# KNOWLEDGE REPRESENTATION

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>Issues in Knowledge Representation</td>
</tr>
<tr>
<td>6.1</td>
<td>A Brief History of AI Representational Systems</td>
</tr>
<tr>
<td>6.2</td>
<td>Conceptual Graphs: A Network Language</td>
</tr>
<tr>
<td>6.3</td>
<td>Alternatives to Explicit Representation</td>
</tr>
<tr>
<td>6.4</td>
<td>Agent Based and Distributed Problem Solving</td>
</tr>
<tr>
<td>6.5</td>
<td>Epilogue and References</td>
</tr>
<tr>
<td>6.6</td>
<td>Exercises</td>
</tr>
</tbody>
</table>

---

George F Luger

**ARTIFICIAL INTELLIGENCE**

Structure and Strategies for Complex Problem Solving

Fourth Edition
Associationist Theories

- The meaning of an object in terms of a network of associations with other objects
- When humans perceive and reason about an object, that perception is the first mapped into a concept
- This concept is part of the entire knowledge of the world and is connected through appropriate relationships to other concepts
- The relationships form an understanding of the properties and behavior of objects
- For example snow can be associated with cold, white, snowman, slippery and ice
Figure 6.1: Semantic network developed by Collins and Quillian in their research on human information storage and response times (Harmon and King 1985).
Knowledge Organization

- Knowledge organization has been formalized in inheritance systems.
- Inheritance systems allow to store information at the highest level of abstraction which reduces the size of knowledge base and helps prevent update inconsistencies.
- Inheritance also helps to maintain the consistency of the knowledge base when adding new classes and individuals.
Semantic network

- “semantic network” encompasses a family of graph-based representations
- These representations differ chiefly in the names that are allowed for nodes and links and the inferences that may be performed on these structures
- A common set of assumptions and concerns is shared by all network representation languages
- Represents knowledge as a graph with the nodes corresponding to facts or concepts and the arcs to relations or associations between concepts
- Both nodes and links are generally labeled
Network representation of properties of snow and ice
Quillian’s semantics approach to natural language understanding

- Determine the meaning of a body of English text by building up collections of these intersection nodes

- Choose between multiple meanings of words by finding the meanings with the shortest intersection path to other words in the sentence

- Answer a flexible range of queries based on associations between word concepts in the queries and concepts in the system
Figure 6.3: Three planes representing three definitions of the word “plant” (Quillian 1967).

Plant: 1) Living structure that is not an animal, frequently with leaves, getting its food from air, water, earth.
2) Apparatus used for any process in industry.
3) Put (seed, plant, etc.) in earth for growth.
Standardization of Network Relationships

• Case relationships include
  – agent
  – object
  – instrument
  – location
  – time
• A sentence is represented as a verb node, with various case links to nodes representing other participants in the action
• It is called a *case frame*
• In parsing a sentence, the program finds the verb and retrieves the case frame for that verb from its knowledge base
• It then binds the values of agent, object, etc. to the appropriate nodes in the case frame
Figure 6.4: Intersection path between “cry” and “comfort” (Quillian 1967).
Figure 6.5: Case frame representation of the sentence “Sarah fixed the chair with glue.”
Benefit of Conceptual dependency

- By providing a formal theory of natural language semantics, it reduces problems of **ambiguity**

- The representation itself directly captures much of natural language semantics, by attempting provide **canonical form** for the meaning of sentences. That is, all sentences that have the same meaning will be represented internally by syntactically identical, not just semantically equivalent, graphs.
Conceptual dependency theory of four primitive conceptualizations

<table>
<thead>
<tr>
<th>ACTs</th>
<th>actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>PPs</td>
<td>objects (picture producers)</td>
</tr>
<tr>
<td>AAs</td>
<td>modifiers of actions (action aiders)</td>
</tr>
<tr>
<td>PAs</td>
<td>modifiers of objects (picture aiders)</td>
</tr>
</tbody>
</table>

For example, all actions are assumed to reduce to one or more of the primitive ACTs. These primitives, listed below, are taken as the basic components of action, with more specific verbs being formed through their modification and combination.

