KNOWLEDGE REPRESENTATION

Chapter #3

Representation and Mapping

- **Facts**: things we want to represent.
- **Representations of facts**: things we can manipulate.

- Eg Sky is blue Sky(BLUE)

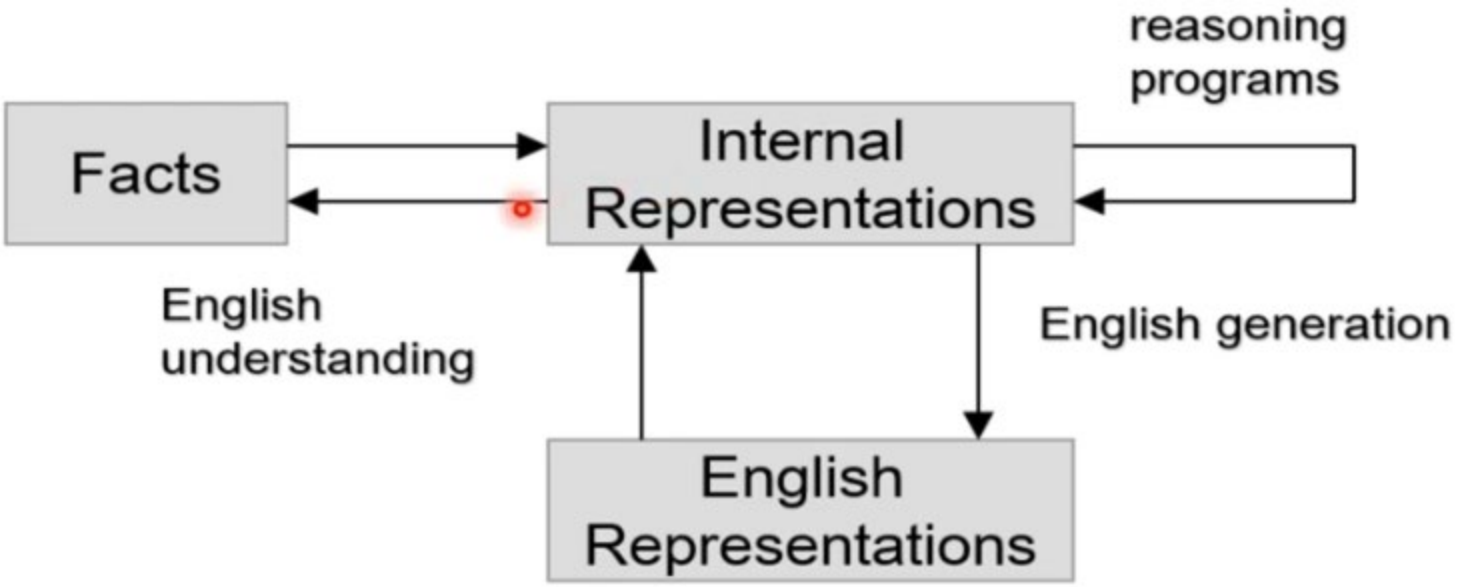
- Thus, knowledge representation can be considered at two levels :
- Knowledge level at which facts are described, and
- Symbol level at which the representations of the objects, defined in terms of symbols, can be manipulated in the programs.

Mapping between Facts and Representation

- Knowledge is a collection of "facts" from some domain.
- We need a representation of "facts" that can be manipulated by a program.
- Normal English is insufficient, too hard currently for a computer program to draw inferences in natural languages.
- Therefore, we must be able to map "*facts to symbols*" and "*symbols to facts*" using *forward and backward representation mapping*.

