

บทความวิชาการประจำปี 2531

สถาบันบัณฑิตพัฒนบริหารศาสตร์

การสร้างระบบผู้ชำนาญการ :

ผลการวิเคราะห์ระบบต้นแบบ 8 ระบบ

ดร. อนุมงคล ศิริเวทิน

คณะสถิติประยุกต์

สถาบันบัณฑิตพัฒนบริหารศาสตร์

**NIDA PAPER 1988**

THE NATIONAL INSTITUTE OF  
DEVELOPMENT ADMINISTRATION

**การสร้างระบบผู้ชำนาญการ :**  
**ผลการวิเคราะห์ระบบต้นแบบ 8 ระบบ**

เรียบเรียงโดย

**ดร. อนุมงคล ศิริเวทิน**

รองศาสตราจารย์-คณบดี

**คณะสถิติประยุกต์**  
**สถาบันบัณฑิตพัฒนบริหารศาสตร์**



## ประกาศสถาบันบัณฑิตพัฒนบริหารศาสตร์ เรื่อง ผลการคัดเลือกบทความวิชาการดีเด่น ประจำปี 2531

ตามประกาศสถาบันบัณฑิตพัฒนบริหารศาสตร์ ลงวันที่ 14 กรกฎาคม 2531 ได้ประกาศเชิญชวนให้ข้าราชการของสถาบันส่งบทความวิชาการเข้ารับการพิจารณาคัดเลือกเป็นบทความวิชาการดีเด่น ประจำปี 2531 ใน 5 สาขาวิชา คือ สาขาวิชารัฐประศาสนศาสตร์ บริหารธุรกิจ พัฒนาการเศรษฐกิจ สถิติประยุกต์ และสังคมศาสตร์อื่น ๆ โดยบทความวิชาการที่ได้รับการคัดเลือกเป็นบทความวิชาการดีเด่นจะได้รับเงินรางวัลในแต่ละสาขาวิชาละ 10,000 บาท นั้น

สถาบันบัณฑิตพัฒนบริหารศาสตร์ ได้รับบทความวิชาการที่ส่งเข้ารับการคัดเลือกเป็นบทความวิชาการดีเด่น จำนวนทั้งสิ้น 5 บทความ เป็นบทความในสาขาวิชารัฐประศาสนศาสตร์ 2 บทความ สาขาวิชาบริหารธุรกิจ 1 บทความ และสาขาวิชาสถิติประยุกต์ 2 บทความ ซึ่งคณะกรรมการดำเนินงานคัดเลือกบทความวิชาการดีเด่นได้พิจารณาบทความวิชาการดังกล่าวเสร็จเรียบร้อยแล้ว ปราบทว่าบทความในสาขาวิชาสถิติประยุกต์ เรื่อง การสร้างระบบผู้ชำนาญการ : ผลการวิเคราะห์ระบบต้นแบบ 8 ระบบ ของรองศาสตราจารย์ ดร. อุ่นมงคล ศิริเวทิน เป็นบทความวิชาการที่ได้รับการคัดเลือกให้เป็นบทความวิชาการดีเด่น ประจำปี 2531

จึงประกาศให้ทราบทั่วกัน

ประกาศ ณ วันที่ 17 มกราคม พ.ศ. 2532

(นายสมศักดิ์ ชูโต)

อธิการบดีสถาบันบัณฑิตพัฒนบริหารศาสตร์

# สารบัญ

|                                   | หน้า |
|-----------------------------------|------|
| บทนำ                              | 1    |
| ระบบผู้จำหน่ายการ                 | 1    |
| ระบบผู้จำหน่ายการที่นำมาวิเคราะห์ | 3    |
| หัวข้อพิจารณาในการพัฒนาระบบ       | 6    |
| บทสรุป                            | 15   |
| เอกสารอ้างอิง                     | 17   |

# การสร้างระบบผู้เชี่ยวชาญ :

## ผลการวิเคราะห์ระบบต้นแบบ 8 ระบบ

ดร. อนุมงคล ศิริเวทิน\*

### 1. บทนำ

เนื่องจากปัญญาประดิษฐ์ และโดยเฉพาะอย่างยิ่งระบบผู้เชี่ยวชาญ หรือระบบผู้เชี่ยวชาญ (expert system) ซึ่งผู้ทรงคุณวุฒิในสาขานั้นบางคน เช่น Bobrow et. al. (1986 – หน้า 880) และ Davis (1986 – หน้า 957) เรียกว่าระบบเชิงฐานความรู้ (knowledge-based system) หรือระบบเชิงความรู้ (knowledge system) (แต่ผู้อยู่ในวงการบางคนเช่น Waterman (1986 – หน้า 18) ก็ให้คำนิยามที่ละเอียดขึ้นไปอีกคือนิยามว่าระบบผู้เชี่ยวชาญเป็นเซ็ทย่อยของระบบเชิงความรู้) กำลังเป็นเรื่องที่ได้รับความสนใจและกล่าวถึงอย่างสูง นอกจากนี้ก็มีผลผลิตที่เกี่ยวข้องกับระบบผู้เชี่ยวชาญออกมาในตลาดไม่ขาดระยะไม่ว่าจะเป็นตัวระบบผู้เชี่ยวชาญเอง ภาษาสั่งงานสำหรับใช้สร้างระบบผู้เชี่ยวชาญ บทความ และหนังสือที่กล่าวถึงหัวข้อนี้

ในประเทศไทย ความสนใจในหัวข้อนี้ก็มีเพิ่มขึ้นอย่างเห็นได้ชัด แต่การเผยแพร่ความรู้และประสบการณ์ในการสร้างระบบผู้เชี่ยวชาญยังมีน้อยเกินไป โดยเฉพาะในรูปแบบที่ผู้สนใจอาจนำมาใช้เป็นตัวอย่างหรือเกณฑ์ในการสร้างระบบดังกล่าวของตนเองขึ้นมาบ้าง ในฐานะที่ผู้เขียนได้มีประสบการณ์ในการแนะนำ

และควบคุมการสร้างระบบผู้เชี่ยวชาญ ระดับต้นแบบหลายระบบโดยเป็นส่วนหนึ่งของการเรียนการสอนวิชาคอมพิวเตอร์ที่สถาบันบัณฑิตพัฒนบริหารศาสตร์ ผู้เขียนมีความรู้สึกว่าการนำประสบการณ์ดังกล่าวมาเผยแพร่ น่าจะเป็นประโยชน์แก่นักวิชาการ และผู้ที่อยู่ในวงการคอมพิวเตอร์ที่สนใจวิทยาการด้านนี้ และผู้ที่ประสงค์จะสร้างระบบผู้เชี่ยวชาญแต่ยังไม่แน่ใจในขั้นตอนการดำเนินงาน หรือยังไม่แน่ใจในการเลือกใช้เทคนิคและวิธีการในแต่ละขั้นตอนของการดำเนินงาน จะได้มีตัวอย่างเพื่อการเปรียบเทียบประกอบการตัดสินใจมากขึ้น

ดังนั้นบทความนี้จึงมีลักษณะผสมระหว่างการสำรวจ (survey) (อย่างมีขอบเขตจำกัด) การวิจารณ์ และเป็นบทความเชิงสอน (tutorial) และบทความแสดงจุดยืน (position paper) ด้วยอีกส่วนหนึ่งในขณะเดียวกัน

### 2. ระบบผู้เชี่ยวชาญ

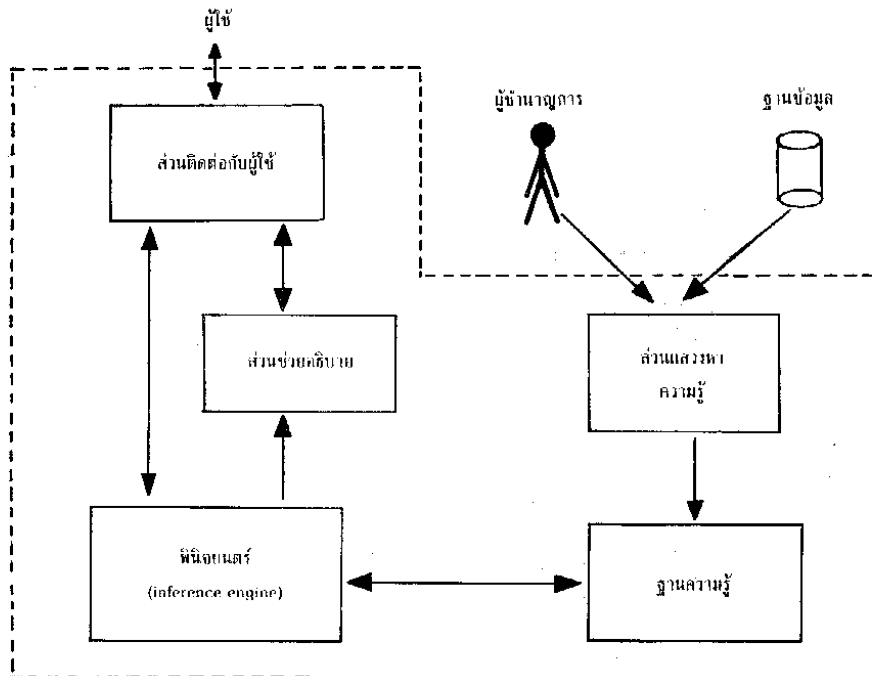
เพื่อความเข้าใจร่วมกัน สำหรับผู้อ่านบทความนี้และเพื่อผู้ที่อาจมีพื้นฐานในเรื่องระบบผู้เชี่ยวชาญไม่เพียงพอ ผู้เขียนขอให้นิยามคำว่าระบบผู้เชี่ยวชาญ และคำที่เกี่ยวข้องตลอดจนลักษณะทั่วไปของระบบผู้เชี่ยวชาญดังต่อไปนี้

\* คณบดี – รองศาสตราจารย์ คณะสถิติประยุกต์ สถาบันบัณฑิตพัฒนบริหารศาสตร์

ระบบผู้เชี่ยวชาญเป็นระบบของชุดคำสั่งงานที่พัฒนาขึ้นมาโดยมุ่งหวังที่จะให้สามารถทำงานได้เสมือนผู้เชี่ยวชาญในหัวข้อปัญหาใดหัวข้อหนึ่ง

