Expert System Shells - Examples

- EMYCIN: Shell taken from MYCIN, backward chaining, uncertainty factors, explanation facility
- CLIPS: Public doman ESS from NASA, combines three paradigms: rule-base, object-oriented, procedural
- JESS: Successor to CLIPS, entirely in Java, can execute Java code

JESS

- LISP-like notation – expressions are written as whitespace separated lists delimited by parentheses, e.g., (a b c), (+ 2 3)
- Variables are identifiers that begin with a question mark, e.g., ?color
- Every fact has a *template* (something like a Java class). Fields are called *slots* and are unordered.

Template Example

The *template* construct creates templates:

```
(deftemplate automobile
  "a specific car"
  (slot make)
  (slot model)
  (slot year (type INTEGER))
  (slot color (default white)))
```

Jess (cont'd)

- To add a fact to working memory, we assert it, e.g., (assert (spicy chili))
- To assert several facts at once, we can use the deffacts construct:

```
(deffacts my-fact
  (spicy chili)
  (spicy thai)
  (sweet candy))
```

Jess (cont'd)

- To add a fact to working memory, we assert it, e.g., (assert (spicy chili))
- To assert several facts at once, we can use the deffacts construct:

```
(deffacts my-fact
  (spicy chili)
  (spicy thai)
  (sweet candy))
```

Defining Rules

The `defrule` constructs creates a new rule:

```
(defrule
   (spicy ?x)
   (sweet ?x)
  =>
   (assert (delicious ?x)))
```

Note that the if-part is not executed, but is a pattern that is matched against the working memory.

Special Functions

- The *printout* function is used to print out information: `(printout t "I chose " ?x "." crlf)`
  The "t" means standard output.
- The `reset` function re-initializes the working memory.
- The `run` function cause the inference engine to start.
- `(watch all)` traces rule execution.

Forward Chaining – Data Driven

In a forward chaining, or data driven, expert system, inference starts with the know facts, and then uses the rules to derive new facts, in the hope of reaching a particular conclusion.

The search may be haphazard.

Once the search finishes, the path from the start to the goal may be listed and other paths ignored.

Forward Chaining (cont'd)

Search may be undirected if care is not taken.

To direct the search along a particular set of inferences, we can put markers in the working memory along the way. Rules could be of the form:

```
   (= ?stage 1)
   <other facts>
=>
   <stage 1 deduction>
   (assert (stage 2)))
```

which would restrict the firing of this rule to a particular stage in the inference.

Heuristic Control

Similarly, heuristic control of the rules could be built in. Rules can be group, and only certain rules fired under given conditions by asserting the appropriate facts in the working memory (and retracting them when necessary).

Uncertainly

Some expert systems allow for uncertainty in their deductions. Both evidence and rules have uncertainly factors. The uncertainty of the facts that trigger a rule is combine with the uncertainty of the rule itself to derive the uncertainty of the conclusion. See MYCIN.

Model-Based Reasoning

An alternative to rule-based reasoning is *model-based reasoning*. Here, a *model* or *simulation* is used to test hypotheses under different conditions. The model is using a computer program whose parameters are accessible and changeable by the expert system. An example might be a computer simulation of a logic device where a faulty component may be creating an error.

Model-Based Reasoning (cont'd)

If the observations differ from the expectations that the model provides, the parameters may be changed until the two coincide.

Models may have different levels of granularity.

A problem is that it must be decided beforehand what features are salient to the model and which are ignored by the model (*abstraction*).

Models are based on a theoretical understanding of the problem.

Model-Based Reasoning - Example

- Livingstone is a model-based expert system developed at NASA to detect and isolate failures in space systems.
- It uses a hierarchical model of components.
- Each component is modeled using a finite-state machine.
- Used to model the Deep Space One spacecraft, the X-37 spaceplane, and the EO-1 satellite.

Case-Based Reasoning

- An alternative approach to both rule- and model-based reasoning in case-based reasoning.
- A library, or database, of previously solved cases is kept.
- The current problem is compared to the database by matching salient features. The best match or matches are used.
- The previous solution is adapted to the present case.

CBR (cont'd)

The steps in case-bases reasoning are:

- Retrieve appropriate cases from memory
- Modify a retrieved case to apply to the current problem
- Apply the transformed case
- Save the new solution along with a record of success or failure

CBR (cont'd)

Difficulties include:

- How do you match cases?
- Do the features have to be identified beforehand and indexex?
- What does similarity mean?
- Slows down past a point when cases are added.

CBR - Example

ARCHIE is an interactive design-aiding system for architectural design. It supports construction and evaluation of solutions. Users specify their problem descriptions and/or solution descriptions; the system retrieves and displays past designs and provides suggestions and warnings. (Kolodner)