KNOWLEDGE REASONING & AI ETHICS

Prof.S.N.Panchbhai

EXAMINATION SCHEME AND CREDITS

BTAIHM503

Continuous Assessment: 20 Marks

Mid Semester Exam:20 Marks

End Semester Exam: 60 Marks (Ideally duration 03 hrs.)

COURSE OBJECTIVES

1. To provide a strong foundation of fundamental basics of knowledge reasoning & AI ethics.

2. To provide awareness and understanding of knowledge reasoning.

3. To impart knowledge about AI ethics.

COURSE OUTCOME

CO1: Apply the knowledge and reasoning-based concepts.

CO2: Specify and identify the logical agents.

CO3: Apply Probabilistic Reasoning & Uncertainty along with rules.

CO4: Understand the human psychology and social ethics to use AI

CO5: Apply virtualization concepts and various cloud services to design, develop and deploy cloud applications.

SYLLABUS

Unit 1: Knowledge & Reasoning [7 Hours]

Knowledge representation issues, Representation & mapping, Approaches to knowledge representation, semantic nets- frames and inheritance, Knowledge-Based Agents, The Wumpus World, Logic, Propositional Logic: A Very Simple Logic, Propositional Theorem Proving, Effective Propositional Model Checking, Agents Based on Propositional Logic

Unit 2: Logical Agents [7 Hours]

Using predicate logic: Representing simple fact in logic, Representing instant & ISA relationship, Computable functions & predicates, Resolution, Natural deduction. Representing knowledge using rules: Procedural versus declarative knowledge, Logic programming, forward versus backward reasoning, Matching, Control knowledge.

SYLLABUS

First-order logic: Representation Revisited Syntax and Semantics of First-Order Logic, Knowledge Engineering in First-Order Logic Inference in first-order logic, propositional vs. first-order inference, unification & lifts forward chaining, Backward chaining, Resolution

Unit 3: Probabilistic Reasoning & Uncertainty **[7 Hours]**

Quantifying Uncertainty, Acting under Uncertainty, Basic Probability Notation, Inference Using Full Joint Distributions, Independence, Bayes' Rule, and Its Use, The Wumpus World Revisited, Probabilistic Reasoning, Representing Knowledge in an Uncertain Domain, The Semantics of Bayesian Networks, Efficient Representation of Conditional Distributions Exact Inference in Bayesian Networks, Approximate Inference in Bayesian Networks, Relational and First-Order Probability Models, and Other Approaches to Uncertain Reasoning.

SYLLABUS

Unit 4: Introduction to AI Ethics **[7Hours]**
Artificial intelligence, ways of implementing AI, Advantages and disadvantages of AI, Definition of morality and ethics, Descriptive Ethics, Normative Ethics, Meta-ethics, Applied Ethics, Impact on society, Impact on human psychology, Impact on the legal system, impact on Environment and planet, impact on trust (privacy issues), challenges of AI and data governance, Ethical implications and responsibilities.

Unit 5: Ethical initiatives in the field of artificial intelligence **[7ours]**
International ethical initiatives, Autonomous systems, Ethical harms, Machine Ethics, Artificial moral agents Singularity, AI standard and regulation, IEEE 'human standards' with implications for AI, Ethics in military use of AI: use of weapons, regulations governing an AWS, Ethical Arguments for and Against AI for Military Purposes.

CONTENTS:

- ✓ Knowledge-based agent (KBA)
- ✓Knowledge representation
- ✓What to represent
- ✓Types of knowledge
- ✓Relation between knowledge and intelligence
- ✓AI knowledge cycle
- ✓Approaches to knowledge representation
- ✓Requirements of knowledge representation systems
- ✓Knowledge representation issues

KNOWLEDGE-BASED AGENT (KBA):

- ✓ KBA are those agents which can combine **knowledge with current percepts** to **Infer** hidden aspects of the current state.

- ✓KBA is an agent which has **knowledge with intelligence.**

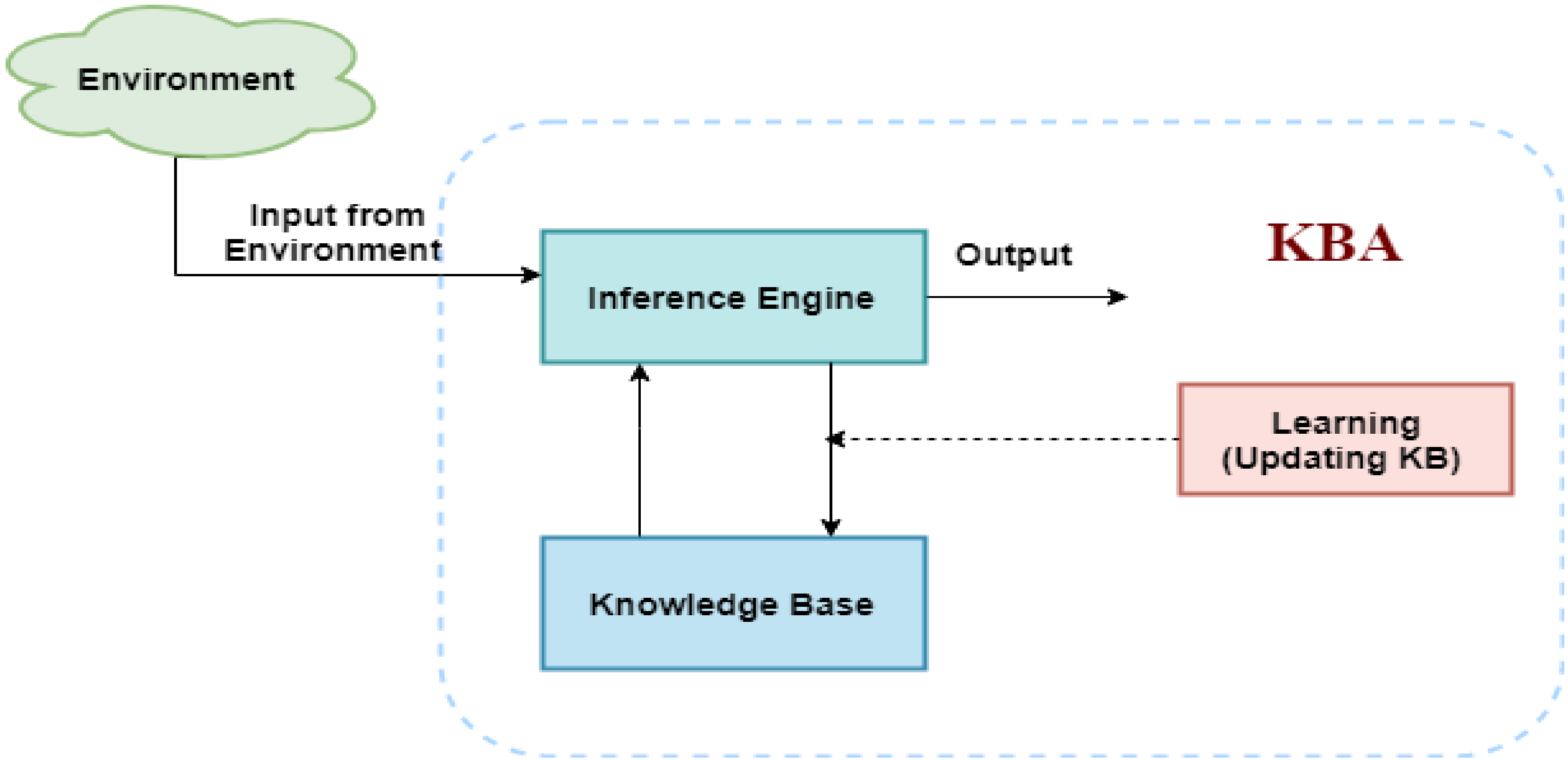
- ✓ Knowledge-based agents have the capability of :

  maintaining an internal state of knowledge, reason over that knowledge, update their knowledge after observations and take action.

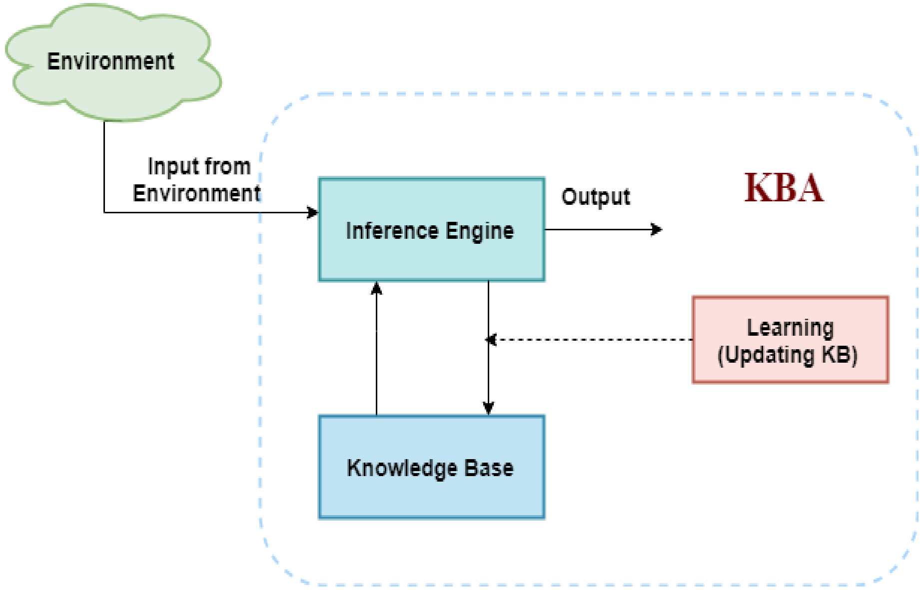
KNOWLEDGE-BASED AGENT (KBA):

- ✓A KBA must be able to do :
  1. Represents the state, actions
  2. Incorporate new percepts
  3. An agent can update the internal representation of the world
  4. An agent can deduce the internal representation of the world
  5. An agent can deduce appropriate actions.

KNOWLEDGE-BASED AGENT (KBA):



KNOWLEDGE-BASED AGENT (KBA):



✓ The knowledge-based agent is composed of two parts

1. Knowledge base
2. Inference system

1. **Knowledge base:**

- Knowledge-base is a central component of a knowledge-based agent, it is also known as KB
- It is a collection of sentences
- These sentences are expressed in a language which is called a knowledge representation language
- The Knowledge-base of KBA stores facts about the world.
- Knowledge-base is required for updating knowledge for an agent to learn with experiences and take action as per the knowledge.

KNOWLEDGE-BASED AGENT (KBA):

2. **Inference System:**

- Inference means deriving new sentences from old ones.
- Inference system allows us to add a new sentence to the knowledge base.
- A sentence is a proposition about the world
- Inference system applies logical rules to the KB to deduce new information.
- Inference system generates new facts so that an agent can update the KB.
- An inference system works mainly in two rules which are given as:

  1. Forward chaining

  2. Backward chaining

OPERATIONS PERFORMED BY KBA:

1. **TELL:** This operation tells the knowledge base what it perceives from the environment.

2. **ASK:** This operation asks the knowledge base what action it should perform.

1. **Perform:** It performs the selected action.

LEVELS OF KBA:

1. Knowledge level
2. Logical Level
3. Implementation level

LEVELS OF KBA:

1. **Knowledge Level:**
   - Knowledge level is the first level of a knowledge-based agent, and in this level, we need to specify what the agent knows, and what the agent's goals are. With these specifications, we can fix its behavior.

2. **Logical Level:**
   - At this level, sentences are encoded into different logics. At the logical level, an encoding of knowledge into logical sentences occurs.

3. **Implementation Level:**
   - This is the physical representation of logic and knowledge. At the implementation level, agent perform actions as per logic and knowledge level.