- ATRANS: transfer a relationship (give)
- PTRANS: transfer physical location of an object (go)
- PROPEL: apply physical force to an object (push)
- MOVE: move body part by owner (kick)
- GRASP: grab an object by an actor (grasp)
- INGEST: ingest an object by an animal (eat)
- EXPEL: expel from an animal’s body (cry)
- MTRANS: transfer mental information (tell)
- MBUILD: mentally make new information (decide)
- CONC: conceptualize or think about an idea (think)
- SPEAK: produce sound (say)
- ATTEND: focus sense organ (listen)
**Figure 6.6:** Conceptual dependencies (Schank and Rieger 1974).

- **PP ↔ ACT**: indicates that an actor acts.
- **PP ↔ PA**: indicates that an object has a certain attribute.
- **O**: indicates the object of an action.
- **R ↔ PP**: indicates the recipient and the donor of an object within an action.
- **D ↔ PP**: indicates the direction of an object within an action.
- **1**: indicates the instrumental conceptualization for an action.
- **X ↔ Y**: indicates that conceptualization X caused conceptualization Y. When written with a C this form denotes that X COULD cause Y.
- **PP ↔ PA2**: indicates a state change of an object.
- **PP1 ↔ PP2**: indicates that PP2 is either PART OF or the POSSESSOR OF PP1.
Some basic conceptual dependencies and their use in representation

1. PP $\leftrightarrow$ ACT
   John $\leftrightarrow$ PTRANS
   John ran.

2. PP $\leftrightarrow$ PA
   John $\leftrightarrow$ height (>average)
   John is tall.

3. PP $\leftrightarrow$ PP
   John $\leftrightarrow$ doctor
   A nice boy

4. PP
   boy
   PP
   nice

5. PP
   dog
   POSS-BY
   John

6. ACT $\leftarrow$ PP
   John $\rightarrow$ PROPEL $\leftarrow$ cart
   John pushed the cart.

7. ACT $\leftarrow$ PP
   John $\rightarrow$ ATRANS $\leftarrow$ book
   John took the book from Mary.

8. ACT
   John $\rightarrow$ INGEST $\leftarrow$ ice cream
   John ate ice cream.

9. ACT $\leftarrow$ PP
   John $\rightarrow$ PTRANS $\leftarrow$ fertilizer
   John fertilized the field.

10. PP $\leftrightarrow$ PA
    plants $\leftrightarrow$ size $>$ x
    The plants grew.

11. (a) $\leftrightarrow$ (b)
    Bill $\leftrightarrow$ PROPEL $\leftarrow$ bullet
    Bill shot Bob.

12. T
    yesterday
    John $\leftarrow$ PTRANS
    John ran yesterday.
Figure 6.9: Conceptual dependency representing “John ate the egg” (Schank and Rieger 1974).

Figure 6.10: Conceptual dependency representation of the sentence “John prevented Mary from giving a book to Bill” (Schank and Rieger 1974).
Script

- A structured representation describing a stereotyped sequence of events in a particular context
- Scripts are used in natural language understanding systems to organize a knowledge base in terms of the situations that the system is to understand
- Components of a script
  - Early conditions or descriptors of the world
  - Results or facts that are true once the script has terminated
  - Props or the “things” that support the content of the script
  - Roles are the actions that the individual participants perform
  - Scenes
Questions

Example

John went to a restaurant last night. He ordered steak. When he paid he was running out of money. He hurried home since it had started to rain.

Questions

Did John eat dinner last night? Did John use cash or a credit? How could John get a menu? What did John buy?

Example

Sue went to lunch. She sat a table and called a waiter, who brought her a menu. She ordered a sandwich.

Questions

Why did the waitress bring Sue a menu? Was Sue in a restaurant? Who paid? Who was the “she” who ordered the sandwich?
More Questions

Example
Kate went to a restaurant. She was shown to a table and ordered steak from the waitress. She sat there and waited for a long time. Finally, she got mad and left.

