Frames:

A frame is a collection of attributes (slots) and associated values that describe some entity in the world.

Frame System:

Its a collection of frames which are connected to each other by virtue of the fact that the value of an attribute of one frame may be another frame.

Ways Of Relating Classes to each Other:

- Classes are nothing but Sets. 'isa' relation is equivalent to subset relation of set theory and "instance" relation is m or element relation of set theory.

- Two sets can be related in following other ways also:

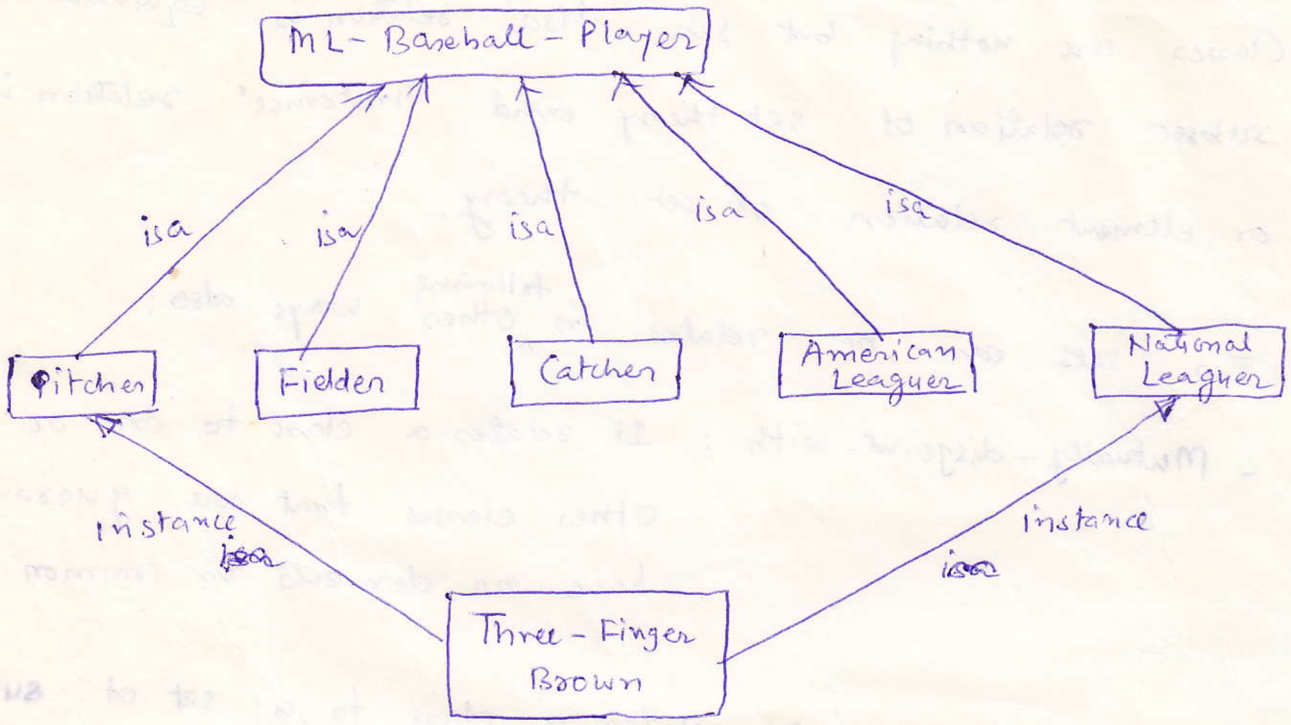
  - Mutually-disjoint-with : It relates a class to one or mor other classes that are guaranteed have no elements in common with th

  - is-covered-by : It relates a class to a set of subclas the union of which is equal to it.

- If a class is is-covered-by a set S of mutually disjoint c then S is called a partition of the class.

Example : Consider the class of major league baseball playe Everyone is either a pitcher, a catcher or a fielde (No one is more than one of these). In addition, everyone plays in either the National League or the American League, but not both.

Representing Relationships among Classes :



ML - Baseball - Player

is-covered-by : 1) { Pitcher, Fielder, Catcher },

2) { American Leaguer, National Leaguer }

(1) and (2) are individually partition of ML - Baseball - Player class.

