

## บทที่ 2

### แนวคิดและงานวิจัยที่เกี่ยวข้อง

ในการพัฒนาระบบผู้เชี่ยวชาญสำหรับเครื่องอัตโนมัติ มีผู้วิจัยได้ศึกษาแนวคิดและงานค้นคว้าวิจัยที่เกี่ยวกับระบบผู้เชี่ยวชาญที่นำมาใช้และการทำงานรวมทั้งปัญหาของเครื่องจักร เพื่อการออกแบบระบบให้สอดคล้องกับขั้นตอนการทำงานและผู้วิจัยได้ผสมผสานแนวความคิดและข้อมูลที่เกี่ยวข้องมีรายละเอียดตามลำดับดังนี้

#### 2.1 ส่วนประกอบของระบบผู้เชี่ยวชาญ

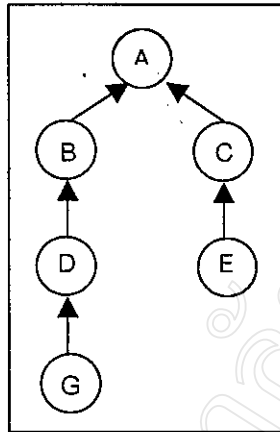
Efraim Turban and Jay E. Aronson (1998) ได้กล่าวถึงส่วนประกอบของระบบผู้เชี่ยวชาญเป็น 5 หัวข้อดังนี้

1. ฐานความรู้ (Knowledge base) หรือแหล่งความรู้จากที่ต่าง ๆ ที่นำเข้าไปในโปรแกรมคอมพิวเตอร์เพื่อจัดเป็นหมวดหมู่ สำหรับแหล่งข้อมูลมีได้หลายแหล่งและการนำความรู้จากผู้เชี่ยวชาญจัดว่าเป็นงานที่ซับซ้อน ดังนั้นในปัจจุบันจึงมีการนำวิศวกรเข้ามาช่วยจัดเรียงความรู้จากผู้เชี่ยวชาญกันใหม่ เช่น การนำคำตอบของผู้เชี่ยวชาญมาเขียนเป็นลำดับขั้นการทำงาน เพื่อการวิเคราะห์ก่อนที่จะนำไปบันทึกลงในคอมพิวเตอร์ต่อไป สำหรับแหล่งข้อมูล ได้แก่
  - 1.1 มนุษย์
  - 1.2 มัลติมีเดีย
  - 1.3 คู่มือ
  - 1.4 ฐานข้อมูล
  - 1.5 การค้นคว้าพิเศษ
  - 1.6 เว็บไซต์
2. สมอของระบบผู้เชี่ยวชาญ (Inference Engine) ทำหน้าที่ควบคุมโครงสร้างของปัญหาและแปลความหมาย ในส่วนนี้นับเป็นปัจจัยพื้นฐานของโปรแกรมคอมพิวเตอร์ที่จะเลือกวิธีการตัดสินใจกับข้อมูลที่ได้รับอย่างมีเหตุผลและเป็นลำดับขั้นตอน เพื่อคำนวณหาข้อสรุปเป็นคำตอบต่อไป วิธีที่นิยมใช้อย่างกว้างขวางแบ่งออกเป็น 2 ประเภทคือ

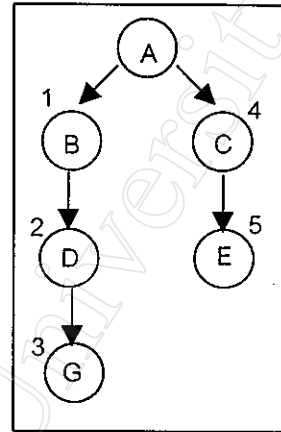
2.1 การวินิจฉัยแบบถอยหลัง (Backward Chaining Inference) วิธีการนี้ไม่ต้องกำหนดเป้าหมายก่อนแต่จะหาเหตุผลมาสนับสนุนเพื่อให้ได้ผลลัพธ์ที่เป็นจริง เช่น การวินิจฉัยโรคของแพทย์ แพทย์จะสอบถามถึงอาการของผู้ป่วยหลาย ๆ อย่างประกอบกัน ก่อนที่จะสรุป

ว่าผู้ป่วยป่วยเป็นโรคอะไร การอนุมานแบบนี้มีชื่อเรียกอีกอย่างว่า Data driven inference หรือ bottom-up inference ดังรูป 2.1

2.2 การวินิจฉัยแบบเดินหน้า (Forward Chaining Inference) วิธีนี้จะกำหนดเป้าหมายซึ่งต้องกำหนดมาก่อน แล้วจึงหาวิธีวินิจฉัยไปสู่เหตุผลที่สนับสนุนให้เป้าหมายที่ต้องการ เช่น ธนาคารต้องสอบประวัติการเงินของลูกค้าก่อนที่จะให้มีการกู้ยืม การอนุมานแบบนี้เรียกอีกอย่างหนึ่งว่า Goal driven inference หรือ bottom-down inference ดังรูป 2.2



รูป 2.1 การวินิจฉัยแบบถอยหลัง  
(วิธีธิ อังการณ,2531:154)



รูป 2.2 การวินิจฉัยแบบเดินหน้า  
(วิธีธิ อังการณ,2531:155)