ระบบผู้เชี่ยวชาญมีโครงสร้างพื้นฐานดังแสดงในภาพที่ 1

ภาพที่ 1 : โครงสร้างพื้นฐานของระบบผู้เชี่ยวชาญ



จะเห็นได้ว่าระบบผู้เชี่ยวชาญจะแยกฐานความรู้เฉพาะเรื่อง (domain knowledge base) ออกจากส่วนที่ใช้ควบคุมการแก้ปัญหา หรือพินิจยนตร์ (inference engine) และเพื่อให้สามารถทำงานได้เหมือนผู้เชี่ยวชาญที่เป็นบุคคล ระบบผู้เชี่ยวชาญจะมีส่วน (module) ที่ทำหน้าที่อธิบายการทำงานของตนโดยเฉพาะเรียกว่าส่วนช่วยอธิบาย (explanation module) ส่วนประกอบที่สำคัญอีกสองส่วนได้แก่ ส่วนแสวงหาความรู้ (knowledge acquisition module) ซึ่งจะช่วยวิศวกรความรู้ (knowledge engineer) ในการแสวงหาและรับถ่ายทอดความรู้จากผู้ชำนาญการ

ส่วนสุดท้ายได้แก่ส่วนติดต่อกับผู้ใช้ (user interface module) ที่มีหน้าที่ช่วยให้การติดต่อกับระบบเหมือนกับติดต่อกับมนุษย์มากที่สุด

ระบบผู้เชี่ยวชาญแตกต่างจากชุดคำสั่งงานปรกติหลายประการเช่น ระบบผู้เชี่ยวชาญจะมีข้อมูลในรูปของความรู้และสัญลักษณ์แทนที่จะเป็นตัวเลข วิธีการแก้ปัญหาจะเป็นแบบหยั่งรู้ (heuristic) แทนที่จะเป็นแบบทำตามขั้นตอน (algorithmic) โดยใช้กระบวนการของการอ้างถึงและสรุป (inferential process) แทนที่จะเป็นการทำซ้ำ ๆ กัน (repetitive) เช่นในชุด

คำสั่งงานธรรมดา และระบบผู้เชี่ยวชาญยังแยกส่วนของความรู้ออกจากส่วนควบคุมดังได้กล่าวมาแล้ว ซึ่งมักจะไม่ค่อยพบกันในชุดคำสั่งงานธรรมดา

ปัญหาที่สำคัญอีกสองประการที่ผู้พัฒนาระบบผู้เชี่ยวชาญจะต้องประสบ ได้แก่ การเลือกหัวข้อปัญหาที่เหมาะสมซึ่งโดยปรกติแล้ว หัวข้อปัญหาที่เหมาะสมจะมีลักษณะแคบแต่ลึกและจะแตกต่างจากปัญหาปรกติที่เรานำคอมพิวเตอร์มาช่วยแก้ คือโดยทั่วไปแล้วเราจะไม่ทราบว่าจะตอบที่ได้ “ถูก” หรือ “ผิด” แต่ผู้เชี่ยวชาญในสาขานั้นจะมีความเห็นส่วนใหญ่ตรงกันว่าคำตอบที่ได้ “ดี” หรือ “ไม่สู้ดี” ปัญหาประการที่สองได้แก่การแสวงหาความรู้ที่จะมาใช้ในการแก้ปัญหาทั้งหมดที่เป็นความรู้จริง ๆ (specific knowledge) และวิธีการที่จะประยุกต์และควบคุมการใช้ความรู้นั้นให้นำไปสู่คำตอบ (inference engine, control structure) ถ้าหัวข้อปัญหาที่เลือกเหมาะสมก็หมายความว่า ความรู้ที่เราแสวงหามีอยู่ในตัวคน คนใดคนหนึ่ง (หรือกลุ่มใดกลุ่มหนึ่ง) แต่การที่ทราบว่าความรู้มีอยู่เช่นนี้แล้ว มิได้หมายความว่าเราจะแสวงหาความรู้นั้นมาได้โดยง่าย กระบวนการนี้มีชื่อเรียกว่า วิศวกรรมความรู้ (knowledge engineering) และดำเนินการโดยวิศวกรความรู้ (knowledge engineer) ซึ่งจะทำหน้าที่คล้ายกับนักวิเคราะห์ระบบ (systems analyst) ในระบบประมวลผลธรรมดา และวิศวกรความรู้จะเป็นส่วนหนึ่งของทีมงานที่พัฒนาระบบผู้เชี่ยวชาญ

### 3. ระบบผู้เชี่ยวชาญที่นำมาวิเคราะห์

ระบบผู้เชี่ยวชาญที่ผู้เขียนบทความได้มีส่วนร่วมในการเลือกและกำหนดหัวข้อปัญหาและผู้ชำนาญการ แนะนำและควบคุมการพัฒนาและนำผลการ

วิเคราะห์มาเสนอในบทความนี้มีอยู่ด้วยกันรวม 8 ระบบ โดยมีรายชื่อและรายละเอียดบางประการดังแสดงในตารางที่หนึ่ง

ระบบเหล่านี้พัฒนาขึ้นโดยนักศึกษาในสาขาวิชาคอมพิวเตอร์ คณะสถิติประยุกต์ สถาบันบัณฑิตพัฒนบริหารศาสตร์ โดยเป็นส่วนหนึ่งของการเรียนวิชา MIS & Expert Systems ในภาค 2 ปีการศึกษา 2529

ระบบทุกระบบเขียนขึ้นเพื่อใช้งานกับเครื่องไมโครคอมพิวเตอร์ IBM PC หรือเครื่องที่มีลักษณะการทำงานเช่นเดียวกัน ที่มีขนาด 512 KB ขึ้นไป และมีเครื่องอ่าน/เขียนจานแม่เหล็ก (disc drive) อย่างน้อย 1 เครื่อง 6 ระบบในจำนวนนี้เขียนขึ้นด้วยภาษา Turbo Prolog โดยบริษัท Borland (1986) หนึ่งระบบใช้ภาษา EXSYS (1984) และอีกหนึ่งระบบใช้ภาษาเบสิกและภาษา Turbo Prolog แทบทุกระบบใช้วิธีการจัดเก็บความรู้โดยเก็บในรูปของกฎ (rule-based) โดยมีกฎประมาณ 40-100 กฎ และใช้เวลาในการพัฒนาระหว่าง 4-8 สัปดาห์ ซึ่งนับได้ว่าเป็นระบบประเภทต้นแบบสาธิต (demonstration prototype) ตามนิยามของ Waterman (1986-หน้า 139-140)

รายละเอียดเพิ่มเติมของแต่ละระบบมีดังต่อไปนี้ “Orthopaedic” : ช่วยแพทย์ในการวินิจฉัยโรคทางกระดูกและข้อเพื่อเป็นแนวทางในการวินิจฉัยโดยละเอียดในขั้นต่อไป ช่วยให้การตรวจและซักประวัติคนไข้เป็นไปอย่างมีระบบ แสดงผลออกเป็น suggested diagnosis เหมาะสมกับสภาพที่แพทย์จำนวนน้อยต้องตรวจรักษาคนไข้จำนวนมากในเวลาอันจำกัด

ตารางที่ 1 ระบบผู้ชำนาญการที่วิเคราะห์

| ชื่อระบบ                       | ชื่อผู้พัฒนาระบบ   | ชื่อผู้ชำนาญการ                                     | ความชำนาญของระบบ                  |
|--------------------------------|--|---|-----------------------------------|
| Orthopaedic                    | สมศรี ตุงคศักดิ์<br>ขจร โรจนเมธินทร์<br>ศิวารักษ์ อัครนิวรรณ           | รศ.นพ. ณรงค์ บูณะะรัตเวช                            | วินิจฉัยโรคข้อ<br>และกระดูก       |
| การทำงานประมาณ<br>การเงินจำลอง | วราภรณ์ กิตติรุ่งเรืองกุล<br>สุภาพร เกษสุคนธ์<br>ไพศรี ตันติเวชวุฒิกุล | คณะเจ้าหน้าที่สินเชื่อ                              | จัดทำงบการเงินจำลอง               |
| Financial Analyst              | รัชนิกร สุทธระพินท์<br>สุมาลี อิศระยังยืน                              | คณะผู้วิเคราะห์สินเชื่อ                             | วิเคราะห์ฐานะการเงิน<br>ของกิจการ |
| จิตเวช                         | สุเมธ ภูวิทย์พันธ์ุ<br>วัชรพร รัตนบรรเจิดกุล                           | นพ. ชะฤทธิ พงศ์อนุตรี<br>นพ. วิภาส พงศ์เที่ยงธรรม   | วินิจฉัยโรคทางจิต                 |
| Package Tour                   | รวีวรรณ เทนอิสสระ<br>ตรีทิพย์ ลือสิริพงศ์<br>สุวานี เชิดเกียรติกุล     | สุจินต์ อ่อนหนู                                     | จัดทำรายการท่องเที่ยว             |
| Cashflow                       | สมชาย ตั้งพัฒนาประดิษฐ์<br>ธงชัย ไชคอำนวยชัย                           | ลินญู เจริญกุล                                      | บริหารการเงินและ<br>สภาพคล่อง     |
| Diagnose                       | เดือนงาม สุดแสง<br>อัจฉรา อรรณานนท์<br>สมชาย วิจักขณ์พันธ์ุ            | นพ. พิชัย วิจักขณ์พันธ์ุ<br>นพ. สมยศ เจริญศิริวัฒน์ | ตรวจรักษาโรคที่มี<br>อาการไข้     |
| Stocks Trader                  | ประสิทธิ์ อ้วนเจริญ<br>นิตยา รุยะวัฒน์                                 | พิชัย พงศ์ไพโรจน์                                   | แนะนำการลงทุนใน<br>ตลาดหลักทรัพย์ |



**“การทำงานการเงินจำลอง”** : ช่วยบุคคลที่เชี่ยวชาญในด้านการสอบถามข้อมูลจากลูกค้า (สินเชื่อ) โดยนำข้อมูลเกี่ยวกับการดำเนินกิจการของลูกค้ามา (ประกอบกับค่าเฉลี่ยอุตสาหกรรม) สร้างงบการเงินจำลองของกิจการของลูกค้า (คืองบกำไรขาดทุน งบดุล และผลวิเคราะห์ทางการเงิน)

เท่าที่เป็นอยู่ยังมีลักษณะคล้ายชุดคำสั่งงานธรรมดาเพราะจะถามคำถามที่เตรียมไว้แล้วจนครบ แล้วนำคำตอบที่ได้มาพิจารณา (ตามเกณฑ์ที่กำหนดไว้แล้ว) ว่าเชื่อถือได้หรือไม่และจะเชื่อตามนั้นหรือปรับแก้อย่างไร อาจปรับปรุงให้แสดงผลโดยทันที (real-time) เพื่อให้ผู้เชี่ยวชาญนำไปประกอบการพิจารณาในการซักถามลูกค้าให้ได้ข้อมูลที่ถูกต้องเพิ่มขึ้นได้