Pitcher

isa : ML - Baseball - Player

mutually disjoint with : { Fielder, Catcher }

Catcher

isa : ML - Baseball - Player

mutually disjoint with : { Pitcher, Fielder }

Fielder

isa : ML - Baseball - Player

mutually disjoint with : { Pitcher, Catcher }

Americal Leaguer
isa : ML-Baseball-Player
mutually disjoint with : {National-Leaguer}

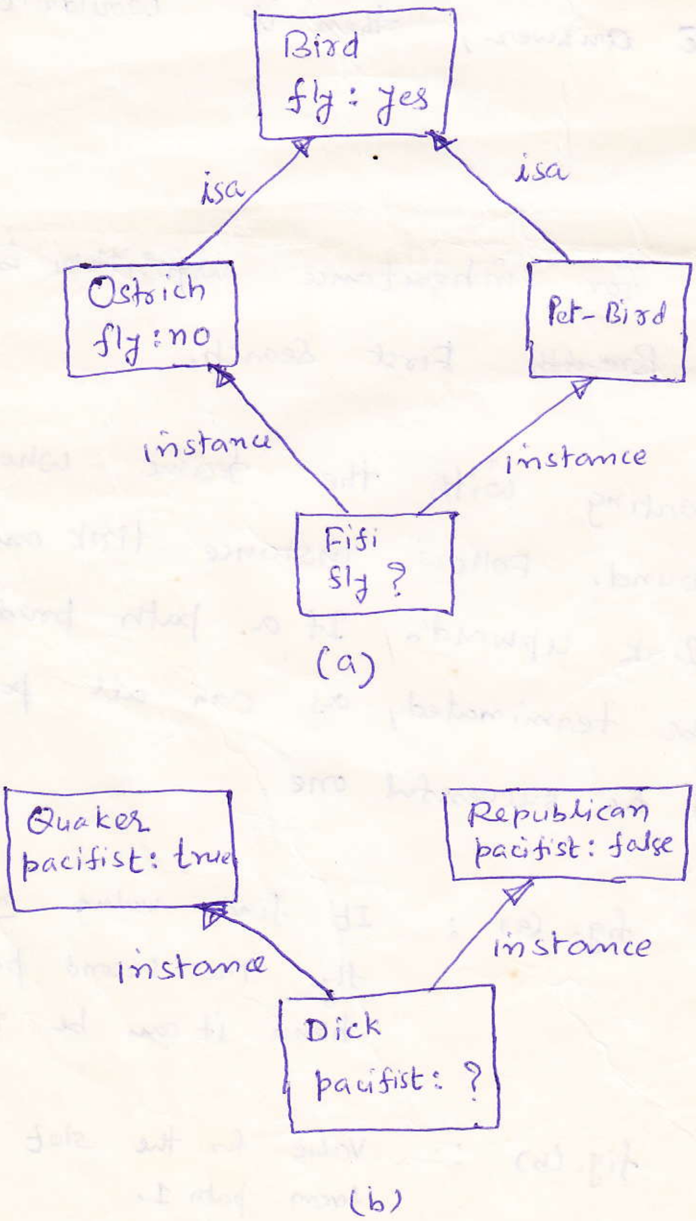
National Leaguer
isa : ML- Baseball - Player
mutually disjoint with : {American Leaguer}

Three-Finger-Brown
~~isa:~~ ~~ML-Ba~~
instance : Pitcher
instance : National-leaguer

Inheritance Algo. Revisited (given in ch. 4)

"isa" hierarchy could be ~~a~~ arbitrary directed acyclic graph (DAG). Hierarchies that are not ~~gra~~ tre are called tangled hierarchies.

- Discussion about algorithm for inheriting values for single-valued slots in a tangled hierarchy.



(a)

(b)

Tangled Hierarchies

In Figure (a), we want to decide whether Fifi can fly. The correct answer is no. If we take Path (2), we get answer as "Fifi can fly"; however if we take Path (1), we get answer as "Fifi can't". So, ~~si~~ we should need ~~think~~ of an algorithms for traversing 'isa' and 'instance' hierarchy that guarantees that specific knowledge will always dominate more general facts.

In Figure (b), we would like to know - Is Dick pacifist ? There are two anewess, and they conflict with each other. If an algorithm finds one of the answers randomly, without looking for alternate answer, then it wouldn't notice ambiguity present.

SO, possible basis for inheritance algorithm is path length while executing Breadth First Search.

Algo. A : DO BFS, starting with the frame whose slot value is to be found. Follow instance link and then follow 'isa' link upward. If a path produces a value, then it can be terminated, as can all paths once their ~~whose~~ length exceeds that of the successful one.

