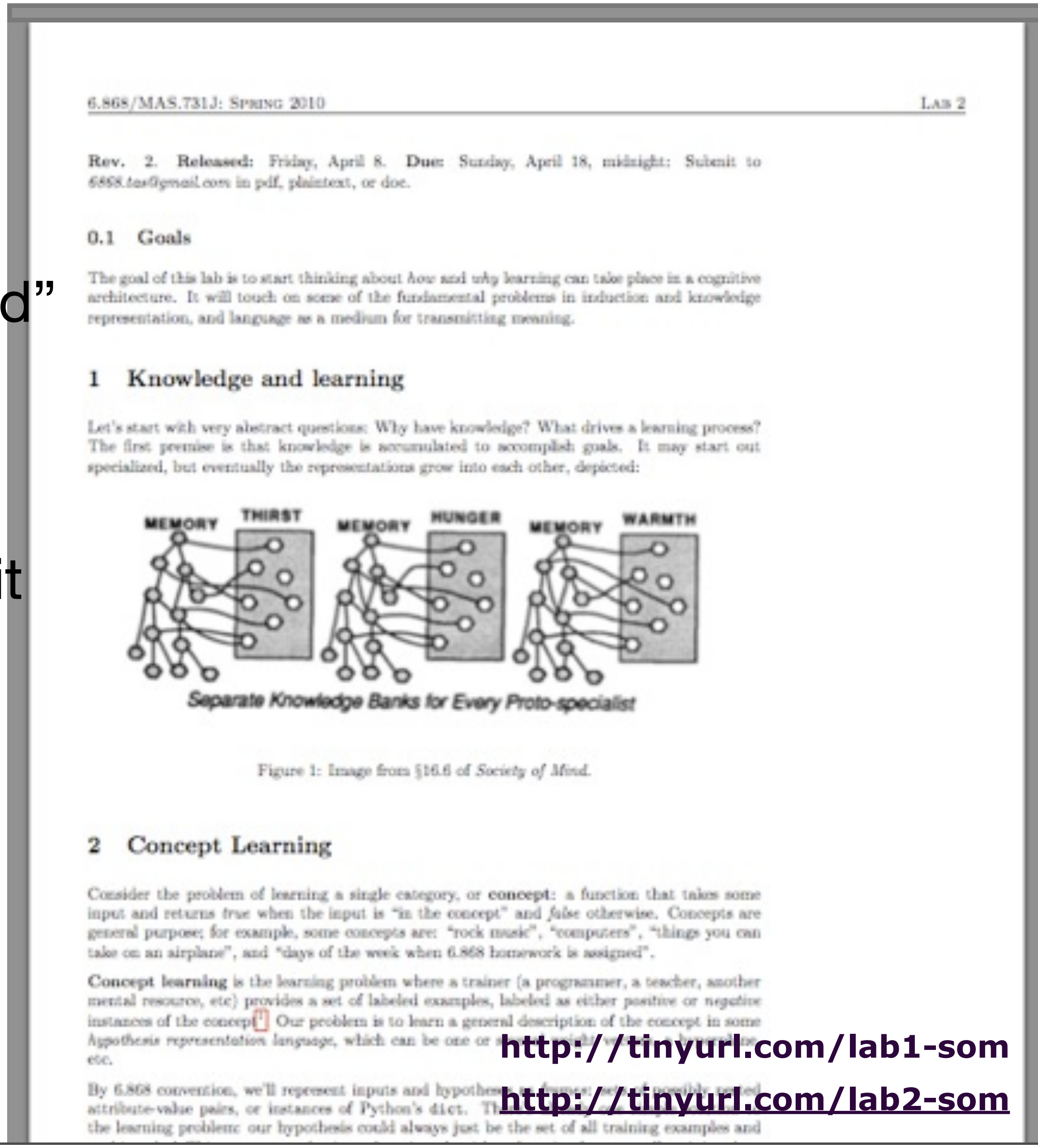
- **Transfer learning:** using knowledge from one domain to solve problems in another:
  - bodily knowledge to spatial knowledge: "*front* of house" "*back* of house"
  - spatial knowledge to time: "before the week", "by today"
- **Knowing which knowledge is relevant:** the more you know, the harder it is to filter.
- **Reflective debugging of plans:** Credit assignment
- **Self models** and **stories:** episodes of thought traces under different conditions and points of view.
- **Social reasoning:** using one's own mental resources to reason about another agent's state of mind

Used in 6.868 Spring 2010

IsisWorld was used in 6.868: Marvin Minsky's "Sociey of Mind" class for two labs assignments

- 30 students installed and ran it

- Binaries for Windows, Mac (ppc,i386), and Linux (64,32)



6.868/MAS.731J: Spring 2010 — Lab 2

Rev. 2. **Released:** Friday, April 8. **Due:** Sunday, April 18, midnight; Submit to *6868.tas@gmail.com* in pdf, plaintext, or doc.