**“Financial Analyst”** : รับข้อมูลเป็นงบดุล งบกำไรขาดทุน และผลวิเคราะห์ทางการเงินของกิจการใด ๆ เพื่อมาประเมินว่า สมควรนำเงินไปลงทุนในกิจการนั้นหรือไม่ และสมควรให้สินเชื่อกับกิจการนั้นหรือไม่

มีลักษณะการทำงานต่อเนื่องกับระบบ “การทำงานการเงินจำลอง” โดยที่ระบบ “การทำงานการเงินจำลอง” เป็นการพยายามจำลองฐานะของกิจการใดกิจการหนึ่งออกมาในรูปแบบของงบการเงิน และ “Financial Analyst” จะทำหน้าที่วิเคราะห์ต่อโดยยึดถือตัวเลขในงบการเงินเป็นหลักเพื่อให้ได้ผลออกมาว่าสมควรหรือไม่ที่จะร่วมลงทุนในกิจการหรือให้สินเชื่อในกิจการ

**“จิตเวช”** : ช่วยแพทย์ในการวินิจฉัยโรคทางจิต โดยครอบคลุมทั้งโรคที่มีสาเหตุทางอารมณ์หรือจิตใจ (functional) และโรคที่มีสาเหตุมาจากความผิดปกติทางกาย (organic)

เหตุที่กำหนดให้ผู้ใช้เป็นแพทย์ (ซึ่งไม่จำเป็นต้องเป็นแพทย์ทางโรคจิตโดยตรง) ก็เพราะคำถามต่าง ๆ เป็นคำถามเทคนิคมาก อีกทั้งผู้ที่อยู่ในข่ายสงสัยว่าจะเป็นโรคจิตก็ไม่สมควรที่จะวินิจฉัยโรคของตนเอง

**“Package Tour”** : จัดรายการทัวร์ภายในประเทศไทยระยะ 3-10 วัน สำหรับกลุ่มลูกค้าจากต่างประเทศ โดยรับข้อมูลเกี่ยวกับจำนวนวัน วัย สุขภาพ และลักษณะความต้องการของกลุ่มลูกค้า รวมทั้งบริษัททัวร์ในต่างประเทศที่เป็นผู้ติดต่อ ผลการทำงานจะแสดงทั้งสถานที่เที่ยว สถานที่พัก ยานพาหนะที่ใช้ในแต่ละช่วงของการเดินทาง และค่าใช้จ่ายภายในกรอบที่ต้องการ

ไม่ใช้ความรู้สึกซึ่งมากนักแต่ก็ช่วยให้การจัดรายการทำได้เร็วขึ้นมาก มีลักษณะคล้ายระบบซอฟต์แวร์ทั่วไปอยู่บ้างแต่การจัดการแบบระบบผู้เชี่ยวชาญช่วยให้การปรับปรุงฐานความรู้ (ซึ่งต้องทำบ่อยมาก) ทำได้สะดวกและรวดเร็วกว่าในกรณีธรรมดา

**“Cashflow”** : ช่วยผู้มีหน้าที่จัดสรรเงินและดูแลสภาพคล่องของบริษัทเงินทุนในการบริหารการเงิน โดยรับข้อมูลสถานะการเงินและสถานะลูกหนี้ ประเภทต่างๆ ของบริษัท ประกอบกับสภาพตลาดเงิน ก่อตัวคืออัตราดอกเบี้ย ช่วงเวลา แหล่งเงินทุนหรือแหล่งที่ต้องการเงินทุน ฯลฯ แล้วตัดสินใจลงทุนและ/หรือเรียกหนี้คืน โดยให้บริษัทได้ประโยชน์สูงสุดในขณะที่ยังคงดำรงสภาพคล่องที่เหมาะสม และอยู่ภายใต้กฎหมายของธนาคารแห่งประเทศไทย

น่าจะมีประโยชน์ในฐานะผู้เชี่ยวชาญที่ปฏิบัติงานจริง โดยเฉพาะถ้าข้อมูลเกี่ยวกับการเงินส่วนมากเป็นข้อมูลที่สามารถรับได้โดยตรงจากคอมพิวเตอร์ (on-line)

“**Diagnose**” : ช่วยในการวินิจฉัยโรคที่มีอาการไข้ (ประมาณ 30–40 โรค) โดยการถามประวัติและอาการ (โดยการพิมพ์ข้อมูลเข้าไปและโดยการตอบคำถามของระบบ) รวมทั้งการติดตามอาการคนไข้หลังการรักษา (follow up) สามารถให้ยาและรายละเอียดในการใช้ยา สามารถใช้ได้กับผู้ใช้ 4 กลุ่มคือ แพทย์ ผู้ช่วยแพทย์ (เช่นพยาบาล พนักงานอนามัย เป็นต้น) บุคคลทั่วไป และเภสัชกร (จริงและเทียม)

ถ้าขยายขึ้นไปให้ครอบคลุมได้ 130 โรค ตามความตั้งใจของผู้พัฒนาได้จริงก็จะมีประโยชน์มากพอควร แต่ด้วยวิธีการที่ใช้อยู่สามารถจัดทำในรูปแบบอื่น เช่นเป็นผังงาน หรือรายการคำถาม-ข้อวินิจฉัย โดยได้ผลใกล้เคียงกันและมีต้นทุนต่ำกว่ามาก อาจสมควรปรับปรุงในรายละเอียดเพื่อเพิ่มความถูกต้องในการวินิจฉัยให้สูงกว่านี้

“**Stocks Trader**” : ให้คำแนะนำในการซื้อ-ขายหลักทรัพย์โดยรับข้อมูลการซื้อขายหลักทรัพย์ของหลักทรัพย์ที่สนใจทุกวันทำการ ได้แก่ข้อมูลราคาต่ำสุด ราคาสูงสุด ราคาปิด ปริมาณการซื้อขาย ฯลฯ

เทคนิคที่ใช้ยังเป็นเทคนิคของชุดคำสั่งงานธรรมดาอยู่ แต่ผลที่ได้จากชุดคำสั่งงานภาษา Prolog (โดยการส่งข้อมูลจากชุดคำสั่งงานภาษาเบสิก ไปยังชุดคำสั่งงานภาษา Prolog) ดูเสมือนผลลัพธ์จากระบบผู้เชี่ยวชาญ

#### 4. หัวข้อพิจารณาในการพัฒนาระบบ

##### 4.1 การเลือกหัวข้อปัญหา

เป็นที่ทราบกันดีว่าการเลือกหัวข้อปัญหาในการสร้างระบบผู้เชี่ยวชาญนั้นเป็นปัญหาที่สำคัญประการหนึ่งในตัวของมันเอง ข้อผิดพลาดที่เกิดขึ้นบ่อยครั้งก็

คือการเลือกหัวข้อปัญหาที่ยากเกินไป สาขาเหตุที่เป็นดังนี้อาจจะเพราะว่าระบบผู้เชี่ยวชาญเป็นเรื่องใหม่และยังไม่มีตัวอย่างที่เข้าใจกันดีมากพอว่า ระบบผู้เชี่ยวชาญสามารถทำงานที่มีระดับความยากได้มากน้อยเพียงไร นอกจากนี้เมื่อมีระบบผู้เชี่ยวชาญที่ทำงานใดงานหนึ่งได้แล้ว ก็มักจะไม่เหลือผลงูใจที่จะพัฒนาระบบผู้เชี่ยวชาญที่มีความสามารถคล้ายคลึงกันขึ้นมาอีกเหมือนดังที่ปฏิบัติกันอยู่เป็นประจำคือวิสัยในการประมวลผลธรรมดา (เช่น มีชุดคำสั่งงานจ่ายเงินเดือน ชุดคำสั่งงานสินค้าคงคลัง และชุดคำสั่งงานบัญชีเป็นจำนวนมาก เป็นต้น) ทั้งนี้เพราะวัตถุประสงค์หลักในการพัฒนาชุดคำสั่งงานธรรมดานั้นเน้นประโยชน์ใช้สอย แต่ในขณะที่วัตถุประสงค์หลักในการพัฒนาระบบผู้เชี่ยวชาญส่วนใหญ่ยังเน้นการวิจัยและทดลองหาความรู้ใหม่ ๆ มากกว่าประโยชน์ใช้สอยจริงจัง กลุ่มผู้พัฒนา DENDRAL (ดู Alty et. al. (1984-หน้า 87-88), Buchanan et. al. (1969) และ Lindsay et. al. (1980)) ซึ่งเป็นระบบผู้เชี่ยวชาญที่ประสบความสำเร็จระบบแรก ๆ แนะนำเกณฑ์ง่าย ๆ ในการเลือกหัวข้อปัญหาไว้ 2 ข้อดังนี้

1. เน้นสาขาเฉพาะที่ต้องใช้ความรู้เฉพาะด้านอย่างมากแต่ไม่ต้องใช้ความรู้สามัญ (Common sense) มากนัก
2. เลือกงานที่ไม่ยาก (คนใช้เวลาคิดมากกว่า 2–3 ชั่วโมงขึ้นไป) และไม่ง่ายจนเกินไป (คนใช้เวลาคิดน้อยกว่า 2–3 นาที)

ระบบผู้เชี่ยวชาญที่พิจารณาเลือกหัวข้อเกี่ยวกับการวินิจฉัยโรค 3 ระบบด้วยกันคือ โรคทั่วไป ที่มีอาการไข้ โรคเกี่ยวกับข้อและกระดูก และโรคเกี่ยวกับจิตใจ ทั้ง 3 หัวข้อนี้เรียกได้ว่าเป็นไปตามแนว

โน้มน้าวไป และมีปัญหาน้อยในช่วงของการพัฒนา ผู้เขียนได้พยายามสนับสนุนให้ผู้พัฒนาระบบที่เชื่อถือเลือกหัวข้อปัญหาในทางธุรกิจ (โดยมุ่งหวังประโยชน์ใช้สอย และผลการวิจัยสาขาที่จะประยุกต์ได้เป็นหลัก) ซึ่งค่อนข้างจะขัดกับแนวโน้มทั่วไป เพราะระบบผู้เชี่ยวชาญทางธุรกิจยังมีอยู่น้อยมากเมื่อเทียบกับด้านอื่น ๆ โดยเฉพาะทางแพทย์และวิทยาศาสตร์ ผลก็คือผู้ทำงานในกลุ่มนี้ประสบปัญหาในช่วงดำเนินงานสูงกว่าผู้พัฒนา 3 ระบบแรก และต้องมีการปรับเปลี่ยนหัวข้อและขอบข่ายของความสามารถของระบบรวมถึงการเปลี่ยนตัวผู้เชี่ยวชาญด้วยหลายครั้ง ในท้ายที่สุดผู้พัฒนาได้เลือกหัวข้อดังต่อไปนี้ การสร้างโปรแกรมท่องเที่ยวประเทศไทย 3-10 วันให้แก่ลูกค้าต่างประเทศ การสร้างงบการเงินจำลอง (โดยได้ข้อมูลบางส่วน) เพื่อประโยชน์ในการพิจารณาปล่อยสินเชื่อ การให้คำแนะนำในการซื้อขายหลักทรัพย์ การพิจารณาให้สินเชื่อและการร่วมลงทุน และการจัดสรรเงินสดประจำวัน