KNOWLEDGE REPRESENTATION

- ✓ Humans are best at understanding, reasoning, and interpreting knowledge. How machines do all these things comes under knowledge representation and reasoning.

- ✓ It is a way that describes how we can represent knowledge in artificial intelligence.

- ✓ Knowledge representation, represents the information from the real world for a computer to understand and then utilize this knowledge to solve complex real-life problems like communicating with human beings in natural language.

KNOWLEDGE REPRESENTATION

- ✓ Knowledge representation is not just storing data into some database, but it also enables an intelligent machine to learn from that knowledge and experiences so that it can behave intelligently like a human.

- ✓ It is a study of how knowledge about the world can be represented and what kinds of reasoning can be done with that knowledge

WHAT TO REPRESENT

- ✓ **Object:** All the facts about objects in our world domain. E.g., Guitars contain strings, and trumpets are brass instruments.
- ✓ **Events:** Events are the actions that occur in our world
- ✓ **Performance:** It describes behavior that involves knowledge about how to do things.
- ✓ **Meta-knowledge:** It is knowledge about what we know.
- ✓ **Facts:** Facts are the truths about the real world and what we represent.
- ✓ **Knowledge-Base:** It is the main component of any human, i.e., having a knowledge base. This refers to a group of information regarding any discipline, field, etc.

TYPES OF KNOWLEDGE

1. Declarative knowledge
2. Procedural Knowledge
3. Meta-Knowledge
4. Heuristic Knowledge
5. Structural Knowledge

TYPES OF KNOWLEDGE

1. Declarative Knowledge:

- What is known about the problem
- Also called descriptive knowledge
- Tells us facts: What things are
- Includes: Concepts, facts, objects

TYPES OF KNOWLEDGE

2. Procedural Knowledge:

- Describes how to solve problems
- Also known as Imperative knowledge
- Provides direction on how to do something
- Can be directly applied to a task
- Includes: Rules, Strategies, procedures

TYPES OF KNOWLEDGE

3. Meta Knowledge:

- Describe knowledge about another knowledge
- Used to pick other knowledge that is best suited for solving problems

TYPES OF KNOWLEDGE

4. Heuristics Knowledge:

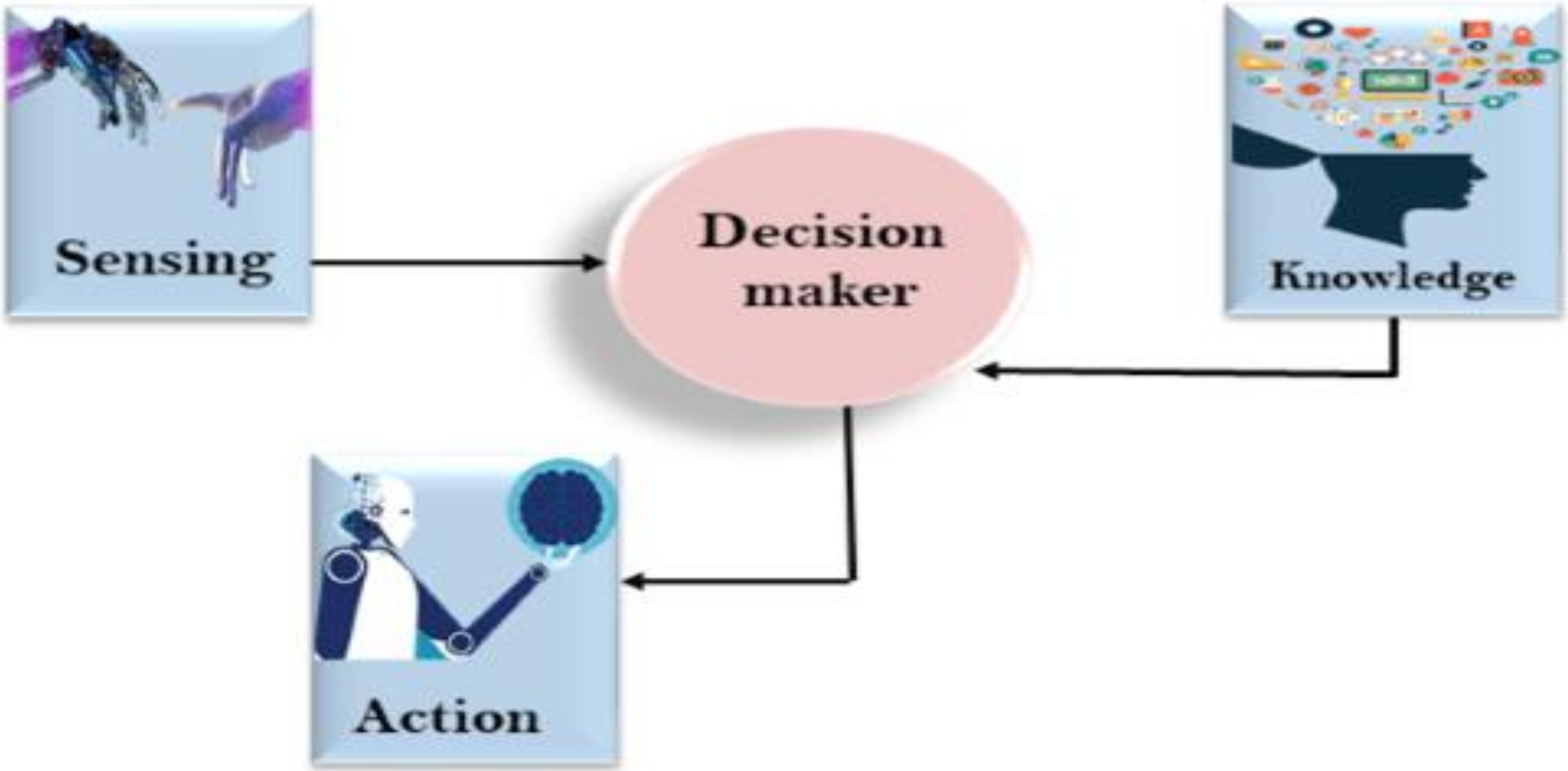
- It is representing knowledge of some experts in a field or subject
- Describes the thumb rules that guide the reasoning process
- Previous experiences, approaches

TYPES OF KNOWLEDGE

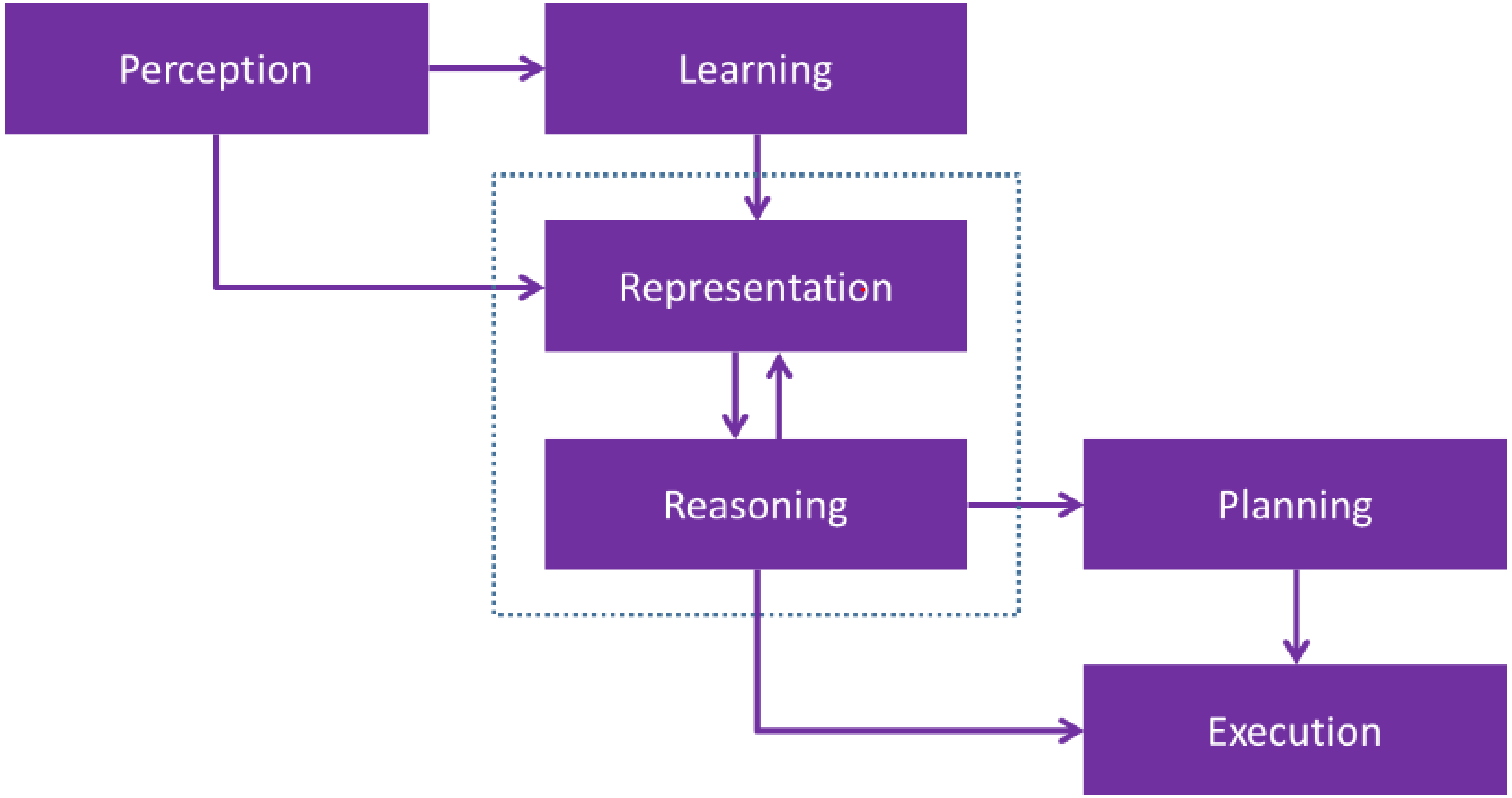
5. Structural Knowledge:

- This type of knowledge helps establish relationships between concepts or objects and their description, acting as the basic form of knowledge to solve real-world problems.

