

Knowledge Engineering in Logic

Knowledge engineering is the methodical construction of knowledge bases. Because the end result is a computer program, it is natural to apply concepts from software engineering.

Nevertheless, the book identifies several steps that especially apply to knowledge engineering. The steps cannot be done independently, or in a simple sequence, but typically one will have to go backward and forward between them. The digital circuits example is taken from the book.

Identify the task.

What questions do we want the knowledge base to answer? What kinds of facts will be available for a specific problem instance.

Example: What is the output of a digital circuit? What inputs results in a given output? Based on some of the inputs and outputs, we might know what the other inputs and outputs should be based on the requirements of the circuit.

Assemble the relevant knowledge.

How does the domain work? There might be a known set of rules that govern the domain. If this an area unfamiliar to the knowledge engineer, *knowledge acquisition* from a human expert is needs.

Example: For digital circuits, the rules for gates are well-known.

Decide on a vocabulary.

What *ontology* (predicates, constants, and functions) will be used to encode the domain?

Example: Should we explicitly represent the connections with predicates and signals with functions (as in Section 8.4) or implicitly with a constant (one way to do Exercise 8.18)?

Encode general knowledge.

What are the *axioms* (rules, sentences) of the domain? This should uncover many of the defects in the vocabulary. It should be “natural” to encode the axioms of the domain.

Example: Encode the relationship between the inputs and output of OR gates.

Encode specific problem instances.

How will the facts of specific problem instances be encoded? For an agent, these come from the sensors?

Example: What are the values of some inputs/outputs of the circuit? What do we want the other inputs/outputs to be?

Test the knowledge base with queries.

Test the knowledge base with the problem instances.

Debug the knowledge base.

If the outcome is not what was expected, trace the result to incorrect/missing knowledge or an incorrect circuit.