Knowledge Representation and Reasoning
Part-II

Dr. Singara Singh Kasana

Associate Professor

Computer Science and Engineering Department

Thapar Institute of Engineering and Technology

Patiala

WEAK SLOT-AND-FILLER STRUCTURES

- The structures that support property inheritance along *instance* and *is a* links are called slot-and-filler structures.
- These structures are useful for support of other reasons besides property inheritance:

1) It indexes assertions by the entities they describe. The binary predicates are indexed by their first argument. So retrieving the value for a particular attribute of an object is fast.
2) It is easy to describe properties of objects which requires some higher order mechanisms in purely logic based system.
3) It is a form of object oriented programming. So it has all advantages that such systems have like modularity, inheritance, ease of use, etc.

- These structures do not say much about the specific knowledge these should contain. So they are called as “**knowledge-poor**” or “**weak-slot-and-filler structures**”.
- These structures include semantic networks and frames.

2. Semantic Network Representation

- In Semantic networks, you can represent your knowledge in the form of graphical networks.
- This network consists of nodes representing objects and arcs which describe the relationship between those objects.
- Also, it categorizes the object in different forms and links those objects.

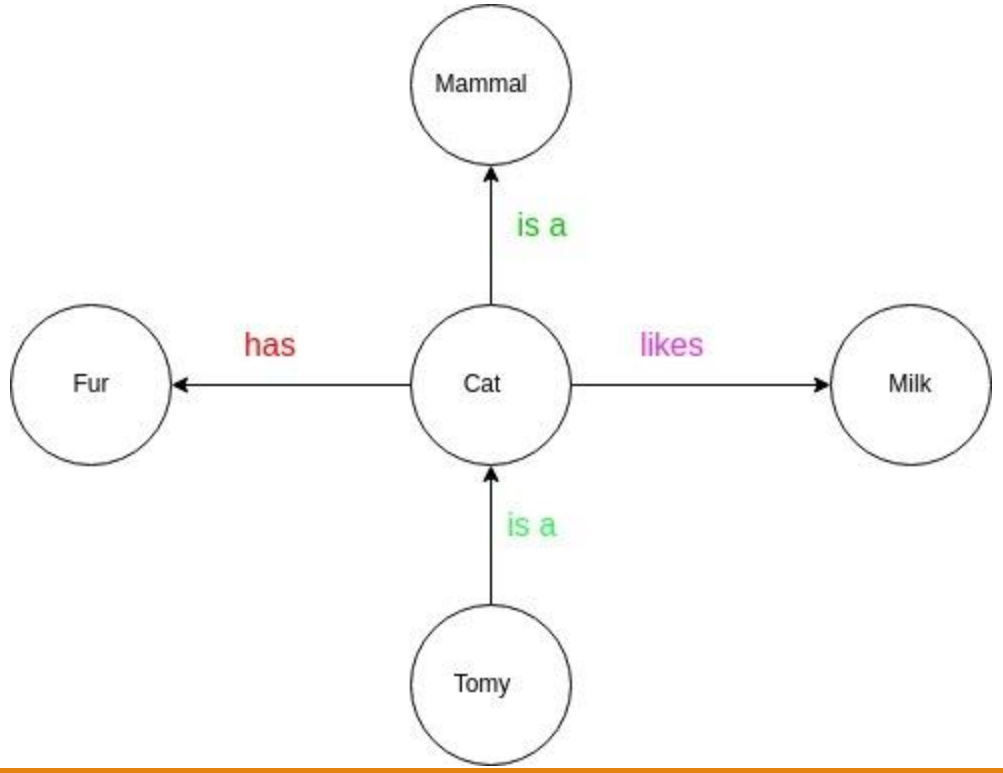
This representation consists of two types of relations:

i. IS-A relation (Inheritance): Child is a subclass of parent class

ii. Has–A relation(Composition)

