DRONACHARYA College of Engineering

INTELLIGENT SYSTEMS (CSE-303-F)

Section B

Knowledge Representation

INTRODUCTION

- Knowledge representation is study of ways of how knowledge is actually pictured and how effectively it resembles the representation of knowledge in human brain.
- Knowledge representation systems provides way of representing complex knowledge and possess following characteristics:
  - Representation scheme should have well defined syntax
  - Representation scheme should have good expressive capacity
  - Representation scheme should have must be efficient (use minimum resources)

INTRODUCTION

- Knowledge representation system and DBMS differ by the fact that DBMS do not hold facility for manipulation of facts.

- E.g. All carnivorous has sharp teeth.

  Cheetah is a carnivore.

  Hence Cheetah has sharp teeth.
- The last statement is inferred by first two.
- In a DBMS, until one specifies that Cheetah has sharp teeth, it is not possible to get this information.

DECLARATIVE REPRESENTATION

- Declarative representation declares every piece of knowledge and permits the reasoning system to use rules of inference like modus ponens, modus tollens etc. to come out with new piece of information.
- "All carnivorous has sharp teeth."

  "Cheetah is a carnivore."
- This can be represented using declarative representation as

$\forall x$ (Carnivore (x) $\rightarrow$ Sharp_Teeth(x))

Carnivore(Cheetah)

Advantages

- Declarative representations are flexible.
- Each piece of knowledge is an independent chunk on its own. Hence modularity is higher.
- It is enough that you represent knowledge only once. E.g. the statement

$$\forall x \, (\text{Carnivore}(x) \rightarrow \text{Sharp\_Teeth}(x))$$

engulfs a wide variety of animals which are carnivorous in nature.

PROCEDURAL REPRESENTATION

- Represents knowledge as procedures.
- Interfacing mechanism manipulate these procedures to arrive at result.
- E.g. for the earlier example, procedural representation can be

```
procedure carnivore (x);
        if (x = Cheetah) then return true
                else return false
end procedure carnivore (x).
procedure sharp_teeth (x);
        if carnivore (x) then return true
                else return false
end procedure sharp_teeth (x).
```

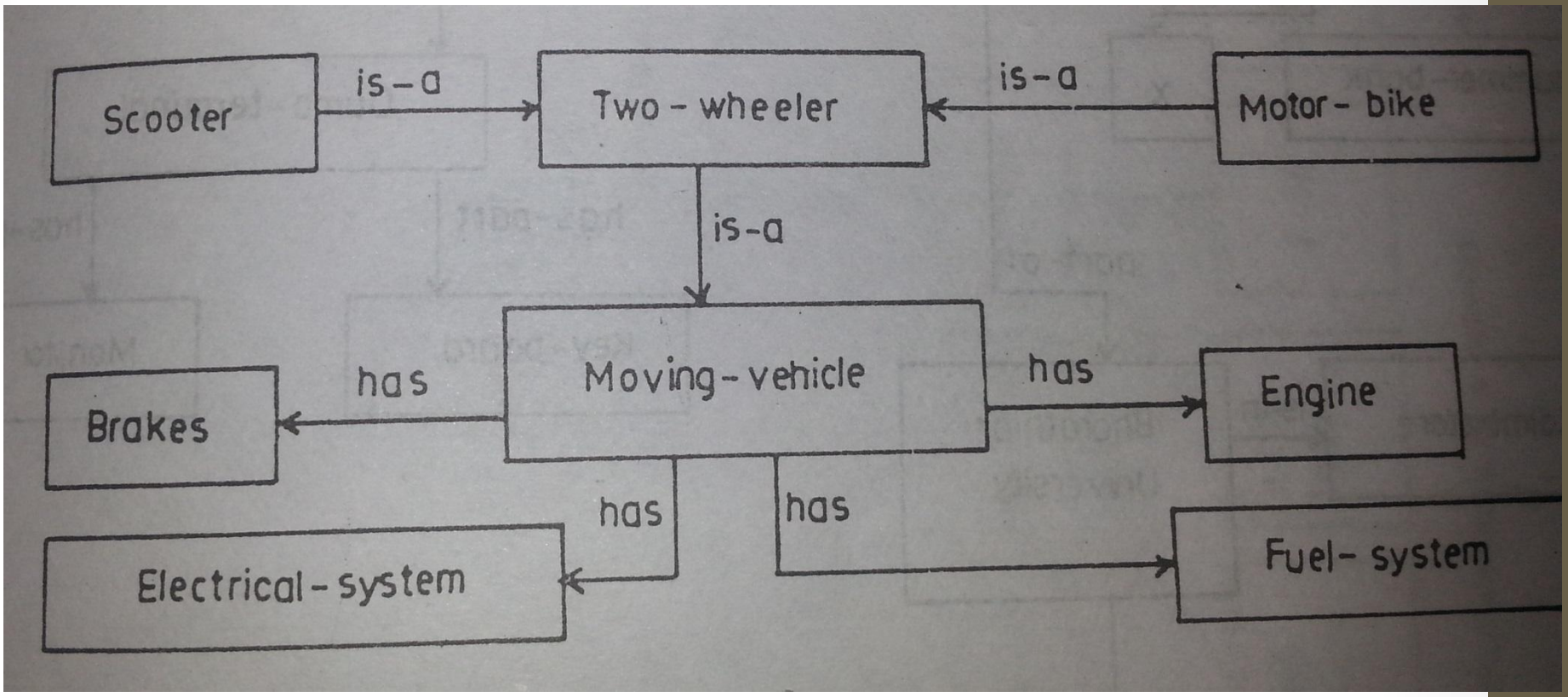
- To check whether Cheetah has sharp teeth, one should activate procedure sharp_teeth with variable x initialized to value Cheetah.
- This procedure calls procedure carnivore(x) with value of x = Cheetah
- Procedure carnivore returns true and so is the procedure sharp_teeth.
- In procedural languages, one has control over the search which is not available in declarative representations.

