

# INTRODUCTION AND ARCHITECTURE OF EXPERT SYSTEM

BY  
PREM DESHMANE

# What is Expert System ?

- An expert system, is an interactive computer-based decision tool that uses both facts and heuristics to solve difficult decision making problems, based on knowledge acquired from an expert.
- Inference engine + Knowledge = Expert system  
( Algorithm + Data structures = Program in traditional computer )
- First expert system, called DENDRAL, was developed in the early 70's at Stanford University.

# INTRODUCTION


Expert systems are computer applications which embody some non-algorithmic expertise for solving certain types of problems. For example :

- Diagnostic applications
- Play chess
- Make financial planning decisions
- Configure computers
- Monitor real time systems
- Underwrite insurance policies
- Perform many services which previously required human expertise.



# Expert System Shells

- Many expert systems are built with products called expert system shells. A shell is a piece of software which contains the user interface, a format for declarative knowledge in the knowledge base, and an inference engine. The knowledge and system engineers use these shells in making expert systems.

- 
- **Knowledge engineer** : uses the shell to build a system for a particular problem domain.
  - **System engineer** : builds the user interface, designs the declarative format of the knowledge base, and implements the inference engine.

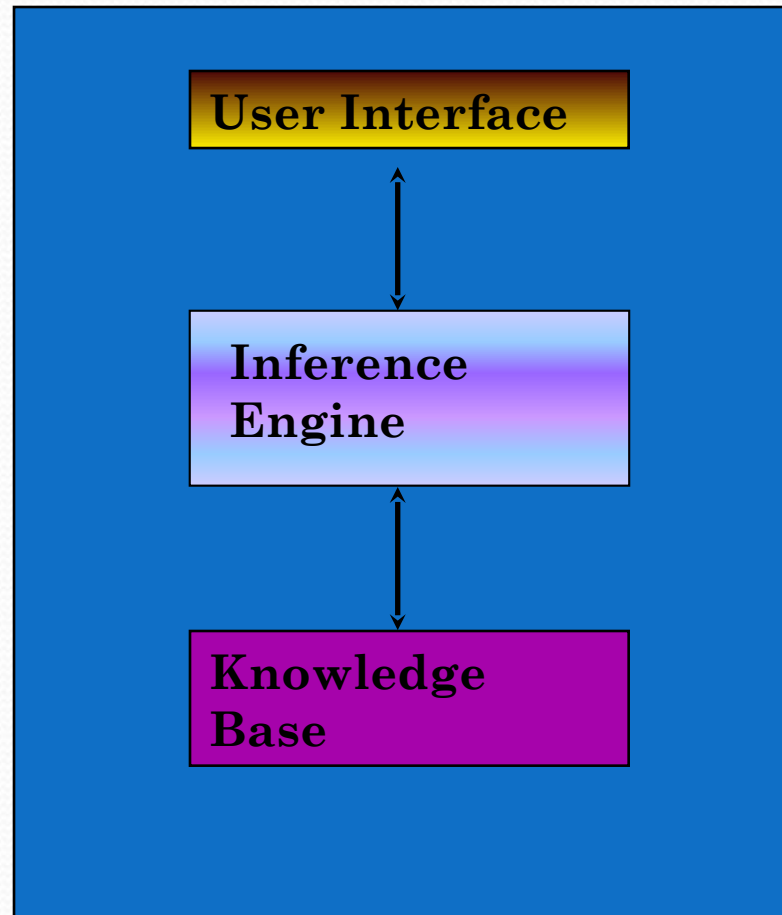
Depending on the size of the system, the knowledge engineer and the system engineer might be the same person.



# Human Expert Behaviors

- Recognize and formulate the problem
- Solve problems quickly and properly
- Explain the solution
- Learn from experience
- Restructure knowledge
- Break rules
- Determine relevance
- Degrade gracefully

# Three Major ES Components



- The knowledge base contains the knowledge necessary for understanding, formulating, and solving problems.
- Two Basic Knowledge Base Elements
  - Facts & Special heuristics, or rules that direct the use of knowledge
- The Inference Engine, *brain* of the ES.
- The control structure (rule interpreter)
- Provides methodology for reasoning



