Knowledge Representation

What is knowledge representation?

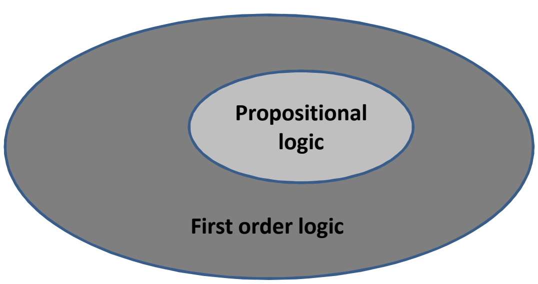
- Structured knowledge base
- Search and inference algorithms

Examples of knowledge representation in AI

- **Logic** for general reasoning
- **Expert systems** for medical diagnosis
- **Semantic networks**, **frames** for reasoning about categories
- **Scripts** for story understanding
- **Description logics, ontologies** for web services, e-mail sorting, spam-detection

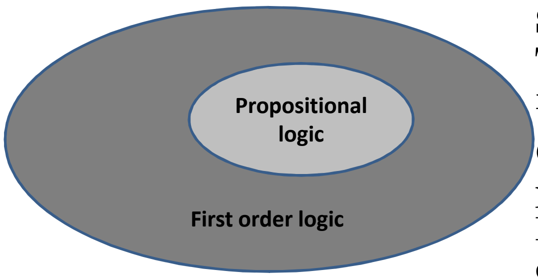
Propositional logic: AND, OR, NOT, IMPLIES

First order logic: add functions (e.g., MotherOf(x)), predicates (e.g., Man(x)), and quantifiers (∃ and ∀)



Propositional logic: AND, OR, NOT, IMPLIES

First order logic: add functions (e.g., MotherOf(x)), predicates (e.g., Man(x)), and quantifiers (∃ and ∀)



Example:

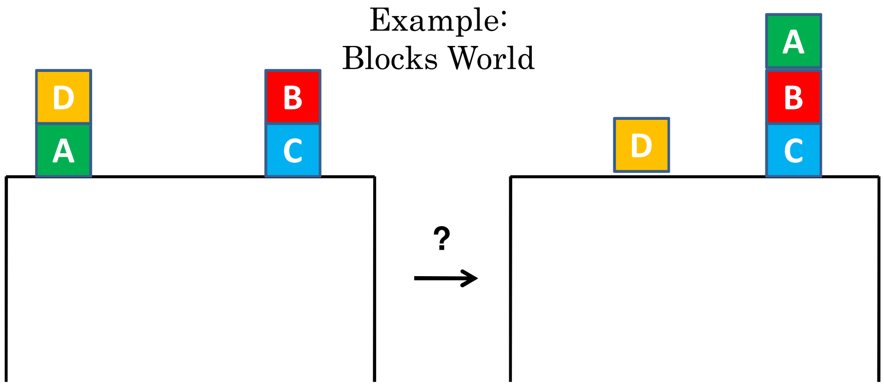
All men are mortal
Socrates is a man
Therefore, Socrates is mortal

Can't do this in propositional logic
In first order logic, can do:

$\forall x$, Man(x) → Mortal(x)
Man(Socrates)

Mortal(Socrates)

Planning via Logic
Example:
Blocks World



Initial state: on(A,TABLE,0), on(D, A, 0), on(C, TABLE,0), on(B,C,0), clear(A,0), clear(B,0), clear(D,0)

Goal state: on(A,B,2), on(D, TABLE, 0), on(C, TABLE,2), on(B,C,2), clear(A,2), clear(D,0)

Predicates:

on (x,y,t)
clear (x,y,t)
move (x,y,z,i)