KNOWLEDGE REPRESENTATION SCHEMES

- Semantic Nets
- Frames
- Conceptual Dependency
- Scripts

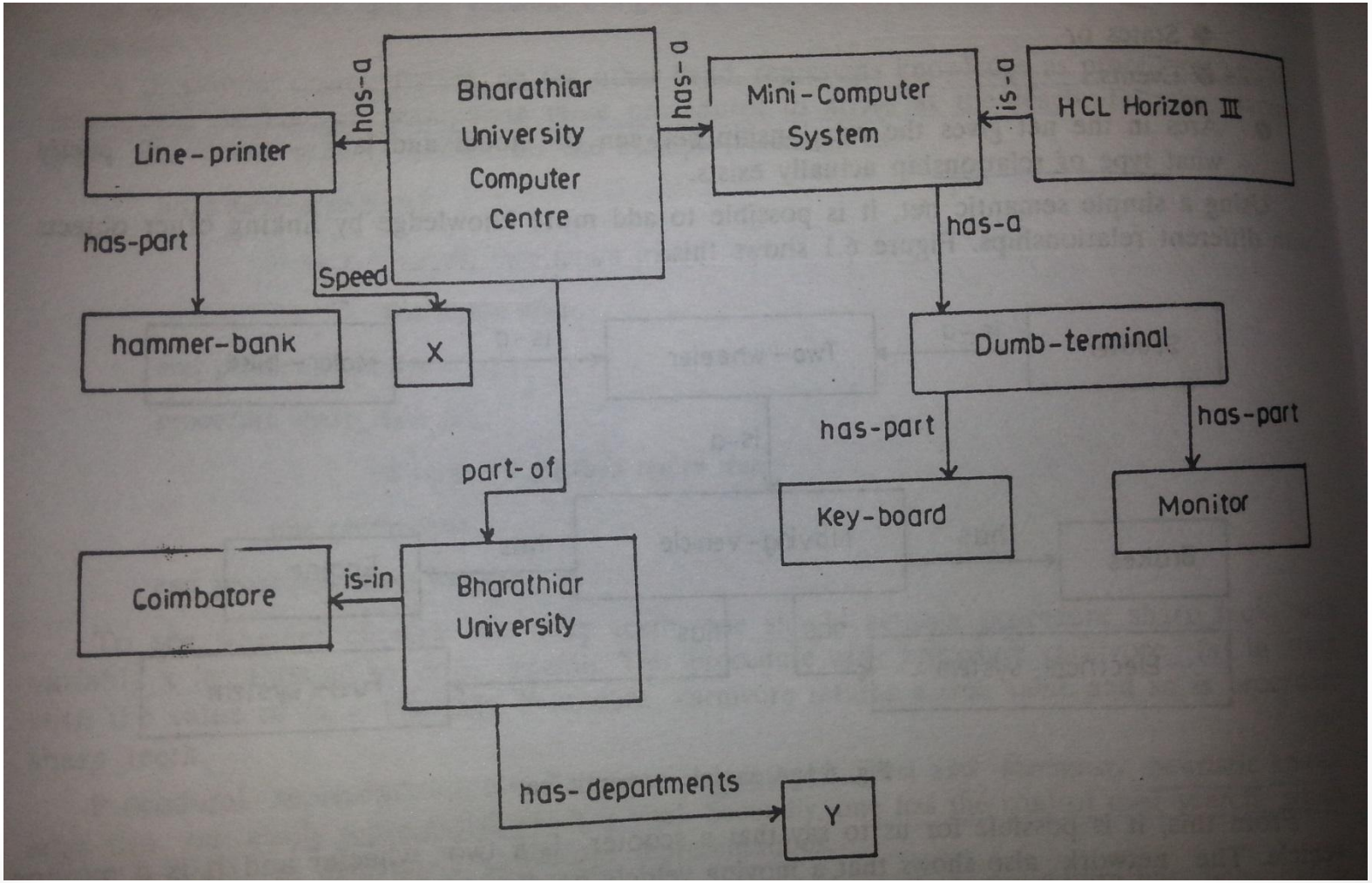
SEMANTIC NETS

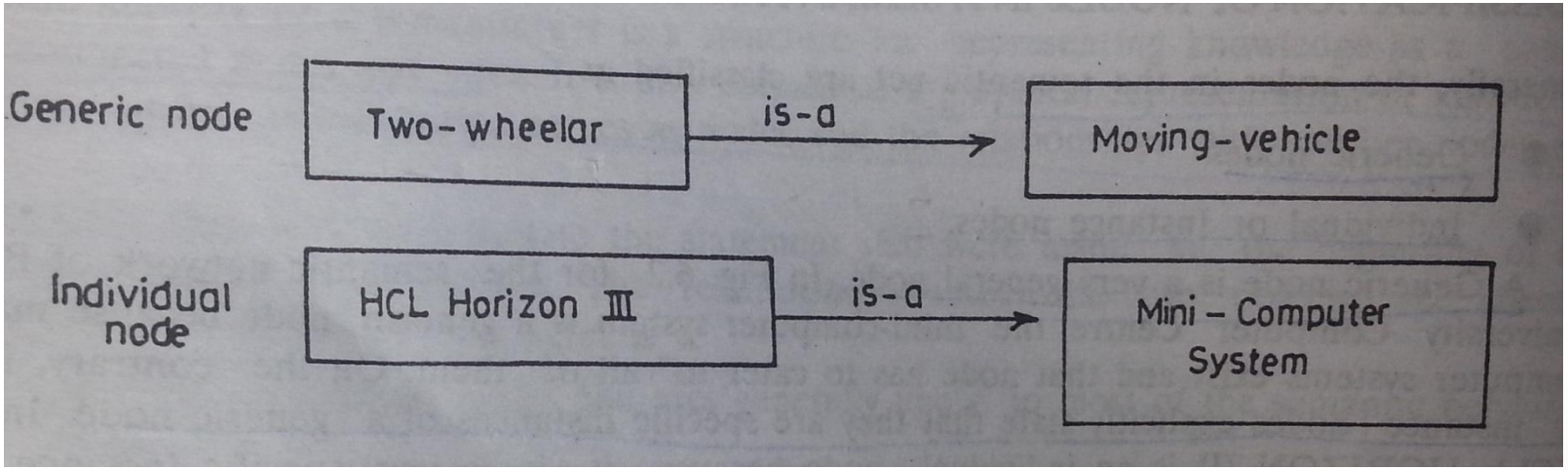
- Represents knowledge as pattern of interconnected nodes and arcs.
- Also defined as graphical representation of knowledge.
- The objects under consideration serves as nodes and relationships with other nodes serves as arcs.