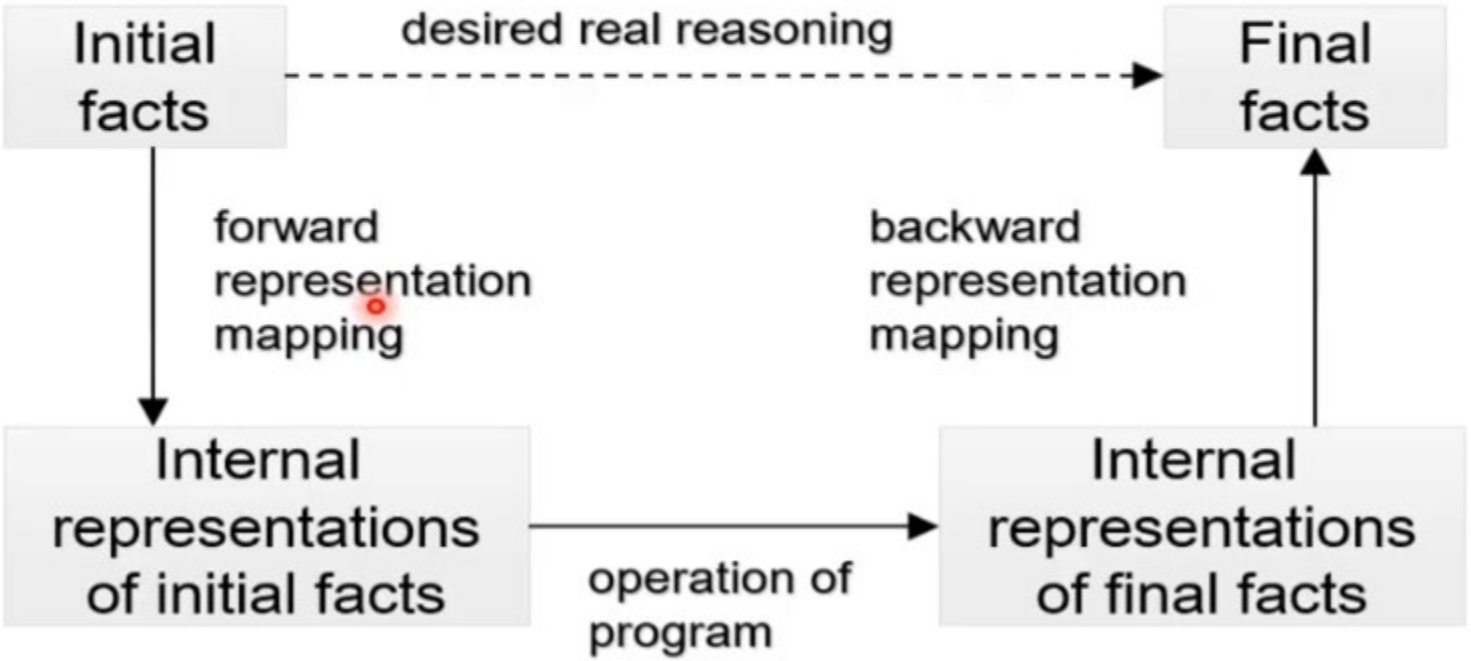

LOVELY PROFESSIONAL UNIVERSITY
PUNJAB (INDIA)
L
P
U

Mapping between Facts and Representation



Forward and Backward Representation

The forward and backward representations are elaborated below



Representation

- Sets of Syntax and Semantic
  - Convention which make it possible to describe things
  - Syntax: Specified Symbols and rules.
  - Semantic: How meaning is associated with symbol arrangement allowed by syntax

Representation and Mapping

- Spot is a dog
- Every dog has a tail
- Spot has a tail

[it is new knowledge]

Using backward mapping function to generate English sentence

Representation and Mapping

- Spot is a dog

  dog(Spot)

- Every dog has a tail

  $\forall x: dog(x) \rightarrow hastail(x)$

- Spot has a tail

  hastail(Spot)

  [it is new knowledge]

Using backward mapping function to generate English sentence

Knowledge Representation Properties

- A good knowledge representation enables fast and accurate access to knowledge and understanding of the content.
- A knowledge representation system should have following properties.

  - Representational Adequacy
  - Inferential Adequacy
  - Inferential Efficiency
  - Acquisitional Efficiency

Properties for Knowledge Representation Systems

1

Representational Adequacy

The ability to represent the required knowledge

2

Inferential Adequacy

The ability to manipulate the knowledge represented to produce new knowledge corresponding to that inferred from the original

3

Inferential Efficiency

The ability to direct the inferential mechanisms into the most productive directions by storing appropriate guides

4

Acquisitional Efficiency

The ability to acquire new knowledge using automatic methods wherever possible rather than reliance on human intervention

Properties of Good Representation

Representational adequacy

- It's a ability to represent required knowledge that are needed in that domain

Inferential adequacy

- It's a ability to manipulate the knowledge represented to produce new knowledge

Properties of Good Representation

Inferential efficiency

- The ability to direct the inferential mechanisms into the most productive directions by storing appropriate guides.

Acquisitional efficiency

- the ability to acquire new knowledge using automatic methods wherever possible rather than reliance on human intervention

Knowledge Representation Schemes

- There are four types of Knowledge representation :
  - Relational
  - Inheritable
  - Inferential
  - Declarative/Procedural.

Simple Relational Knowledge

Simple relational knowledge

- It's a simplest way to represent facts.
- It uses set of relations to store fact as that of database.
- Each fact about a set of objects is set out systematically in columns.
- It stores set of attributes & its associated values
- This approach of knowledge representation is famous in database systems where the relationship between different entities is represented.