Result of this ~~Thus~~ Algo. for fig. (a) : It finds value no for the slot fly. The second path has length 2, hence it can be terminated.

fig. (b) : Value for the slot pacifist is true form path 1.
Value for the slot pacifist is false form path 2. Both have same

Tangled Hierarchies

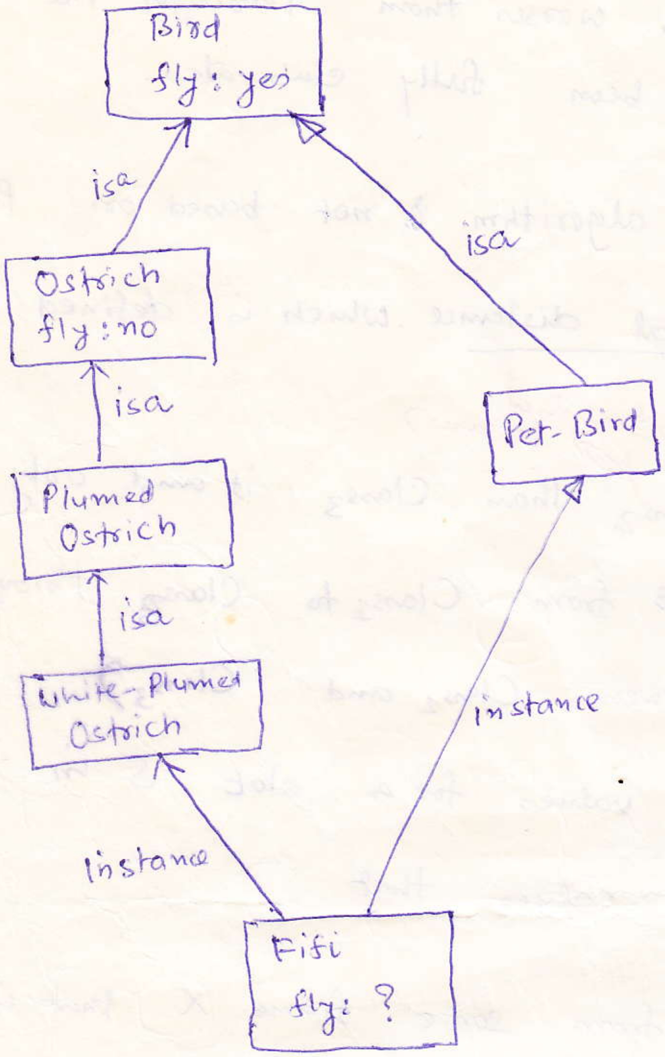


fig. (c)

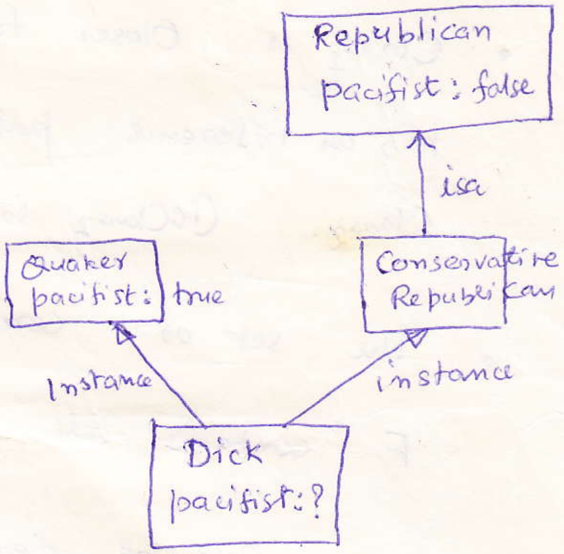


fig. (d)

onsider fig. (c) – Path through Pet-Bird is shorter, and because it is BFS, our algo. would reach Bird before it reaches Ostrich. So Fifi can fly, which is not true.

fig. (d) – From frame Quaker, it finds value for slot pacifist to be true. Now it can't consider other path, as it is lengthier. So, contradict is not found.

Problem : path through the region which ~~has~~ has been d
elaborated looks worser than through the regio
which have not been fully elaborated.