RELATION BETWEEN KNOWLEDGE AND INTELLIGENCE



AI KNOWLEDGE CYCLE



APPROACHES TO KNOWLEDGE REPRESENTATION

1. Simple – Relational approach
2. Inheritable knowledge
3. Inferential knowledge
4. Procedural knowledge

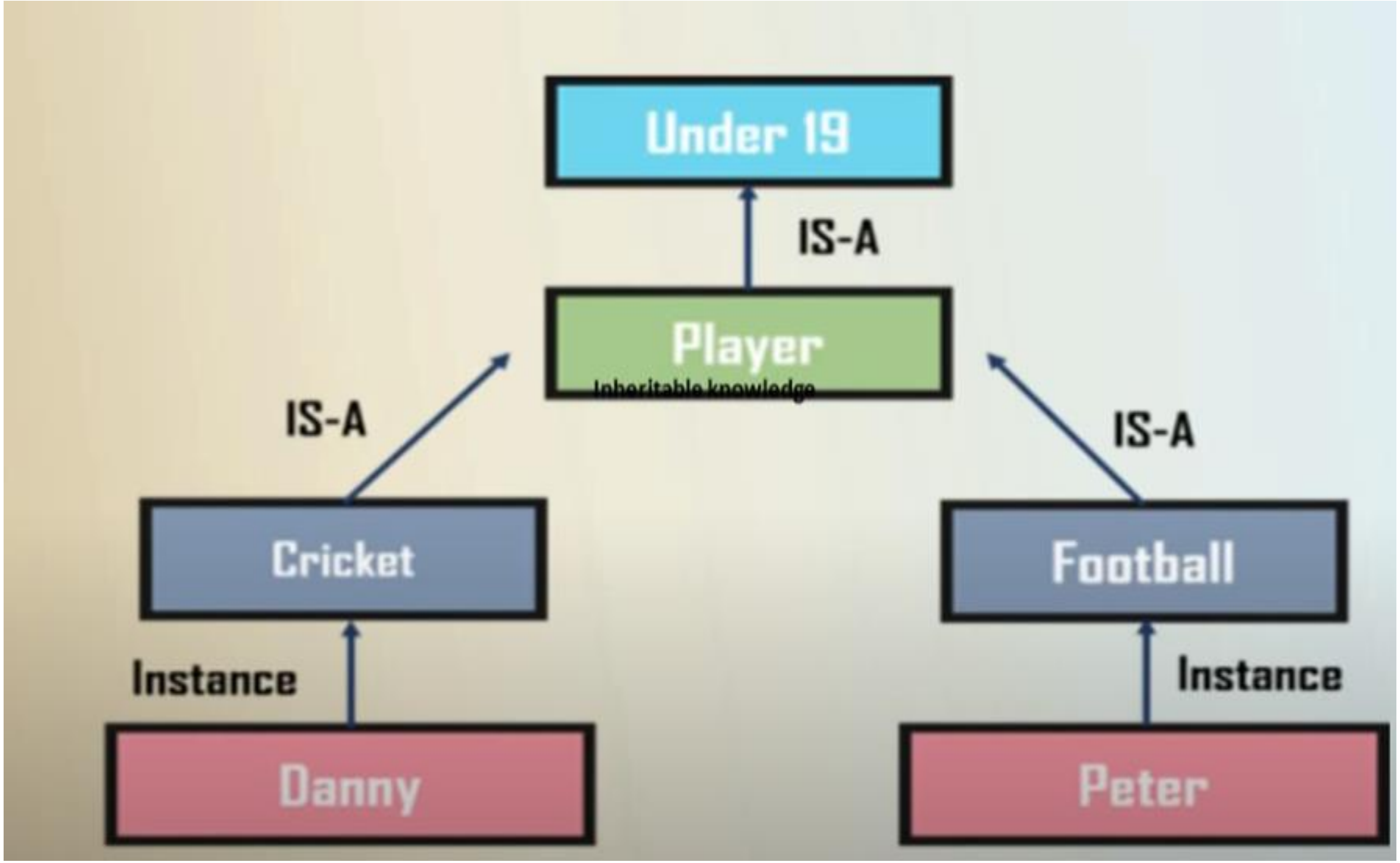
1.SIMPLE RELATIONAL APPROACH

- ✓ It is the simplest way to represent the facts.
- ✓ This is a relational method of storing facts.
- ✓ This method helps in storing facts where each fact regarding an object is provided in columns.
- ✓ Low opportunity for inferences
- ✓ This approach is used in DBMS (database management systems.)

Developer	Language	Experience
Dennis Ritchie	C	15 Years
Guido van Rossum	Python	12 Years

2.INHERITABLE KNOWLEDGE APPROACH

- ✓ In this approach, all data must be stored in the hierarchy of classes.
- ✓ In this approach we apply inheritance property
- ✓ This approach shows the relation between instance and class, and it is called instance relation
- ✓ In this approach, objects and values are represented in boxed nodes.
- ✓ We use arrows which points from objects to their values.


Under 19
IS-A
Player
Inheritable knowledge
IS-A
IS-A
Cricket
Football
Instance
Instance
Danny
Peter

3. 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 : Albert is a man
  All men are mortal

  Then inferential knowledge can be represented as ;

  man (Albert)

  $$\forall X = man\ (X) -> mortal\ (X)$$

4. PROCEDURAL KNOWLEDGE

- ✓ It uses small programs and codes which describe how to do specific things, and how to proceed
- ✓ Most important rule used is the if-then rule.
- ✓ We can easily represent heuristic or domain-specific knowledge using this approach
- ✓ Various coding languages are used such as LISP language and Prolog language.

KNOWLEDGE REPRESENTATION ISSUES:

✓ Several issues must be considered when representing various kinds of real-world knowledge

1. Important Attributes
2. Relationship among attributes
3. Choosing the granularity of representation
4. Representing sets of objects
5. Finding the right structure as needed

1.IMPORTANT ATTRIBUTES

✓ Is any attribute of objects so basic that they occur in almost every problem domain?

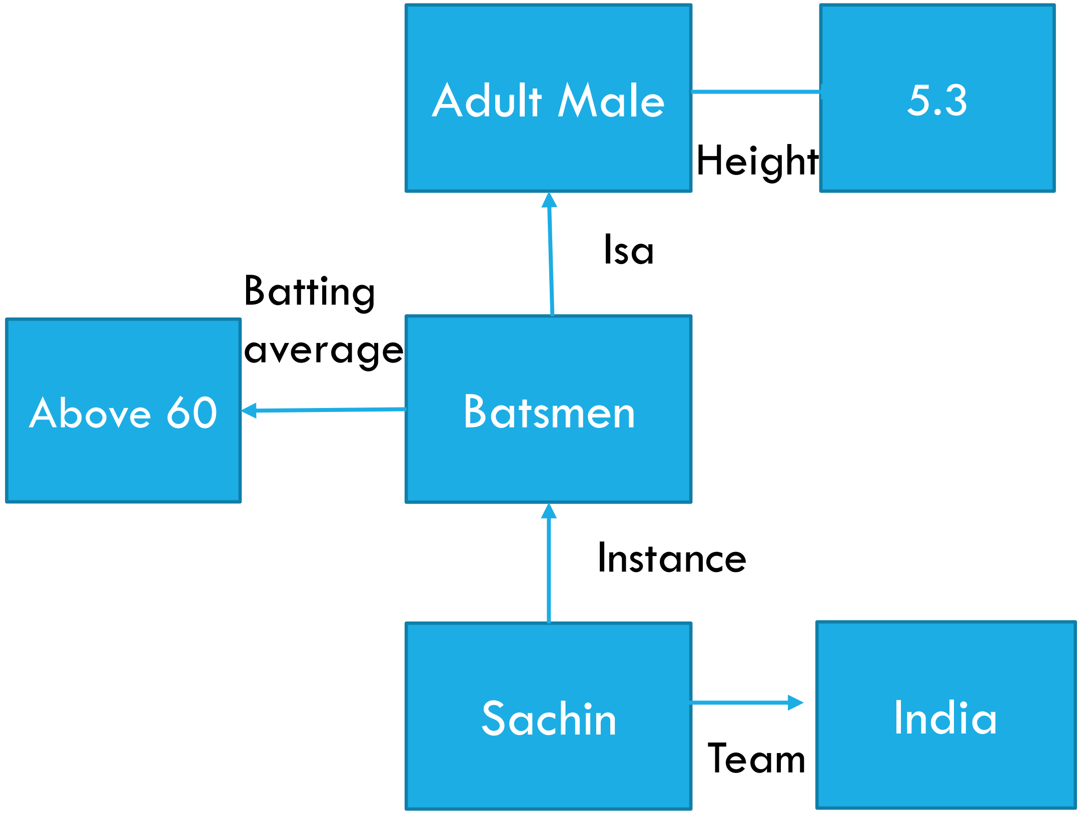
✓ If there are, we need to make sure that they are handled appropriately.

✓ There are two attributes:

1. Instance
2. Isa

These attributes are important because they support property inheritance.

- Instance: Indicates class membership
- Isa: indicates class inclusion

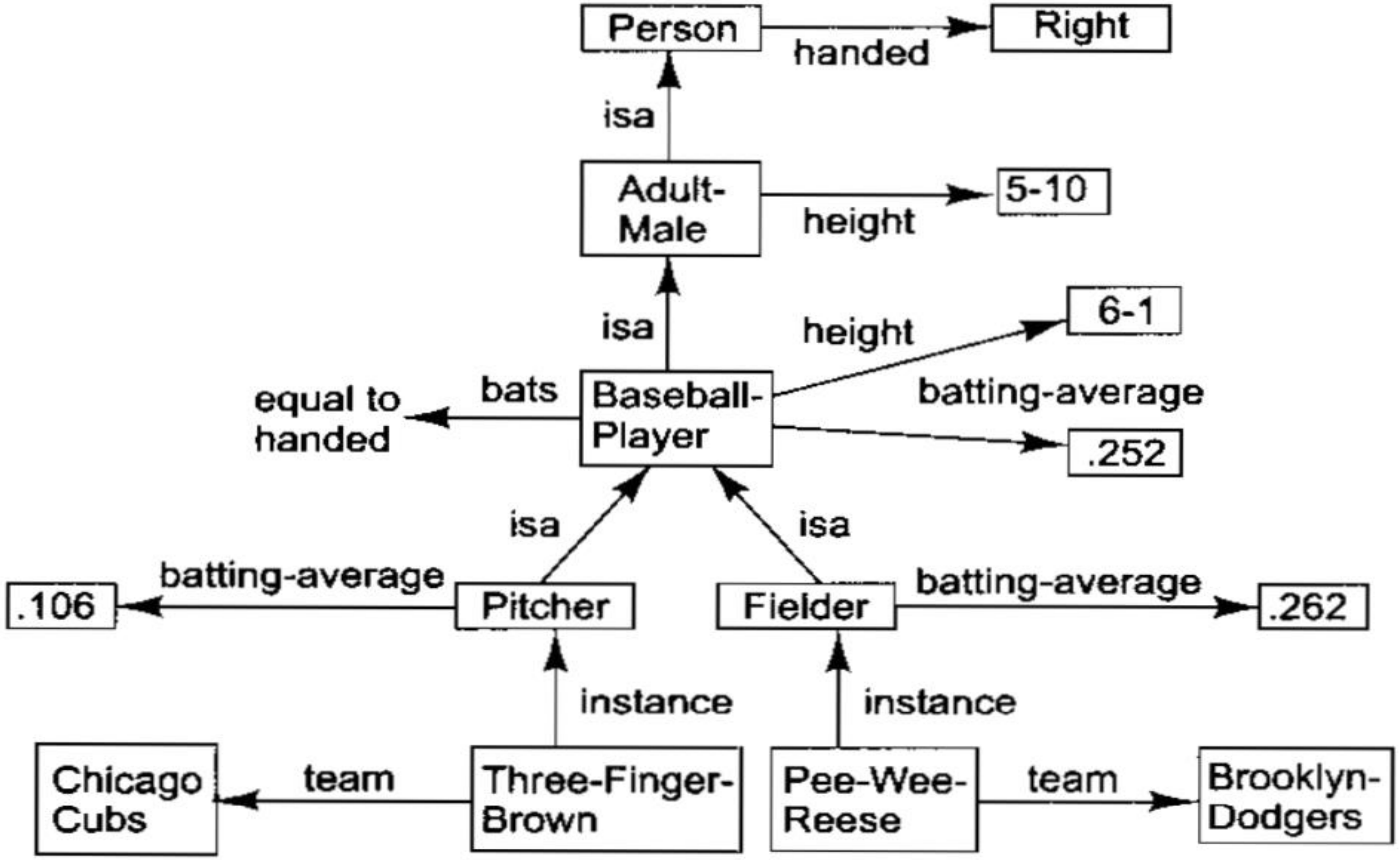


2. RELATIONSHIPS AMONG THE ATTRIBUTES

- ✓ Any important relationships exist among attributes of objects?
- ✓ There are four properties of attributes:

1. Inverse
2. Existence in an isa hierarchy
3. Techniques for reasoning about values
4. Single valued attributes

2. RELATIONSHIPS AMONG THE ATTRIBUTES



2. RELATIONSHIPS AMONG THE ATTRIBUTES

1.Inverse:

✓ We use the attributes instance, isa, and team

✓There are two ways to show the relationship

a) Represent two relationships in a single representation that ignores a focus

team(Pee-Wee-Reese, Brooklyn-Dodgers)

Can equally be interpreted as a statement about Pee-Wee-Reese or about the Brooklyn Dodgers (How it is actually used depends on the other assertion that a system contains)

This is used in the logical representation.