Palyer	Height	Weight
Sachin	5.2	57
Gambhir	5.10	62

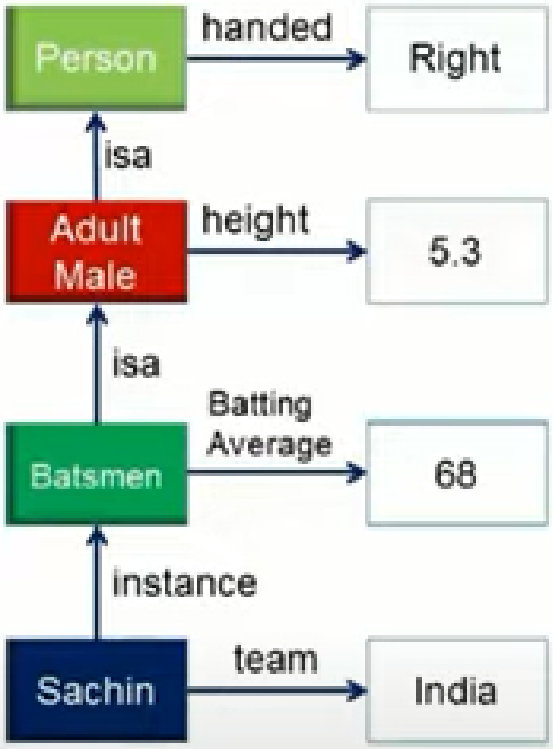
Inheritable knowledge

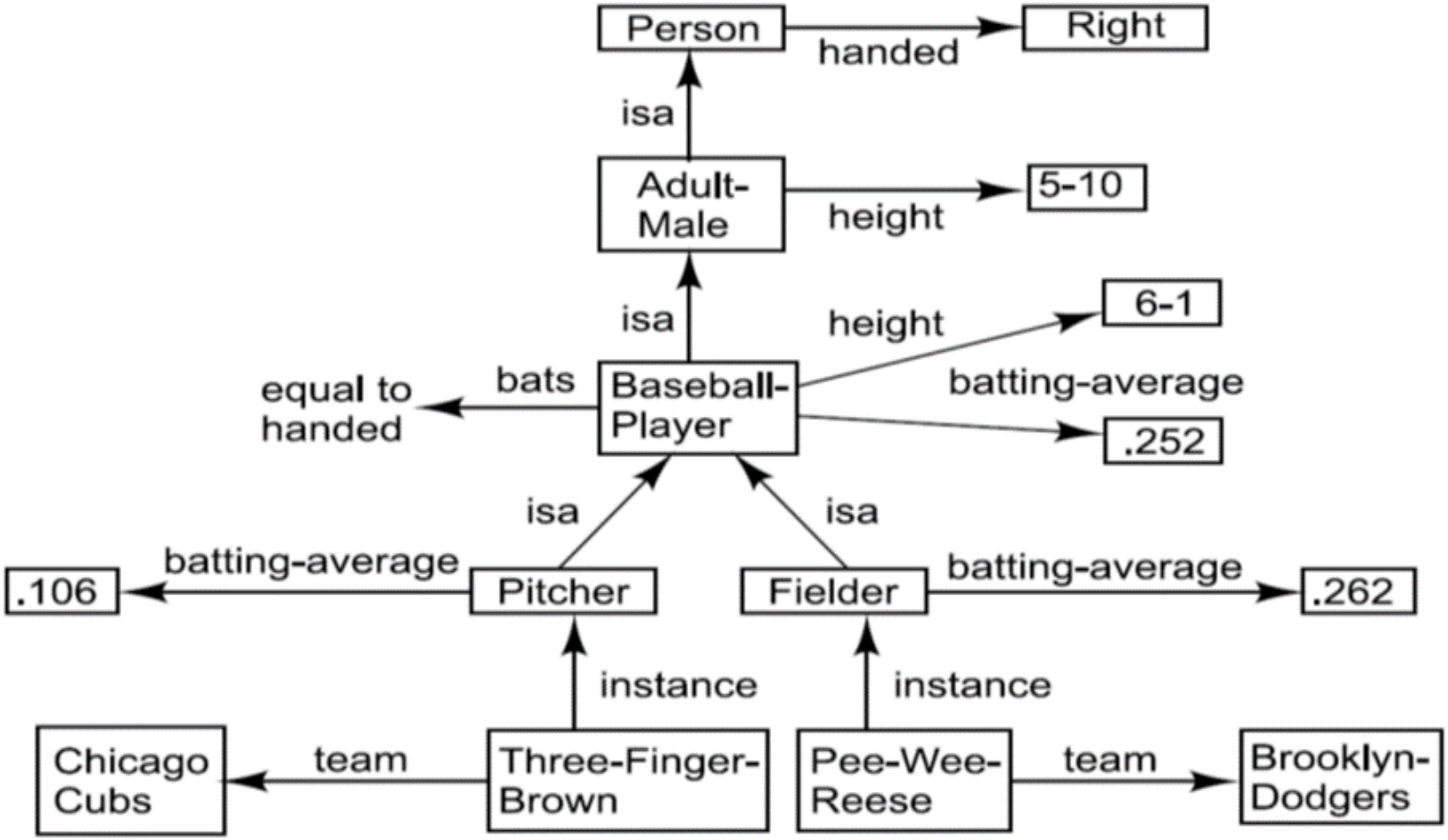
- In the inheritable knowledge approach, all data must be stored into a hierarchy of classes.
- All classes should be arranged in a generalized form or a hierarchal manner.
- In this approach, we apply inheritance property.
- Elements inherit values from other members of a class.
- This approach contains inheritable knowledge which shows a relation between instance and class, and it is called instance relation.
- Every individual frame can represent the collection of attributes and its value.
- In this approach, objects and values are represented in Boxed nodes.
- We use Arrows which point from objects to their values.

Inheritable Knowledge

Inheritable Knowledge

- Relational knowledge is made up of objects consisting of :
  - Attributes
  - Corresponding associated values
- We extend the base more by allowing inference mechanisms:
- Property inheritance:
  - Elements inherit values from being members of a class
  - Data must be organized into a hierarchy of classes





Inferential knowledge

- Inferential knowledge approach represents knowledge in the form of formal logics.
- This approach can be used to derive more facts.
- It guaranteed correctness.
- **Example:** Let's suppose there are two statements:
  - Marcus is a man
  - All men are mortal
    Then it can represent as;

    man(Marcus)
    ∀x = man (x) ----------> mortal (x)s

Inferential Knowledge

- Represent knowledge as formal logic:

- Example:

  Fact : All dogs have tails

  FOL : $\forall x : dog(x) \rightarrow hastail(x)$

- **Advantages:**
  - A set of strict rules.
    - Can be used to derive more facts.
    - Truths of new statements can be verified.
    - Guaranteed correctness.
  - Many inference procedures available to in implement standard rules of logic.

Procedural knowledge

- Procedural knowledge approach uses small programs and codes which describes how to do specific things, and how to proceed.
- In this approach, one important rule is used which is **If-Then rule**.
- In this knowledge, we can use various coding languages such as **LISP language** and **Prolog language**.
- We can easily represent heuristic or domain-specific knowledge using this approach.
- But it is not necessary that we can represent all cases in this approach.

Procedural Knowledge

- **Disadvantages:**
  - Completeness:
    - All cases may not be represented.
  - Consistency:
    - All deductions may not be correct.
  - Modularity is sacrificed.
  - Changes in knowledge base might have far-reaching effects.