# Expert System Architecture

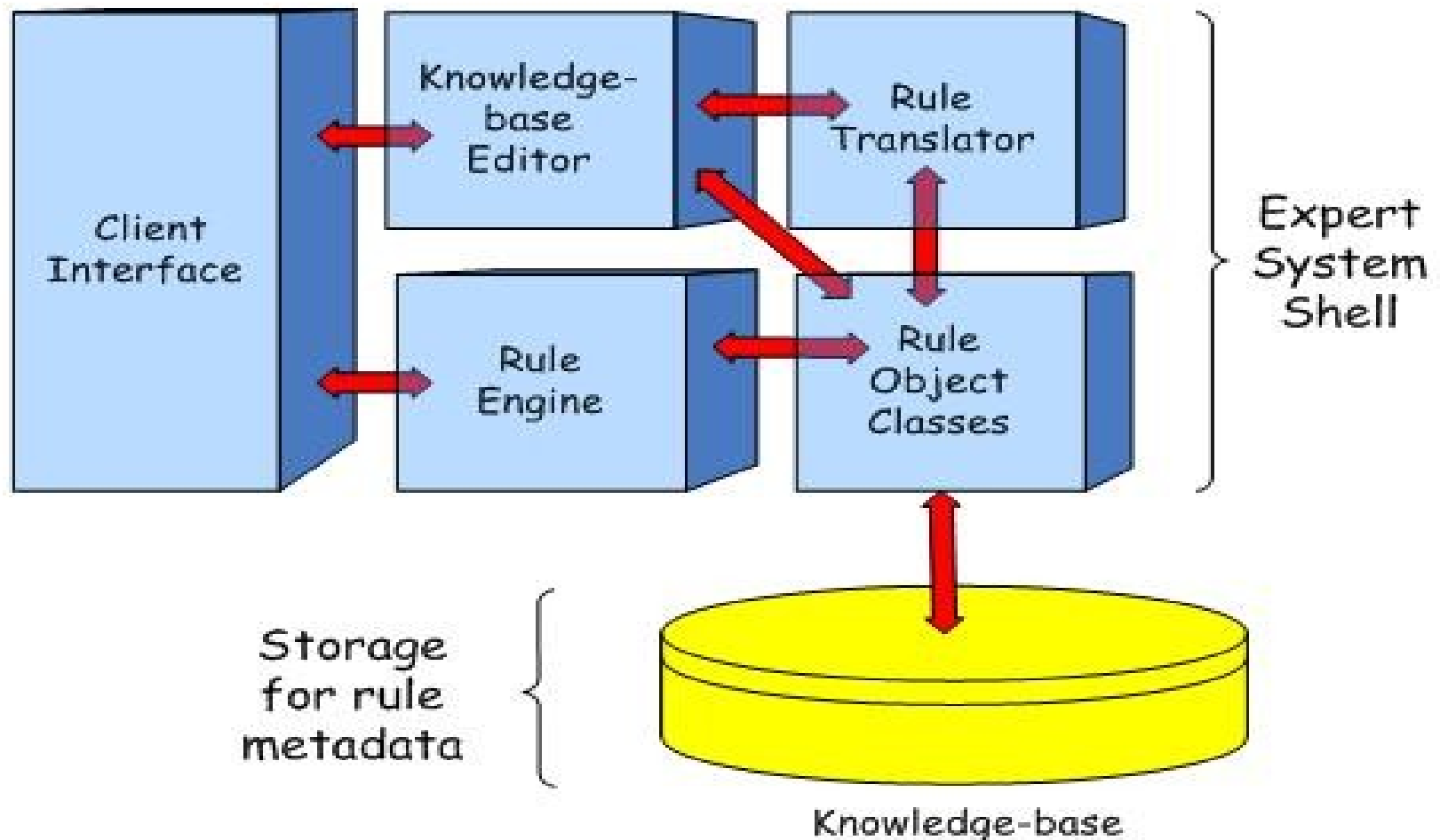
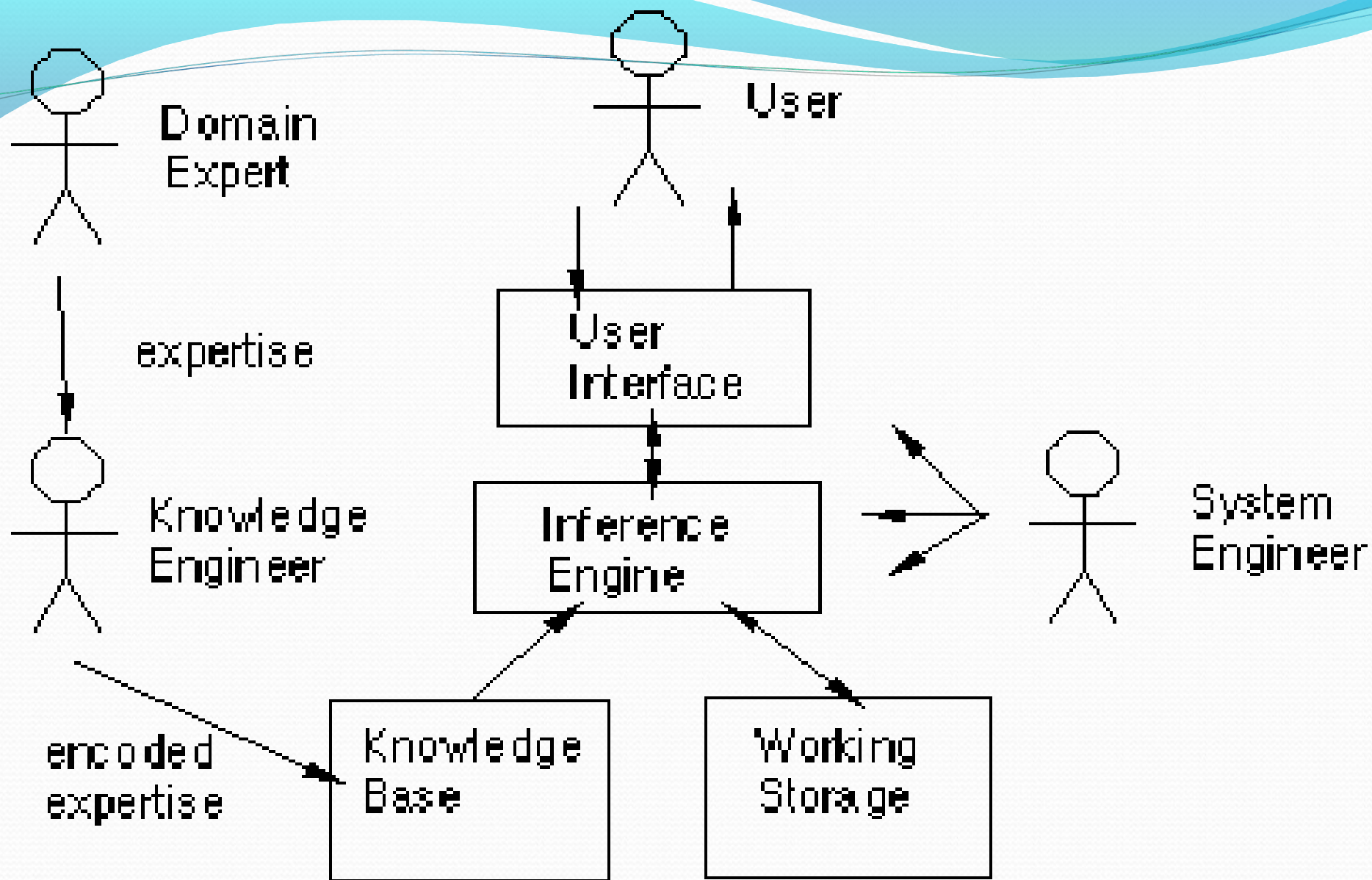


