

# **1**

## **INTRODUCTION 1**

- 1.1 Artificial Intelligence 2**
- 1.2 Expert Systems and Knowledge Engineering 3**
- 1.3 Benefits of Applying Expert System Technology 3**
- 1.4 Expert Systems and Experts 5**
- 1.5 Roles of Expert Systems 6**
- 1.6 Application Areas of Expert Systems 7**
- 1.7 Special Capabilities and Features of Expert Systems 7**
- 1.8 Limitations of Expert Systems 8**
- 1.9 Developing an Expert System 9**

# **2**

## **BASIC CONCEPTS OF EXPERT SYSTEMS 11**

- 2.1 Developing an Expert System 12**
- 2.2 Knowledge Acquisition 12**
- 2.3 Development of an Expert System Program 16**
- 2.4 Structure of an Expert System 17**
- 2.5 AI Knowledge Representation Paradigms 18**
  - 2.5.1 Production Rules 18**
  - 2.5.2 Frames, Inheritance, and Object-Oriented Programming 23**

# 3

## PLANNING AND MANAGING THE DEVELOPMENT 29

- 3.1 Initial Phases 30**
  - 3.1.1 Project Start-up 30**
  - 3.1.2 Selection of the Domain 34**
  - 3.1.3 Selection of the Development Environment 36**
- 3.2 Core Development Phases 38**
  - 3.2.1 Development of a Feasibility Prototype System 38**
  - 3.2.2 Development of a Full Prototype System 42**
- 3.3 Final Development and Deployment Phases 44**
  - 3.3.1 Development of a Production System 45**
  - 3.3.2 System Deployment 48**
  - 3.3.3 System Operation and Maintenance 50**

Checklist 51

# 4

## FORMING THE TEAM 55

- 4.1 Project Technical Leader 56**
- 4.2 Project Manager 58**
- 4.3 Domain Selector 58**
- 4.4 Domain Expert 59**
- 4.5 Consulting Domain Expert 60**
- 4.6 Hardware and Software Selector 60**
- 4.7 Knowledge Acquirer 61**
- 4.8 Knowledge Representer 62**
- 4.9 Knowledge Implementer 63**
- 4.10 Systems Engineer 64**
- 4.11 Project Tool Developer 65**
- 4.12 Corporate/Client Interface 65**
- 4.13 Technical Documenter/Writer 67**
- 4.14 System Tester and Evaluator 68**
- 4.15 System Deployer 69**
- 4.16 Trainer 70**
- 4.17 System Operator 70**
- 4.18 System Maintainer 71**
- 4.19 Consultant 72**
- 4.20 End User 73**

Checklist 74

**5****SELECTING THE DOMAIN: THE PROCESS 77**

**5.1** Importance of Domain Selection **78**

**5.2** A Method for Evaluating Application Domains **79**

Checklist **96**

**6****SELECTING THE DOMAIN: DESIRED DOMAIN ATTRIBUTES 97**

**6.1** Basic Requirements **98**

**6.1.1** Need for the Expert System Approach **98**

**6.1.2** Existence of Experts **99**

**6.1.3** Need to Capture the Expertise **101**

**6.1.4** Limited Success Is Acceptable **102**

**6.1.5** Payoff **102**

**6.1.6** Risk versus Payoff **105**

**6.2** Type of Problem **106**

**6.2.1** Use of Symbolic Reasoning **106**

**6.2.2** Use of Heuristics and Other Task Characteristics **107**

**6.2.3** Widespread Knowledge and Common Sense **107**

**6.2.4** Not Driven by a Particular Technology **108**

**6.2.5** Task Definition **110**

**6.2.6** Task Inputs **110**

**6.2.7** Task Outputs **111**

**6.2.8** Similarity to Successful System **112**

**6.3** Expertise **112**

**6.3.1** Experts Are Better than Novices **113**

**6.3.2** Availability of Expertise **113**

**6.3.3** Utilizing a Single Expert **114**

**6.4** Task Bounds **114**

**6.4.1** Bounds on Task Difficulty **114**

**6.4.2** Estimated Lower Bound on Task Knowledge **115**

**6.4.3** Estimated Upper Bound on Task Knowledge **115**

**6.4.4** Narrowness of Task **116**

**6.5** Domain Area Personnel and Politics **116**

**6.5.1** Domain Personnel's Expectations of Success **116**

**6.5.2** Domain Leaders' Task Agreement and Continuing Involvement **117**

**6.5.3** Problem Previously Identified **118**

**6.5.4** Top-Level Support **118**

6.5.5	Users Want It	118
6.5.6	Introduction of the System	119
6.5.7	Cooperative User Group	120
6.5.8	Political Problems Related to System Control	120
6.5.9	Political Problems Related to System Results	121
6.5.10	Problems Related to the Sensitivity of the Knowledge	121
6.6	Development, Testing, and Deployment	122
6.6.1	Incomplete Coverage	122
6.6.2	Decomposability	123
6.6.3	Teachable Skill	124
6.6.4	Written Material	124
6.6.5	Availability of Test Cases	125
6.6.6	Real-Time and Performance Issues	125
6.6.7	User Interface	126
6.6.8	Long-Term System Need	127
6.6.9	No Alternative	128
6.6.10	Stability	129
6.6.11	Project Dependencies and Milestones	129
6.6.12	Tolerance to Incorrect Results	130
6.6.13	Measurable Payoff	131
6.6.14	Expert Agreement on Correctness	132
Checklist		134

