**Inheritance**

From <Chapter 10 Inheritance> in the book <Knowledge Representation and Reasoning> by Ronald J.Brachman and Hector J.Levesque

Hierarchies allow us to avoid repeating representations—it is sufficient to say that "elephants are mammals" to immediately know a great deal about them. Taxonomies of kinds of objects are so fundamental to our thinking about the world that they are found everywhere, especially when it comes to organizing knowledge in a comprehensible form for human consumption, in encyclopedias, dictionaries, scientific classifications, and so on.

In this chapter, we reduce the frames and descriptions of previous chapters to simple nodes that appear in inheritance networks, like the one expressed in Figure 10.1. We will use the following concepts in our discussion:

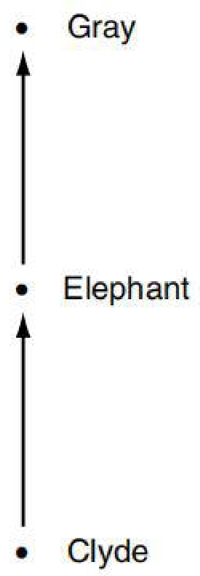


Figure 10.1.

- Edges in the network, connecting one node directly to another. In Figure 10.1, Clyde · Elephant and Elephant · Gray are the two edges.
- Paths included in the network; a path is a sequence of one or more edges. So edges are paths, and in Figure 10.1, Clyde · Elephant · Gray is the only other path.
- Conclusions supported by the paths. In Figure 10.1, three conclusions are supported: Clyde → Elephant; Elephant → Gray; and Clyde → Gray. The last conclusion is supported because the edges represent relationships that are transitive (and so Clyde is a gray thing).

**Strict Inheritance**

In a strict inheritance network, conclusions are produced by the complete transitive closures of all paths in the network. Any traversal procedure for computing the transitive closure will do for determining the supported conclusions.

In a strict inheritance network that is a *directed acyclic graph (DAG)*, the results are the same as for trees: All conclusions you can reach by any path are supported. This includes conclusions found by traversing different branches upward from a node in question. Figure 10.3 illustrates a strict DAG. It says that Ernest is both a student and an employee. The network supports the conclusions that Ernest is an academic, as well as a taxpayer, and salaried. Note that in this figure we introduce a negative edge with a bar through it, between Student and Illiterate, meaning that a student is not an illiterate thing. So edges in these networks have polarity—positive or negative. Thus the conclusion that Ernest is not illiterate is supported by the network in the figure.