เมื่อพิจารณาแต่ละหัวข้อ จะเห็นได้ว่าทุกหัวข้อมีความเหมาะสมในแง่ที่ผู้ที่ปฏิบัติงานแต่ละงานดังกล่าวให้คิดจะต้องเป็นผู้ที่มีความเชี่ยวชาญในแต่ละเรื่องอย่างแท้จริง แต่เมื่อพิจารณาในรายละเอียดจะพบว่านอกจากระบบที่เกี่ยวข้องกับการแพทย์แล้ว หัวข้อปัญหาของระบบอื่น ๆ ยังมีข้อบกพร่องในด้านต่าง ๆ อยู่บ้างดังต่อไปนี้

การสร้างโปรแกรมท่องเที่ยวนั้นแม้จะต้องใช้ผู้เชี่ยวชาญในการจัดทำแต่วิธีการจัดทำค่อนข้างจะมีหลักเกณฑ์ที่แน่นอนและไม่จำเป็นต้องใช้ปรีชาญาณ (intuition) หรือความรู้ในลักษณะ heuristic นัก ดังนั้นระบบนี้จึงมีลักษณะคล้ายชุดคำสั่งงานธรรมดาที่มีขอบเขตการโต้ตอบระหว่างผู้ใช้และระบบสูงกว่า

ปรกติมาก แต่ก็มีความเหมาะสมด้านอื่น ๆ เช่น การแยกส่วนควบคุมออกจากฐานความรู้ เป็นต้น ซึ่งจะได้อธิบายถึงในตอนต่อไป

ระบบที่เกี่ยวกับสินเชื่อทั้งสองระบบนั้น ความจริงแล้วน่าสนใจมากเพราะหากสามารถพัฒนาขึ้นได้จะมีประโยชน์ใช้สอยจริง แต่ปรากฏว่าระบบมีลักษณะขัดกับคุณลักษณะเบื้องต้นของระบบผู้เชี่ยวชาญในยุคหนึ่งเรื่องหนึ่ง คือผู้ให้ข้อมูล (ผู้ขอรับสินเชื่อ) อาจให้ข้อมูลไม่จริง และโดยทั่วไปแล้วอาจพยายามให้ข้อมูลเท็จเพื่อประโยชน์ของตนก็ได้ และผู้เชี่ยวชาญสินเชื่อจะพยายามใช้ความสามารถเชิงสัมพันธ์ในการประเมินความเชื่อถือของคำตอบ เช่นจับคำว่าผู้ตอบมีความกระวนกระวายในขณะตอบแต่ละคำถามเพียงไร ฟังว่าน้ำเสียงที่พูดสม่ำเสมอหรือเปลี่ยนแปลงในลักษณะใด การตอบคำถามตอบได้ทันทีหรือไม่ ตอบตรงคำถามหรืออ้อมค้อมเพียงไร ฯลฯ นอกจากนี้ผู้เชี่ยวชาญจริงยังต้องใช้เทคนิคการถามซึ่งขึ้นกับผลตอบสนองจากการสังเกตดังกล่าวอีกด้วย ซึ่งคอมพิวเตอร์ไม่มีความสามารถในการสังเกตดังกล่าวจึงทำงานในลักษณะนี้ไม่ได้ผลดีจริงได้ยาก

ระบบจัดสรรเงินสดประจำวันนั้นความจริงนับได้ว่าเป็นระบบที่เหมาะสมระบบหนึ่ง แต่ในการทำงานจริงผู้เชี่ยวชาญจะโทรศัพท์ติดต่อกับสถาบันการเงินประเภทต่าง ๆ เพื่อขอทราบและต่อรองอัตราดอกเบี้ยซึ่งในสภาพปัจจุบันในประเทศไทยยังอาศัยความสัมพันธ์เชิงบุคคลอยู่มาก ดังนั้นระบบผู้เชี่ยวชาญจึงถูกจำกัดด้วยความสามารถขั้นพื้นฐานในการทำงานนี้ (เพราะการจะเขียนชุดคำสั่งงานให้คอมพิวเตอร์สามารถโทรศัพท์ติดต่อได้เช่นเดียวกับคนธรรมดา นั้นเป็นปัญหาขนาดใหญ่มากในตัวของมันเองอยู่แล้ว)

ระบบสุดท้ายเกี่ยวกับการซื้อขายหลักทรัพย์นั้น ก็เป็นหัวข้อปัญหาที่น่าสนใจมาก แต่ความเป็นผู้เชี่ยวชาญในด้านนี้ ยังไม่มีนิยามและการยอมรับที่เป็นมาตรฐานใช้ได้ทั่วไป และผู้พัฒนาได้เลือกวิธีการวิเคราะห์ที่เรียกกันว่า การวิเคราะห์เชิงเทคนิค (technical analysis) มาเป็นเกณฑ์ในการพัฒนาระบบ เป็นผลให้ระบบมีลักษณะเป็นชุดคำสั่งงานธรรมดาที่มีการทำงานในลักษณะโต้ตอบสูงกว่าปกติ แทนที่จะเป็นระบบผู้เชี่ยวชาญที่แท้จริง

#### 4.2 การแสวงหาความรู้ (knowledge acquisition) และวิศวกรรมความรู้ (knowledge engineering)

ในเรื่องการแสวงหาความรู้และวิศวกรรมความรู้ นั้นอาจแบ่งเป็นหัวข้อย่อยได้อีก 3 หัวข้อคือ การเลือกผู้เชี่ยวชาญ วิธีการถ่ายทอดความรู้จากผู้เชี่ยวชาญ การและเครื่องมือในการถ่ายทอดความรู้

ในการเลือกผู้เชี่ยวชาญนั้น ผู้ดำเนินงานทุกระบบได้ใช้เกณฑ์ที่พึงประสงค์ทั่วไปเกี่ยวกับลักษณะของผู้เชี่ยวชาญ กล่าวคือ เป็นผู้ที่เกี่ยวข้องชาญในหัวข้อปัญหาอย่างแท้จริง มีความสนใจและกระตือรือร้นที่จะทำงานนี้ มีทักษะสูงในการอธิบายแนวความคิดของตน ทราบดีหรือเข้าใจถึงประโยชน์ของคอมพิวเตอร์ในงานลักษณะนี้ สามารถและเต็มใจที่จะอุทิศเวลาของตนเพื่อการนี้ได้อย่างแท้จริงในเวลา เป็นจริงแต่ละระบบก็ประสบความสำเร็จในระดับที่แตกต่างกันออกไปแต่ไม่มีชุดใดสามารถหาผู้เชี่ยวชาญที่มีลักษณะที่ตรงกับทุกประการ โดยเฉพาะอย่างยิ่งในคุณสมบัติสุดท้ายเรื่องเวลา โดยส่วนรวมผู้เชี่ยวชาญที่เป็นแพทย์ให้เวลากับโครงการได้มากที่สุด ทั้งนี้อาจเป็นเพราะ 2 ใน 3 เป็นญาติกับผู้พัฒนาระบบ และ

อีกคนหนึ่งมีความรู้ทางด้านคอมพิวเตอร์ค่อนข้างดี ผู้ชำนาญการที่เป็นนักธุรกิจมักจะให้เวลากับโครงการได้ไม่มากนัก (สัปดาห์ละไม่เกิน 2 ครั้ง ๆ ละไม่เกิน 2 ชั่วโมง) และส่วนมากไม่มีความเข้าใจ หรือเชื่อใจในความสามารถของคอมพิวเตอร์ว่าจะมาทำงานแทนตนได้จริง ซึ่งก่อให้เกิดผลเสียในการทำงานในช่วงนี้

สำหรับวิธีการถ่ายทอดความรู้ นั้น วิศวกรรมความรู้ส่วนใหญ่ต้องใช้เวลาในช่วงแรกศึกษาตำราเบื้องต้นของหัวข้อปัญหาซึ่งเป็นความรู้ใหม่ของแต่ละกลุ่มแทบทั้งสิ้น ในช่วงต่อมาเป็นการศึกษาวิธีที่เกิดขึ้นแล้ว โดยผู้เชี่ยวชาญเป็นผู้อธิบายถึงลำดับขั้นตอนและเหตุผลในการตัดสินใจดำเนินการต่างๆ เช่น การสรุป การตั้งคำถาม การตอบรับ หรือตอบปฏิเสธ เป็นต้น ในช่วงนี้วิศวกรรมความรู้และผู้ชำนาญการจะพยายามมองหา (formulate) กฎเกณฑ์ต่าง ๆ ที่ใช้ในการแก้ปัญหา และกลุ่มหนึ่งได้สร้างระบบต้นแบบ (prototype) ขึ้นตามเกณฑ์ที่กำหนดเพื่อให้ผู้เชี่ยวชาญทดลองใช้และประเมินผลซึ่งเป็นการกระทำที่พึงปรารถนาแต่กระทำได้ลำบากในสถานการณ์จริง เนื่องจากขาดแคลนเครื่องมือ ดังจะได้กล่าวต่อไปในหัวข้อที่สาม นอกจากนี้เพียง 2-3 กลุ่มเท่านั้น ที่ผู้เชี่ยวชาญได้มีโอกาสลองใช้ระบบที่พัฒนาขึ้นมากับการมีข้อมูลจริงในระหว่างการพัฒนา ทั้งนี้ด้วยข้อจำกัดด้านเวลาของผู้เชี่ยวชาญ ลักษณะงานของหัวข้อปัญหาต่าง ๆ และการขาดแคลนอุปกรณ์คอมพิวเตอร์ในการทำงานจริง

ในเรื่องเครื่องมือสำหรับถ่ายทอดความรู้ นั้น เนื่องจากผู้พัฒนาส่วนใหญ่เลือกใช้ภาษา Prolog ซึ่งเป็นภาษา logic-based เท่านั้นจึงไม่มีส่วนของภาษาที่ช่วยทำงานนี้โดยอัตโนมัติ ผู้พัฒนากลุ่มหนึ่งเลือกใช้ภาษา EXSYS ซึ่งมีลักษณะเป็นเปลือกกระบวนผู้เชี่ยวชาญ