Graph is intended to represent the data like

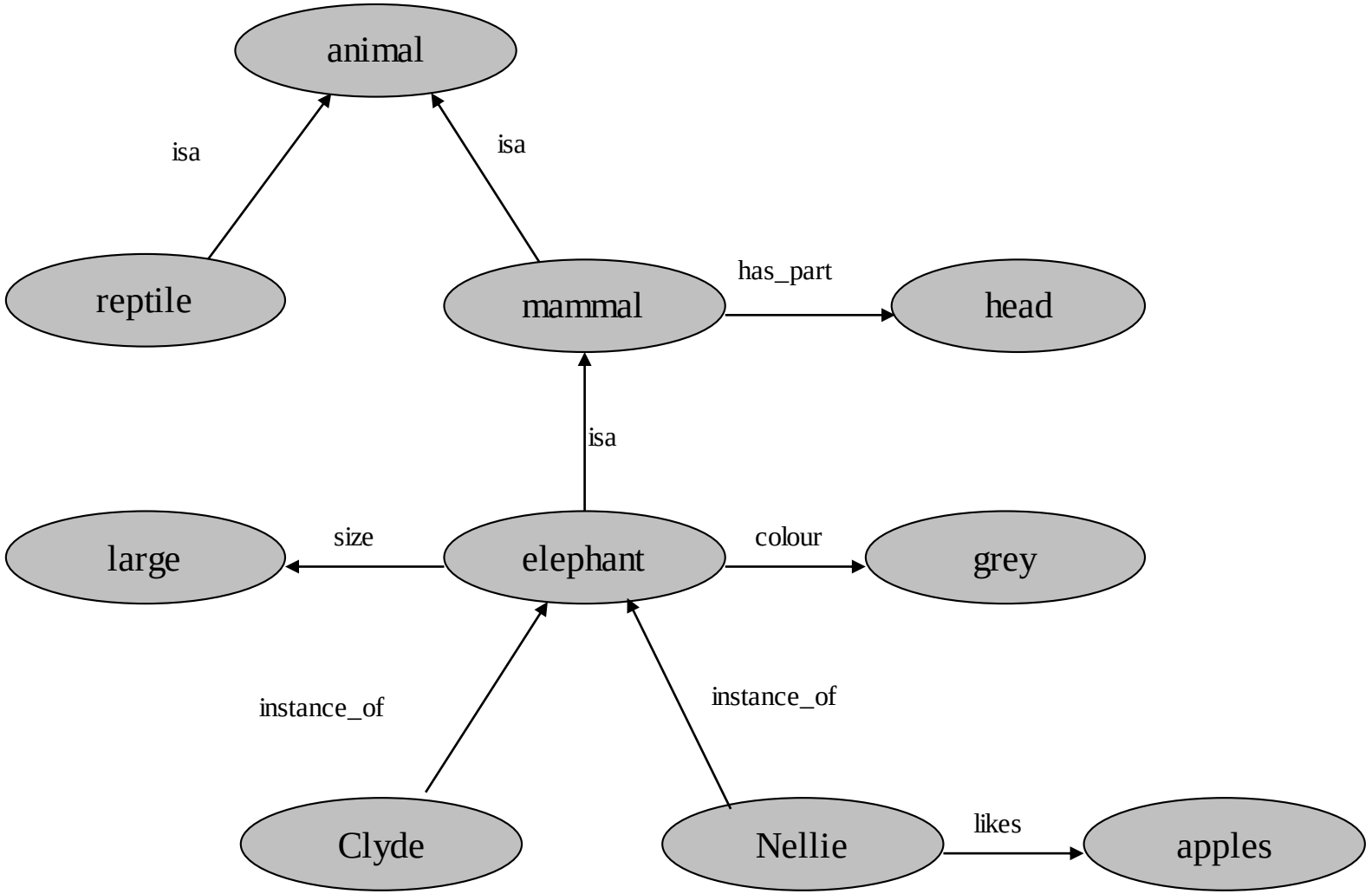
- Tomy is a cat
- Tomy has fur
- Cat likes milk
- Tomy is a mammal



Consider the following sentences:

1) Clyde and Nellie are elephants.
2) Nellie likes apples.
3) All elephants are mammals.
4) All elephants are large in size and grey colored.
5) All mammals and reptiles are animals.
6) Mammals have head.

