The Architecture of Knowledge-based Expert System

Knowledge Base / Rules Facts

Inference Engine

Knowledge Acquisition Subsystem

Explanation Subsystem

User Interface

User

Expert or Knowledge Engineer
Knowledge Acquisition Subsystem

We can get expert’s knowledge from the following resource, then put them into knowledge base.

1. Tech Watch Columns

   Such as: Ted Lewis, Computer Magazine
            Bob Glass, Journal of Systems and Software
            Steve McConnell, Software Magazine

   For example:

   We can get the following rules from Ted Lewis’s articles:

   **Rule 1:** Innovation + high risk $\rightarrow$ fail
   **Rule 2:** Innovation + low risk $\rightarrow$ success
   **Rule 3:** There is venture fund in innovation $\rightarrow$ low risk
Knowledge Acquisition Subsystem

**Rule 4:** Technology + Before market $\rightarrow$ success
**Rule 5:** Technology + After market $\rightarrow$ fail

**Fact 1:** Digital Radio Broadcast Satellites (DARS) is after market
**Fact 2:** Old economy company is transferring into Agile, e-business economy;
**Fact 3:** Venture capital is increasing very quickly for research.
**Fact 4:** Traditional product development is a sequential model.
**Fact 5:** Viral money system is being formed.
**Fact 6:** The current e-commerce is too overcrowded.
Knowledge Acquisition Subsystem

2. Tech News

For example:

- IEEE tech news
- ACM tech news
- Gartner tech news
- MCC tech news
- Software Productivity Consortium tech trends

*From these tech news, we can get a lot of facts about current tech trends.*

Fox example:

**Fact 1.** B2B is long on promise, but still short on results.
Knowledge Acquisition Subsystem

Fact 2: Sales of desktop computers and servers are slowing down

Fact 3: U.S. home Internet access penetration levels jumped significantly during September

3. Tech Watch Companies/ Institutes

   For example:
   
   Gartner Group
   
   Chasm Group
   
   MCC
   
   Software Productivity Consortium

We can also get rules and facts from them.
Inference Engine

Inference Engine compares the user’s information with the knowledge in knowledge base, and derives whatever conclusions may logically follow.

There are two common inference method:

Backward-chaining: Inference Engine guesses a conclusion and then attempts to prove that this guess is correct.

Forward-Chaining: Inference Engine compares the facts with the IF part of a rule, and fires a conclusion from the THEN part of that rule.

According to the goal of tech watch, Our Expert system will chose the Forward-Chaining as the main inference Engine.
Explanation subsystem and User Interface

An appropriate explanation subsystem and user interface should be constructed, to help user query expert system and understand the result.

There are variety of interface techniques available today, such as Web-based, graphics, natural language, We can construct an expert system based on Web.
Expert System Development Tools

There are lots of expert system development tools available:

LISP
PROLOG
CLIPS
JESS

and lots of other tools on http://www.pcai.com/pcai