So, let us think of an algorithm ~~b~~ not based on Path
but based on inferential distance which is defined as t

- Class$_1$ is Closer (shorter inferential Distance) to Class$_2$ than Class$_3$ if and only if
has an inference path ~~to~~ from Class$_1$ to Class$_3$ through
Class$_2$. (i.e. Class$_2$ is in between Class$_1$ and Class$_3$)

- The set of competing values for a slot S in a f
F contains all those values that
  - Can be derived from some frame X that is ab
F in the isa hierarchy.
  - Are not contradicted by some frame Y that
a shorter inferential distance to F than X d

For fig. (a) we have two candidate classes from which to get answ
But Ostrich has shorter inferential distance to Fifi than Bi
does. So single answer no for the slot fly of Fifi fo

fig. (b) We get two answers and neither class is closer to
Dick than other and hence contradiction is identifi

fig. (c) Class Ostrich has shorter inferential distance than cla
Bird and hence only answer is no.

No class is closer to Dick. Hence possible values for

The 'Inferential Distance' is defined in such a manner that as long as 'Ostrich is a subclass of Bird', it will be closer to all its instances, no matter how many other classes are added to the system.

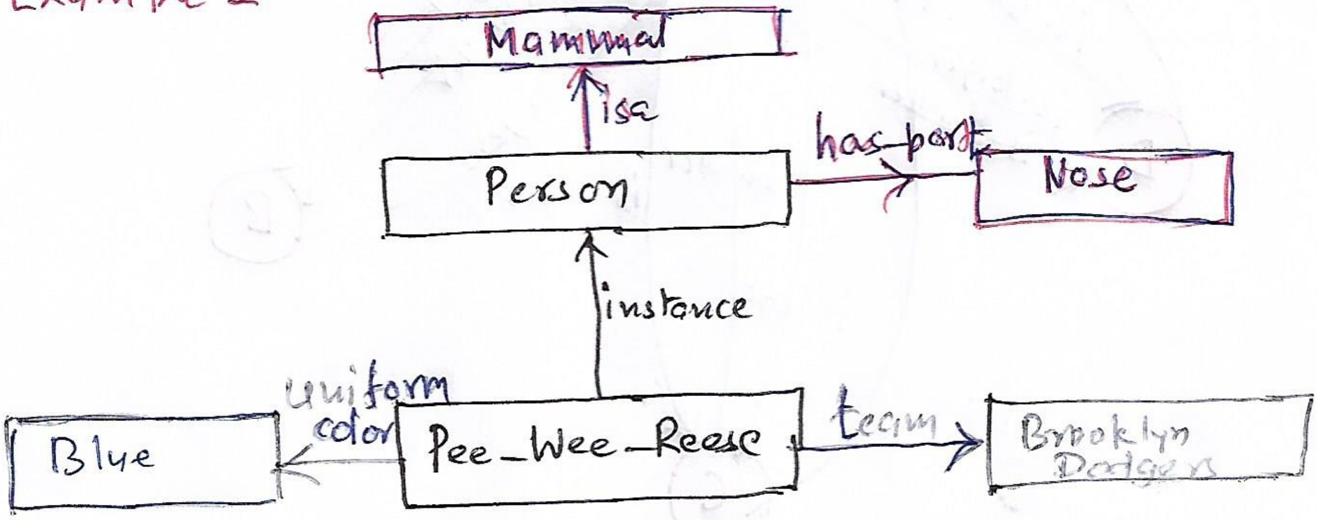
Algo. Property Inheritance - Refer book, ch. 9 Weak slot and filler structures.

Semantic Nets

- The main idea behind Semantic Nets is that the meaning of a concept comes from the ways it is connected to other concepts.

- In a Semantic Net, information is represented as a set of nodes connected to each other by a set of labelled arcs, which represent relationship among nodes.

Examble 1



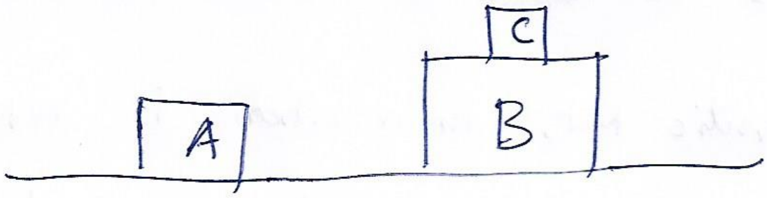
Relations ; isa, instance (General Relations)