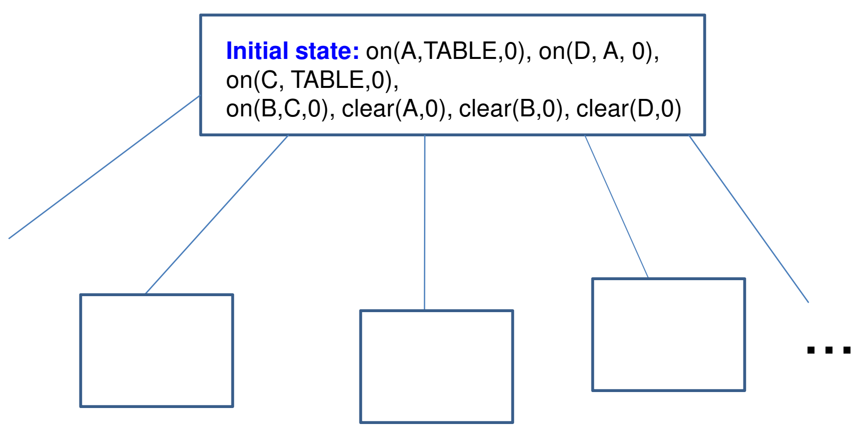
Axioms:

∀t: clear(TABLE,t)
∀x, y, z, t: on(x, y, i) ∧ clear (x,i) ∧ clear(z,i) ∧
move(x,y,z,i) ⊃ on(x,z, i+1) ∧ clear(y, i+1)
…

"Frame" Axioms:

Describe the propositions an action does *not* affect.

Planning as Search



Planning as Deduction

Initial state: on(A,TABLE,0), on(D, A, 0), on(C, TABLE,0), on(B,C,0), clear(A,0), clear(B,0), clear(D,0)

Goal state: on(A,B,2), on(D, TABLE, 0), on(C, TABLE,2), on(B,C,2), clear(A,2), clear(D,0)

Planning as Deduction

Initial state: on(A,TABLE,0), on(D, A, 0), on(C, TABLE,0), on(B,C,0), clear(A,0), clear(B,0), clear(D,0)

- **Add initial state to axioms.**
- **Use axioms to prove, via an automatic theorem proving method, that goal state can follow from initial state**
- **Plan is extracted from sequence of statements use in proof.**

Goal state: on(A,B,2), on(D, TABLE, 0), on(C, TABLE,2), on(B,C,2), clear(A,2), clear(D,0)

Planning as Satisfiability

(Kautz & Selman, 1992)

- Encode planning problem as a conjunction of axioms plus initial state plus goal state. Expand axioms with

  on(A,TABLE,0) ∧ on(D, A, 0) ∧ on(C, TABLE,0) ∧
  on(B,C,0) ∧ clear(A,0) ∧ clear(B,0) ∧ clear(D,0) ∧
  on(A,B,2) ∧ on(D, TABLE, 0) ∧ on(C, TABLE,2) ∧
  on(B,C,2) ∧ clear(A,2) ∧ clear(D,0) ∧...

- Determine if this formula is satisfiable
- If so, extract plan from assignments to variables that satisfy formula
- If not, increment goal *t* and try again.

To convert actions to propositions:

move(d,a,b,1) => on(d, a, 1) ∧ clear (d,1) ∧ clear(b,1)
=> on(d,b, 2) ∧ clear(a, 2)

To convert quantified formulas as propositions:

∀t: clear(TABLE,t) →
clear(TABLE,0) ∧ clear(TABLE,1) ∧ clear(TABLE,2)

∃: create disjunction of all possibilities

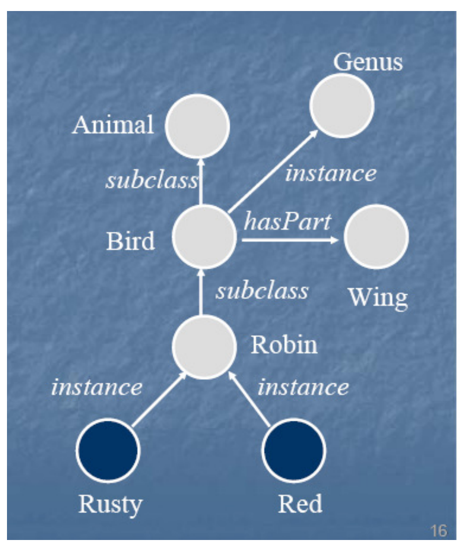
In short, formula has propositions for actions and states variables at each possible timestep

Problems with First Order logic

- Hard to express many concepts
- Hard to express associations
- Hard to keep consistent
- Takes a long time to do inference
- …