## 7 **SELECTING THE HARDWARE AND THE SOFTWARE DEVELOPMENT TOOL** 139

7.1	A Process for Selecting the Development Environment	141
7.2	Criteria and Parameters for Comparing Software Tools or Hardware	146
7.3	Obtaining the System	147
7.3.1	Availability	147
7.3.2	Cost	148
7.3.3	Legal Arrangements	148
7.3.4	Integration into Environment	149
7.3.5	Uniformity with Others	149
7.4	Background and Prospects	150
7.4.1	Vendor Track Record and Prospects	150
7.4.2	System History	151
7.4.3	Stage of Development	151
7.4.4	System Upgrading	152
7.4.5	System Maintenance by the Vendor	152

<b>7.5</b>	<b>Learning to Use the System</b>	<b>153</b>
7.5.1	Training Materials	153
7.5.2	Training Courses	153
<b>7.6</b>	<b>System Usability and General Features</b>	<b>154</b>
7.6.1	System Interface	154
7.6.2	Supportive Programming Environment	154
7.6.3	Efficiency of the Programming Environment	155
7.6.4	Software Engineering-Related Facilities	155
7.6.5	Documentation	155
7.6.6	Performance	156
7.6.7	Size	157
7.6.8	System Interfaces	157
7.6.9	Security	158
7.6.10	User Support and Consultants	158
<b>7.7</b>	<b>Special Features Related to Software</b>	<b>159</b>
7.7.1	Building or Buying a Software Tool	159
7.7.2	Knowledge Representation Paradigms	160
7.7.3	Incorporation of a Computer Language	162
7.7.4	Match to Problem	162
7.7.5	Access to Source Code	163
<b>7.8</b>	<b>Special Features Related to Hardware</b>	<b>163</b>
7.8.1	Computer Type	164
7.8.2	System Software and File System	164
7.8.3	Support for Multiple Tools	165
<b>7.9</b>	<b>Issues Related to Deployment</b>	<b>165</b>
7.9.1	Use as Deployment Vehicle	165
7.9.2	Easy Transfer to Deployment Vehicle	166
7.9.3	Operation and Maintenance of the Deployed Expert System	167
	<b>Checklist</b>	<b>169</b>

## **8**

### **SELECTING THE DOMAIN EXPERTS 173**

<b>8.1</b>	<b>Responsibility for Selecting the Experts</b>	<b>174</b>
<b>8.2</b>	<b>Attributes of Good Domain Experts</b>	<b>175</b>
8.2.1	Existence of Domain Expertise and Experience	175
8.2.2	Level of Expertise	175
8.2.3	Extensiveness of Experience	176
8.2.4	Reputation	176
8.2.5	Finding Experts with the Right Experience	177

8.2.6	Communication Skills	178
8.2.7	Temperament	179
8.2.8	Cooperativeness	180
8.2.9	Working Relations	181
8.2.10	Availability	181
8.2.11	Management Support for Expert Involvement	182
8.2.12	Computer and AI Background	183
8.2.13	Experts as Domain Representatives	184
8.2.14	Expectations	186
8.3	The Process of Selecting Domain Experts	186
8.4	When No Experts Are Available	190
8.5	The Number of Domain Experts	192
8.5.1	Using a Single Expert	192
8.5.2	Using Multiple Experts	194
8.6	Consulting Domain Experts	195
	Checklist	197

## **9** **ACQUIRING THE KNOWLEDGE** 199

9.1	Considering Knowledge Acquisition at the Project's Beginning	201
9.1.1	Selecting the Domain with a View toward Knowledge Acquisition	201
9.1.2	Selecting Experts with a View toward Knowledge Acquisition	202
9.2	Knowledge Acquisition Meetings	203
9.2.1	Maximizing Access to the Experts	203
9.2.2	Allowing the Experts to Demonstrate their Expertise	204
9.2.3	Minimizing Interruptions	205
9.2.4	Accessing the Implementation	206
9.2.5	Locating Meetings at the Project Team's Site	207
9.3	Beginning the Knowledge Acquisition	208
9.3.1	Meeting Atmosphere	208
9.3.2	Focusing the Knowledge Acquisition	209
9.3.3	Getting Background Domain Knowledge	209
9.3.4	Preparing a Tutorial Document	210
9.3.5	Giving the Domain Experts Some AI Background	211
9.3.6	Using Written Materials for Initial Knowledge	211
9.3.7	Initial Steps	212
9.4	Documenting the Knowledge	213
9.4.1	Using Quasi-English Knowledge Acquisition Rules	213