2. RELATIONSHIPS AMONG THE ATTRIBUTES

b) To use attributes that focus on a single entity but to use them in pairs, one the inverse of the other. In this approach, we would represent the team information with two attributes:

- one associated with Pee-Wee-Reese:

    team = Brooklyn Dodgers

- one associated with Brooklyn Dodgers

    (Instance)team-members = Pee-Wee-Reese

This second approach is followed in semantic net and frame-based systems

2. RELATIONSHIPS AMONG THE ATTRIBUTES

2. 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.
- ✓ For example, the attribute "height" is a specialization of the 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.

2. RELATIONSHIPS AMONG THE ATTRIBUTES

3. Techniques for reasoning about the values:

- ✓ This is about reasoning values of attributes not given explicitly. Several kinds of information are used in reasoning, like,
- ✓ height: must be in a unit of length
- ✓ age: of person can not be greater than the age of a person's parents.

2. RELATIONSHIPS AMONG THE ATTRIBUTES

4. Single valued attributes:

- ✓ This is about a specific attribute that is guaranteed to take a unique value.

- ✓ Example: A baseball player can at times have only a single height and be a member of only one team.

3. CHOOSING THE GRANULARITY OF REPRESENTATION

- ✓ At what level should knowledge be represented?
- ✓ is there a good set of primitives in which all knowledge can be broken down?
- ✓ is it helpful to use such primitives?
- ✓Primitives are fundamental concepts such as holding, seeing, and playing…..

3. CHOOSING THE GRANULARITY OF REPRESENTATION

Example :

Fact:	Vetal spotted Vikram
Representation:	Spotted(agent(Vetal),object(Vikram))
Question 1:	Who spotted Vikram?
Question 2:	Did Vetal see Vikram? (We can not answer)

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

Spotted(x,y) → Saw (x,y)

3. CHOOSING THE GRANULARITY OF REPRESENTATION

- ✓ Alternate solution could be:
- ✓ Represent the fact that spotting is really a special type of seeing.

saw(agent(Vetal),object(Vikram),timespan(briefly))

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

4.REPRESENTING SETS OF OBJECTS

- ✓ How should a set of objects be represented?
- ✓ It is very important to represent a set of objects because of two reasons:

1. There are some properties that are true for the sets but not true for the individual member of a set.

Example: Consider the assertion:

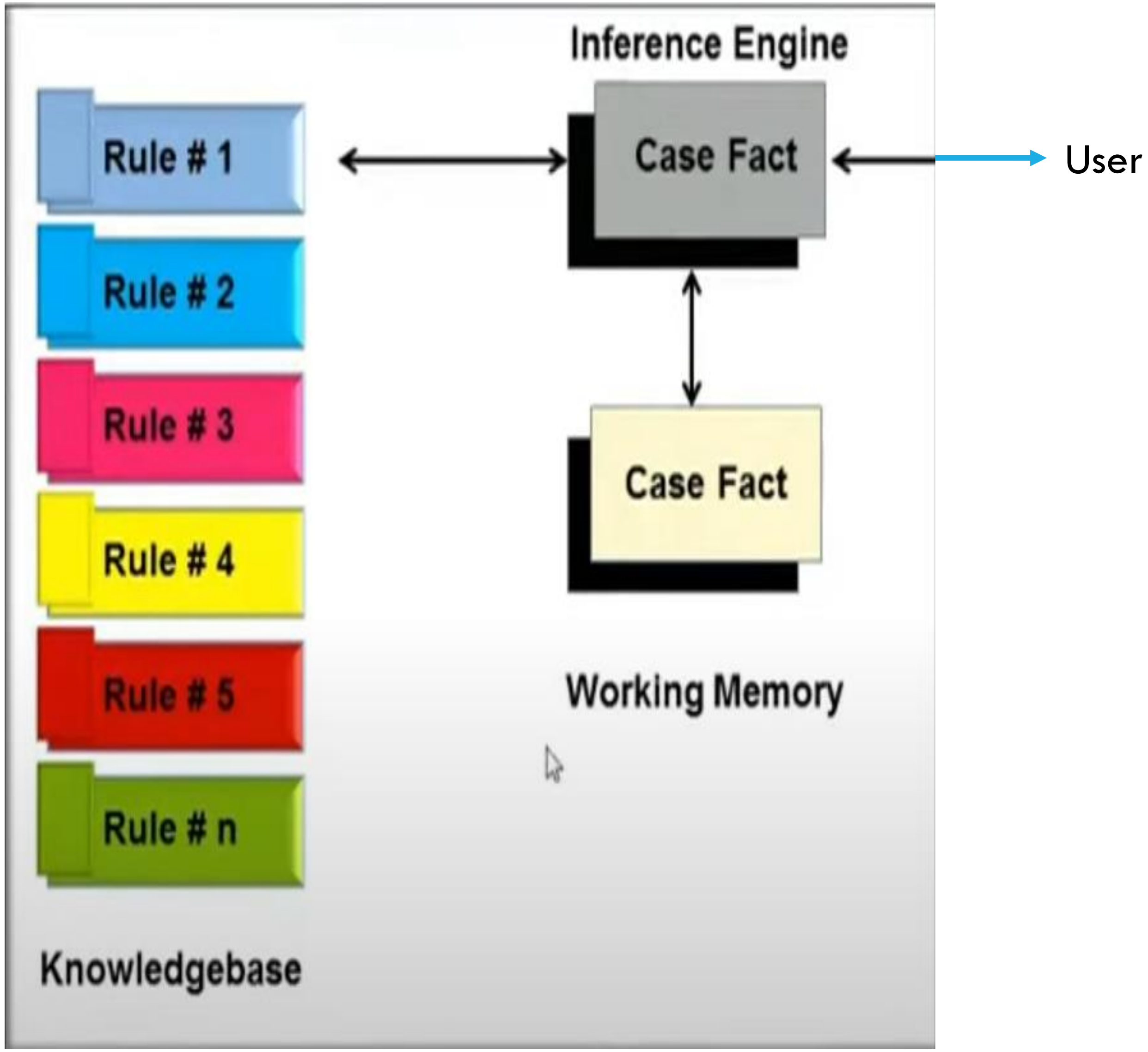
There are more sheep than people in Australia.

4.REPRESENTING SETS OF OBJECTS

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

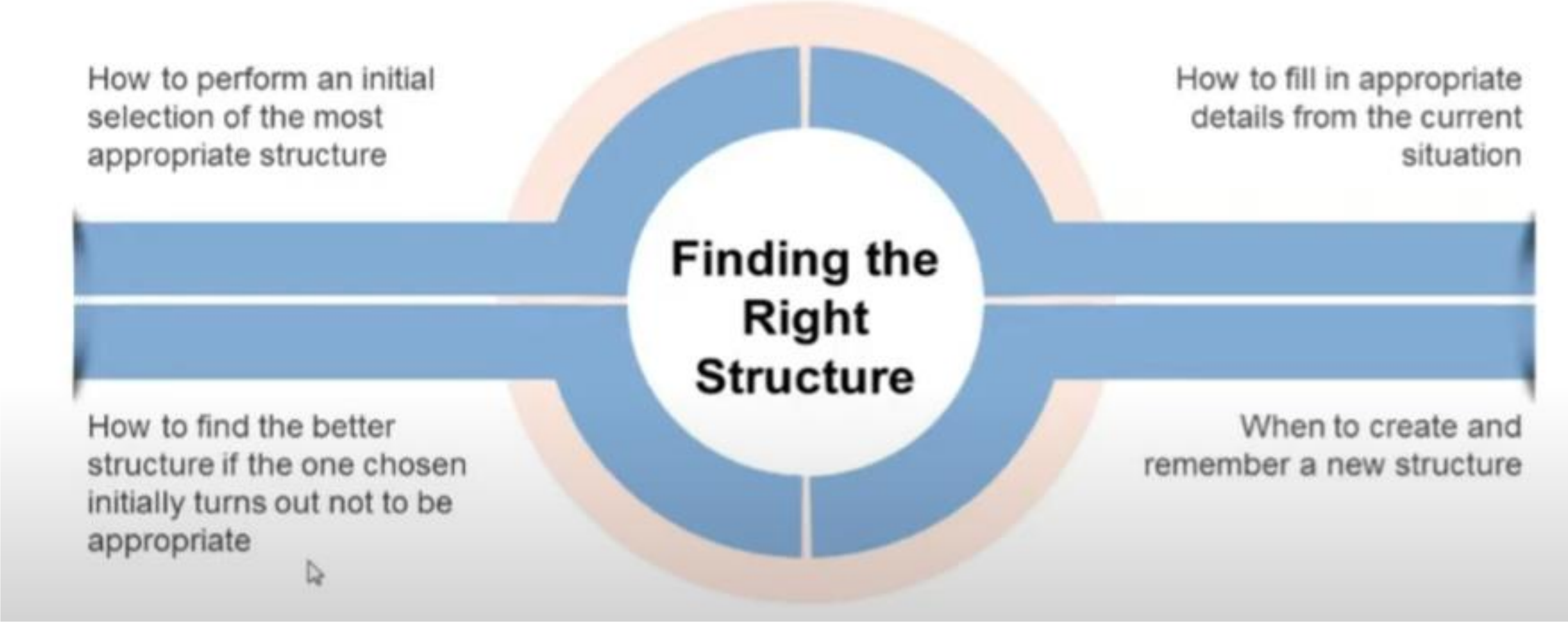
5. FINDING THE RIGHT STRUCTURE AS NEEDED

- ✓ Given a larger amount of knowledge stored in a knowledge-based, how can relevant parts be accessed when they are needed?
- ✓ This is about accessing the right structure for describing a particular situation
- ✓ Problem of matching rules against state descriptions during the problem-solving process.



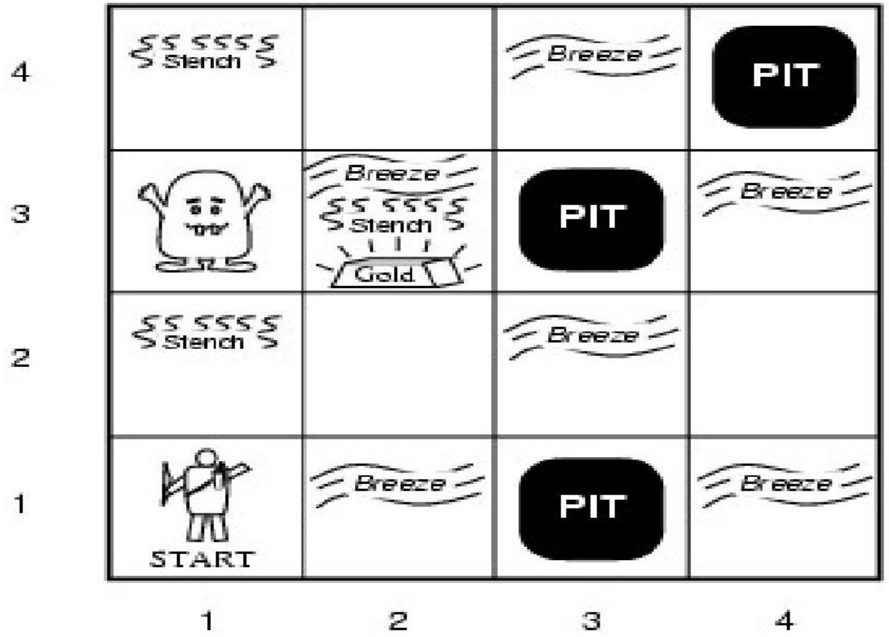
Runtime problem solving :

1. Content of working memory and content of knowledge base should match.
2. How to perform this matching and how to search for particular knowledge?