0.1 Goals

The goal of this lab is to start thinking about *how* and *why* learning can take place in a cognitive architecture. It will touch on some of the fundamental problems in induction and knowledge representation, and language as a medium for transmitting meaning.

1 Knowledge and learning

Let's start with very abstract questions: Why have knowledge? What drives a learning process? The first premise is that knowledge is accumulated to accomplish goals. It may start out specialized, but eventually the representations grow into each other, depicted:

Figure 1: Image from §16.6 of *Society of Mind*.

2 Concept Learning

Consider the problem of learning a single category, or **concept**: a function that takes some input and returns *true* when the input is "in the concept" and *false* otherwise. Concepts are general purpose; for example, some concepts are: "rock music", "computers", "things you can take on an airplane", and "days of the week when 6.868 homework is assigned".

Concept learning is the learning problem where a trainer (a programmer, a teacher, another mental resource, etc) provides a set of labeled examples, labeled as either positive or negative instances of the concept[1]. Our problem is to learn a general description of the concept in some *hypothesis representation language*, which can be one or … etc.

By 6.868 convention, we'll represent inputs and hypothes… attribute-value pairs, or instances of Python's dict. Th… the learning problem: our hypothesis could always just be the set of all training examples and

http://tinyurl.com/lab1-som

http://tinyurl.com/lab2-som

Upcoming success stories

The simulator is used for our research and PhD theses.

- **Bo** is building and evaluating a model of transferring goal structures between privileged parent-child interactions.

- **Dustin**'s research is about connecting imperative commands with action representations.

Download it!

- **Download IsisWorld** (BSD 3-clause)

  http://mmp.mit.edu/isisworld

- **Fork the project, make changes, send them back!**

  http://github.com/dasmith/IsisWorld

Try it yourself!

Sense-act cycle in 5 lines of Python:

5.2 Building an agent

A simple implementation of an agent in Python:

```python
 import xmlrpclib as xml
# connect to environment via XML-RPC
e = xml.ServerProxy('http://localhost:8001')
# sense world
perceptions = e.do('sense')
# do something
e.do('say',{'statement':"Hello world!"})
# simulator is paused by default
# run for X=0.02 seconds
e.do('step_simulation',{'seconds':0.02})
```

What’s missing in AI?

Acknowledgements

- **Other simulator developers**: Yotam Aron, Rahul Rajagopalan, Chris M. Jones, Gleb Kuznetsov,

- **Sponsors**: the MIT Media Lab and the Mind Machine Project

- **Colleagues and Advisors**: Ken Arnold, Catherine Havasi, Rob Speer, Jason Alonso, Henry Lieberman, Marvin Minsky

- **Testers**: 6.868 students

Extra Slides
(about more specific ideas)

Qualities of a good task

- **Easily understandable**: *i.e.* a researcher studying language acquisition can have gross understanding of how his model affects planning and learning.

- **Tests only relevant components** of intelligence, abstracting away “irrelevant” parts.

- **Continual/Scalable benchmarks** that extend from easy to difficult.

- **Modular whenever possible**, recognizing the merits of specialization (and the limitations of humans).

Relevant components of intelligence

- **Adaptive**: can learn to solve many problems and goals, which may not be fixed in advance.

- **Resourceful**: can solve problems in many ways

- **Reflective**: reasons about itself and is pragmatically rational (will reason about itself, to spend time on problems proportionate to expected utility)

- **Communicational**: can communicate with humans in natural languages.

- **“Useful”**: helps us to live and better understand Earth (intelligence of a human form, not just any intelligence).