LOVELY PROFESSIONAL UNIVERSITY
PUNJAB (INDIA)
L
P
U

Knowledge and reasoning also play a crucial role in dealing with _____ environment.

a) Completely Observable
b) Partially Observable
c) Neither Completely nor Partially Observable
d) Only Completely and Partially Observable

Treatment chosen by doctor for a patient for a disease is based on _____

a) Only current symptoms
b) Current symptoms plus some knowledge from the textbooks
c) Current symptoms plus some knowledge from the textbooks plus experience
d) None of the mentioned

A knowledge-based agent can combine general knowledge with current percepts to infer hidden aspects of the current state prior to selecting actions.

a) T*r*ue

b) False

A) Knowledge base (KB) is consists of set of statements.
B) Inference is deriving a new sentence from the KB.
Choose the correct option.
a) A is true, *B* is true
b) A is false, B is false
c) A is true, B is false
d) A is false, B is true

Inference algorithm is complete only if _____

a) It can derive any sentence

b) It can derive any sentence that is an entailed version

c) It is truth preserving

d) I*t* can derive any sentence that is an entailed version & It is truth preserving

A knowledge representation system should have which of the following properties.
i. Representation Adequacy
ii. Inferential Adequacy
iii. Inferential Efficiency
A) i and ii only
B) ii and iii only
C) i and iii only
D) All i, ii and iii

... is the ability to represent all kinds of knowledge that are needed in that domain.

A) Representation Adequacy
B) Inferential Adequacy
C) Inferential Efficiency
D) Acquisitional Efficiency

... is the ability to manipulate the representational structures to derive new structures corresponding to new knowledge inferred from old.

A) Representation Adequacy
B) Inferential Adequacy
C) Inferential Efficiency
D) Acquisitional Efficiency

... is the ability to acquire new knowledge using automatic methods wherever possible rather than reliance on human intervention.
A) Representation Adequacy
B) Inferential Adequacy
C) Inferential Efficiency
D) Acquisitional Efficiency

State whether the following statements about the relational knowledge are True or False.
i. It is the simplest way to represent declarative facts.
ii. It provides a framework to compare two objects based on equivalent attributes.
A) i-True, ii-True
B) i-True, ii-False
C) i-False, ii-False
D) i-False, ii-True

Any instance in which two different objects are compared is a ... type of knowledge.
A) inheritable
B) relational
C) inferential
D) procedural

State whether the following statements about inheritable knowledge are True.

i. Here the knowledge elements inherit attributes from their parents.

ii. The basic knowledge representation not required to be augmented with an inference mechanism.

iii. The classes are organized in a generalized hierarchy.

A) i and ii only

B) ii and iii only

C) i and iii only

D) All i, ii and iii


LOVELY PROFESSIONAL UNIVERSITY
PUNJAB (INDIA)
L
P
U

Issues in Knowledge Representation

- The main objective of knowledge representation is to draw the conclusions from the knowledge,
- but there are many issues associated with the use of knowledge representation techniques.

- Important Attributes.
- Relationship Among Attributes.
- Choosing the granularity of representation.
- Representing Set of objects.
- Finding right structure as needed.

Important Attributes

- Are there any attributes so basic, that occur in many different types of problem?
- If there are we need to make sure that they are handled appropriately in each of problem.
- There are two such attributes as **instance & isa** , which are important because each supports property of inheritance.