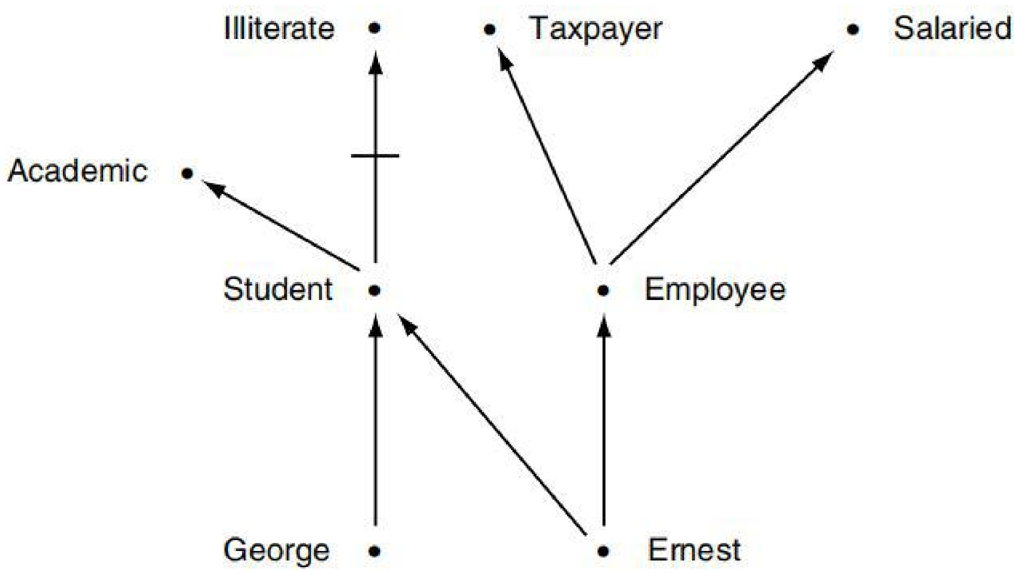


Figure 10.3

**Defeasible Inheritance**

Frame systems such as those that we covered in Chapter 8 would suggest that in fact virtually all properties (and procedures) can be overridden. We call the kind of inheritance in which properties can be defeated defeasible inheritance.

In fact different paths in a network can support conflicting conclusions and a reasoning procedure needs to decide which conclusion should prevail, if any. For example, there is an argument for Clyde being gray: He is an elephant and elephants are gray; however, there is a "better" argument for concluding that he is not gray, because this has been asserted of him specifically. In some cases, we will not be able to say which conclusion is better or worse. In Figure 10.5 there is nothing obvious that tells us how to choose between the positive or negative conclusions about Nixon's pacifism. The network tells us that by virtue of his being a Quaker he is a pacifist; it also tells us that by virtue of his being a Republican, he is not. This type of network is said to be ambiguous.

When exploring different accounts for reasoning under this kind of circumstance, we typically see two types of approaches: *credulous* (or brave or choice) accounts allow us to choose arbitrarily between conclusions that appear equally well supported; skeptical (or cautious) accounts are more conservative, often accepting only conclusions that are not contradicted by other paths. In the Nixon case, a credulous account would in essence flip a coin and choose one of Nixon → Pacifist or Nixon → ¬Pacifist, because either conclusion is as good as the other. A skeptical account would draw no conclusion about Nixon's pacifism.

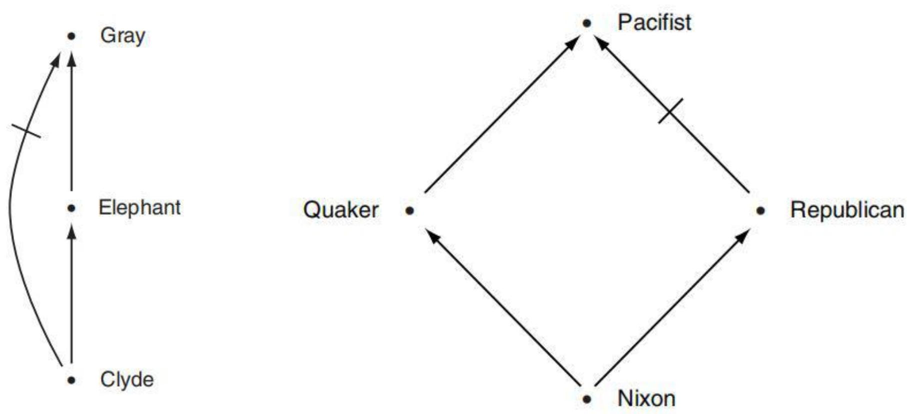


Figure 10.5

**STRATEGIES FOR DEFEASIBLE INHERITANCE**

For DAGs with defeasible inheritance, we need a method for deciding which conclusion to choose (if any) when there are contradictory conclusions supported by different paths through the network. In this section, we examine two possible ways of doing this.

**The Shortest Path Heuristic**

To decide in an automated way, the shortest path heuristic says that we should prefer conclusions resulting from shorter paths in the network. In Figure 10.6 Because there are fewer edges in the path from Clyde to Gray that includes the negative edge than in the path that includes the positive edge, the negative conclusion prevails. In the network on the right, we see the opposite polarity conclusion being supported. Whales are mammals, but mammals are typically not aquatic creatures. Whales are exceptional in that respect, and are directly asserted to be aquatic creatures. We infer using the shortest path heuristic that BabyBeluga is an AquaticCreature.