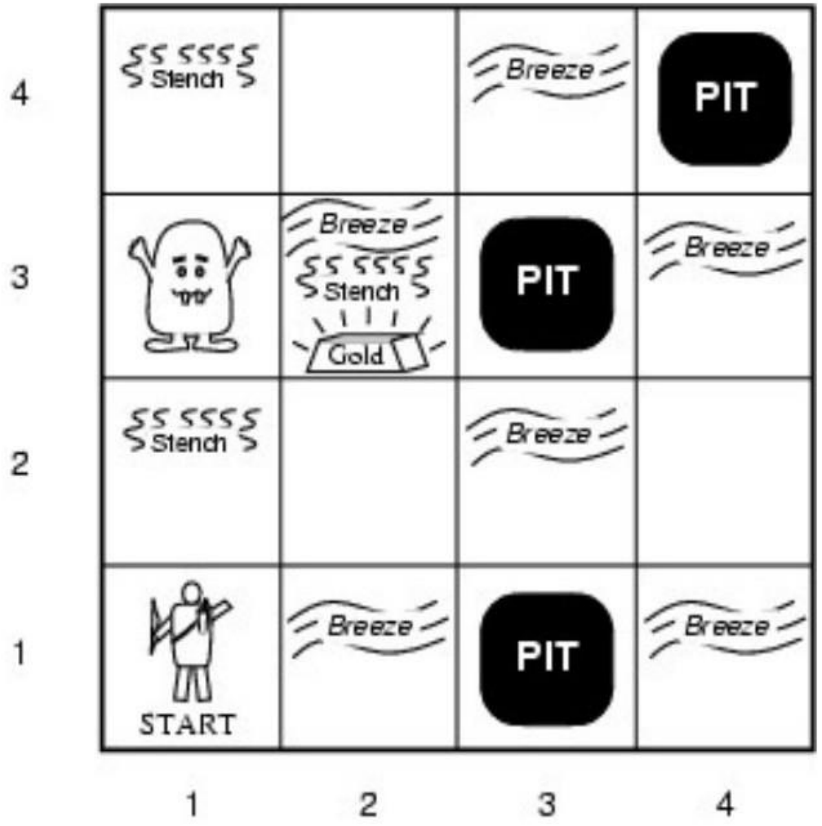
There is no good, general-purpose method for solving all these problems. Some knowledge representation techniques solve some of these issues.

WUMPUS WORLD PROBLEM



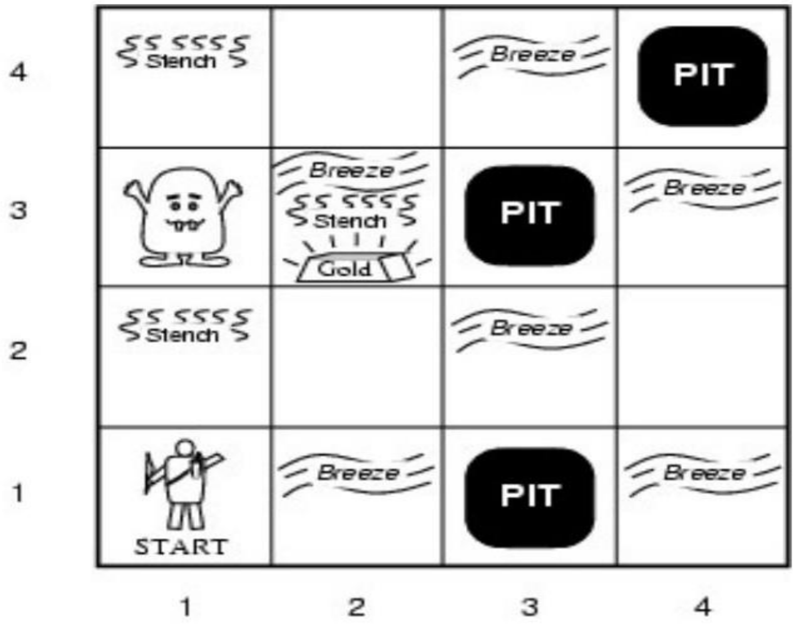
Wumpus World PEAS

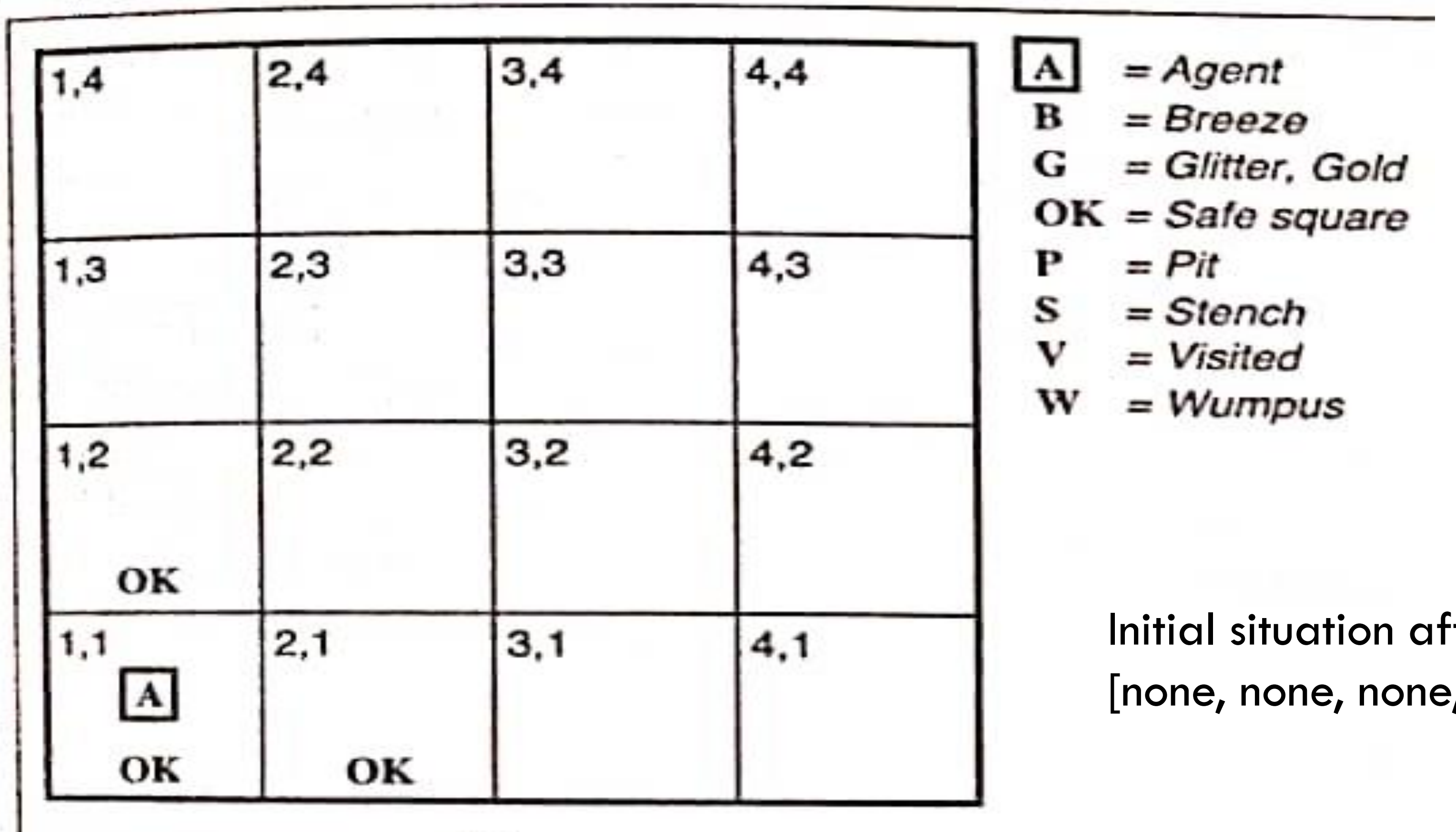
- Performance measure
  - gold +1000, death -1000
  - -1 per step, -10 for using the arrow
- Environment
  - 4x4 grid of rooms
  - Agent always starts at [1,1]
  - Wumpus and gold randomly chosen
  - Each square other than start is pit with 0.2 probability



Wumpus World PEAS

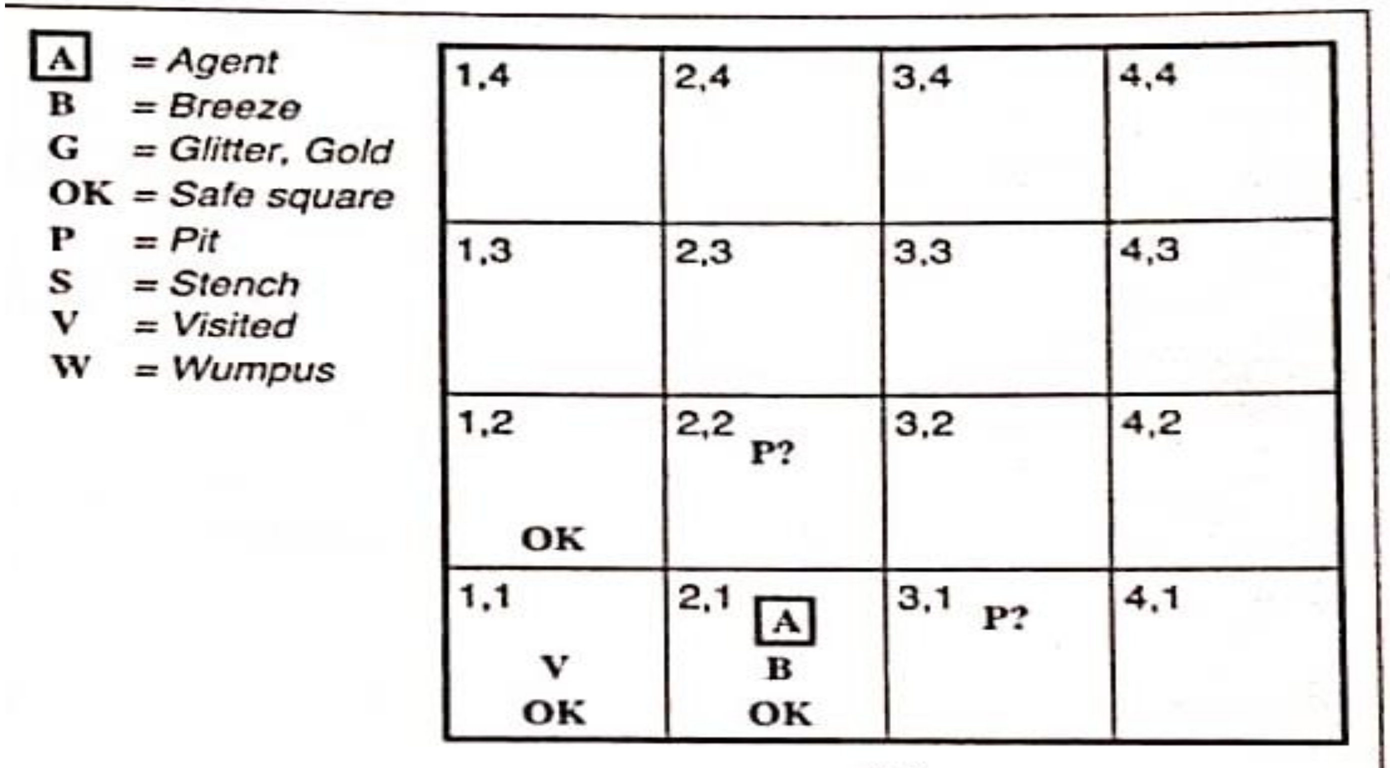
- Actuators: Left turn, Right turn, Forward, Grab, Release, Shoot, Climb
  - Shooting kills wumpus if you are facing it
  - Shooting uses up the only arrow
  - Grabbing picks up gold if in same square
  - Releasing drops the gold in same square
  - Climbing out of the cave from [1, 1]
- Sensors (percepts): Stench, Breeze, Glitter, Bump, Scream
  - Squares adjacent to wumpus are smelly
  - Squares adjacent to pit are breezy
  - Glitter iff gold is in the same square
  - Bump when agent walks into a wall
  - When wumpus is killed, it screams