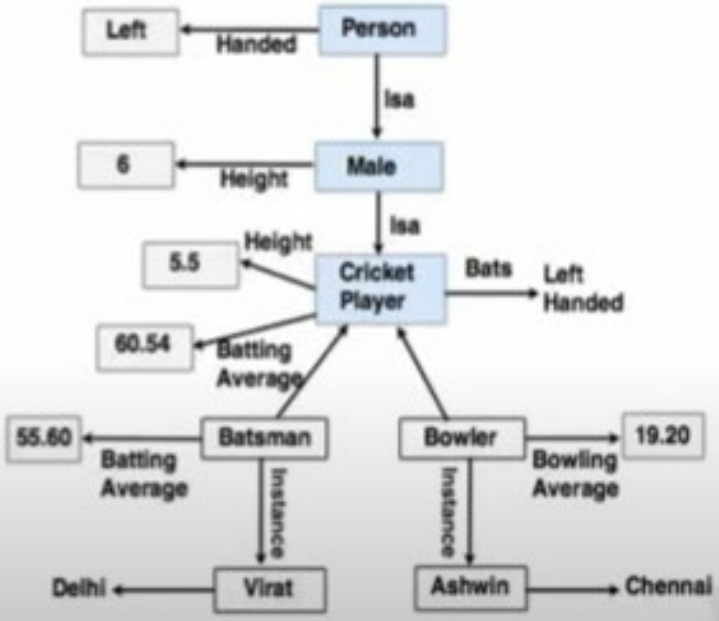
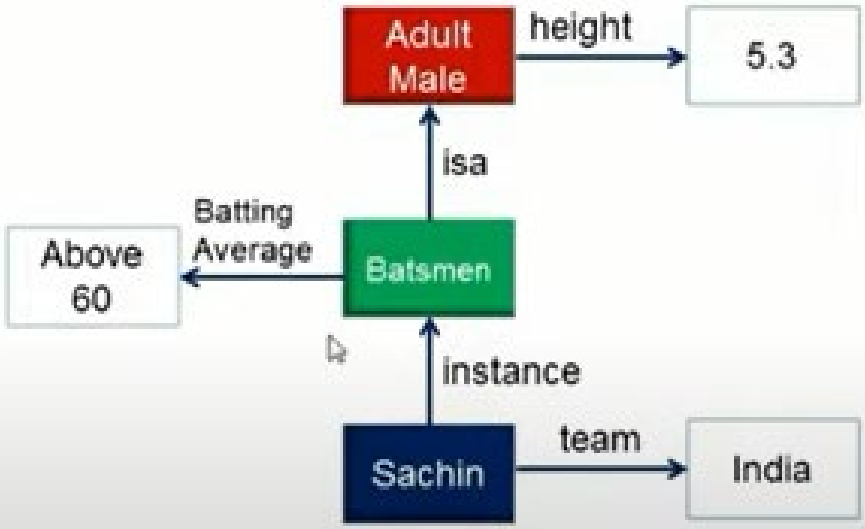
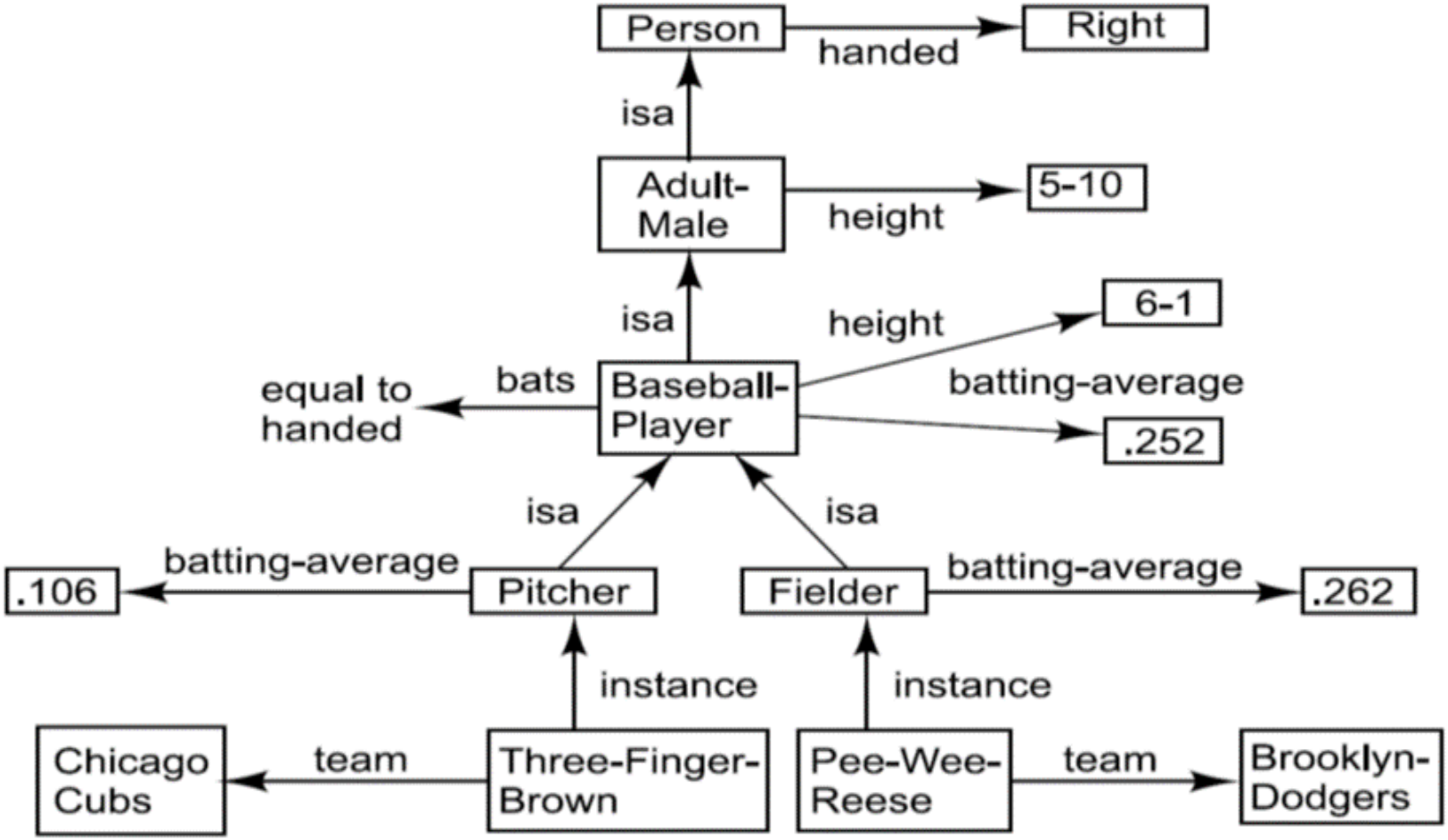


Fig: Inheratable Knowledge Representation

- These attributes are important because they support property inheritance.
- instance – indicates Class Membership
- isa – indicates class inclusion






LOVELY PROFESSIONAL UNIVERSITY
PUNJAB (INDIA)
L
P
U

Relationship Among Attributes

- The attributes we use to describe object are themselves act as entity.
- The relationship between attributes of an object, is independent of specific knowledge they encode, & may hold properties like
  - Inverse
  - Existence in a isa hierarchy
  - Technique for reasoning about values.
  - Single value attributes.