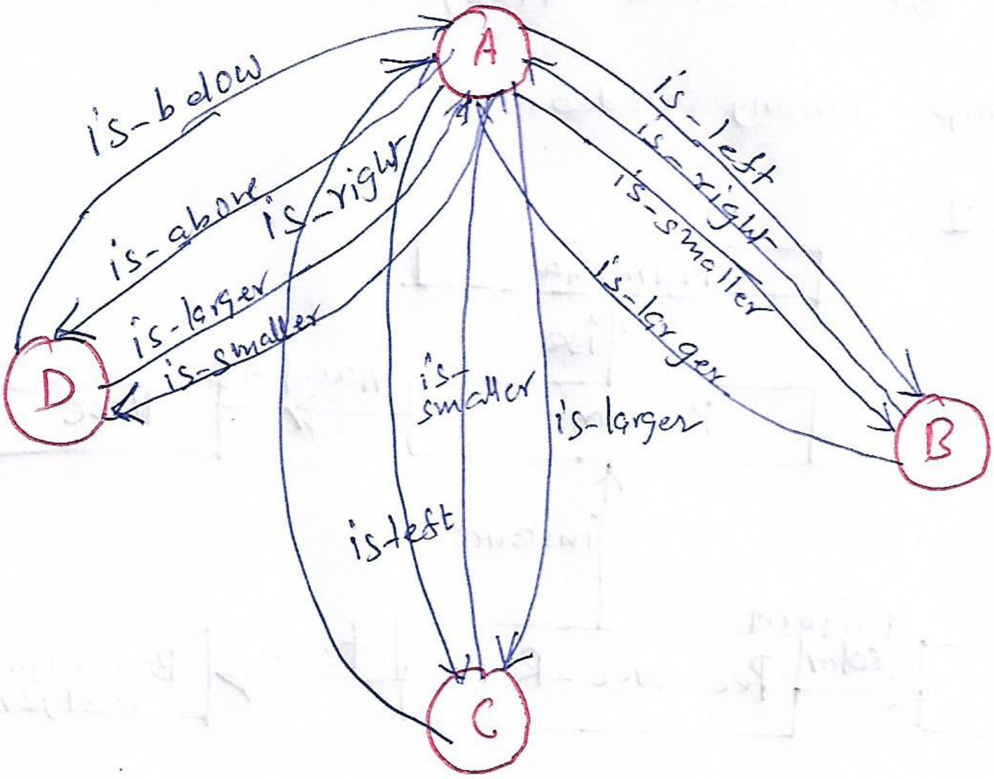
uniform- color, has-part, team (Domain Speific Relations)

Additional relation can be derived through <u>inheritance</u>. eg. has-part (Pee-Wee-Reese, Nose)

Example 2

Semantic Net Representation Of "Block World" system





Here, samples relationship is given among A-B, A-C, A-D.

We need to construct arcs betn B-C, B-D and C-D

- A Semantic Net is a knowledge representation technique
- Mathematically, it can be defined as Labeled Directed Graph.
- Semantic Nets are also called Associative Nets/ Propositional Net.
- Semantic Nets allow Multiple Inheritance. So, an object can belong to more than one category and a category can be a subset of more than one category.

Advantages

- Semantic Nets have the ability to represent default values for categories. This may be overridden by specific values.
- Meaning is transparently conveyed.
- Simple and easy to understand
- Can be easily translated into PROLOG

Disadvantages

- No standard definition for link names.

Representation of relationship (arcs) of Semantic Nets in Logic

Binary Predicates

isa (person, mammal)

instance (Pee Wee Reese, Person)

team (Pee Wee Reese, Brooklyn_Dodgers)

- Predicates with non binary arity can also be represent in Semantic Nets, using general purpose predicates such as "instance".