(expert system shell) และมีเครื่องมือสำหรับช่วยในการสร้างระบบผู้เชี่ยวชาญการอยู่ข้างแต่เนื่องจากผู้พัฒนากลุ่มนี้มิได้ลงทะเบียนเรียนวิชาเพื่อหน่วยกิต จึงไม่ได้พยายามใช้ความสามารถด้านนี้ที่มีอยู่ สำหรับโครงการระยะยาวกว่านั้นไม่ว่าผู้พัฒนาจะเลือกใช้ภาษาใดในการพัฒนาระบบ เมื่อพัฒนาขึ้นมาได้ระดับหนึ่ง ระบบดังกล่าวควรมีความสามารถที่จะช่วยในการถ่ายทอดและรับความรู้ซึ่งผู้พัฒนาสามารถใช้ในการแสวงหาความรู้จากผู้ชำนาญการได้ในช่วงต่อไป มีระบบผู้เชี่ยวชาญการบงการระบบที่เขียนด้วยภาษา Prolog ได้แยกส่วนของความรู้ในเรื่องต่าง ๆ ออกอย่างชัดเจนเพื่อช่วยให้สามารถเปลี่ยนแปลงปรับปรุงฐานความรู้ได้โดยไม่กระทบกระเทือนส่วนอื่น ๆ ของระบบ เช่นใน Diagnose สามารถเพิ่ม/ลด/เปลี่ยนชื่อยาได้ และใน Orthopaedic สามารถเปลี่ยนน้ำหนักสำหรับสมมุติฐานต่าง ๆ ในแต่ละโรค (ซึ่งมีผลเท่ากับเปลี่ยนกฎที่ใช้สรุปว่าผู้ใดเป็นโรคอะไร) ได้ เป็นต้น

#### 4.3 ภาษาสั่งงาน

จากระบบผู้เชี่ยวชาญทั้งสิ้น 8 ระบบ 6 ระบบเลือกใช้ภาษา Turbo Prolog โดย Borland (1986) เป็นภาษาสั่งงาน ระบบหนึ่งเลือกใช้ภาษา EXSYS (1984) และระบบหนึ่งใช้ภาษา Basic ร่วมกับภาษา Prolog ระบบทุกระบบเขียนขึ้นใช้งานกับเครื่องไมโครคอมพิวเตอร์ ผู้พัฒนามีชุดคำสั่งแปลภาษา Prolog ให้เลือกใช้ถึง 3 ชุดด้วยกันคือ Turbo Prolog, Micro Prolog และ Prolog 86 แต่เนื่องจากมีการสอนภาษา Turbo Prolog ในชั้นเรียนประกอบกับการทำงานของชุดคำสั่งแปลภาษานี้ทำงานได้อย่างมีประสิทธิภาพเป็นที่น่าพอใจ ผู้พัฒนาส่วนใหญ่จึงเลือกใช้ภาษานี้ ทั้งๆ ที่เมื่อเทียบแล้วภาษา Turbo Prolog เป็นเพียงภาษาที่

เรียกว่าภาษาเชิงตรรก (logic-based) ทั่วไปและนับไม่ได้ว่าเป็นเครื่องมือในการพัฒนาระบบผู้เชี่ยวชาญที่ดีเท่าภาษาที่มีลักษณะเป็นเปลือกระบบผู้เชี่ยวชาญ เช่น ภาษา Expert, EMycin (Buchanan and Shortliffe 1984) และ OPS5 (Forgy 1981) หรือมีลักษณะเป็นชุดเครื่องมือ (tool kit) กล่าวคือ มีวิธีการที่ใช้ในหลายเปลือกระบบผู้เชี่ยวชาญ เช่น ESE (Hirsch et. al. 1985) KEE (Pikes and Kehler 1985) และ LOOPS (Stefik et. al. 1983) แต่เนื่องจากผู้พัฒนาไม่คุ้นเคยกับภาษาดังกล่าวและพูดคำสั่งแปลภาษาเหล่านั้นส่วนมากก็ไม่แพร่หลาย ผลจึงเป็นดังกล่าวมาแล้ว

จากผลการใช้งานได้พบว่า Turbo Prolog ทำงานได้เร็ว สร้างสภาพแวดล้อมสำหรับการโต้ตอบ (interactive environment) ได้ดีมาก สร้างและใช้จอภาพ (screen) และหน้าต่าง (window) ได้สะดวก เหมือนดังที่ Shamma (85) ได้รายงานไว้ นอกจากนี้ยังสามารถทำงานกับภาษาไทย (ในระดับรับข้อมูล-แสดงผล แต่ไม่นำไปประมวลผล) ได้โดยง่าย ซึ่งมีประโยชน์สามารถใช้สร้างระบบผู้เชี่ยวชาญที่ถามคำถามและแสดงผลเป็นภาษาไทยได้ เพียงแต่ผู้ใช้ต้องตอบคำถามด้วยอักษร "y" หรือ "n" หรือรหัสเป็นตัวเลข (แทนที่จะใช้ภาษาไทยซึ่งก็มิสะดวกไปกว่านั้น) เท่านั้น นอกจากนั้นการใช้ภาษา Prolog ยังทำได้ในลักษณะประกาศ (declarative) ซึ่งถ้าใช้ในลักษณะนี้ ภาษา Prolog จะทำงานเป็นพินิจยนตร์ (inference engine) โดยอัตโนมัติโดยใช้วิธีการมุ่งหาเป้า (goal-seeking) ซึ่งจัดเป็นวิธีการแบบโยกถอยหลัง (backward chaining) แต่ถ้ามีเหตุผลที่ต้องการทำงานแบบกำหนดวิธีการ (procedural) ที่ใช้วิธีการวินิจฉัยแบบอื่นเช่น โยกก้าวหน้า (forward chaining) ก็สามารถ

ทำได้ดังเช่นในระบบ Diagnose และระบบ Package Tour เป็นต้น ระบบผู้เชี่ยวชาญระบบอื่นเช่น ระบบจิตเวช และระบบ Orthopaedic ก็ใช้วิธีการหาคำตอบที่เป็นแบบกำหนดวิธีการแบบอื่นที่มีใช้การโยงถอยหลัง หรือการโยงก้าวหน้า และก็สามารถใช้ภาษา Prolog ได้โดยสะดวกเช่นกัน โดยสรุปแล้วผู้เขียนเห็นด้วยกับ Webster (86) ว่า ภาษานี้เหมาะสมสำหรับการใช้สร้างระบบผู้เชี่ยวชาญ

สำหรับภาษา EXSYS นั้นก็นับว่าเป็นเลือกระบบผู้เชี่ยวชาญที่น่าสนใจมากภาษาหนึ่ง เพราะมีความต้องการต่ำคือสามารถใช้ได้กับเครื่องไมโครระดับ IBM PC ที่มีขนาดความจำตั้งแต่ 256 KB ขึ้นไป ต้องการเครื่องอ่าน/เขียนจานแม่เหล็กเพียงหนึ่งเครื่อง และใช้ชุดคำสั่งงานควบคุม version DOS 2.0 ขึ้นไป ภาษานี้ใช้วิธีการจัดเก็บความรู้แบบกฎ (rules) และสามารถรับกฎได้ประมาณ 700 กฎต่อทุกๆ 64 KB ที่เกิน 192 KB ขึ้นไป จึงมีศักยภาพที่จะสร้างระบบผู้เชี่ยวชาญขนาดใหญ่มากได้ในเครื่องไมโครคอมพิวเตอร์ขนาดเพียง 512 KB หรือ 640 KB เท่านั้น นอกจากนี้ยังมีคุณลักษณะที่เป็นประโยชน์สมกับที่เป็นระบบผู้เชี่ยวชาญกรอกรหัสหลายประการเช่น ทุก ๆ ครั้งที่รับกฎข้อใหม่เข้าไปจะตรวจสอบความสมนัย (consistency) กับกฎต่างๆ ที่มีอยู่แล้ว และสามารถรับและทำงานกับความรู้ที่มีความไม่แน่นอนได้โดยกำหนดให้เป็นความน่าจะเป็นระดับต่างๆ และมีวิธีการให้เลือกจัดการกับความน่าจะเป็น นี้น่ามากกว่าหนึ่งแบบรวมทั้งแบบ Bayes ด้วย เป็นต้น

#### 4.4 การจัดเก็บความรู้เฉพาะเรื่อง

เป็นที่ทราบกันดีว่าในปัจจุบันนี้มีวิธีการจัดเก็บความรู้เฉพาะเรื่องที่น่าสนใจอยู่ 3 วิธีด้วยกัน (ดู