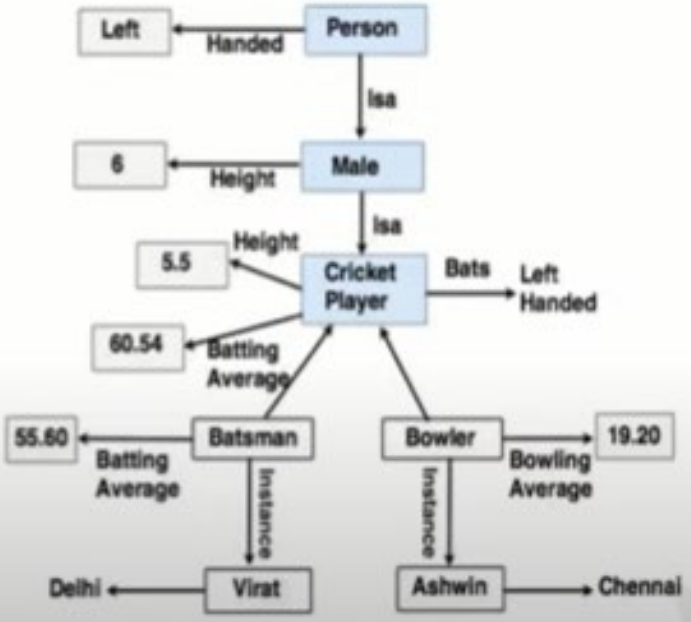
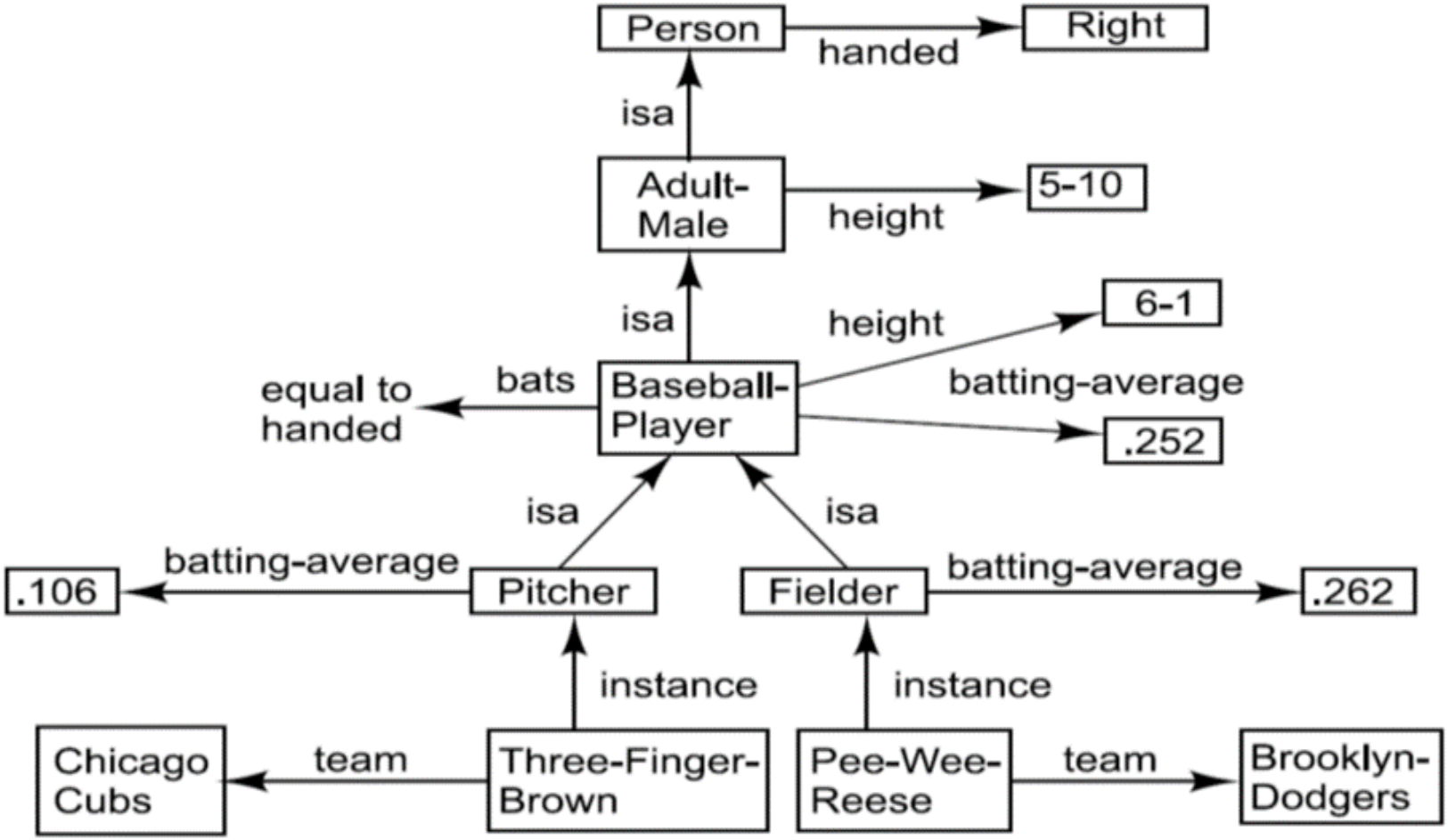


Fig: Inheratable Knowledge Representation

Inverses :

This is about consistency check, while a value is added to one attribute.
The entities are related to each other in many different ways. The figure shows attributes (isa, instance, and team), each with a directed arrow, originating at the object being described and terminating either at the object or its value.



Inverse: There are two ways of realizing this:

(a) first, represent two relationships in a single representation; e.g., a logical representation, team(Pee-Wee-Reese, Brooklyn–Dodgers), that can be interpreted as a statement about Pee-Wee-Reese or Brooklyn–Dodger.

second, use attributes that focus on a single entity but use them in pairs, one the inverse of the other; for e.g., one,

Dodgers , and the other, team = Pee-Wee-Reese, ...

(b) This second approach is followed in semantic net and frame-based systems, accompanied by a knowledge acquisition tool that guarantees the consistency of inverse slot by checking, each time a value is added to one attribute then the corresponding value is added to the inverse.

Existence in an "isa" hierarchy :

This is about **generalization-specialization**, like, classes of objects and specialized subsets of those classes. There are attributes and specialization of attributes

Example: the attribute "height" is a specialization of general attribute

"physical-size" which is, in turn, a specialization of "physical-attribute". These generalization-specialization relationships for attributes are important because they support inheritance.

Relationship Among Attributes

- **Technique for reasoning about values.**
  - Some constraints has to be followed by the value of an attributes.
  - Information about type of value.
  - Eg Value of height must be a number measured in unit of length.
  - Constraints on the value
  - Eg Age of person can not be greater than his parents age.
- **Single value attributes**
  - This is about specific attribute that is guaranteed to take unique value.
  - Eg Baseball player can at a time have single height.

