

# BEST

## Blackboard-based Expert System Toolkit

**BEST** is a toolkit aimed at developing complex expert systems. This tool is based on the blackboard architecture and enables the integration of various artificial intelligence techniques with conventional programming techniques. It can be used to solve various problems including diagnostics, classification, planning, allocation, and scenario-making, in different domains such as economy, medicine, mechanical engineering, electrical engineering, quality control, environment protection, etc.

### Features

**BEST** provides the following features:

- **Prolog/Rex** knowledge representation language
- Inference engine (deductive, inductive, hybrid)
- Several search strategies (A\*, best first, depth first, hill climbing, branch-and-bound, etc.)
- Uncertainty handling (fuzzy logic, probability)
- **MEKON** - truth maintenance system
- Multiple paradigms integration
- Explanation and justification capability

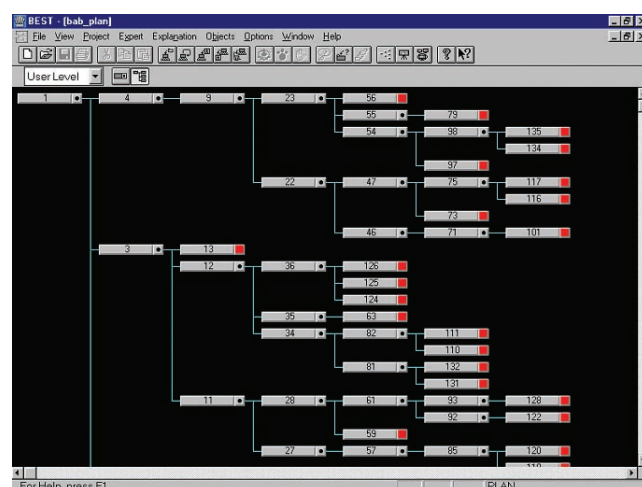
**Prolog/REX** facilitates the representation of declarative and procedural knowledge. The declarative knowledge can be represented using concepts (frames), relations, demons, facts, hypotheses and contexts. On the other hand, the procedural knowledge can be defined using forward-and backward-chaining rules, constraint rules, Domain Knowledge Source Activation Rules (DKSAR), set-control rules and other programming language procedures.

Inference engine provides deductive, inductive and mixed initiative reasoning, different search strategies (depth-first, breadth-first, branch-and-bound, hill-climbing, beam, A\*), and uncertainty handling (probability, certainty factors, fuzzy logic).

**MEKON** represents a truth maintenance system implemented within **BEST**. Integrated with **Prolog/Rex** and **BEST**'s inference engine, **MEKON**

provides a simple and efficient means for the examination of search spaces. **MEKON** facilitates the solution of different kinds of real search problems concerning planning, diagnosis, allocation tasks, classification tasks or decision making.

A complex problem can be decomposed into several subproblems represented by knowledge sources in **BEST**. These knowledge sources can be implemented using one of the following programming paradigms: rule-based, logic, procedural, object-oriented or access-oriented programming, model-based reasoning, and intelligent databases.



A decision tree for the branch-and-bound search strategy

Depending on the user, **BEST** distinguishes two types of explanations: for the end user, and for the system builder. **BEST** provides two standard types of explanations HOW and WHY, while the level of details can be adapted to the user's knowledge and experience. WHAT IF type of explanation is also provided, which is crucial for sensitivity analysis.

### Technical information

**BEST** is implemented using MS Visual C++ and Arity Prolog. It runs in the MS Windows NT, 2000 and XP environment.

[www.pupin.com](http://www.pupin.com)