Relevant components of intelligence

- **Adaptive**: can learn to solve many problems and pursue goals, which may not be fixed in advance. **Test the agent on its learning (generate new problems and constrain agent's life span)**

- **Resourceful**: can solve problems in many ways. **Complex environment favors ingenuity in problem solving; evaluation penalizes stubbornness.**

- **Reflective**: reasons about itself and is pragmatically rational (will reason about itself, to spend time on problems proportionate to expected utility) **Give agent many goals, evaluate <u>overall</u> achievement**

- **Communicational**: can communicate with humans in natural languages. **Evaluate ability to learn and communicate knowledge by how well the agent/listener can use the knowledge to solve problems.**

- **"Useful"**: helps us to live and better understand our world. **Generate problems that are relevant to human-level tasks. Use human languages.**

R.A.L.P.H

Rational **A**gents with **L**imited **P**erformance **H**ardware

Principles of Metareasoning *

Stuart Russell and Eric Wefald
Computer Science Division
University of California
Berkeley, CA 94720

This paper is dedicated to the memory of Eric Wefald

Abstract

In this paper we outline a general approach to the study of metareasoning, not in the sense of explicating the semantics of explicitly specified meta-level control policies, but in the sense of providing a basis for selecting and justifying computational actions. This research contributes to a developing attack on the problem of resource-bounded rationality, by providing a means for analysing and generating optimal computational strategies. Because reasoning about a computation without doing it necessarily involves uncertainty as to its outcome, probability and decision theory will be our main tools. We develop a general formula for the utility of computations, this utility being derived directly from the ability of computations to affect an agent's external actions. We address some philosophical difficulties that arise in specifying this formula, given our assumption of *limited* rationality. We also describe a methodology for applying the theory to particular problem-solving systems, and provide a brief sketch of the resulting algorithms and their performance.

Phase 2: Language Learning

Implicit goal #1: the agent is rewarded when it *utters* the correct label when presented the with the corresponding example. Envision a Rosetta Stone kind of interface, iterating through examples and counter-examples of:

- *basic properties:* map between regions in geometric perceptual space and color/size/etc words (adjectives)

- *object labels:* map between collections of properties and nouns

- *object relation labels:* map between (objects re-presented in visual space?) and prepositions denoting spatial configurations.

- *action labels:* map between actions (possibly with relations to objects, etc) and verbs

- *event labels:* map between states or transitions between states and verbs.

- *event relation labels:* map between event states; temporal event relations (durative, punctual, composition, sequence)

Language labels are structured (taxonomic) but doesn't matter yet.

Phase 2: Language Learning

Implicit goal #1: the agent is rewarded when it *utters* the correct label when presented the with the corresponding example. Envision a Rosetta Stone kind of interface, iterating through examples and counter-examples of:

- *basic properties:* map between regions in geometric perceptual space and color/size/etc words (adjectives)

- *object labels:* map between collections of properties and nouns

- *object relation labels:* map between (objects re-presented in visual space?) and prepositions denoting spatial configurations.

- *action labels:* map between actions (possibly with relations to objects, etc) and verbs

- *event labels:* map between states or transitions between states and verbs.

- *event relation labels:* map between event states; temporal event relations (durative, punctual, composition, sequence)

Phase 1: Exploration

Agent builds model of own actions, differentiating the world in order to achieve goals and avoid anti- goals.

Many ways to learn:

- By observing the effects of your actions (blind search, experimentation,...)

- By observing others (their direct actions, inferring implicit intentions)

- By being told (by someone you trust)

- By reasoning (finding and using patterns in knowledge)

Qualities of a good task

- Easily understandable: i.e. a researcher studying language acquisition can have gross understanding of how his model affects planning and learning.

- Tests only relevant components of intelligence, abstracting away "irrelevant" parts.

- Continual/Scalable benchmarks that extend from easy to difficult.

- Modular whenever possible, recognizing the merits of specialization (and the limitations of humans).

Continuously Scaling

This simple task framework can improve as researchers build better agents.

- Objects in environment become richer, with textures, parts, mass, shapes, ...

- Agents get more detailed sensory input, and motor controls (actions they can accomplish)

- Researchers hand code common goal structures as benchmarks:

- Other agents in environment, leading to model other agent's beliefs and goals.

- Extends to the full set of prepositions (spatial and temporal relations) and increases vocabulary.

- Lexicon becomes structured, hierarchical labels ("thing"/"moveable"/"dog")

Language Learning

An agent is plopped in a strange new world, **needs to learn about the environment**, **its own actions and goals,** **intentions and beliefs of other agents**, **a common language,** and **how to use the language**. Finally, it is tested on its ability to solve problems in explained to it by other agents in the common language.

Language: A reflective level operation

- "situated" in the context of problem solving (semantics of language is now the mapping to the agent's acquired planning knowledge)

- *Adjectives*: map to sub-spaces of geometric spaces, for example perceptual topologies.

- *Nouns*: map to combinations of properties (the regions denoted by adjectives)

- *Verbs:* map to categories of events (composite relations between objects and properties) and orderings between events.