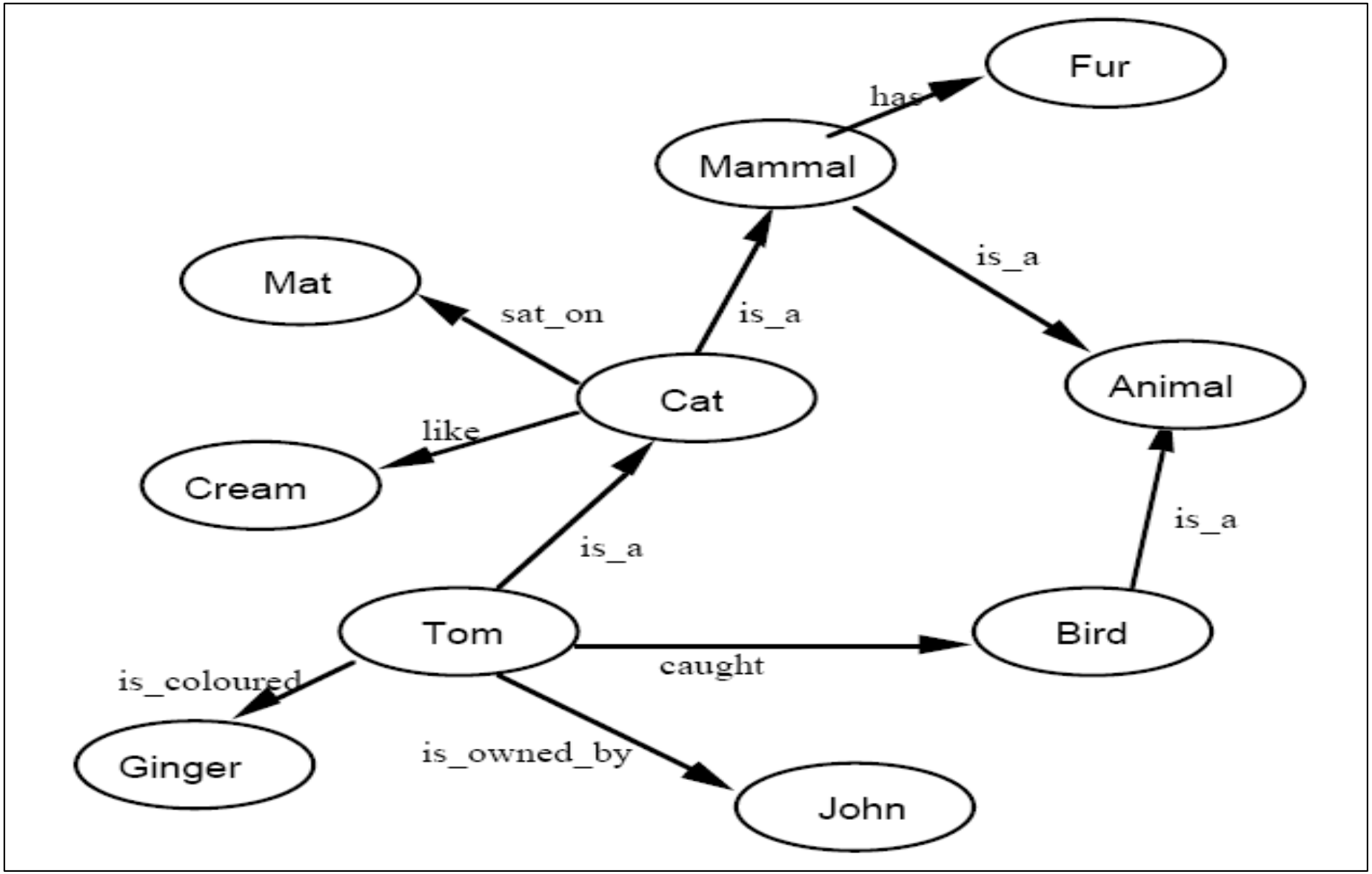
Represent the knowledge in the form of semantic networks.


animal
isa
isa
reptile
mammal
has_part
head
isa
large
size
elephant
colour
grey
instance_of
instance_of
Clyde
Nellie
likes
apples

Semantic Network Example 2

Represent following KB as semantic networks.

- Tom is a cat.
- Tom caught a bird.
- Tom is owned by John.
- Tom is ginger in colour.
- Cats like cream.
- The cat sat on the mat.
- A cat is a mammal.
- A bird is an animal.
- All mammals are animals.
- Mammals have fur.