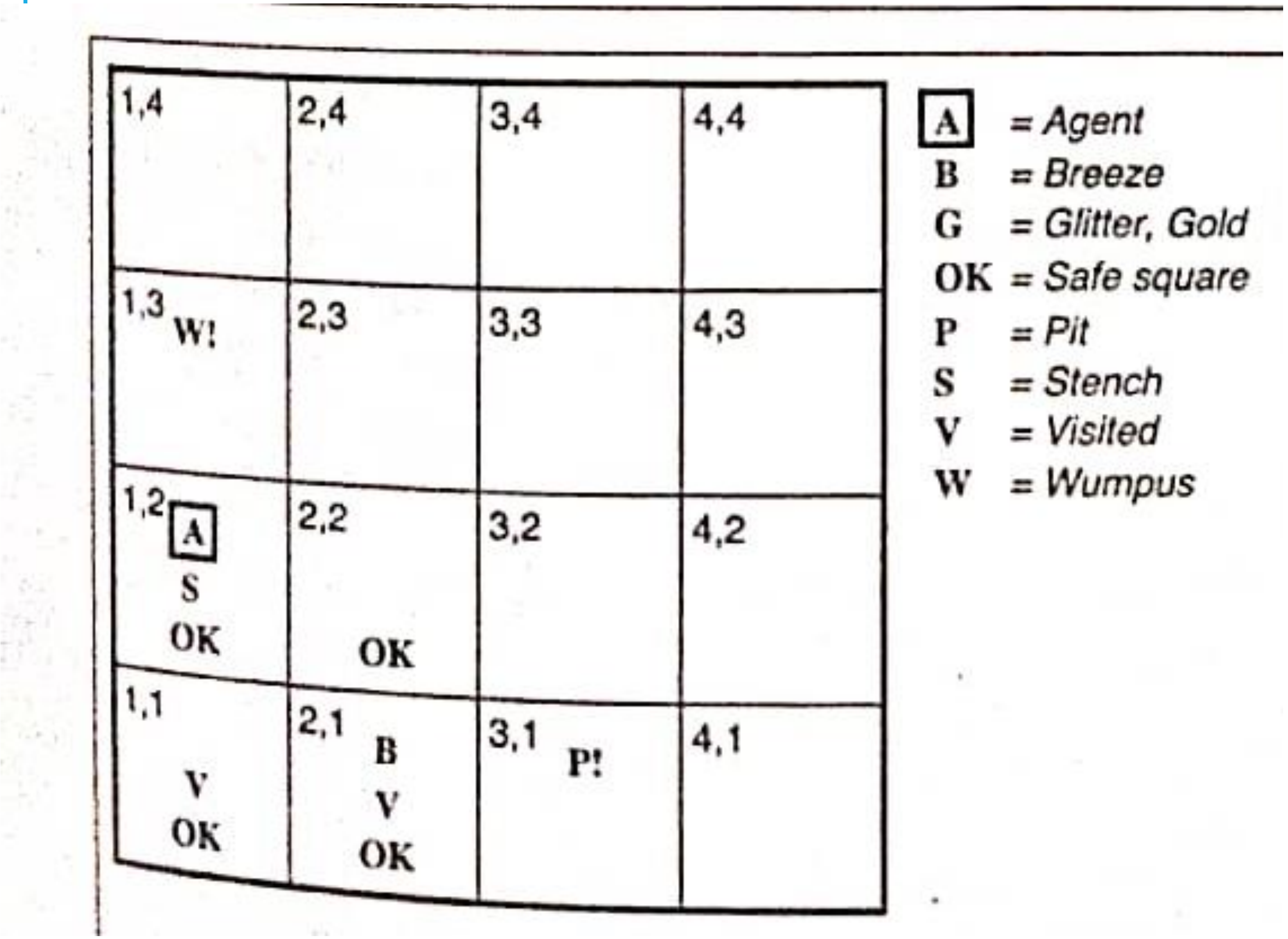
(a)

Initial situation after percept
[none, none, none, none, none]

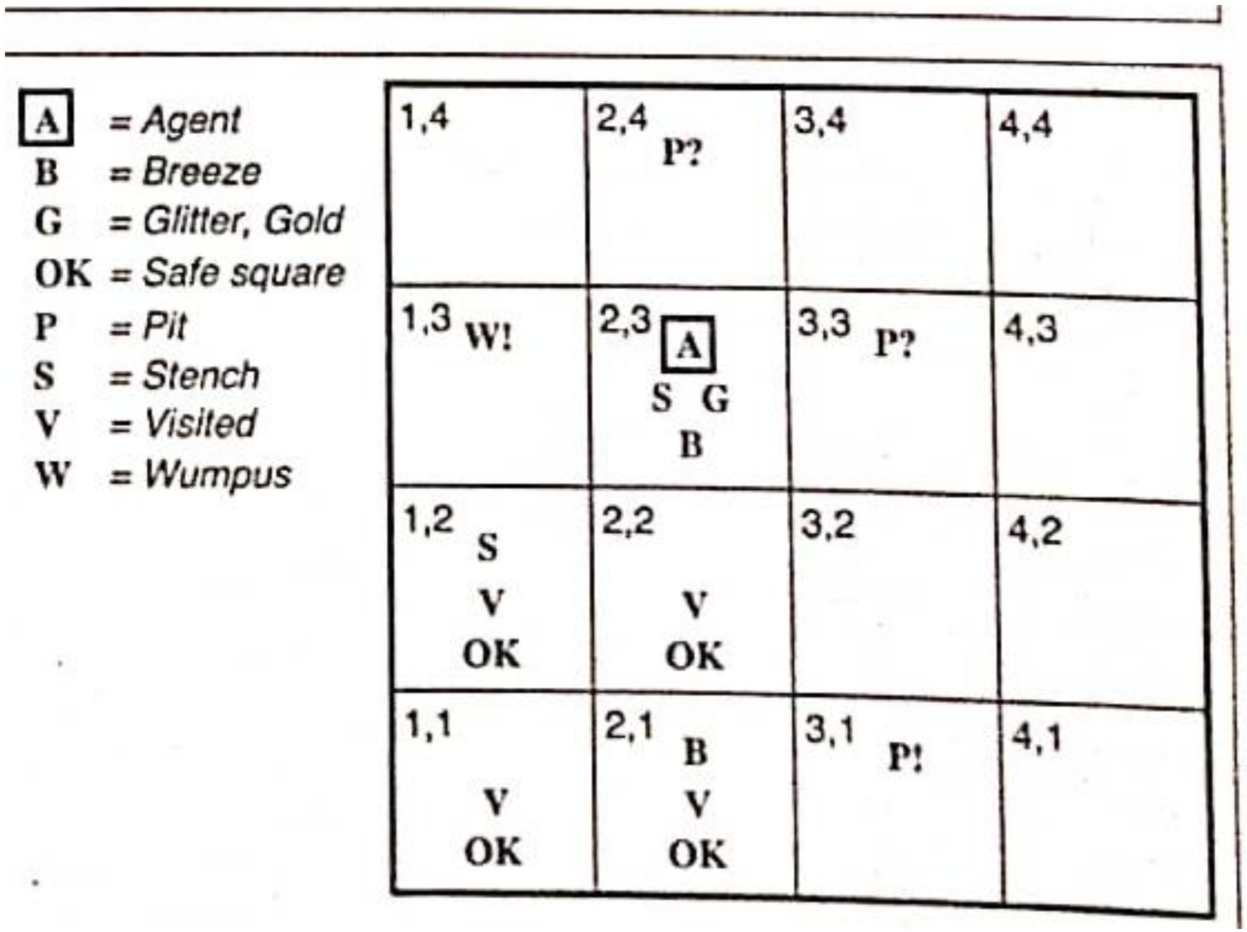


(b)

After one move percept
[none, Breeze, none, none, none]



After the third move percept [Stench, none, none, none, none]



After the fifth move percept
[Stench, Breeze, Glitter, none, none]

LOGIC:

- ✓ **Logics** are formal languages for representing information such that conclusions can be drawn
- ✓ **Syntax** defines the sentences in the language

  X + Y = 4 (This is well-formed sentence)

  X4Y+ = (This is not a well-formed sentence)
- ✓ **Semantics** define the meaning of sentences i.e. define the truth of a sentence in a world.
- ✓ When we need to be precise, we use the term **model** in place of 'Possible world'.

ENTAILMENT

- ✓ Now, we have a notion of truth, we are ready to study logical reasoning.
- ✓ Logical reasoning involves the relation of logical **entailment** between sentences – the idea that a sentence follows logically from another sentence.

$$\alpha \vDash \beta$$

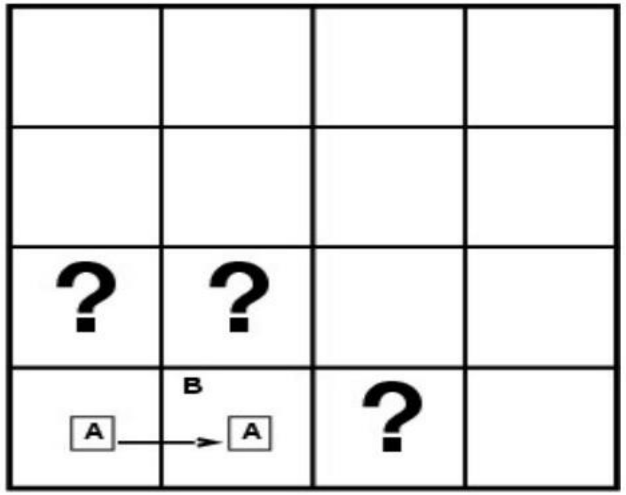
- ✓ $\alpha \vDash \beta$ if in every model α is true, β is also true

$$\alpha \vDash \beta \text{ if } M(\alpha) \subseteq M(\beta)$$

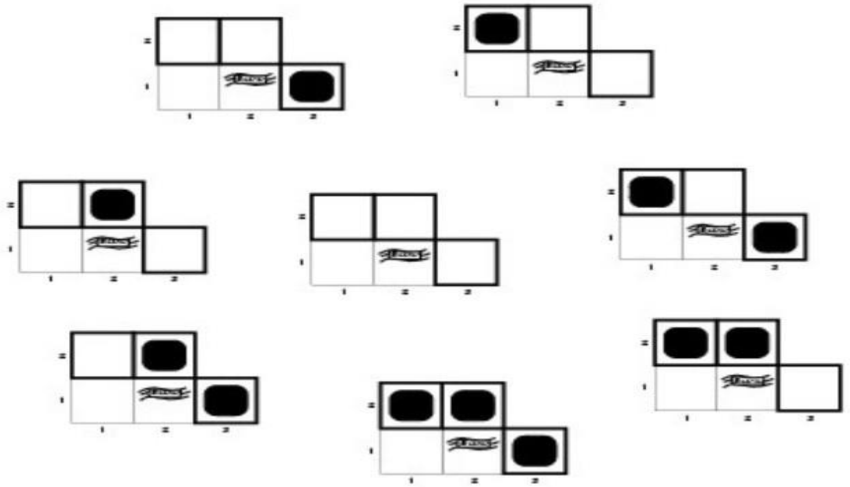
- ✓ Example (X=0) ⊨ (XY=0)
- ✓ Entailment is a relationship between sentences.