Scooter
is-a
Two-wheeler
is-a
Motor-bike
is-a
Moving-vehicle
has
Brakes
has
Engine
has
Electrical-system
has
Fuel-system

CLASSIFICATION OF NODES IN SEMANTIC NETS

- Nodes in semantic nets are classified as
  - Generic nodes
  - Individual or instance nodes
- Generic node is very general node
- Individual or instance nodes state that they are specific instances of generic nodes.


Bharathiar University Computer Centre
has-a
Line-printer
has-a
Mini-Computer System
is-a
HCL Horizon III
has-part
Speed
hammer-bank
X
has-a
Dumb-terminal
has-part
has-part
Key-board
Monitor
part-of
Coimbatore
is-in
Bharathiar University
has-departments
Y

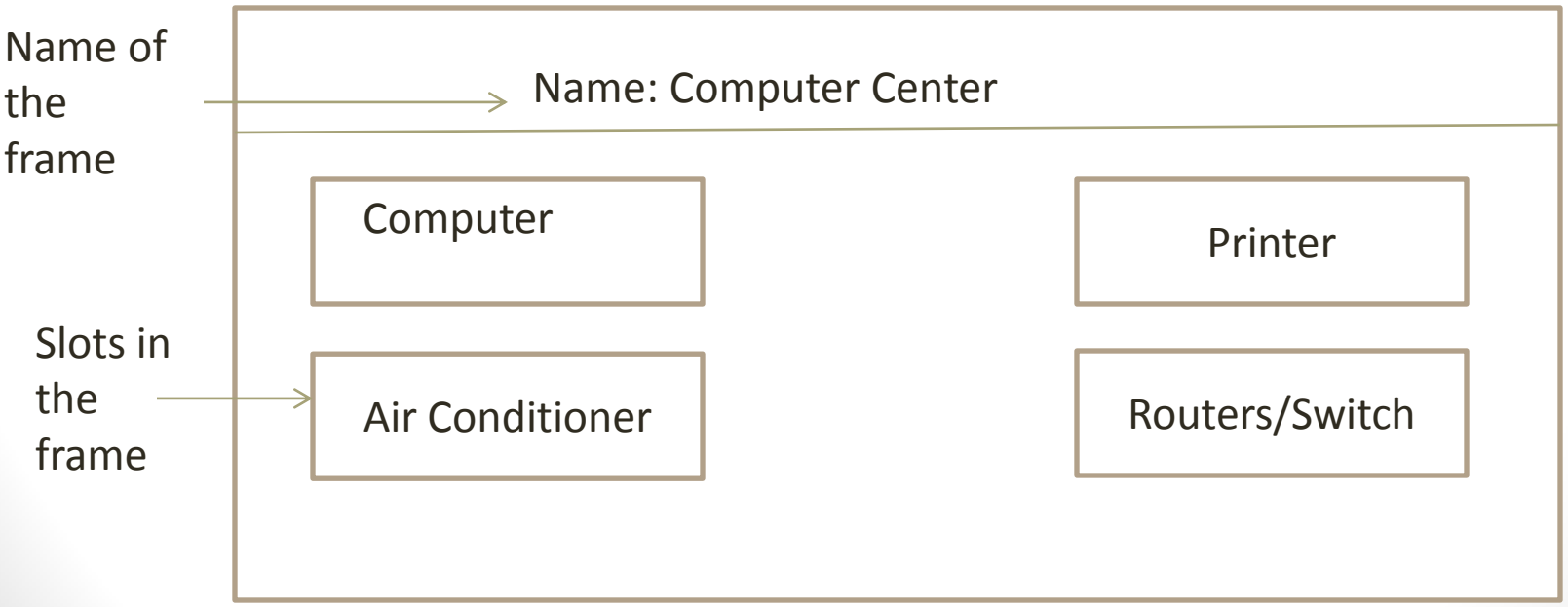

Generic node
Two-wheelar
is-a
Moving-vehicle
Individual node
HCL Horizon III
is-a
Mini – Computer System