Semantic networks

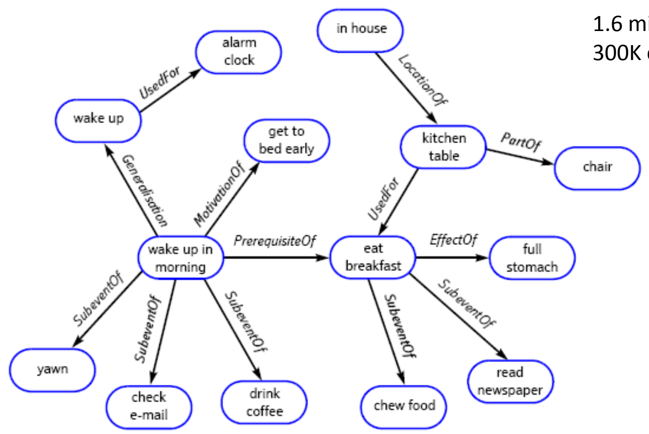
- Retrieving knowledge = graph search problem
- E.g., "is Rusty an animal?"
-
- Inheritance: "does Rusty have a wing?"



Example: ConceptNet (Openmind Commonsense)

- Large-scale semantic net mined from OMCS corpus

Teaching machines about everyday life



1.6 million links interrelating 300K concepts

Fig 2 A subset of ConceptNet.

Example: ConceptNet (Openmind Commonsense)

- Openmind Commons home page
- Demo

Table 1 ConceptNet's relational ontology of 20 link types.

ConceptuallyRelatedTo	IsA	FirstSubeventOf	DesirousEffectOf
ThematicKLine	MadeOf	SubeventOf	UsedFor
SuperThematicKLine	DefinedAs	LastSubeventOf	LocationOf
CapableOfReceivingAction	CapableOf	PrerequisiteEventOf	MotivationOf
PropertyOf	PartOf	EffectOf	DesireOf

Table 2 Ontology of concept types.

Events	Things	Places	Properties
Eat sandwich	Orange juice	At zoo	Furry
Sell car	Morning coffee	On table	Very expensive
Tell story	Policeman	Near school	Dark
Go to zoo	Leaf blower	Inside oven	Quickly
Type letter	Laptop computer	In closet	Dark

- **Reasoning in ConceptNet** = traversing links (or "spreading activation")

- **Given an initial set of concepts, can infer:**
  - What events might come next
  - What might have happened earlier
  - What objects might be required to perform an action
  - What properties of objects are
  - Where object might be found
  - Goals of people with respect to events or objects

Example: A story-understanding program reads "He woke up and turned off his alarm clock", then is asked "Why did he turn off his alarm clock?" "How did he turn off his alarm clock?"

Wordnet

- Wordnet (current projects)

Problems with semantic networks

- Can they scale?
- How to glom together entire "situations"? (e.g., alarm clock, morning, etc.) and to represent temporal order?
- How to learn the network automatically?

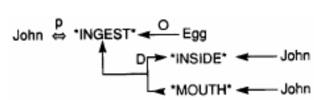
Scripts and Conceptual Dependency (Schank, 1970s-1980s)

- CD is a language that builds up meaning from primitives:
  - ACT action
  - PP objects (picture produces)
  - AA modifiers of actions (action aiders)
  - PA modifiers of objects (picture aiders)

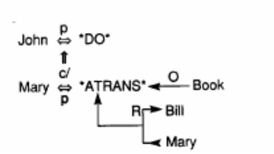
  - Primitive actions
    - ATRANS (transfer a relationship)
    - PTRANS (transfer a phyical location)
    - MTRANS (transfer mental information)
    - Etc.

From http://www.wiziq.com/educational-tutorials/presentation/485-Introduction-to-Artificial-Intelligence

CD examples



John ate an egg.