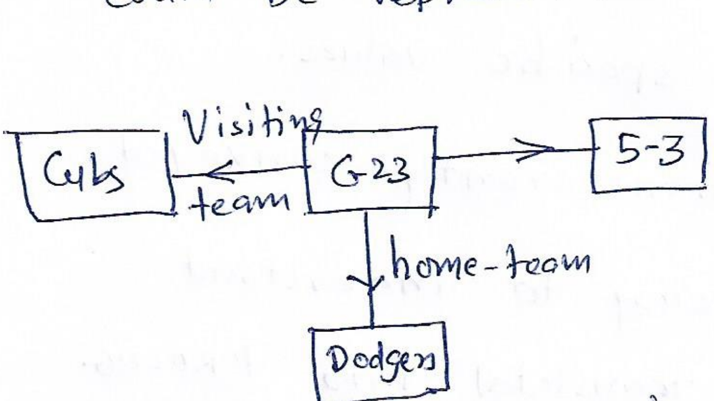
Examples

1. man (Marcus) ← a unary predicate

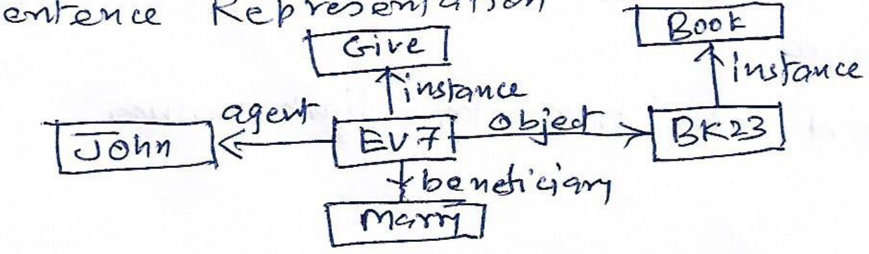
may be represented as instance (Marcus, man)

2. Score (Cubs, Dodgers, 5-3)

Could be represented as:

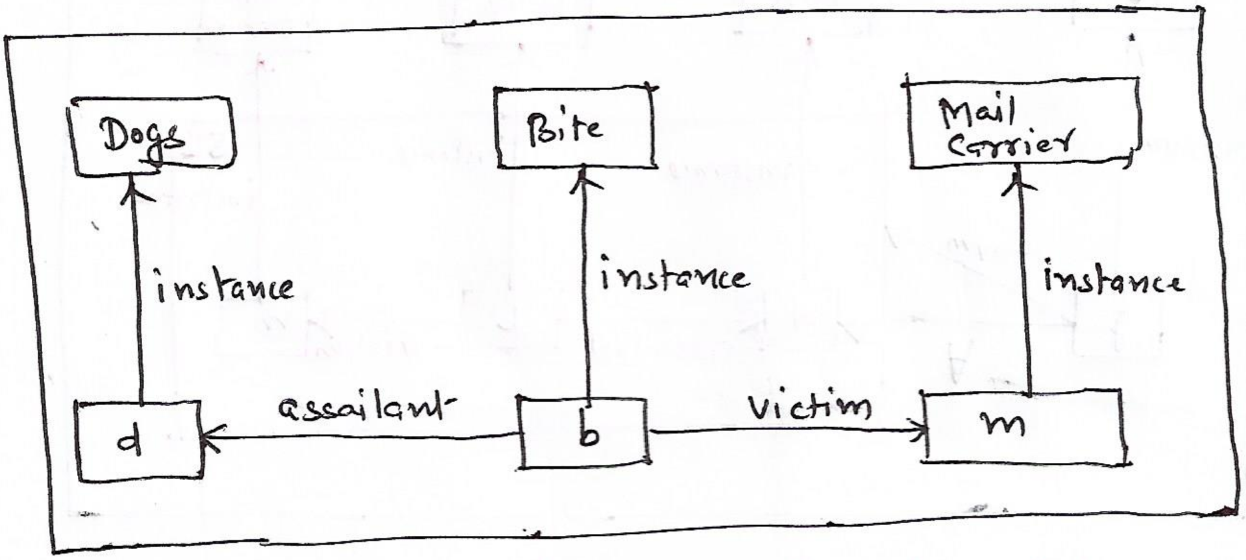


3. Sentence Representation in a Semantic Net



consider the statements

- The dog bit the mail carrier.