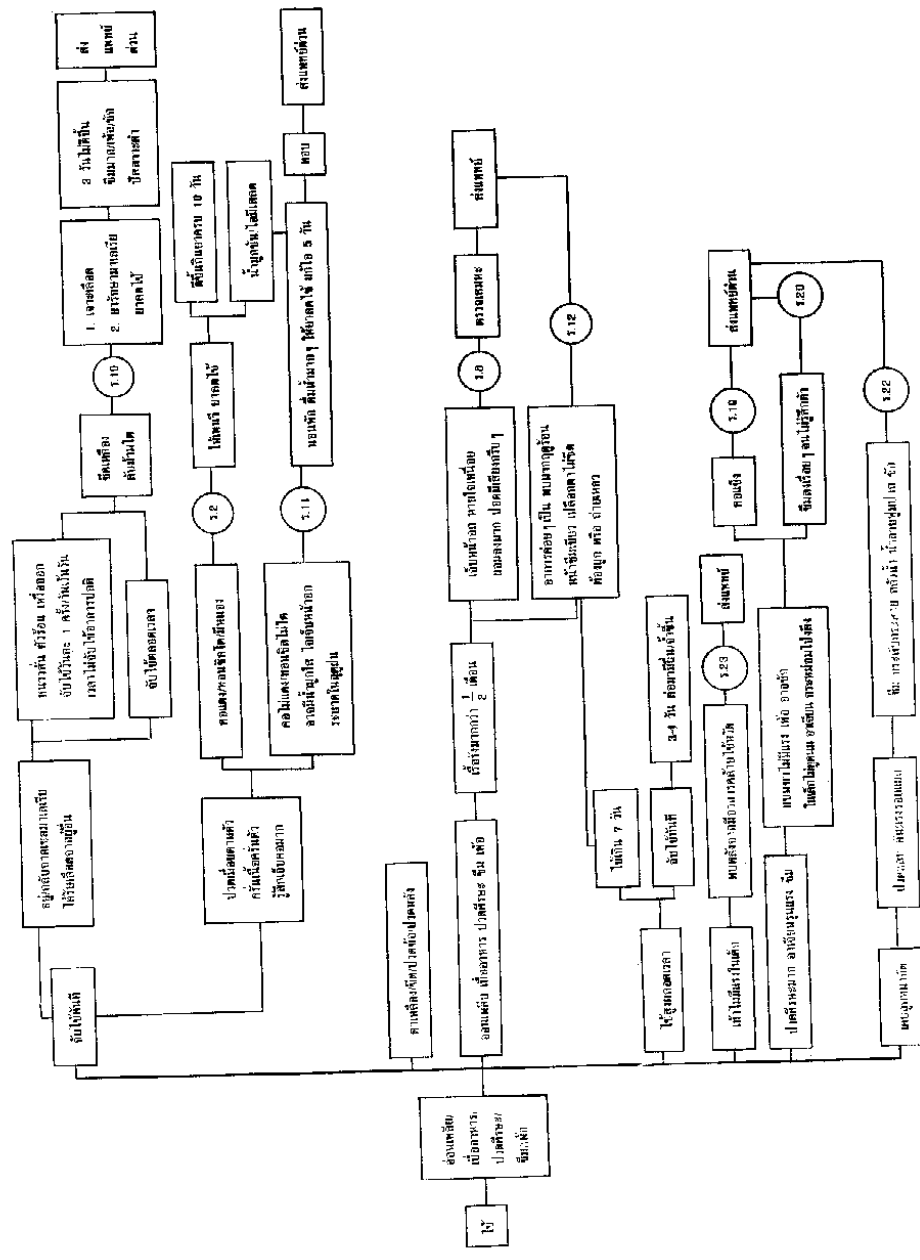
Waterman (1986) หน้า 63-79 และ Alty et. al. (1984) หน้า 60-76) คือ กฎ (rules) หรือกฎการผลิต (production rules) ข่ายความหมาย (semantic nets) หรือ เครือข่ายความหมาย (semantic networks) และกรอบ (frames) ในบรรดาวิธีที่ผู้เชี่ยวชาญที่กล่าวถึง ผู้พัฒนาส่วนใหญ่เลือกใช้วิธีการจัดเก็บแบบกฎ ซึ่งมีวิธีการจัดเก็บจริงแตกต่างกันหลายรูปแบบจนมีบางกลุ่มเข้าใจคลาดเคลื่อนว่าตนได้จัดเก็บโดยวิธีข่ายความหมาย แต่ความจริงแล้วควรจะเรียกว่าเป็นแบบกฎมากกว่า (ดูภาพ 2)

ผู้พัฒนาบางระบบยังคงคุ้นเคยกับวิธีการเขียนคำสั่งงานแบบกำหนดวิธีการ และ algorithmic อยู่ จึงใช้วิธีการดังกล่าวในการจัดเก็บความรู้ (โดยเฉพาะอย่างยิ่งในระบบที่พัฒนาด้วยภาษา Basic) ซึ่งเป็นสาเหตุให้หลายๆ ระบบไม่มีการแยกส่วนของความรู้ออกจากส่วนควบคุม หรือพินิจยนต์ (inference engine) อย่างแจ่มชัด ทำให้สูญเสียลักษณะของระบบผู้เชี่ยวชาญไป

สาเหตุที่ผู้พัฒนาส่วนใหญ่นิยมใช้กฎมากกว่าข่าย หรือกรอบในการจัดเก็บความรู้ นั้น ส่วนหนึ่งจะเป็นเพราะภาษาที่ใช้ เพราะทั้ง Prolog และ EXSYS ต่างก็เป็นภาษาที่เอื้ออำนวยต่อการจัดเก็บแบบกฎมากกว่าแบบอื่น และอีกส่วนหนึ่งน่าจะเกิดจากความเคยชิน และความเข้าใจของผู้พัฒนาเพราะกฎเข้าใจได้ง่ายกว่าอีก 2 แบบ และตัวอย่างระบบผู้เชี่ยวชาญที่พัฒนาแล้วและมีชื่อเสียงถูกยกมาเป็นตัวอย่างส่วนใหญ่จะใช้กฎ เช่น MYCIN, DENDRAL, PROSPECTOR, RI, XCON และ YES/MVS เป็นต้น

ผู้เขียนมีความรู้สึกว่าการจัดเก็บแบบกฎสามารถใช้งานได้ดีกับปัญหาทุกประเภท แต่ปัญหาที่สามารถจัดกลุ่ม (group) หรือชั้น (class) ของสิ่งของ

ภาพที่ 2 : ตัวอย่างการจัดเก็บความถี่เฉพาะเรื่อง (บางส่วน) ใน Diagnose



(objects) ในเซตที่เราสนใจได้อย่างค่อนข้างชัดเจน และในแต่ละกลุ่มหรือชั้นมีรายละเอียดร่วมกันจำนวนมาก การจัดเก็บแบบกฎจะมีประสิทธิภาพต่ำกว่าการจัดเก็บแบบข้อความหมายมาก ส่วนการจัดเก็บแบบกรอบนั้น น่าจะเหมาะสมกับปัญหาประเภทที่สามารถจัดกลุ่มหรือชั้นของสิ่งของได้ แต่ในแต่ละกลุ่มหรือชั้นนั้น นอกจากจะมีรายละเอียดร่วมกันจำนวนมากแล้ว ยังมีรายละเอียดที่แตกต่างกันอยู่มากอีกด้วย อย่างไรก็ตามวิธีการเลือกวิธีการจัดเก็บส่วนความรู้ยังขึ้นอยู่กับความต้องการที่ผู้พัฒนาสามารถเลือกใช้ได้ โดยเฉพาะอย่างยิ่งถ้าเป็นภาษาประเภท เลือกกระบวนผู้ชำนาญการที่มีวิธีการจัดเก็บเพียงแบบเดียว เช่น EMycin, Expert หรือ OPS 5 ผู้พัฒนา ก็คงไม่มีทางเลือกนอกจากจะใช้วิธีการจัดเก็บที่มีอยู่ในภาษานั้น

#### 4.5 ฟินิจันตรี (inference engine) หรือ ส่วนควบคุม (control structure)

เนื่องจากผู้พัฒนาส่วนใหญ่เลือกใช้ภาษา Prolog จึงมีฟินิจันตรีของภาษา Prolog โดยอัตโนมัติอยู่แล้ว แต่เมื่อพิจารณาชุดคำสั่งงานดังกล่าวในรายละเอียดแล้ว ปรากฏว่า ผู้พัฒนาส่วนใหญ่ไม่ได้ใช้ความสามารถส่วนนี้ของภาษา Prolog มากนัก แต่กลับเขียนชุดคำสั่งงานให้ทำงานแบบโยงก้าวหน้า หรือแบบผสมอื่นๆ เสียเป็นส่วนใหญ่ (ตัวอย่างเช่นในภาพที่ 2) และหลายระบบก็มีวิธีการทำงานที่เป็นขั้นตอน (algorithmic) มากกว่าวิธีการหยั่งรู้ (heuristic) ซึ่งคงจะเกิดมาจากสาเหตุใหญ่ 2 ประการ ได้แก่ ความเคยชินของผู้พัฒนาประการหนึ่ง และเกณฑ์ที่ได้จากผู้ชำนาญการที่อาจมีลักษณะเป็นขั้นตอนอย่างมากอีกประการหนึ่ง ปัญหาประการหลังนี้แท้จริงแล้วเป็นปัญหาที่สืบเนื่องมาจากเรื่องของการแสวงหาความรู้และธรรมชาติของระบบ

ผู้ชำนาญการ ซึ่งควรจะมีพัฒนาการในทางที่ซับซ้อนยิ่งขึ้นไปพร้อมกับการเติบโตและก้าวหน้าด้านวิศวกรรมความรู้ รวมถึงเครื่องมือต่างๆ สำหรับวิศวกรรมความรู้

ดังได้กล่าวแล้วว่าเกณฑ์หรือวิธีการทำงานที่วิศวกรรมความรู้แสวงหามาได้จากผู้ชำนาญการนั้นจะมีอิทธิพลกับฟินิจันตรีที่วิศวกรรมความรู้จะเลือกใช้ในบรรดาระบบผู้ชำนาญการที่ศึกษา เกณฑ์เหล่านี้มีตั้งแต่ที่มีรูปแบบง่าย ๆ (แต่ไม่ได้หมายความว่า จะหามาได้โดยง่าย) เช่นเป็นเซตของสมมูลฐานกลุ่มหนึ่งสำหรับแต่ละโรค หรือเป็นเซตที่มีเงื่อนไขทางอ้อมๆ (เช่นต้องมีอย่างน้อย 3 สมมูลฐาน จากในเซตนี้รวมกับอีกอย่างน้อย 2 สมมูลฐานจากอีกเซตหนึ่ง) จนกระทั่งเกณฑ์ที่มีโครงสร้างเป็นลำดับขั้นที่สลับซับซ้อนเช่นในการพิจารณาให้สินเชื่อโดยดูจากงบดุล และงบกำไรขาดทุน จะพยายามหาระดับสภาพคล่องของบริษัทที่ขอสินเชื่อ โดยพิจารณาจาก quick ratio, current ratio ฯลฯ พยายามหาความมีกำไร (profitability) จาก profit margin และการเปรียบเทียบอัตราส่วนผลตอบแทนต่อทรัพย์สิน และผลตอบแทนต่อเงินกองทุน กับอัตราเฉลี่ยอุตสาหกรรม หา leverage ของบริษัทในลักษณะเดียวกัน แล้วนำ leverage, ความมีกำไร และระดับสภาพคล่องมาประเมินความสามารถในการจัดการประเมินความเสี่ยงของบริษัทจากความเต็มใจในการชำระหนี้ ความสามารถในการชำระหนี้ และเงินลงทุน จนท้ายที่สุดตัดสินใจว่าควรให้สินเชื่อหรือไม่จากความเสี่ยงของบริษัทและหลักทรัพย์สินที่ใช้ค้ำประกันหนี้ และในระหว่างนี้ก็สามารให้คำแนะนำเกี่ยวกับการดำเนินกิจการแก่บริษัทตามสภาพของค่าต่างๆ ที่ประเมินได้ เช่น บริษัทควรเพิ่มสินค้าในสต็อก บริษัทควรจ่ายเงินปันผลเพราะส่วนของผู้ถือหุ้นมีมูลค่าสูง และบริษัท