Entailment in the Wumpus World

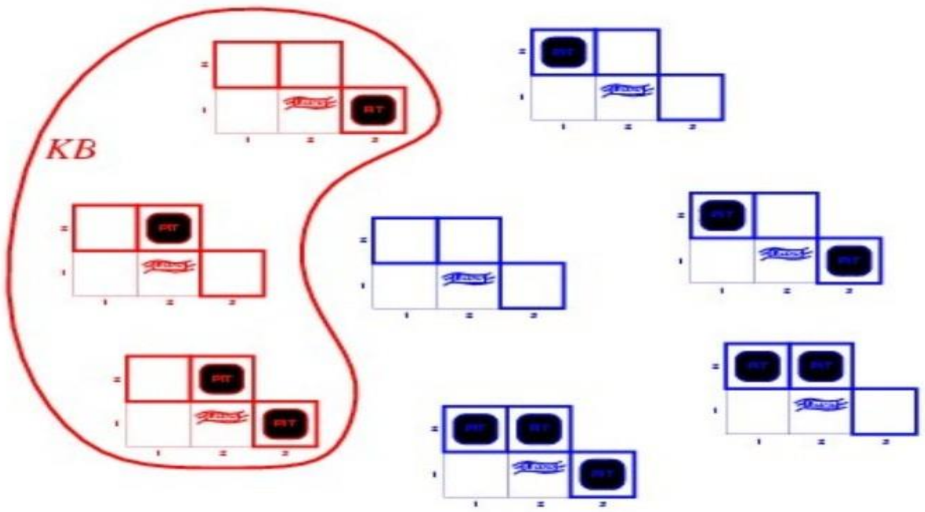
- Situation after detecting nothing in [1,1], moving right, breeze in [2,1]
- Consider possible models for *KB* assuming only pits
- 3 Boolean choices $\Rightarrow$ 8 possible models



Wumpus Models

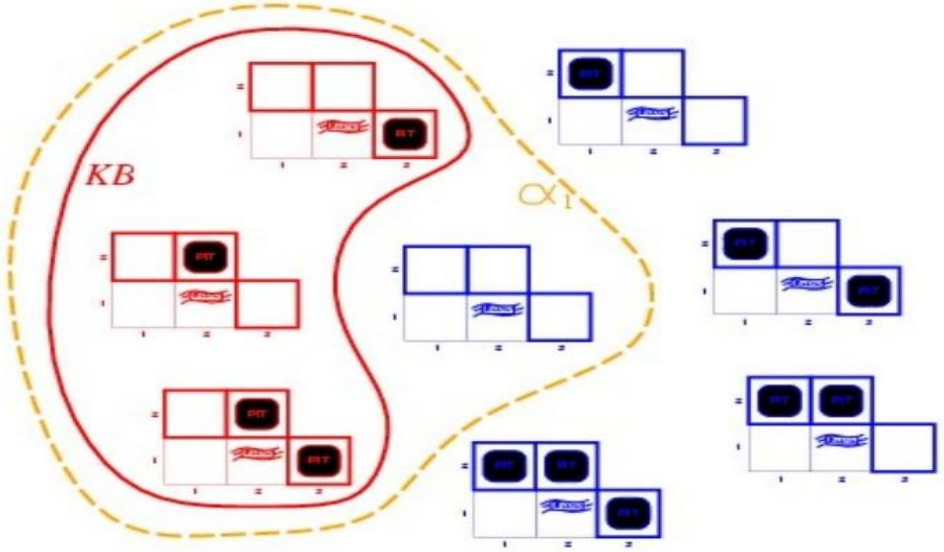


Wumpus Models



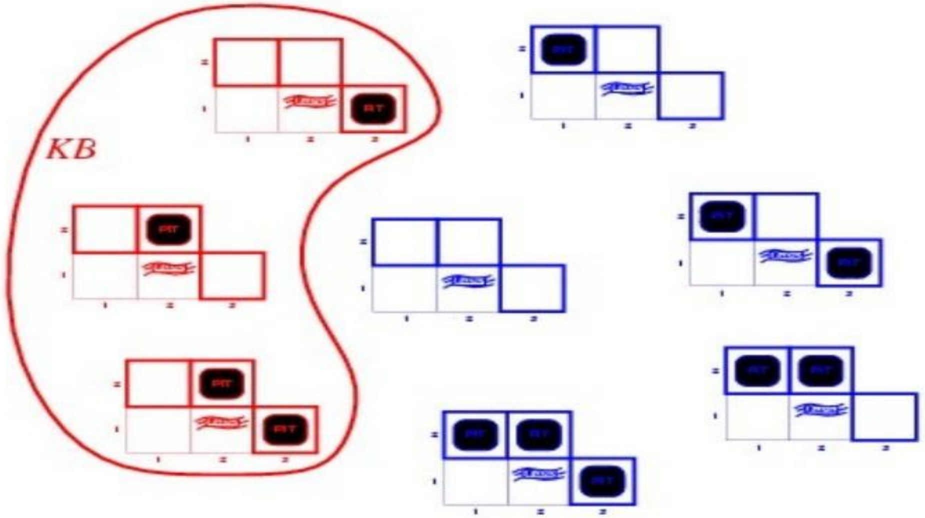
- KB = wumpus-world rules + observations

Wumpus Models



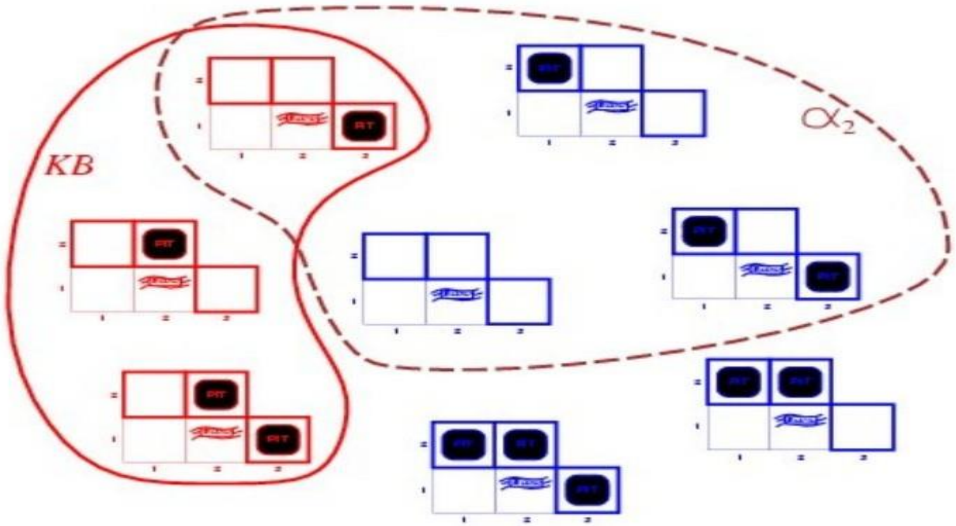
- *KB* = wumpus-world rules + observations
- α_1 = "[1,2] is safe", $KB \models \alpha_1$, proved by model checking
  - enumerate all possible models to check that α is true in all models in which KB is true

Wumpus Models



- KB = wumpus-world rules + observations
- α_2 = "[2,2] is safe", $KB \models \alpha_2$?

Wumpus Models



- KB = wumpus-world rules + observations
- α_2 = "[2,2] is safe", $KB \not\models \alpha_2$

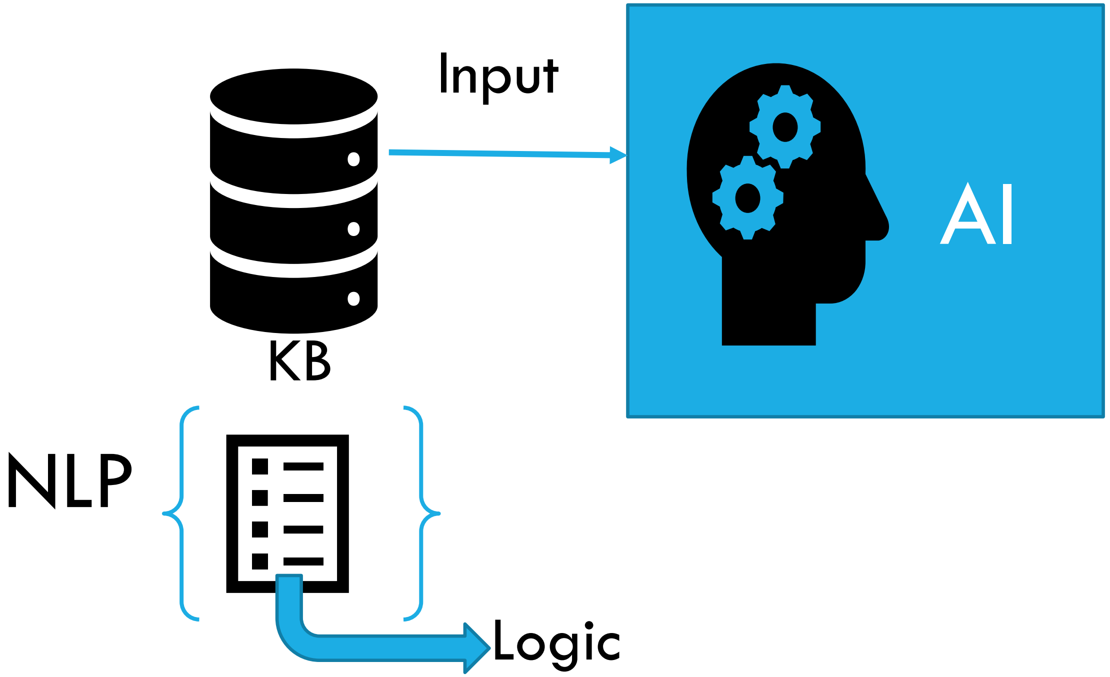
Soundness and Completeness

- An inference algorithm that derives only entailed sentences is called sound

- An inference algorithm is complete if it can derive any sentence that is entailed

- If KB is true in the real world, then any sentence derived from KB by a sound inference procedure is also true in the real world

PROPOSITIONAL LOGIC



Logic in AI

Propositional Logic

Predicate Logic

PROPOSITIONAL LOGIC

- ✓ Propositional Logic is a Declarative statement. (True/False)

Example:

1. It is Sunday
2. The sun rises from the West
3. 6+4 =12
4. Open the door

FACTS ABOUT PROPOSITIONAL LOGIC

- ✓ Propositional logic is also called Boolean Logic.
- ✓ In propositional logic, we use symbolic variables to represent the logic
- ✓ Propositions can be either true or false, but they cannot be both.
- ✓ Propositional logic consists of an object, relations, and logical connectives.
- ✓ These connectives are also called logical operators.
- ✓ The propositions and connectives are the basic elements of propositional logic.
- ✓ Connectives can be said to be logical operator which connects two sentences.

FACTS ABOUT PROPOSITIONAL LOGIC

- ✓ A proposition formula that is always true is called tautology, and it is also called a valid sentence.
- ✓ A proposition formula that is always false is called Contradiction.

❑ **Types of Propositions:**

1. Atomic Proposition
2. Compound Proposition

PROPOSITIONAL LOGIC

1. Atomic Proposition :

Atomic propositions are simple propositions. It consists of a single proposition symbol. These are the sentences that must be either true or false.

Example: a) Earth is square

b) 2 + 3 = 5

2. Compound Proposition :

Compound propositions are constructed by combining simpler or atomic propositions, using parenthesis and logical connectives.