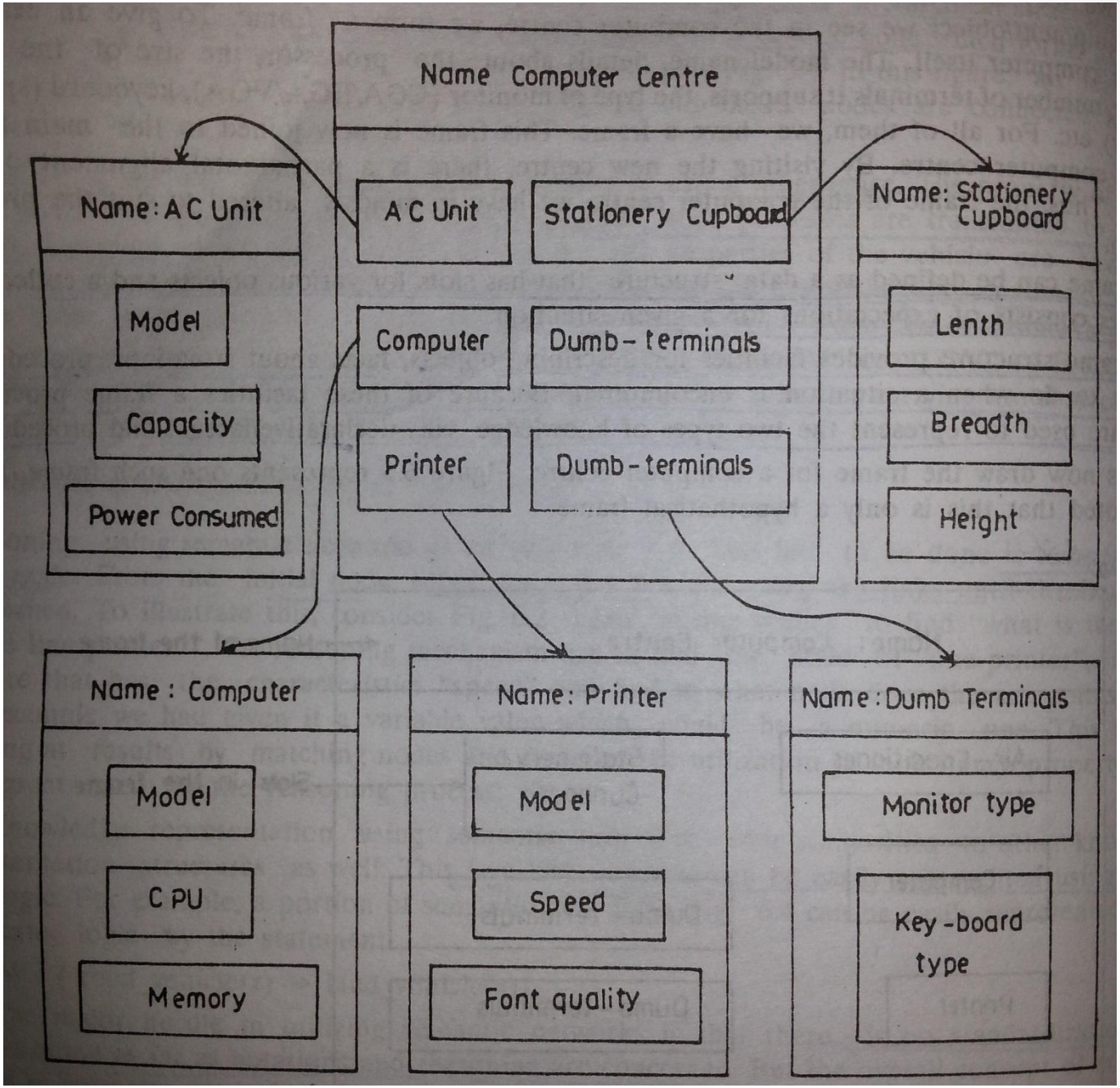
REASONING UNDER SEMANTIC NETS

- Reasoning under semantic nets is very easy task.
- From the initial nose, all nodes are processed using links.
- E.g. in the previous example if one wishes to find “what is the speed of line printer”, the reasoning mechanism needs to find node “line printer”, identify the arcs that has characteristics speed.