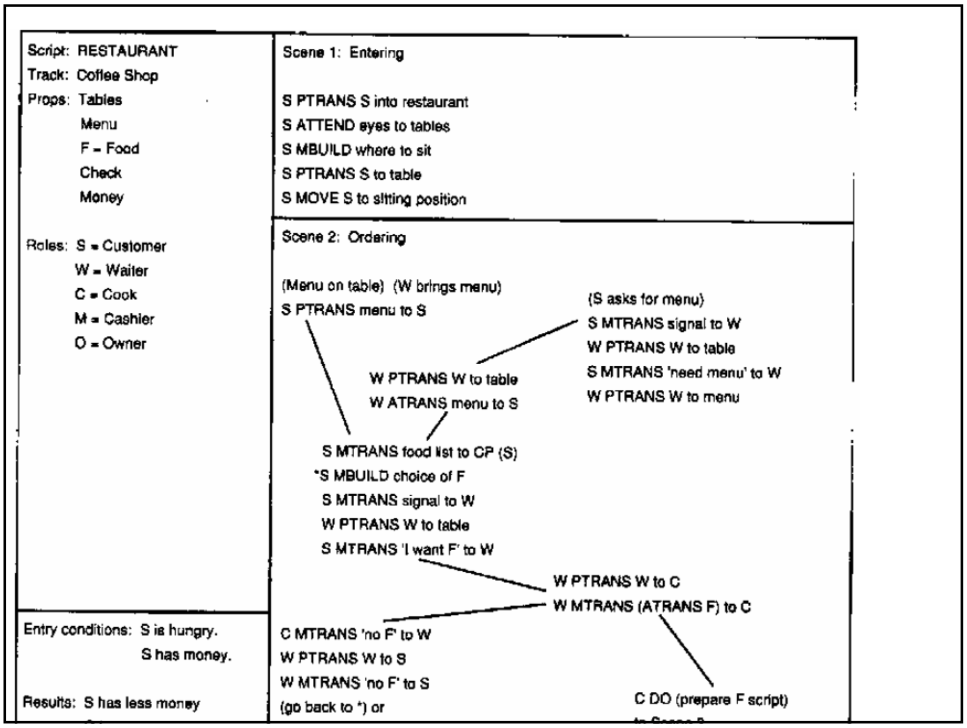


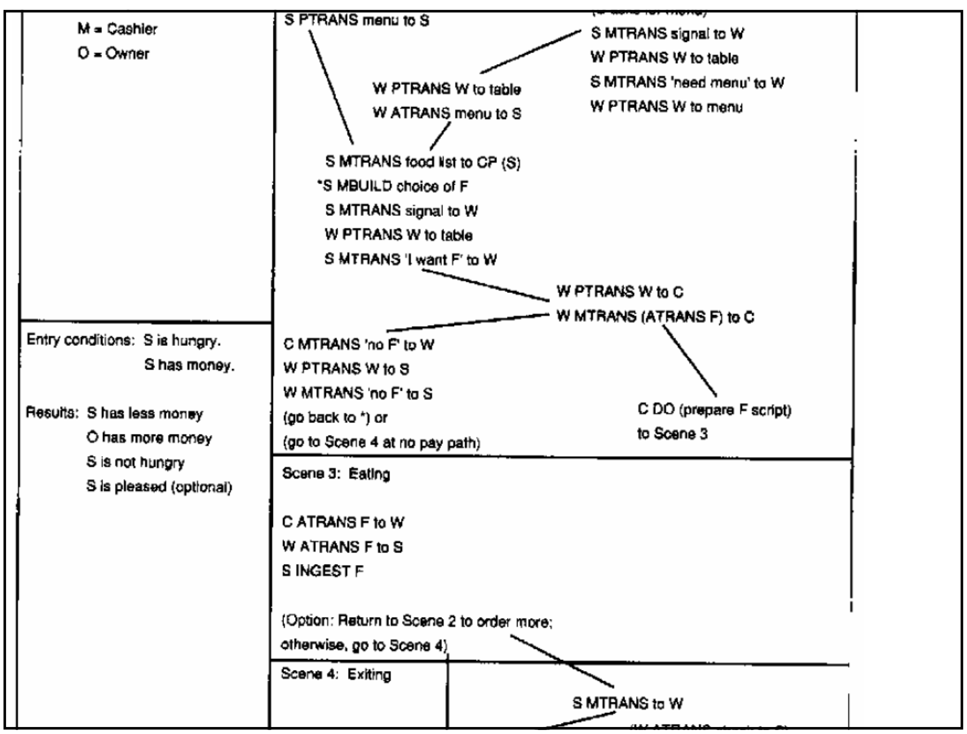
John prevented Mary from giving a book to Bill.