Choosing Granularity of Representation

- High-level facts may not be adequate for inference.
- Low-level primitives may require a lot of storage.
- At what level of detail the knowledge should be represented & what should be our primitive.
- Eg Depending on the level of audience we should represent knowledge.

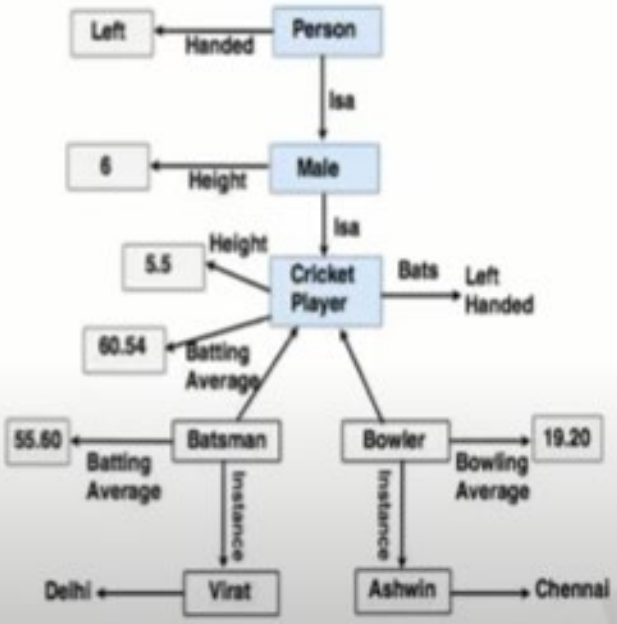


Fig: Inheratable Knowledge Representation

- **Example :**

Consider the following fact :

Raju spotted Rani.

We could represent this as :

Spotted (agent(Raju),object(Rani))

Now we have question :

Who spotted Rani? ----- We can discover the answer as Raju

Did Raju see Rani? ----- We cannot discover the answer……

In order to discover the answer ,we have add other facts such as :

Spotted (x,y)→ Saw(x,y)

- An alternative solution to this problem is to represent the fact that spotting is really a special type of seeing explicitly in the representation of fact.

  We might write something as :

  Saw (agent(Raju),object(Rani),timespan(briefly))

- In this representation we have broken the idea of spotting apart into more primitive concepts of seeing and timespan.

Representing Set of objects

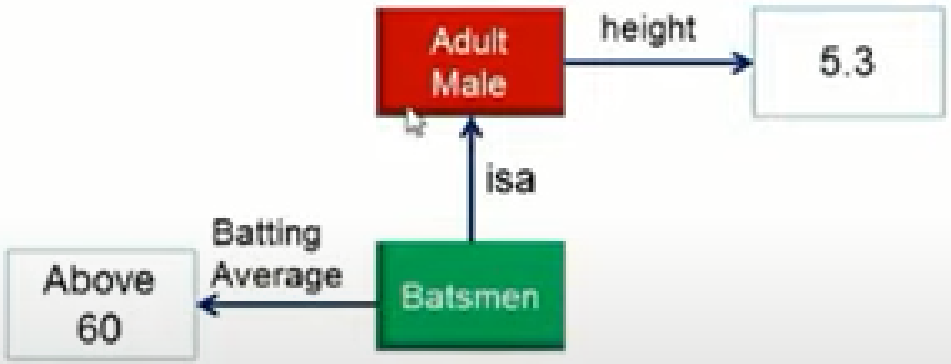
- There are certain properties of an object which are true as member of set but not as individual.
- Eg A person is very strict at a company but he is very calm at his home.
- Eg Page number written on page of book.

- How should set of objects be represented?
- It is very important to represent sets of objects because of two reasons :
  - **Reason 1:** There are some properties that are true for the sets but not true for the individual member of a set.
  - **Example :**

    Consider the assertions :
    There are more sheep than people in Australia.
    English speakers can be found all over the world.

- **Reason 2 :** It is important to represent sets of objects is that if a property is true for all (or even most) elements of a set, then it is more efficient to associate it once with the set rather than to associate it explicitly with every element of the set.

- **Example :**



Finding right structure as needed

- Selection of appropriate structure is necessary in order to represent knowledge effectively.