9.4.2	The Knowledge Document	214
9.4.3	Readability of the Knowledge Documentation	216
9.4.4	Terminology Used in the Knowledge Documentation	216
9.4.5	Devising Terminology for Documentation and Discussion	217
9.4.6	Identifying the Knowledge Acquisition Rules and Procedures	219
9.4.7	Organizing the Knowledge Acquisition Rules and Procedures	220
9.4.8	Utilizing an Explanatory Clause	220
9.4.9	Domain Description and Glossary	221
9.5	Acquiring the Knowledge	221
9.5.1	Basic Knowledge Acquisition Cycle before Implementation	221
9.5.2	Basic Knowledge Acquisition Cycle after Implementation Has Begun	222
9.5.3	Using Test Cases to Elicit Expert Knowledge	225
9.5.4	Using Acquired Knowledge to Guide Related Knowledge Acquisition	226
9.5.5	Using Knowledge Acquisition Formalisms Directly	227
9.5.6	Updating the Knowledge Documentation	227
9.5.7	Deferring Specification of Certain Details	227
9.5.8	Generating Test Cases from Test Cases	228
9.5.9	Establishing a Default for Close Decisions	229
9.5.10	Finding the Extent of Rules	229
9.6	Recording the Knowledge	230
9.6.1	Flexibility	230
9.6.2	Recording Reminders	231
9.6.3	Recording Benefits of the Expert System	231
	Checklist	233

## 10

## REPRESENTING THE SYSTEM KNOWLEDGE 237

10.1	Decomposing the Problem	238
10.2	Paradigm(s) for Representation	248
10.3	Frames, Inheritance, and Demons	249
10.4	Production Rules	254
10.5	Object-Oriented Programming	257
10.6	Programming Language Code	258
10.7	Representing and Rerepresenting	259
	Checklist	263

# **11**

## **IMPLEMENTING THE SYSTEM 265**

- 11.1 Expert System Implementation Compared with the Implementation of Conventional Programs 266**
  - 11.1.1 Implementation without a Full Specification 266**
  - 11.1.2 Reimplementing the Implementation 269**
- 11.2 Some Techniques for Knowledge Implementation 277**
  - 11.2.1 Correspondence of Knowledge Acquisition Rules and Implementation Rules 277**
  - 11.2.2 Grouping Rules in Rulesets 278**
  - 11.2.3 Implementing and Debugging 279**
  - 11.2.4 Documentation 280**
- 11.3 Managing the Implementation and Promoting Maintainability 282**
  - 11.3.1 Software Modularity and Task Assignment 283**
  - 11.3.2 Uniformity of Style 284**
  - 11.3.3 Use of Paradigms that Promote Maintainability 286**
  - 11.3.4 Configuration Management and Control 288**
  - 11.3.5 Data Flow and Access 290**
  - 11.3.6 Control Flow 292**
  - 11.3.7 Input/Output 293**
- Checklist 296**

# **12**

## **TESTING AND EVALUATING THE SYSTEM 299**

- 12.1 Validation and Verification during Expert System Development 300**
  - 12.1.1 Validation and Verification as Inherent Parts of Knowledge Acquisition 300**
  - 12.1.2 Validation Testing by Consulting Experts 302**
- 12.2 Validation of the Developed Expert System 303**
  - 12.2.1 Validating Absolutely 303**
  - 12.2.2 Validating against Expert Performance 306**
  - 12.2.3 Validation by Field Testing 307**
- 12.3 Verification of an Expert System Program 308**
- 12.4 Evaluation Effort and Standards 310**
- Checklist 312**



## **13**

### **TRANSFERRING AND DEPLOYING THE SYSTEM 315**

- 13.1** Developing an Expert System with Technology Transfer and Deployment in Mind **316**
  - 13.1.1** Issues Related to Domain Selection **316**
  - 13.1.2** Issues Related to Knowledge Representation and Implementation **318**
- 13.2** Technology Transfer **319**
- 13.3** Organizational Roles and Activities in Technology Transfer **321**
- 13.4** Transferring AI Expertise **324**
  - 13.4.1** General Training Requirements **324**
  - 13.4.2** AI Techniques, Tools, and Systems Training **325**
  - 13.4.3** Domain Training **326**
  - 13.4.4** Training on the Expert System Program **327**
  - 13.4.5** Knowledge Acquisition Training **328**
- 13.5** Deployment **328**
  - 13.5.1** Deployment Environment **328**
  - 13.5.2** Reliability, Maintainability, and Security **331**
  - 13.5.3** Integration of the Expert System **333**
- 13.6** Gaining User Acceptance of the Deployed Expert System **334**
  - 13.6.1** Educating the Users **334**
  - 13.6.2** Introducing the Expert System **336**
  - 13.6.3** Use of the Expert System **337**
- Checklist **339**

## **14**

### **EXPERT SYSTEM TRENDS AND DIRECTIONS 341**

Glossary **347**

Index **353**