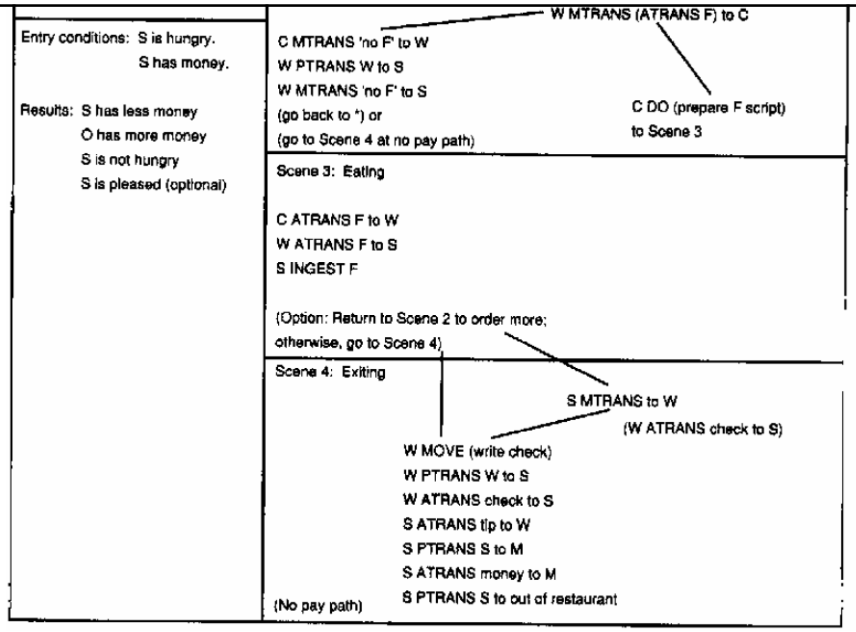
13

- **Scripts:** Stereotyped situations with temporal order, represented in CD.
- **Most famous example:** Restaurant script

- Scene 1: Entering
  S PTRANS S into restaurant, S ATTEND eyes to tables, S MBUILD where to sit, S PTRANS S to table, S MOVE S to sitting position
- Scene 2: Ordering
  S PTRANS menu to S (menu already on table), S MBUILD choice of food, S MTRANS signal to waiter, waiter PTRANS to table, S MTRANS 'I want food' to waiter, waiter PTRANS to cook
- Scene 3: Eating
  Cook ATRANS food to waiter, waiter PTRANS food to S, S INGEST food
- Scene 4: Exiting
  waiter MOVE write check, waiter PTRANS to S, waiter ATRANS check to S, S ATRANS money to waiter, S PTRANS out of restaurant


Script: RESTAURANT
Track: Coffee Shop
Props: Tables
Menu
F = Food
Check
Money
Roles: S = Customer
W = Waiter
C = Cook
M = Cashier
O = Owner
Entry conditions: S is hungry.
S has money.
Results: S has less money
Scene 1: Entering
S PTRANS S into restaurant
S ATTEND eyes to tables
S MBUILD where to sit
S PTRANS S to table
S MOVE S to sitting position
Scene 2: Ordering
(Menu on table) (W brings menu)
S PTRANS menu to S
(S asks for menu)
S MTRANS signal to W
W PTRANS W to table
S MTRANS 'need menu' to W
W PTRANS W to menu
W PTRANS W to table
W ATRANS menu to S
S MTRANS food list to CP (S)
*S MBUILD choice of F
S MTRANS signal to W
W PTRANS W to table
S MTRANS 'I want F' to W
W PTRANS W to C
W MTRANS (ATRANS F) to C
C MTRANS 'no F' to W
W PTRANS W to S
W MTRANS 'no F' to S
(go back to *) or
C DO (prepare F script)


M = Cashier
O = Owner
Entry conditions: S is hungry.
S has money.
Results: S has less money
O has more money
S is not hungry
S is pleased (optional)
S PTRANS menu to S
S MTRANS signal to W
W PTRANS W to table
S MTRANS 'need menu' to W
W PTRANS W to menu
W PTRANS W to table
W ATRANS menu to S
S MTRANS food list to CP (S)
*S MBUILD choice of F
S MTRANS signal to W
W PTRANS W to table
S MTRANS 'I want F' to W
W PTRANS W to C
W MTRANS (ATRANS F) to C
C MTRANS 'no F' to W
W PTRANS W to S
W MTRANS 'no F' to S
(go back to *) or
(go to Scene 4 at no pay path)
C DO (prepare F script)
to Scene 3
Scene 3: Eating
C ATRANS F to W
W ATRANS F to S
S INGEST F
(Option: Return to Scene 2 to order more;
otherwise, go to Scene 4)
Scene 4: Exiting
S MTRANS to W



- Story:
  - John went to a restaurant. John ordered a hamburger. When the hamburger came it was cooked perfectly. John was very happy and left a big tip for the waiter.
- Query:
  - Did John eat the hamburger?