ควรลดหนี้สินระยะยาวโดยการขายทรัพย์สินที่ไม่ให้  
ดอกเบี้ย เป็นต้น

สำหรับวิธีการในการสรุปเพื่อเสนอข้อแนะนำ  
หรือข้อวินิจฉัยนั้นส่วนใหญ่ได้ใช้วิธีการโยงก้าวหน้า  
หรือวิธีการที่ค่อนข้างเป็นขั้นตอนอื่นๆ ดังได้กล่าวแล้วมี  
กรณีหนึ่งที่น่าสนใจและอาจนำไปใช้ได้ในสภาพการณ์  
อื่นๆ คือ ระบบผู้เชี่ยวชาญ Orthopaedic ใช้วิธีการ  
ให้เติมแก่สมมุติฐานเพื่อสรุปโรค (ดูตารางที่ 2 เป็น  
ตัวอย่าง)

ในกรณีพิเศษ (special case) ที่คะแนน  
เป็นหนึ่งในห้าทั้งหมดทุกสมมุติฐาน เช่น สมมติว่ามี 10  
สมมุติฐาน และเกณฑ์ในการสรุปเป็น 3 การใช้ข้อ  
สรุปด้วยวิธีนี้จะลดภาระในการเขียนคำสั่งลงไปมาก  
เพราะกรณีที่จะเกิดขึ้นได้ทั้งหมดคือ  $10C_3 = 120$  กรณี  
แต่สามารถเขียนครอบคลุมได้ทุกกรณีโดยเขียนคำสั่ง  
ครั้งเดียวเท่านั้น สำหรับเกณฑ์ในการสรุป Gouty  
arthritis และ Osteoarthritis ที่ยกมาเป็นตัวอย่าง  
นี้อาจจะไม่มีปริมาณถึง 120 แต่การหากรณีให้ครบจะ  
ทำได้ค่อนข้างยุ่งยากและอาจตกหล่นไม่ครบถ้วนถ้าไม่  
ใช้วิธีการให้คะแนน ส่วนปัญหาที่ว่าสมมุติฐานดังกล่าว  
จะไม่มี auto-correlation ซึ่งกันและกัน และ  
สามารถนำคะแนนมาบวกกันได้โดยตรงหรือไม่ ตลอดจน  
จนการให้คะแนนนั้นถูกต้องเพียงไร เป็นปัญหาในเรื่อง  
วิศวกรรมความรู้ที่กล่าวถึงแล้ว สังเกตว่า สมมุติฐาน  
(๑) บางคู่ไม่สามารถเกิดขึ้นพร้อมกันได้ เช่น Female  
และ Male เป็นต้น

#### 4.6 ส่วนช่วยอธิบาย (explanation module)

ระบบผู้เชี่ยวชาญส่วนมากที่ศึกษามีส่วนช่วย  
อธิบายแบบง่าย ๆ กล่าวคือ สามารถตอบได้ว่าสาเหตุ  
ที่สรุปได้คำตอบออกมาเป็นเพราะว่าได้รับข้อเท็จจริง

อะไรเข้าไป และสรุปอะไรเพิ่มเติม (โดยใช้กฎข้อใด)  
ซึ่งวิธีทำก็ตรงไปตรงมาโดยเพียงแต่เก็บ trace ของการ  
สรุปทุกครั้งเข้าไว้ และเป็นส่วนที่ทำได้โดยอัตโนมัติใน  
ภาษา EXSYS นอกจากนี้ในภาษา EXSYS ยัง  
สามารถตอบคำถาม “ทำไม” เมื่อผู้ใช้สงสัยว่าทำไม  
ระบบจึงตามคำถามใดคำถามหนึ่งได้โดยอัตโนมัติอีกด้วย  
โดยที่ EXSYS ใช้พินิจยนตร์แบบโยงถอยหลัง หรือมุ่ง  
สู่เป้า คำตอบที่ได้จึงเป็นเพียงคำอธิบายเฉพาะหน้าว่า  
ระบบกำลังพยายามจะเข้าสู่กฎข้อใด ซึ่งมีข้อสมมติและ  
ข้อสรุปอะไรบ้างเท่านั้น ซึ่งอาจมีประโยชน์หรือไม่  
ประโยชน์ก็ได้ ขึ้นอยู่กับว่า chain of reasoning ยาว  
เพียงไรและผู้ถามคำถาม “ทำไม” ติดตามขั้นตอนของ  
การทำงานโยงถอยหลังได้ใกล้ชิดเพียงใด

นอกจากนั้นบางระบบก็สามารถแสดงรายการ  
ข้อเท็จจริงต่าง ๆ ที่ผู้ใช้ป้อนให้กับระบบ เพื่อใช้ตรวจ  
สอบความถูกต้องของคำตอบสุดท้าย และความถูกต้อง  
ของคำตอบที่ผู้ใช้ได้ตอบไปได้อีกด้วย

โดยสรุปแล้วผู้พัฒนาระบบที่ศึกษาไม่ได้ใช้  
ความพยายามกับส่วนนี้มากนักเพราะเห็นว่าเป็นส่วนที่  
มีความสำคัญในระดับรอง ประกอบกับมีเวลาและ  
ทรัพยากรจำกัดในการพัฒนาระบบ

#### 4.7 ส่วนติดต่อกับผู้ใช้ (user interface module)

ระดับของความสามารถของส่วนนี้ขึ้นอยู่กับว่า  
จะใช้อะไรเป็นเกณฑ์วัด ถ้าตั้งระดับความคาดหวังไว้  
สูง เช่น Walker (ed) (1987) ว่าจะให้ระบบผู้เชี่ยวชาญ  
สามารถติดต่อกับผู้ใช้ได้ด้วยภาษาคน (natural  
language) ก็ต้องนับว่าระบบที่ศึกษาทุกระบบล้มเหลว  
ในส่วนนี้โดยสิ้นเชิง แต่ในความเป็นจริงแล้วน่าจะ  
ต้องพิจารณาเป็นรายกรณีไปว่าระบบผู้เชี่ยวชาญนั้นมี  
ความชำนาญในเรื่องอะไร มีวัตถุประสงค์จะใช้งานกับ

ตารางที่ ๒ ตัวอย่างการให้คะแนนเพื่อสรุปโรค

**Disease :** Gouty arthritis.

**Criteria :** Sum of score greater or equal 3.00

| Information            | Score |
|------------------------|-------|
| Male                   | 1.00  |
| Subcutaneous nodule    | 0.25  |
| Monoarthritis          | 1.00  |
| Polyarthritis          | 0.50  |
| Symmetry pattern       | 0.25  |
| Asymmetry pattern      | 0.50  |
| Ankle joint            | 0.50  |
| First metatarsal joint | 1.00  |

**Disease :** Osteoarthrosis.

**Criteria :** Sum of score greater or equal 7.25

| Information                            | Score |
|--|-------|
| Female                                 | 2.00  |
| Male                                   | 1.00  |
| Weight bearing joint                   | 1.00  |
| Small joint                            | 0.50  |
| Large joint                            | 0.50  |
| Distal interphalangeal joint of finger | 0.50  |
| Monoarthritis                          | 0.50  |
| Oligoarthritis                         | 1.00  |
| Polyarthritis                          | 1.00  |
| Stiffness                              | 1.00  |
| Crepitus                               | 1.50  |
| Locking                                | 1.00  |
| Pain on standing                       | 1.00  |
| Pain all time                          | 0.50  |
| Restrict movement                      | 1.00  |
| Herberden's node                       | 2.00  |
| Effusion                               | 0.50  |

คนกลุ่มใด และคนกลุ่มนั้นโดยปรกติจะสนทนากับผู้ชำนาญการในลักษณะและระดับใด ถ้าการสนทนาเป็นระดับเทคนิคก็น่าจะเพียงพอที่ระบบผู้เชี่ยวชาญจะติดต่อกับผู้ใช้ในระดับเทคนิค แต่ถ้าปรกติติดต่อยู่ในระดับชาวบ้าน (เช่นหมอกับคนไข้) ระบบผู้เชี่ยวชาญก็ไม่ควรจะถามคำถามเทคนิค เป็นต้น

ดังได้กล่าวแล้วในส่วนที่รายงานเกี่ยวกับภาษาสั่งงานว่า ระบบบางระบบสามารถโต้ตอบกับผู้ใช้เป็นภาษาไทย ส่วนมากจะใช้เทคนิค หน้าต่าง และรายการ (menu) ซึ่งก็ค่อนข้างสะดวกสำหรับการทำงานกับคอมพิวเตอร์ แต่อาจไม่สอดคล้องกับสภาพความเป็นจริง (เช่นหมอก็มักจะไม่ยื่นรายการอาการโรคให้คนไข้เลือกตอบว่ามีหรือไม่ แต่มักจะให้คนไข้แจ้งอาการเองและถามเพิ่มเติม) ประเด็นนี้ความจริงแล้วคงเป็นทางเลือกของผู้พัฒนาเพราะไม่ว่าจะเลือกวิธีใดก็มีทั้งข้อดีและข้อเสีย การยอมให้ผู้ใช้แจ้งอาการด้วยตนเองทำได้สะดวกในสภาพความเป็นจริง แต่ทำใยากถ้าผู้ใช้ไม่เคยชินกับการพิมพ์ดีด และจะทำงานได้ช้ากว่าทำกับรายการ แต่ถ้าหัวข้อในรายการไม่ตรงกับที่ตนต้องการจะบอกก็จะทำให้หน้าเบื่อ เป็นต้น นอกจากนี้บางระบบที่ยอมให้ผู้ใช้พิมพ์อาการเข้าไปในระบบได้พยายามแก้ปัญหาด้วยค่าเบื้องต้นระดับหนึ่ง เช่น รับคำว่า “เป็นไข” “มีไข้” และ “จับไข้” ด้วยความหมายเดียวกัน และสำหรับคำตอบเรื่องเพศ (ซึ่งในสภาพความเป็นจริงคงไม่ต้องถาม) ก็รับคำตอบ “ผู้ชาย” “ชาย” และ “ช” เป็นคำตอบเดียวกัน เป็นต้น นอกจากนี้บางระบบก็ยอมให้มีการทบทวนและแก้ไขคำตอบเพื่อเพิ่มความสะดวกในการใช้ระบบ

ความจริงแล้วขอบข่ายความสามารถในการทำงานของส่วนนี้มีความสัมพันธ์กับส่วนช่วยอธิบายอยู่