Questions
Who is the “she” who sat and waited? Why did she wait? Who was the “she” who got mad and left?

Example
John visited his favorite restaurant on the way to the concert. He was pleased by the bill because he like Mozart.
Scripts Selections

- Script selection is usually based on matching "key" words
- It is difficult to determine which of two or more potential scripts should be used
- The script match problem is "deep" in the sense that no algorithm exists for guaranteeing correct choices
- It requires heuristic knowledge about the organization of the world
- Scripts assist only in the organization of that knowledge
Figure 6.11: A restaurant script (Schank and Abelson 1977).
A restaurant script (Schank and Abelson 1977)

**Scene 1: Entering**
- **S** enters restaurant
- **S** looks around the tables
- **S** chooses a table
- **S** sits down
- **S** waits for the waiter

**Roles:**
- **S** = Customer
- **W** = Waiter
- **C** = Cook
- **M** = Cashier
- **O** = Owner

**Props:**
- Tables
- Menu
- Food
- Check
- Money

**Entry conditions:**
- **S** is hungry
- **S** has money

**Results:**
- **S** has less money
- **S** has more money
- **S** is not hungry
- **S** is pleased (optional)

**Scene 2: Ordering**
- **S** looks at the menu
- **W** provides the menu
- **S** chooses food
- **W** takes order

**Scripts:**
- **S** asks for menu
- **W** brings menu
- **W** helps **S** choose
- **W** writes order

**Scene 3: Eating**
- **C** prepares food
- **W** delivers food
- **S** eats

**Options:**
- Return to Scene 2 to order more
- Go to Scene 4 (exit)

**Scene 4: Exit**
- **W** takes payment
- **S** leaves

**Scripts:**
- **W** writes check
- **W** gives receipt
- **S** leaves

---

**A R T I F I C I A L  I N T E L L I G E N C E:** Structure and Strategies for Complex Problem Solving, 4th Edition

© 2002 Addison Wesley
Frames

- Another representation scheme
- Similar to scripts
- Capture in explicitly organized data structures the implicit connections of information in a problem domain
- Support the organization of knowledge into more complex units that reflect the organization of objects in the domain
- According to Minsky, a frame is a static data structure used to represent well-understood stereotyped situations
The slots in the frame contains information

- Frame identification information
- Relationship of this frame to other frames
- Descriptors of requirements for a frame
- Procedural information on use of the structure described
- Frame default information
- New instance information
A frame includes:

1. *Frame identification information.*
2. *Relationship of this frame to other frames.* The “hotel phone” might be a special instance of “phone,” which might be an instance of a “communication device.”
3. *Descriptors of requirements for a frame.* A chair, for instance, has its seat between 20 and 40 cm from the floor, its back higher than 60 cm, etc. These requirements may be used to determine when new objects fit the stereotype defined by the frame.
4. *Procedural information on use of the structure described.* An important feature of frames is the ability to attach procedural code to a slot.
5. *Frame default information.* These are slot values that are taken to be true when no evidence to the contrary has been found. For instance, chairs have four legs, telephones are pushbutton, or hotel beds are made by the staff.
6. *New instance information.* Many frame slots may be left unspecified until given a value for a particular instance or when they are needed for some aspect of problem solving. For example, the color of the bedspread may be left unspecified.
Figure 6.12: Part of a frame description of a hotel room. “Specialization” indicates a pointer to a superclass.
Figure 6.13: Spatial frame for viewing a cube (Minsky 1975).
Conceptual Graphs: A Network Language

• An example of a network representation language

• Conceptual Graph
  – finite, connected, bipartite graph

• Nodes of the graph
  – concepts or conceptual relations

• Do not use labeled arcs; instead the conceptual relation nodes represent relations between concepts
Figure 6.14: Conceptual relations of different arities.

- Flies is a 1-ary relation.
- Color is a 2-ary relation.