The intuition behind the shortest path heuristic is that it makes sense to inherit from the most specific subsuming class.

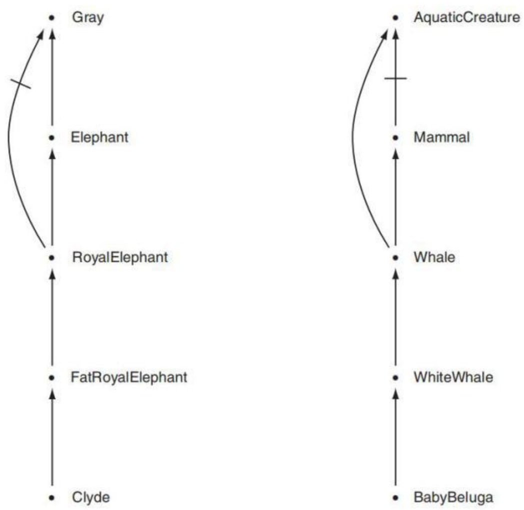


Figure 10.6

However, the shortest path heuristic has serious flaws. Firstly, it can produce intuitively incorrect answers in the presence of redundant edges—those that are already implied by the basic network. For example Figure 10.7, by creating an edge directly from Clyde to Elephant we have inadvertently changed the polarity of the conclusion about Clyde's color. The path from Clyde to Gray that goes through edge q is now shorter (length = 2) than the one with the negative edge from RoyalElephant to Gray (length = 3). So the inclusion of an edge that is already implicitly part of the network undermines the shortest path heuristic.

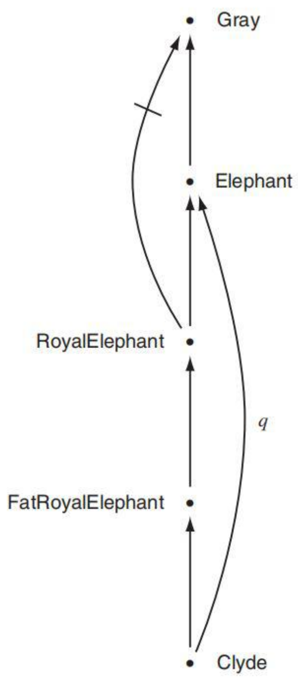


Figure 10.7

Another problem with the shortest path heuristic is the fact that the length of a path through the network does not necessarily reflect anything salient about the domain. To show an extreme example, in the Figure 10.8 network, the shortest path heuristic supports the positive conclusion. But if we were to add another two edges—anywhere in the path—to the left-hand side, the conclusion would be reversed. This seems rather silly; the network should be considered ambiguous in the same manner as the one in Figure 10.5.

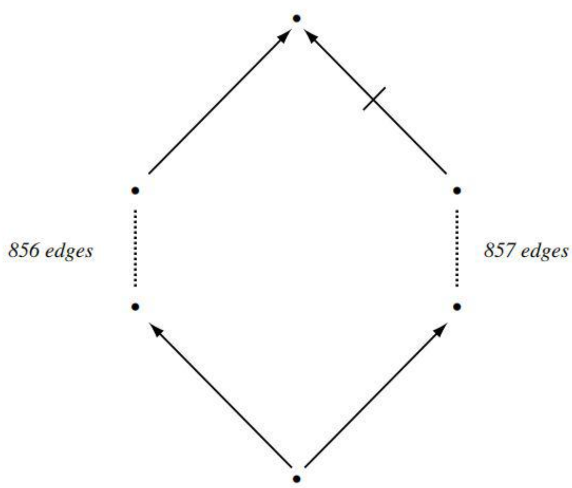


Figure 10.8

**Inferential Distance**

A more plausible strategy is inferential distance, which rather than being linear distance based, is topologically based. Consider Figure 10.7 once again. Starting at the node for Clyde, we would like to say that RoyalElephant is more specific than Elephant despite the redundant edge q because there is a path to Elephant that passes through RoyalElephant. Because it is more specific, we then prefer the negative edge from RoyalElephant to Gray over the positive one from Elephant to Gray. This criterion handles the earlier simple cases of inheritance from Figure 10.6. In the case of the ambiguous network of Figure 10.8, inferential distance prefers neither conclusion, as desired.