มาก และในการทำงานบางหน้าที่ก็แทบจะแยกไม่ออกว่าเป็นหน้าที่ของส่วนนี้หรือส่วนช่วยอธิบาย โดยหลักการแล้ว ประเด็นในการพิจารณาออกแบบและพัฒนาตลอดจนวิธีการที่ใช้ของส่วนนี้ก็ไม่แตกต่างไปจากประเด็นและวิธีการที่ใช้ในการออกแบบส่วนที่ทำหน้าที่นี้ในซอฟต์แวร์ระบบอื่น ๆ ไม่ว่าจะเป็นเรื่องของการใช้งานและชวนให้ใช้ ความมีประสิทธิภาพในการใช้ การลดอัตราความผิดพลาดในระหว่างการติดต่อระหว่างคนและระบบ ฯลฯ หรือการใช้ภาษาไทย การใช้หน้าต่างหลายอัน การใช้หน้าต่างซ้อนซ้อน (pop-up window) การใช้รายการ การใช้กราฟิกส์หรือรูปภาพ การใช้เสียง ฯลฯ ก็ตาม ดังนั้นจึงอาจถือได้ว่าส่วนนี้เป็นส่วนที่มีลักษณะ “เฉพาะตัว” น้อยที่สุดในรอบเวลาหัวข้อต่าง ๆ เกี่ยวกับการพัฒนาระบบผู้เชี่ยวชาญที่ได้นำมาพิจารณาในบทความนี้

## 5. บทสรุป

ประสบการณ์จากการศึกษาระบบผู้เชี่ยวชาญระดับต้นแบบสาธิต (demonstration prototype) ทั้ง 8 ระบบนี้แสดงให้เห็นอย่างชัดเจนถึงความสำคัญเกี่ยวกับการเลือกหัวข้อปัญหา และการเลือกผู้เชี่ยวชาญ ผู้พัฒนาควรจะต้องใช้เวลาพิจารณาหาคำตอบใน 2 เรื่องนี้ ให้เพียงพอ มิฉะนั้นแล้วจะต้องเสียอย่างสูงกับการกลับมาเริ่มต้นใหม่ครั้งแล้วครั้งเล่าหรือประสบกับความล้มเหลวโดยสิ้นเชิง โดยเฉพาะอย่างยิ่งในเรื่องเกี่ยวกับเวลาที่ผู้เชี่ยวชาญจะให้แก่โครงการได้

ในการดำเนินงานนั้น ผู้พัฒนาควรเตรียมตัวเตรียมใจที่จะพบกับอุปสรรคสำคัญยิ่งในช่วงของการแสวงหาความรู้ โดยเฉพาะอย่างยิ่งถ้าต้องการจะจำลองขั้นตอนการคิดให้เหมือนกับผู้เชี่ยวชาญ แม้แต่ระบบที่มีชื่อเสียงและใช้อ้างอิงกันทั่วไป เช่น MYCIN

(Shortliffe (1976)) และ PROSPECTOR (Gaschnig (1981)) ก็ไม่ได้ทำเช่นนั้น (Alty et. al. (1984 หน้า 199)) ในบรรดาระบบที่ศึกษาในครั้งนี้มีเพียงไม่กี่เกินครึ่งที่ทำเช่นนั้นได้ แต่ลักษณะงานของระบบเหล่านี้ก็เอื้ออำนวยและมีลักษณะเป็น ขั้นตอนค่อนข้างมาก (เช่น ระบบ Tour Package, Financial Analyst, Diagnose และ Stocks Trader เป็นต้น) ระบบที่เหลือโดยเฉพาะอย่างยิ่งระบบ Orthopaedic ใช้วิธีการของ MYCIN และ PROSPECTOR ที่มุ่งหวังผลสรุปที่ถูกต้องยอมรับได้แต่อาจจะ (และคงจะ) ไม่ตรงกับที่ผู้ชำนาญการใช้จริง

ภาษาสั่งงานที่มีใช้อยู่ในประเทศไทยขณะนี้แม้จะยังไม่มีความที่เป็นเปลือกหุ้มระบบผู้เชี่ยวชาญ และชุดเครื่องมือ (toolbox) มากนักแต่ผู้เชี่ยวชาญมีเพียงพอที่จะใช้พัฒนาระบบระดับวิจัยได้อย่างสบาย ๆ และภาษา เช่น Turbo Prolog, LISP หรือ EXSYS ก็น่าจะเพียงพอที่จะใช้เป็นเครื่องมือในการพัฒนาระบบขนาดใช้งานจริง (production) โดยเฉพาะ EXSYS ซึ่งมีเครื่องมืออำนวยความสะดวกหลายอย่างอยู่ในระบบเรียบร้อยแล้ว

การแยกส่วนของความรู้ออกจากปัญญาเป็นเทคนิคที่ผู้เชี่ยวชาญคำสั่งงานไม่ได้ใช้เป็นประจำ และอาจต้องการวินัยในการดำเนินการอยู่บ้างแต่ก็เป็นวิธีการที่รับมาปฏิบัติได้ไม่ยาก และความจริงแล้วน่าจะเป็นเทคนิคมาตรฐานที่ใช้ในการเขียนคำสั่งงานโดยทั่วไปได้เพราะน่าจะช่วยให้การบำรุงรักษาชุดคำสั่งงานทำได้ง่ายมีประสิทธิภาพมากขึ้น

การจัดการกับความไม่แน่นอน (fuzzy knowledge) คงเป็นประเด็นที่ต้องพบในการปฏิบัติงานจริง แม้ว่าระบบที่ศึกษาจะไม่มีระบบใดที่ยอมรับความรู้ประเภทนี้แต่ผู้เชี่ยวชาญคิดว่า หากจะทำก็ทำได้ไม่ยากนัก

โดยเฉพาะถ้าสามารถกำหนดความไม่แน่นอนให้เป็นความน่าจะเป็นได้

การเลือกใช้ปัญญาแบบต่าง ๆ นั้นผู้เขียนเห็นว่าปัญหาในขั้นตอนการสร้างที่ลึกซึ้งกว่าปัญหาอื่น ผู้ชำนาญการในเรื่องนี้ยังคงลังเลไม่ได้ว่าระบบผู้เชี่ยวชาญที่คัดสรรจะมีปัญญาที่มีรูปแบบเดียวหรือมีหลายรูปแบบ ซึ่งก็เป็นปัญหาที่เกี่ยวข้องกับรูปแบบของการจัดเก็บความรู้ ภาษาสั่งงานที่ใช้และวิธีการแก้ปัญหาที่ได้กล่าวมาแล้ว วิธีหนึ่งที่จะหาความเข้าใจและความชำนาญในเรื่องนี้อาจจะเป็นการทดลองใช้ปัญญาแบบต่าง ๆ กับระบบผู้เชี่ยวชาญระบบเดียวกัน เพื่อให้เห็นถึงความแตกต่างอย่างแท้จริง และไม่เพิ่มงานในด้านอื่น ๆ อย่างไรก็ดีปัญหานี้เป็นปัญหาในเรื่องประสิทธิภาพของระบบมากกว่าปัญหาเรื่องความเป็นไปได้ของระบบ กล่าวคือไม่ว่าจะเลือกใช้ปัญญาแบบใดก็ควรจะสร้างระบบได้ แต่อาจไม่มีประสิทธิภาพสูงนัก

โดยสรุปแล้วผู้เชี่ยวชาญรู้สึกว่าการหาปัญหาของการหาผู้เชี่ยวชาญที่สามารถให้เวลากับโครงการได้แล้ว การสร้างระบบผู้เชี่ยวชาญระดับต้นแบบหรือระดับวิจัย มิได้เป็นงานที่ยากจนเกินไป ตรงกันข้ามกลับเป็นงานที่น่าสนใจและคงมีผลประโยชน์ไม่น้อยสำหรับผู้สนใจงานด้านนี้ และพร้อมที่จะต่อสู้กับความลำบากในช่วงของการดำเนินงานด้านวิศวกรรมความรู้สำหรับระบบทั้ง ๘ ระบบที่ศึกษากันนับว่าประสบความสำเร็จค่อนข้างดีในระดับระบบต้นแบบ กล่าวคือสามารถทำงานได้ถูกต้องหลายกรณี (แม้จะมีความแปรปรวนสูง) มีลักษณะของระบบผู้เชี่ยวชาญค่อนข้างสมบูรณ์ มีการติดต่อกับผู้ใช้ในระดับที่พอรับได้ และบางระบบก็มีประโยชน์ใช้สอยจริงอยู่บ้าง

๑๐๓๑๑๑๑๑๑

- Alty, J.L. & M.J. Coombs (1984) "Expert Systems—Concepts and Examples," NCC Publications.
- Bobrow, G.D., S. Mittal & M.J. Stefik (1986) "Expert Systems : Perils and Promise," CACM Vol. 29, 880–894.
- Borland International (1986) "Turbo Prolog Owner's Handbook," Borland International, Inc.
- Buchanan, B.G. & E. Shortliffe (1984) "Rule-Based Expert Systems," Addison-Wesley.
- Buchanan, B.G., G.L. Sutherland & E.A. Feigenbaum (1969) "Heuristic DENDRAL : a program for generating explanatory hypotheses in organic chemistry," in B. Weltzer & D. Michie (eds), Machine Intelligence, 4, Edinburgh University Press.
- Davis, R. (1986) "Knowledge based systems," Science V231, 4741 (February).
- EXSYS Inc. (1984) EXSYS-Expert System Development Package, P.O. Box 75158, Albuquerque, NM 87194, USA.
- Fikes, R. & T. Kehler (1985) "The role of frame-based representation in reasoning," CACM Vol. 28, No. 9, 904–920.
- Forgy, C.L. (1981) "OPS 5 User's Manual," report CMU-CS-81-135, Dept. of Computer Science, Carnegie-Mellon University.
- Gaschnig, J. (1981) "PROSPECTOR : An expert system for mineral exploration," Machine Intelligence, Infotech State of the Art Report, series 9, No. 3.
- Hirsch, P., M. Meier, S. Snyder & R. Stillman (1985) "PRISM : Prototype inference System," AFIPS Conference Proceeding 54, 121–124.
- Lindsay, R.K., B.G. Buchanan, E.A. Feigenbaum & J. Lederberg (1980) "Applications of Artificial Intelligence in Organic Chemistry : The DENDRAL Project," McGraw-Hill.
- Shammas, N.C. (1986) "Software Review-Turbo Prolog," BYTE, September, 293–295.
- Shortliffe, E.H. (1976) "Computer-Based Medical Consultations : MYCIN.," Elsevier.
- Stefik, M., D.G. Bobrow, S. Mittal & L. Conway (1983) "Knowledge programming in Loops : Report on an experimental course," The Artificial Intelligence Magazine, Fall issue, 3–13.
- Walker, A. (ed.), M. McCord, J.F. Sowa & W.G. Wilson, (1987), "Knowledge Systems and Prolog," Addison-Wesley.
- Waterman, D.A. (1986) "A Guide to Expert Systems," Addison-Wesley.
- Webster, B. (1986) "Two fine Products," BYTE, September, 335–340.