Fur
has
Mammal
Mat
sat_on
is_a
is_a
Animal
Cat
like
Cream
is_a
is_a
Tom
Bird
caught
is_coloured
Ginger
is_owned_by
John

Representing Non-Binary Predicates Relations

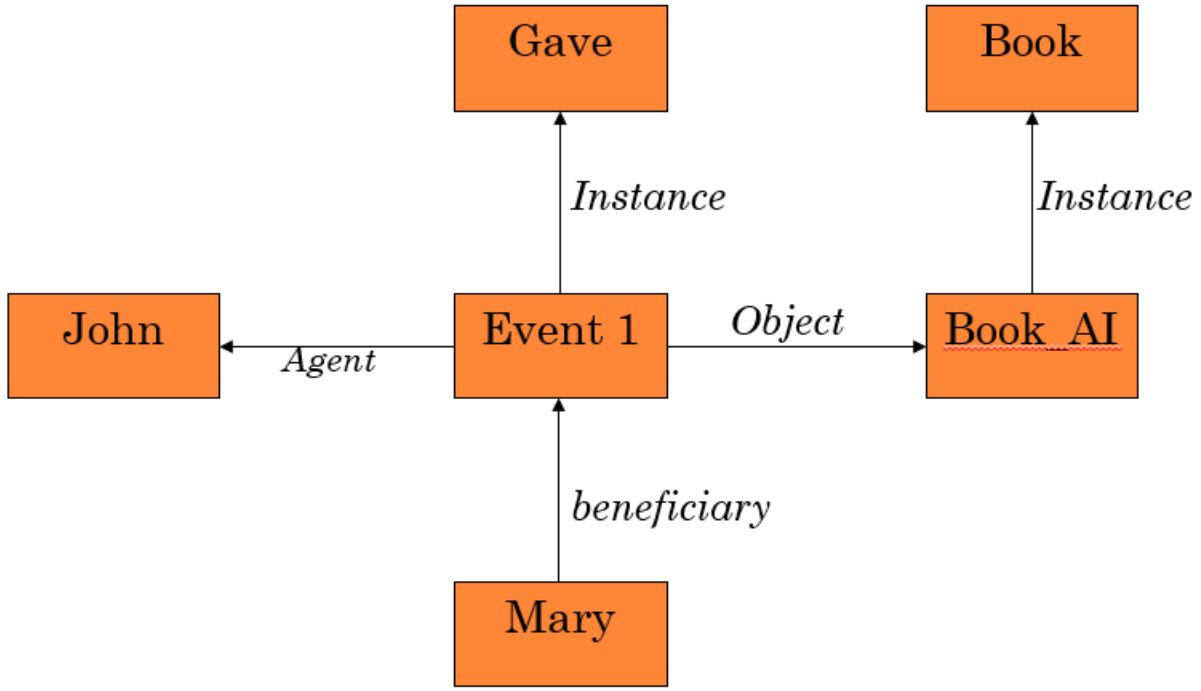
- Semantic nets are natural ways of representing binary predicates of predicate logic like:
  - Team(Pee-Wee-Resse,Brooklyn Dodgers)
  - Uniform_color(Pee-Wee-Resse,blue)

- Unary predicates can be converted in binary predicates using instance and is a relations.

  For example man(marcus) can be converted into semantic network by representing it as instance(marcus,man)
- Three or more arity predicates can be converted into binary form by creating new objects representing the entire predicate statement and then representing relationships of all old objects with this new object.

Representing Non-Binary Predicates Relations Contd..

- John gave Mary the AI book



Advantages:

- Semantic networks are a natural representation of knowledge.
- Also, it conveys meaning in a transparent manner.
- These networks are simple and easy to understand.

Disadvantages:

- Semantic networks take more computational time at runtime we need to traverse the complete network tree to answer some questions..
- Also, these are inadequate as they do not have any equivalent quantifiers e.g., for all, for some, none etc.
- These networks are not intelligent and depend on the creator of the system.
- Semantic networks do not have any standard definition for the link names.

3. Frame Representation

- A frame is a **record** like structure that consists of a **collection of attributes** and **values** to describe an entity in the world.
- These are the AI data structures that divides knowledge into substructures by representing stereotypes situations.
- Basically, it consists of a collection of slots and slot values of any type and size. Slots have names and values which are called facets.