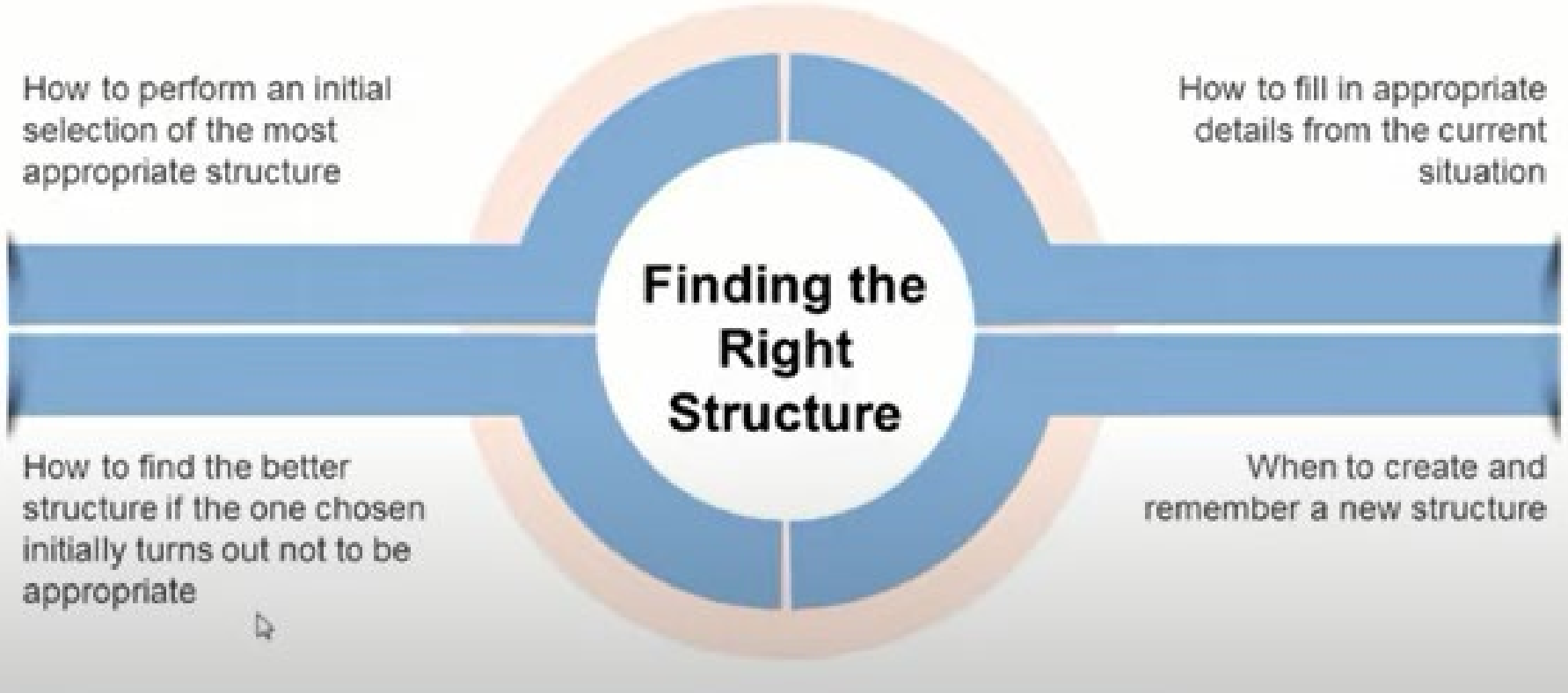
Following problems were raised while selecting appropriate structure.

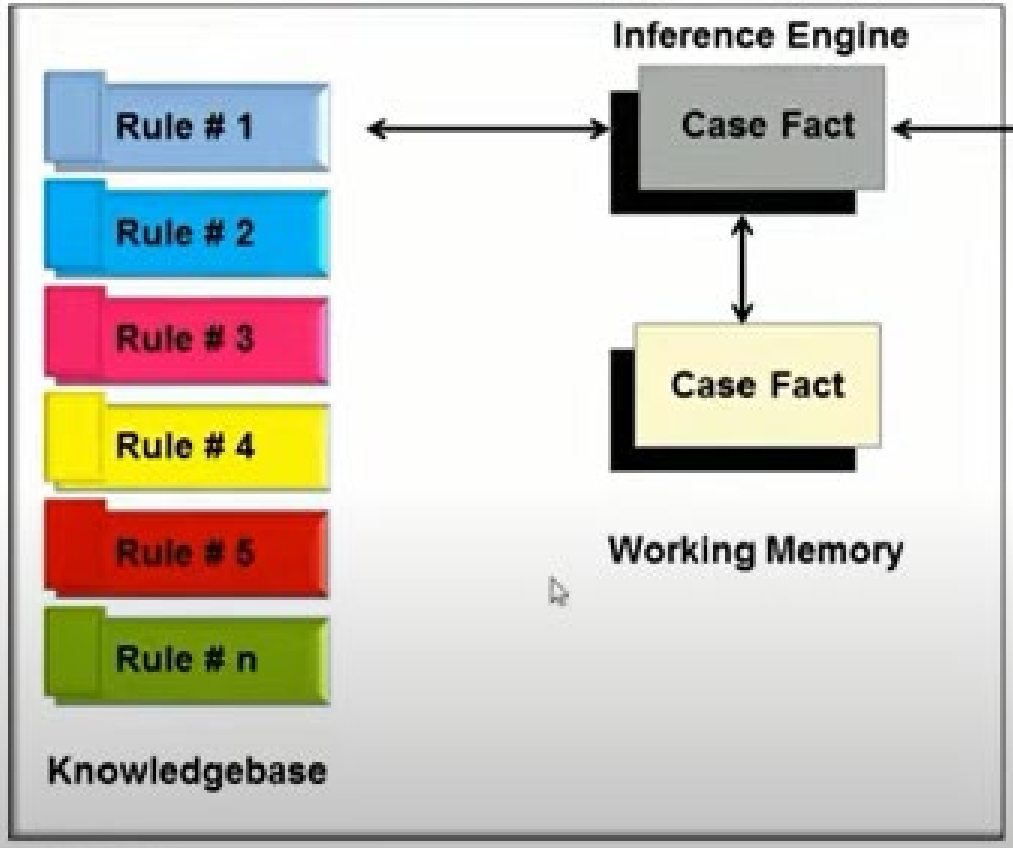
- How to perform initial selection of most appropriate structure.
- How to fill appropriate details from current situation.
- How to find better structure if one chosen is not appropriate.
- What to do if none of the available structure is appropriate

Finding Right Structure

It requires, selecting an initial structure and then revising the choice. While doing so, it is necessary to solve following problems :

- how to perform an initial selection of the most appropriate structure.
- how to fill in appropriate details from the current situations.
- how to find a better structure if the one chosen initially turns out not to be appropriate.
- what to do if none of the available structures is appropriate.
- when to create and remember a new structure





- Given a large amount of knowledge stored in a database, How can relevant parts be accessed when they are needed?
- Problem of matching rules against state descriptions during the problem solving process.