Unfortunately, inferential distance has its own problems. What should happen, for example, when the path from a through b to c is itself contradicted by another path?

**Extensions**

We do not expect an ambiguous network to specify a unique set of conclusions. We use the term extension to mean a possible set of beliefs supported by the network. Ambiguous networks will have multiple extensions. Figure 10.11 illustrates an ambiguous network and Figure 10.12 shows its two credulous extensions.

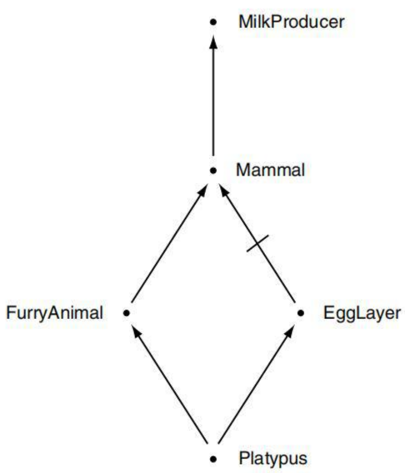


Figure10.11

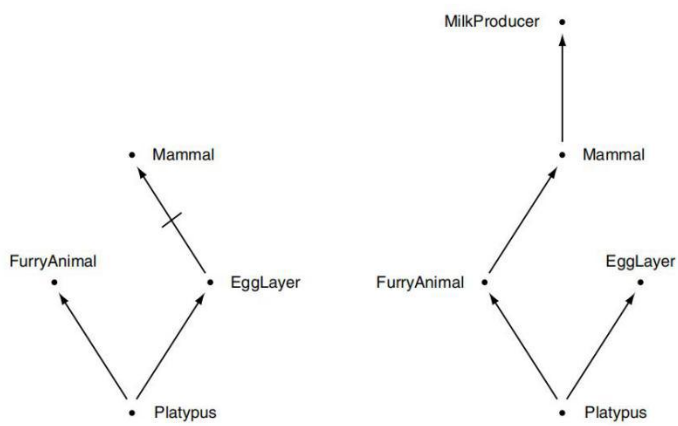


Figure 10.12

Although we have detailed some reasonable formal definitions that allow us to distinguish between different types of extensions, an agent still needs to make a choice based on such a representation of what actually to believe. The extensions offer sets of consistent conclusions, but one’s attitude toward such extensions can vary. Different forms of reasoning have been proposed based on the type of formalization we have presented here:

- credulous reasoning: Choose a preferred extension, perhaps arbitrarily, and believe all of the conclusions supported by it.
- skeptical reasoning: Believe the conclusions supported by any path that is present in all preferred extensions.

Inheritance networks, as described in the text, grew out of a more general-purpose representation formalism that used inheritance, known as **semantic networks**. Early and influential work in this area, inspired by the semantics of verbs and nouns in natural language, was done by Quillian and, for the application of computer vision, Winston.

**Extended paper reading:**

**<OKBC: A Programmatic Foundation for Knowledge Base Interoperability>**

OKBC substantially advances our abil  ity to achieve interoperability between KRSs and client  side KB tools. Its success is shown by its use with a broad range of systems, one of which is being used in a commercial environment. Availability of OKBC imple  mentations in multiple programming languages makes it an attractive choice for the developers of applica  tions that make use of the content and services pro  vided by KRSs. They can have increased confidence that their applications will interoperate with multiple KRSs.