FRAMES

- Knowledge is organized into small packets called “frames”.
- The contents of frame are slots which have values.
- Whenever one encounters a situation, a series of related frames are activated and reasoning is done.



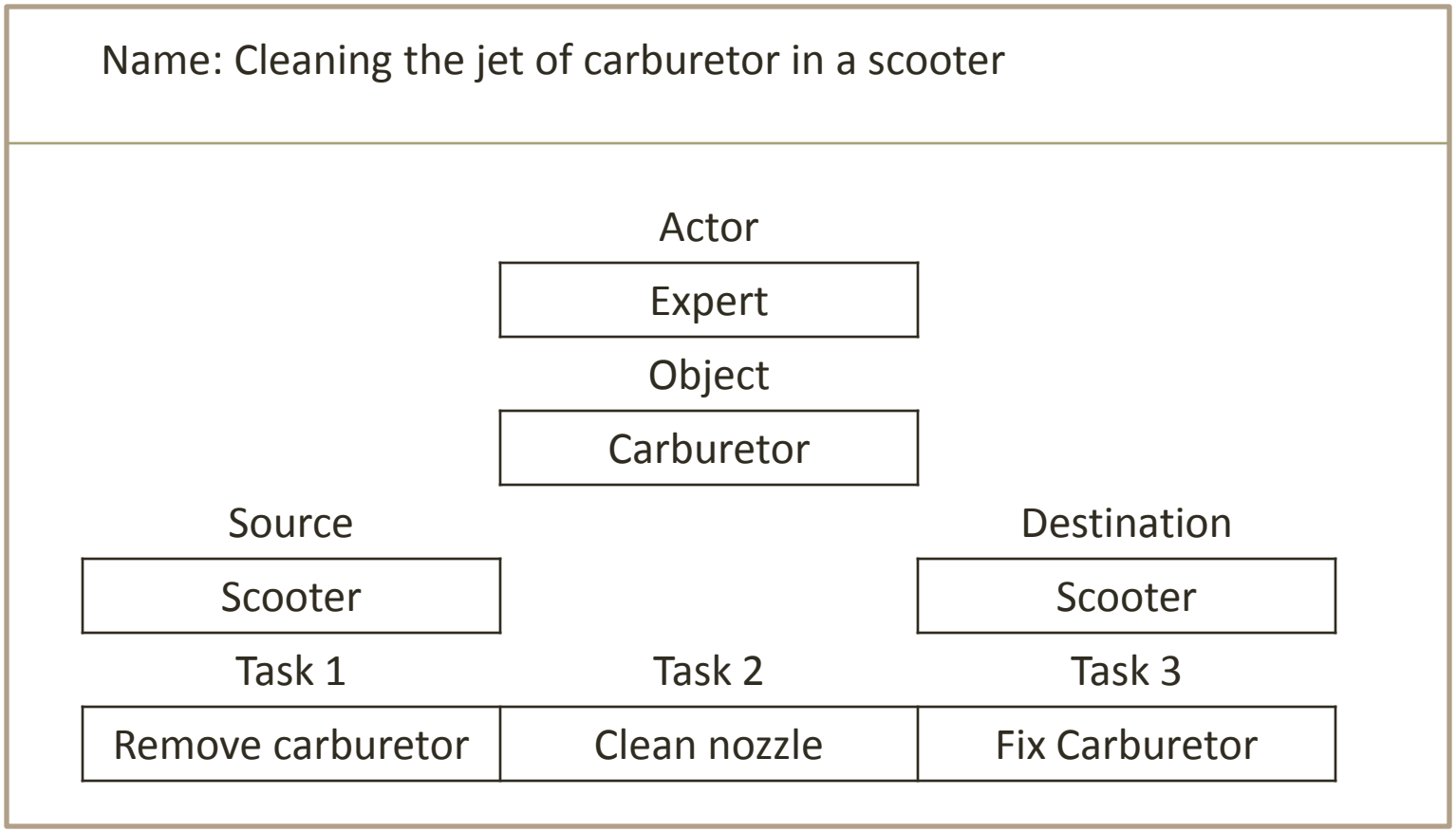

Name Computer Centre
A C Unit
Stationery Cupboard
Computer
Dumb-terminals
Printer
Dumb-terminals
Name: A C Unit
Model
Capacity
Power Consumed
Name: Stationery Cupboard
Lenth
Breadth
Height
Name : Computer
Model
C PU
Memory
Name: Printer
Model
Speed
Font quality
Name: Dumb Terminals
Monitor type
Key-board type