Example: a) It is raining today, and the street is weight

b) Mr.Amte is a doctor and his clinic is at Hemalkasa.

LOGICAL CONNECTIVES

1	Not	¬	Negation	¬ X
2	And	Λ	Conjunction	X ∨ Y
3	Or	∨	Disjunction	X ∨ Y
4	Implies	→	Implication	(X→Y) if X then Y
5	If and only if	←→	Biconditional	(X ←→Y)

LOGICAL CONNECTIVES

Sr.no	Word	Symbol	Technical Term	Example
1	Not	$\neg$	Negation	$\neg$ X
2	And	$\wedge$	Conjunction	X $\wedge$ Y
3	Or	$\vee$	Disjunction	X $\vee$ Y
4	Implies	$\rightarrow$	Implication	(X$\rightarrow$Y) if X then Y
5	If and only if	$\leftarrow\rightarrow$	Biconditional	(X $\leftarrow\rightarrow$Y)

LOGICAL CONNECTIVES

X: It is hot

Y: It is Humid

Z: It is Raining

1. If it is humid then it is hot

$$Y \rightarrow X$$

2. It is hot and humid then it is not raining

$$X \land Y \neg Z$$

TRUTH TABLE

For negation:

P	¬P
True	False
False	True

TRUTH TABLE

For Conjunction:

P	Q	P ∧ Q
True	True	True
Ture	False	False
False	True	False
False	False	False

TRUTH TABLE

For Disjunction:

P	Q	P ∨ Q
True	True	True
True	False	True
False	True	True
False	False	False

TRUTH TABLE

For Implication:

P	Q	$P \rightarrow Q$
True	True	True
True	False	False
False	True	True
False	False	True

TRUTH TABLE

For Biconditional

P	Q	P ←→ Q
True	True	True
True	False	False
False	True	False
False	False	True

PRECEDENCE OF CONNECTIVES

Precedence	Operators
First Precedence	Parenthesis
Second Precedence	Negation
Third Precedence	Conjunction (AND)
Fourth Precedence	Disjunction (OR)
Fifth Precedence	Implication
Six Precedence	Biconditional

LOGICAL EQUIVALENCE

A	B	¬A	¬A ∨ B	A→B
True	True	False	True	True
True	False	False	False	False
False	True	True	True	True
False	False	True	True	True

LIMITATIONS OF PROPOSITIONAL LOGIC

1. With propositional logic, we cannot represent relations like ALL, some, or none.

Example:

- All the girls are intelligent.
- Some apples are sweet.

2. Propositional logic has limited expressive power.

PROPOSITIONAL THEOREM PROVING

✓ so far, we have shown how to determine entailment by model checking: enumerating models and showing that the sentence must hold in all models.

✓ in this section we show how entailment can be done by theorem proving: applying rules of inference directly to the sentences in our knowledge base to construct the proof of the desired sentence without consulting models.

1. Inference and proofs
2. Proof by resolution

PROPOSITIONAL THEOREM PROVING

1. **Commutativity:**

$$P \wedge Q = Q \wedge P$$

or

$$P \vee Q = Q \vee P$$

2. Associativity:

$$(P \wedge Q) \wedge R = P \wedge (Q \wedge R)$$

$$(P \vee Q) \vee R = P \vee (Q \vee R)$$

PROPOSITIONAL THEOREM PROVING

3. Identity Element:

$$P \land True = P$$

$$P \lor True = True$$

4. Distributive:

$$P \land (Q \lor R) = (P \land Q) \lor (P \land R)$$

$$P \lor (Q \land R) = (P \lor Q) \land (P \lor R)$$

PROPOSITIONAL THEOREM PROVING

5. DE Morgan's Law:

$$\neg (P \land Q) = (\neg P) \lor (\neg Q)$$

$$\neg (P \lor Q) = (\neg P) \land (\neg Q)$$

6. Double Negation Elimination:

$$\neg (\neg P) = P$$

PROPOSITIONAL THEOREM PROVING

Absorption law:

$$P \land (P \lor Q) = P$$

$$P \lor (P \land Q) = P$$

Law of contradiction:

$$P \land \neg P = \text{False}$$

Law of excluded middle:

$$P \lor \neg P = \text{True}$$

A simple knowledge base(Wumpus World):

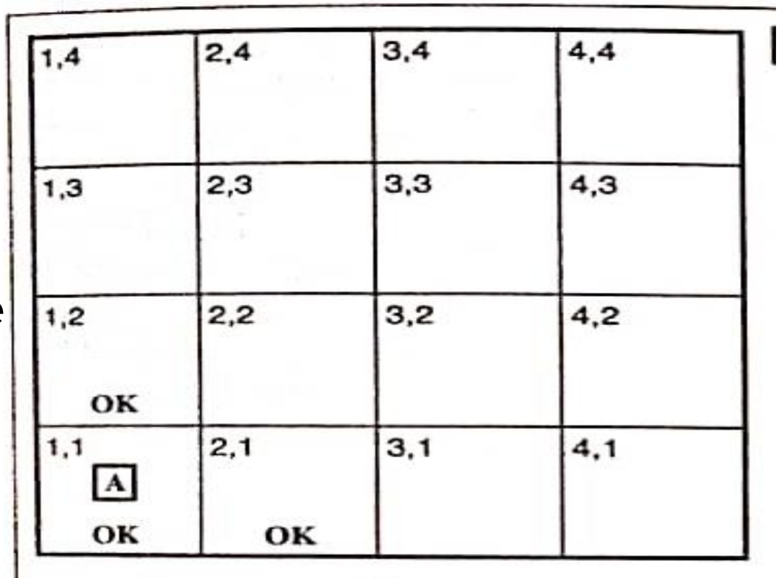
Goal: there is no pit in [1, 2] : $\neg P_{1, 2}$

$P_{x, y}$ is true if there is a pit in [x, y].

$W_{x, y}$ is true if there is a Wumpus in [x, y] dead or alive

$B_{x, y}$ is true if the agent perceives a breeze in [x, y]

$S_{x, y}$ is true if the agent perceives a stench in [x, y]



(a)

- There is no pit [1, 1]:

  R_1: $\neg P_{1, 1}$

- A square is breezy if and only if there is a pit in the neighboring square. This has to be stated for each square; for now, we include just the relevant squares:

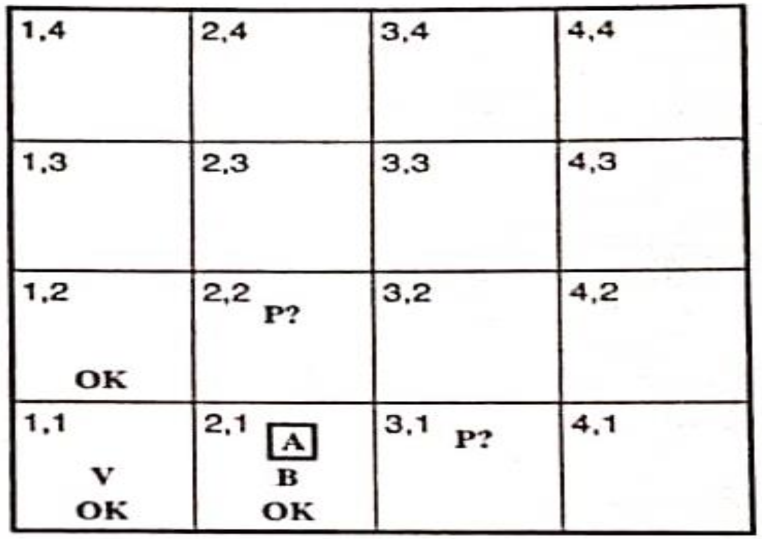
$$R_2 : B_{1,1} \leftrightarrow (P_{1,2} \vee P_{2,1})$$

$$R_3 : B_{2,1} \leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$$

- Now if we include the breeze percepts for the first two squares visited in the specific world the agent is in leading up to the situation as shown in figure

$$R_4: \neg B_{1,1}$$

$$R_5: B_{2,1}$$



(b)

1. INFERENCE AND PROOFS:

- Generating the conclusion from evidence and facts is termed inference

P	Q	$P \to Q$	$Q \to P$	$\neg Q \to \neg P$	$\neg P \to \neg Q.$
T	T	T	T	T	T
T	F	F	T	F	T
F	T	T	F	T	F
F	F	T	T	T	T

1.Modus Ponens:

Whenever any sentence of the form $P \rightarrow Q$ and P is given, then the sentence Q can be inferred

Notation for Modus ponens: $$\frac{P \rightarrow Q, \quad P}{\therefore Q}$$

Statement-1: "If I am sleepy then I go to bed" ==> P→ Q
Statement-2: "I am sleepy" ==> P

Conclusion: "I go to bed." ==> Q.
Hence, we can say that, if P→ Q is true and P is true then Q will be true.

P	Q	P→Q
0	0	0
0	1	1
1	0	0
1	1	1

2. Modus Tollens:

The Modus Tollens rule state that if P→ Q is true and **¬ Q is true, then ¬ P** will also true.

Notation for Modus Tollens: $\dfrac{P \rightarrow Q, \ \sim Q}{\sim P}$

Statement-1: "If I am sleepy then I go to bed" ==> P→ Q
Statement-2: "I do not go to the bed."==> ~Q
Statement-3: Which infers that "**I am not sleepy**" => ~P

3. Conjunction (AND) elimination:

Conjunction elimination states that from conjunction, we can derive either of the conjuncts.

$$\frac{P \wedge Q}{P}$$

Satement1: Sailee is an artist and a women (P Λ Q)

Conclusion: Sailee is an artist

$R_2 : B_{1,1} \leftrightarrow (P_{1,2} \vee P_{2,1})$

$R_6 : (B_{1,1} \rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1})$

Now we apply AND Elimination to R_6 to obtain

$R_7: ((P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1}$

Logical equivalence gives

$R_8 : (\neg B_{1,1} \rightarrow \neg(P_{1,2} \vee P_{2,1}))$

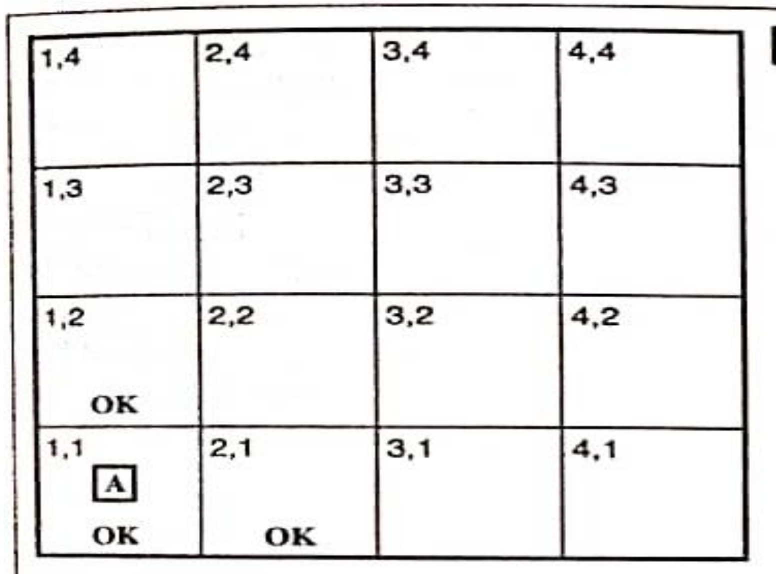
Now we can apply Modus Ponens with R8 and percept R4

$R_9 : \neg(P_{1,2} \vee P_{2,1})$

Apply De Morgan's rule

$R_{10}: \neg P_{1,2} \wedge \neg P_{2,1}$

That is, neither [1,2] nor [2,1] contains a pit.



(a)

THANK YOU