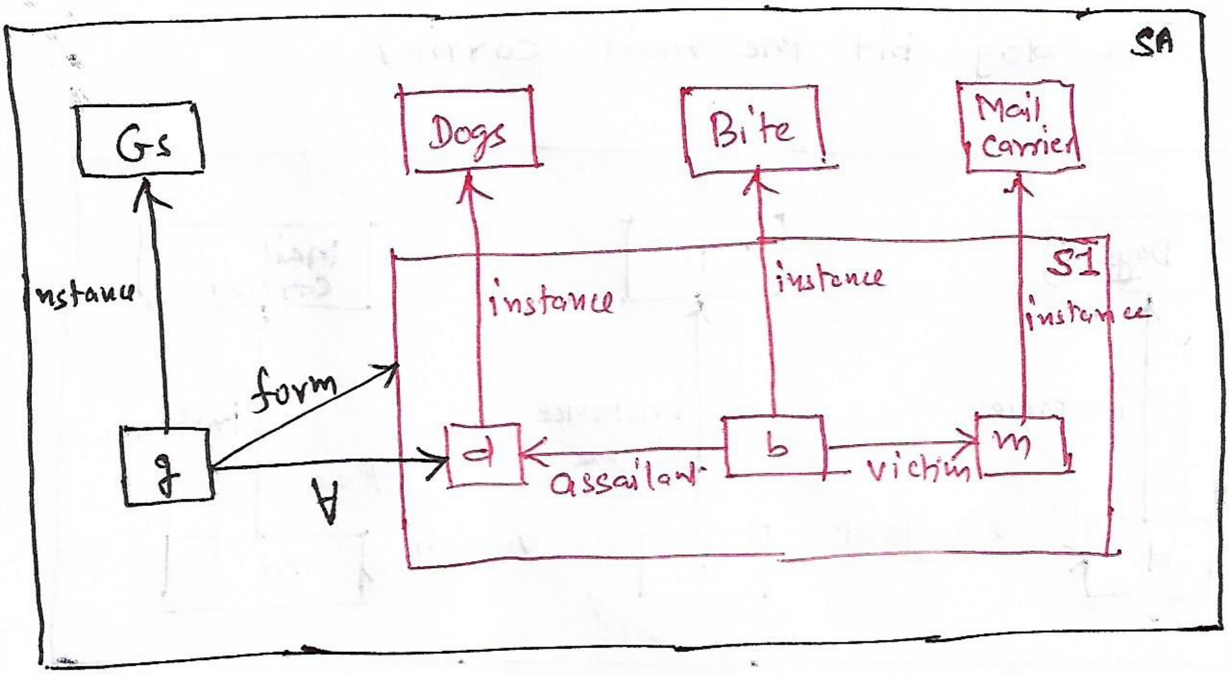


No partition of Semantic Nets

Here, Dogs, Bite and Mail-Carrier are classes of dogs, bitings and mail-carriers respectively. While, nodes d, b and m represent a particular dog, a particular biting and a particular mail carrier.

- Every dog has bitten a Mail Carrier

$$\forall x: dog(x) \longrightarrow \exists y: Mail\_Carrier(y) \land Bite(x, y)$$



Gs : General Statement

Every elements of Gs has atleast two attributes: a form, which states the relation that is being asserted, and one or more ∀ connections, one for each of the universally quantified variable.

Every dog in town has bitten the constable

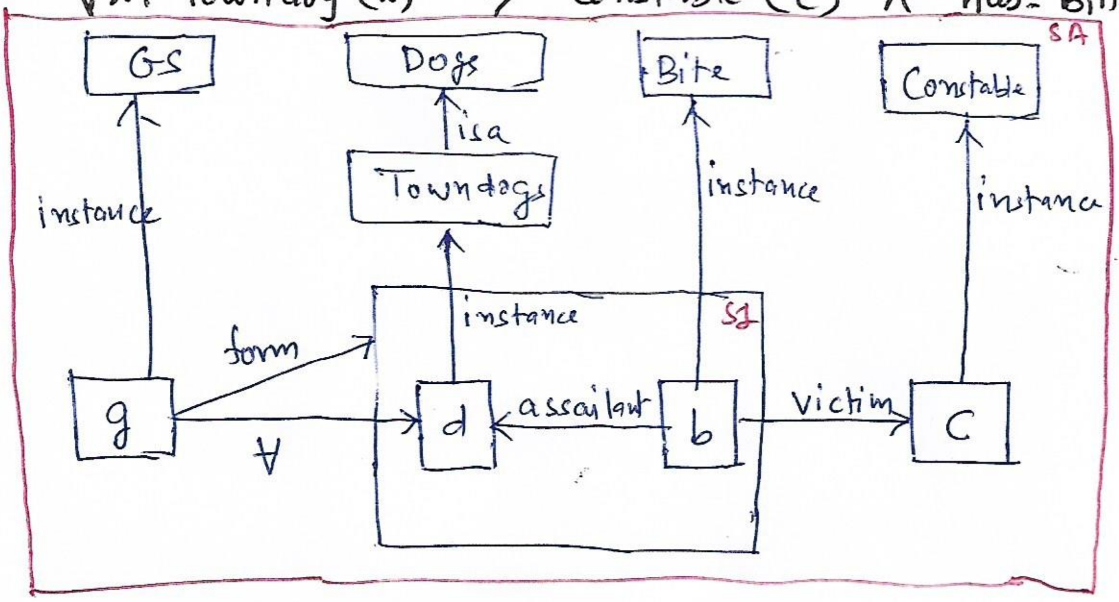
{ In this sentence, we are talking of all dogs of all towns, whereas in the previous one is applicable to all dogs — both from towns, villages etc. }