Figure 2 - Expert System Architecture

- The Client Interface processes requests for service from system-users and from application layer components.
- The Knowledge-base Editor is a simple editor that enable a subject matter expert to compose and add rules to the Knowledge-base.
- Rule Translator converts rules from one form to another i.e; their original form to a machine-readable form.
- The Rule Engine(*inference engine*) is responsible for executing Knowledge-base rules.
- The shell component, Rule Object Classes, is a container for object classes supporting.



## Expert System Components And Human Interfaces



# Components and Interfaces

- **User interface** : The code that controls the dialog between the user and the system.
- **Knowledge base** : A declarative representation of the expertise often in IF THEN rules .
- **Inference engine** : The code at the core of the system which derives recommendations from the knowledge base and problem specific data in working storage.
- **Working storage** : The data which is specific to a problem being solved.

# Roles of Individuals who interact with the system

- **Domain expert** : The individuals who currently are experts in solving the problems; here the system is intended to solve.
- **Knowledge engineer** : The individual who encodes the expert's knowledge in a declarative form that can be used by the expert system.
- **User** : The individual who will be consulting with the system to get advice which would have been provided by the expert.
- **System engineer** : builds the user interface, designs the declarative format of the knowledge base, and implements the inference engine.



# **Expert System Benefits**

- **Increased Output and Productivity**
- **Decreased Decision Making Time**
- **Increased Process and Product Quality**
- **Reduced Downtime**
- **Capture Scarce Expertise**
- **Flexibility**
- **Easier Equipment Operation**
- **Elimination of Expensive Equipment**



- **Operation in Hazardous Environments**
- **Accessibility to Knowledge and Help Desks**
- **Integration of Several Experts' Opinions**
- **Can Work with Incomplete or Uncertain Information**
- **Provide Training**
- **Enhancement of Problem Solving and Decision Making**
- **Improved Decision Making Processes**
- **Improved Decision Quality**
- **Ability to Solve Complex Problems**
- **Knowledge Transfer to Remote Locations**
- **Enhancement of Other MIS**

# Expert System Limitations

- Knowledge is not always readily available
- Expertise can be hard to extract from humans
- Each expert's approach may be different, yet correct
- Hard, even for a highly skilled expert, to work under time pressure
- Expert system users have natural cognitive limits
- ES work well only in a *narrow domain* of knowledge



- **Most experts have no independent means to validate their conclusions**
- **Experts' vocabulary often limited and highly technical**
- **Knowledge engineers are rare and expensive**
- **Lack of trust by end-users**
- **Knowledge transfer subject to a host of perceptual and judgmental biases**
- **ES may not be able to arrive at valid conclusions**
- **ES sometimes produce incorrect recommendations**



# References

- *"Artificial Intelligence", by Elaine Rich and Kevin Knight, (2006), McGraw Hill*
- *"Introduction To Artificial Intelligence & Expert Systems", by Dan W Patterson.*
- *"Expert Systems: Introduction To First And Second Generation And Hybrid*
- *Knowledge Based Systems", by Chris Nikolopoulos.*
- *"Artificial intelligence and expert systems for engineers", by C. S. Krishnamoorthy.*
- *S. Rajeev, (1996), CRC Press INC, page 1-293.*