- Frames are a variant of semantic networks
- All the information relevant to a concept is stored in a single complex entity called a frame.
- Superficially, frames look like record data structures or class.

```
class Book {
Person author;
String title;
int price;
}
```

- A single frame is rarely useful.
- We build a collection of frames called *frame systems* that are connected by virtue of the fact that the value of an attribute of one frame is another frame.

- In frame-based systems we refer to
  - **objects** – *Mammal*, *Elephant*, and *Nellie;*
  - **slots** – properties such as *color* and *size*;
  - **slot-values** – values stored in the slots, e.g. *grey* and *large*.

- Slots and the corresponding slot-values are inherited through the class hierarchy

FRAMES - EXAMPLE

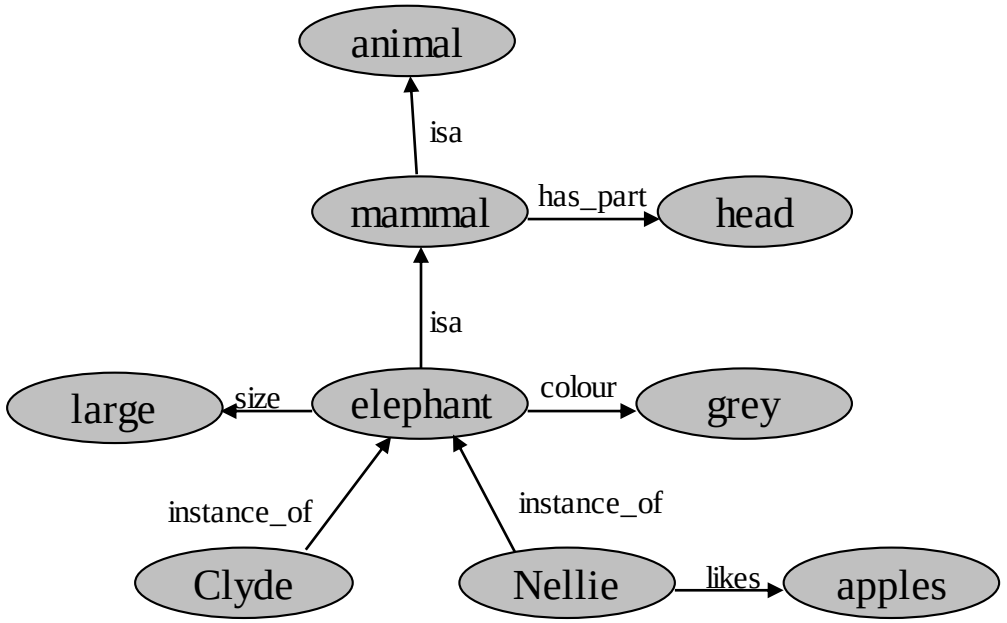
- The previous example can be also represented as frames:

Mammal:
subclass: Animal
has-part: head

Elephant:
subclass: Mammal
colour: grey
size: large

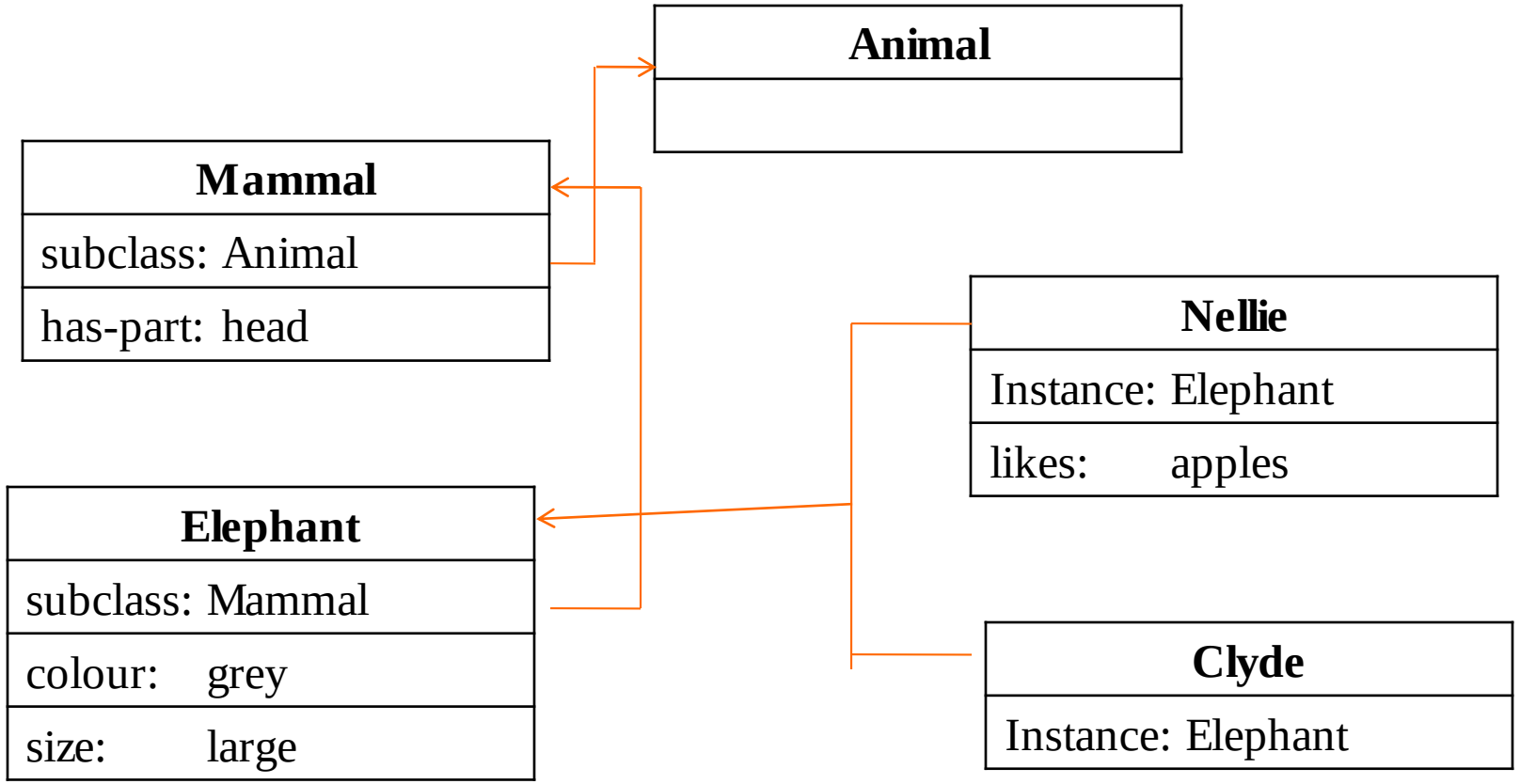
Nellie:
instance: Elephant
likes: apples

Clyde:
instance: Elephant



Frames – example Contd....

- The above frame system can also be represented as:



Example of a frame for a Student

Slots	Filters
Roll No	101
Branch	BE
Name	Ram
Age	20
Address	Patiala
AGPA	9.8

Advantages:

- It makes the programming easier by grouping the related data.
- Frame representation is easy to understand and visualize.
- It is very easy to add slots for new attributes and relations.
- Also, it is easy to include default data and search for missing values.

Disadvantages:

- In frame system, inference mechanism cannot be easily processed.
- It has a very generalized approach.

4. Production Rules

- In production rules, agent checks for the **condition** and if the condition exists then production rule fires and corresponding action is carried out.
- The condition part of the rule determines which rule may be applied to a problem. Whereas, the action part carries out the associated problem-solving steps. This complete process is called a recognize-act cycle.

The production rules system consists of three main parts:

i. **The set of production rules**
ii. **Working Memory**
iii. **The recognize-act-cycle**

Example:

IF (at station AND train arrives) THEN action (get into the train)
IF (on the train AND empty seat) THEN action (sit down).
IF (on train AND unpaid) THEN action (pay charges).
IF (train arrives at destination) THEN action (get down from the train).

The working memory contains the description of the current state of problems-solving and rule can write knowledge to the working memory. This knowledge match and may fire other rules.

If there is a new situation (state) generation, then multiple production rules may be fired together, this is called **conflict set**. In this situation, the agent needs to select a rule from these sets, and it is called a **conflict resolution**.

Advantages of Production rule:

1. The production rules are expressed in natural language.
2. The production rules are highly modular, so we can easily remove, add or modify an individual rule.

Disadvantages of Production rule:

1. Production rule system does not exhibit any learning capabilities, as it does not store the result of the problem for the future uses.
2. During the execution of the program, many rules may be active hence rule-based production systems are inefficient.