Problems with scripts

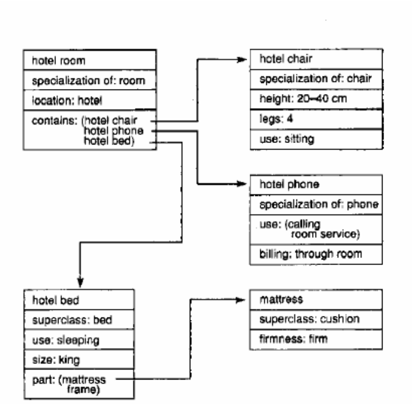
Frames (Minsky, 1970s)

- "When one encounters a new situation (or makes a substantial change in one's view of a problem one selects from a memory structure called a 'frame'. This is a remembered framework to be adapted to fit reality by changing details as necessary."
- (Minsky, 1975)

From http://www.wiziq.com/educational-tutorials/presentation/485-Introduction-to-Artificial-Intelligence

Frame example

a frame is a structured collection of data
- has *slots* (properties) and *fillers* (values)
- fillers can be links to other frames



This work led to the ideas for object-oriented programming!

17

Description logics (1990s, 2000s)

- Combination of semantic networks, frames, and first-order logic.
- Allows both structured conceptual representation and logical inference methods.
- Major current application: Language for "ontologies" for semantic web.

Semantic web

- Extension of html, xml for marking up web resources with “semantics”, via “shared ontologies”
- Inference and reasoning algorithms for using this metadata.
- “Intelligent search” versus “intelligent data”

```
<HTML>
<HEAD>
<META HTTP-EQUIV="SHOE" CONTENT="VERSION=1.0">
<TITLE>Prof. James A. Hendler</TITLE>
</HEAD>
<BODY>
<INSTANCE KEY="http://www.cs.umd.edu/users/hendler/">
<USE-ONTOLOGY ID="cs-dept-ontology" VERSION="1.0" PREFIX="cs" URL=
"http://www.cs.umd.edu/projects/plus/SHOE/cs.html" />
<CATEGORY NAME="cs.Professor" FOR="http://www.cs.umd.edu/users/hendler/"/>
<RELATION NAME="cs.member">
 <ARG POS=1 VALUE="http://www.cs.umd.edu/projects/plus/">
  <ARG POS=2 VALUE="http://www.cs.umd.edu/users/hendler/">
</RELATION>
<RELATION NAME="cs.name">
 <ARG POS=2 VALUE="Dr. James Hendler">
</RELATION>
<RELATION NAME="cs.doctoralDegreeFrom">
 <ARG POS=1 VALUE="http://www.cs.umd.edu/users/hendler/">
  <ARG POS=2 VALUE="http://www.brown.edu">
</RELATION>
<RELATION NAME="cs.emailAddress">
 <ARG POS=2 VALUE="hendler@cs.umd.edu">
</RELATION>
<RELATION NAME="cs.head">
  <ARG POS=1 VALUE="http://www.cs.umd.edu/projects/plus/">
  <ARG POS=2 VALUE="http://www.cs.umd.edu/users/hendler/">
</RELATION>
```

Markup demo

Info on demo

Ontologies

- "An ontology is a document or file that formally defines the relations among terms." (Berners-Lee et al., 2001)
- Consists of *taxonomy* and set of inference rules

Ontology examples

http://ebiquity.umbc.edu/ontology

- What will the semantic web be good for?
- What kinds of knowledge and reasoning will be necessary?
- Douglas Lenat – Cyc project, 1+ hour video
- http://video.google.com/videoplay?docid=7704388615049492068

A few other well-known knowledge representation methods

- Situation calculus
- Neural networks
- Decision trees
- Bayesian networks