FOPL - Sentence I

$$\forall x: \text{town}(x) \rightarrow ((\forall y: \text{dog}(y) \wedge \text{lives}(y, x)) \rightarrow \text{constable}(c) \wedge \text{has\_bitten}(y, c))$$

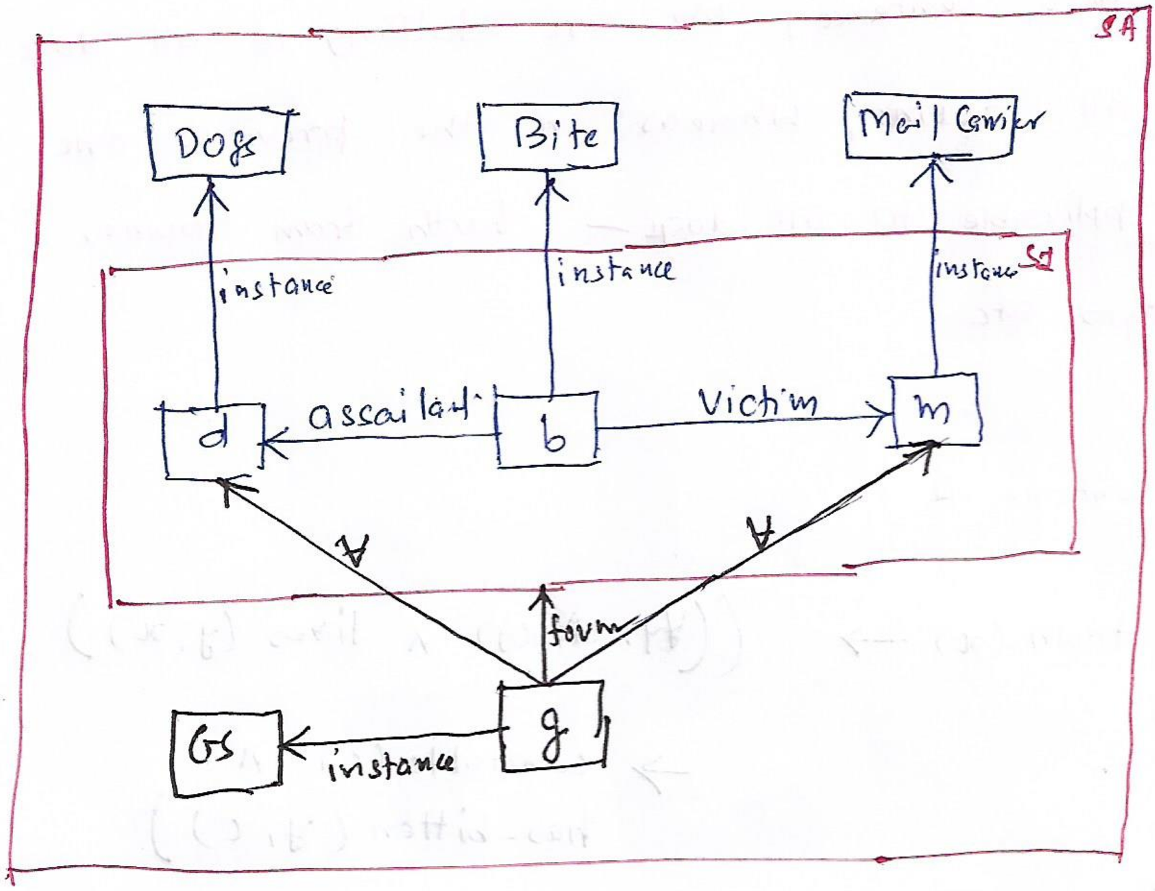
FOPL - Sentence II

$$\forall x: \text{towndog}(x) \rightarrow \text{constable}(c) \wedge \text{has\_bitten}(x, c)$$



- Here, c lies outside the form of General Statement. Therefore, not existentially quantified. Here, it stands for a particular constable.

- Every dog has bitten every mail carrier


SA
Dogs
Bite
Mail Carrier
instance
instance
instance
S1
d
assailant
b
victim
m
∀
∀
form
GS
instance
g