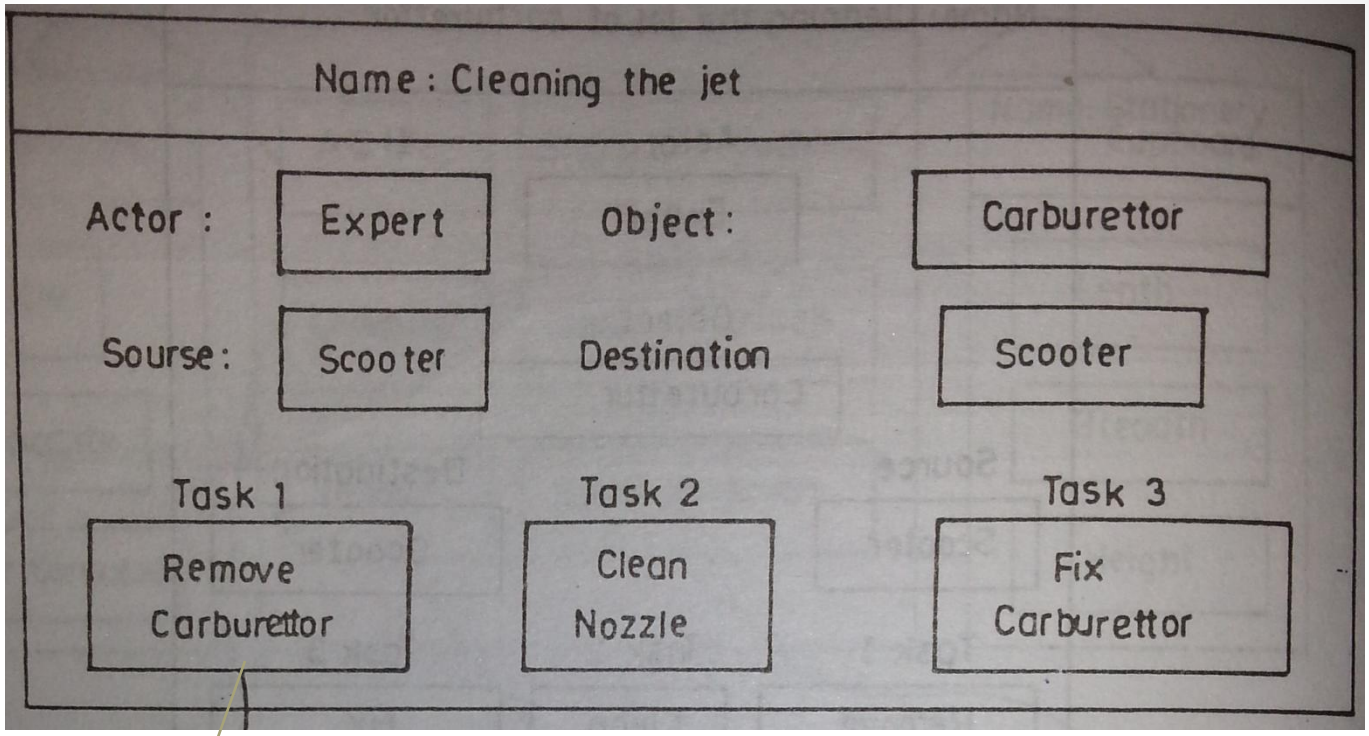
DECLARATIVE AND PROCEDURAL FRAMES

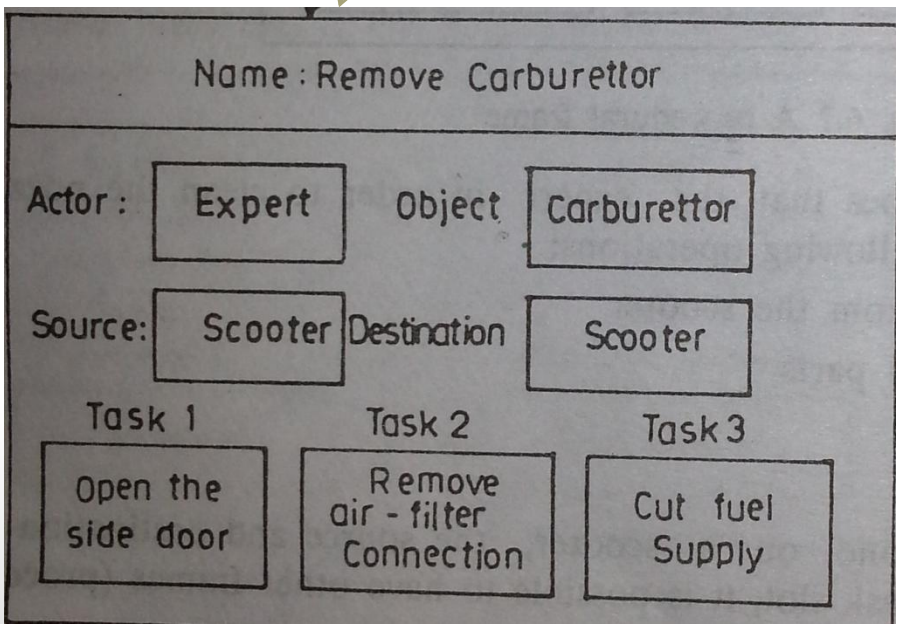
- A frame that merely contains description about objects is called declarative frame.
- E.g. previous example
- Procedural knowledge representation makes in possible to attach slots which explains how to perform things.
- Such frames which have procedural knowledge embedded in it are called action procedure frames.

- The action frame has following slots:
  - Actor Slot : holds information about who is performing the activity
  - Object Slot : information about item to be operated
  - Source Slot : information from where action has to begun
  - Destination Slot : information from where action has to end.
  - Task Slot : generates necessary sub frames required to perform the action.

- E.g. procedural frame for cleaning the jet of carburetor in a scooter.




Name : Cleaning the jet
Actor :
Expert
Object :
Carburettor
Sourse :
Scooter
Destination
Scooter
Task 1
Remove Carburettor
Task 2
Clean Nozzle
Task 3
Fix Carburettor


Name : Remove Carburettor
Actor :
Expert
Object
Carburettor
Source:
Scooter
Destination
Scooter
Task 1
Open the side door
Task 2
Remove air - filter Connection
Task 3
Cut fuel Supply

SCRIPTS

- Frames represents general knowledge representation structure.
- Scripts help exclusively in representing events that takes place in day-to-day activities.
- E.g.
  - Going to hotel, eating something, paying the bill and exiting.
  - Going to super market with list of things to be purchased, putting items needed on a trolley, paying for them.
  - Going to bank for withdrawal, filling the form, presenting to cashier, getting money and leaving the bank.
- Scripts tell people that what can happen in a situation, what events follow and what role every factor plays.

Script : Going to a restaurant

Props : Food
Tables
Menu
Money

Roles: Owner
Customer
Waiter
Cashier

Entry Conditions:
Customer is hungry
Customer has money
Owner has food

Results:
Customer is not hungry
Owner has more money
Customer has less money
Owner has less food

Scene 1: Entering the restaurant
Scans the table
Choose the best one
Goes there
Occupies seat

Scene 2: Ordering the food
Customer asks for menu
Chooses what to eat
Orders that item

Scene 3 : Eating the food
Waiter brings the food
Customer eats it

Scene 4 : Paying the bill
Customer asks for bill
Customer pays bill
Customer moves out

- The simple scripts throws enough light on important components:
  - Entry Conditions
  - Result
  - Props (indicates object that exist in scrip)
  - Roles
  - Track(represents specific instance of generic pattern. Restaurant is specific instance of hotel)
  - Scenes (sequence of activities described in detail)