3. ส่วนดึงข้อมูล (Knowledge acquisition Module) เป็นการดึงข้อมูลจากตำราหรือผู้เชี่ยวชาญเพื่อเพิ่มเติมให้กับฐานข้อมูล นอกจากนี้ยังทำหน้าที่
  - 3.1 เลือกหัวข้อของปัญหามาตรวจสอบให้ตรงกับกฎเกณฑ์ของข้อมูล (Interpreter)
  - 3.2 ควบคุมลำดับความสำคัญของปัญหา (Scheduler)
- 4 ส่วนอธิบายเหตุผล (Explanation Module) ทำหน้าที่อธิบายรายละเอียดของขั้นตอนการวินิจฉัยต่อผู้ใช้งานข้อสรุปหรือคำตอบนั้นได้มาอย่างไร
- 5 ส่วนติดต่อกับผู้ใช้ (User interface) เนื่องจากระบบผู้เชี่ยวชาญเป็นการประมวลผลทางภาษา ดังนั้นผลลัพธ์ที่แจ้งให้ผู้ใช้จึงต้องเข้าใจง่าย หรือแสดงเป็นกราฟได้

วัตถุประสงค์ของการใช้ระบบผู้เชี่ยวชาญ

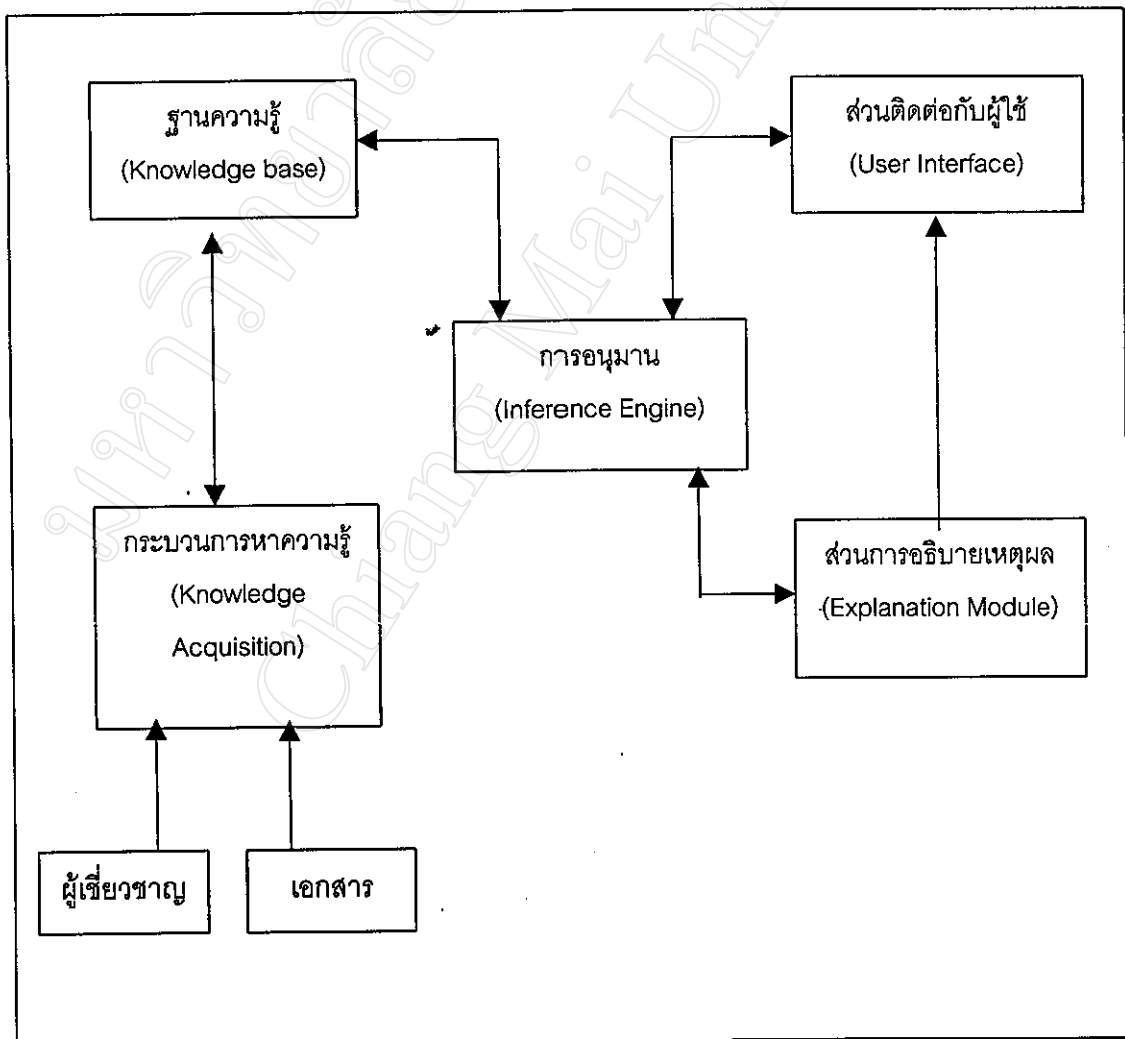
ระบบผู้เชี่ยวชาญที่ใช้กันในวงการต่าง ๆ ผู้ใช้มีวัตถุประสงค์ดังนี้

1. ใช้ระบบผู้เชี่ยวชาญเป็นที่ปรึกษา
2. ใช้ระบบผู้เชี่ยวชาญเป็นผู้สอน
3. ใช้ระบบผู้เชี่ยวชาญเพื่อเพิ่มความรู้ในสิ่งที่ตนยังไม่ทราบ
4. ใช้ระบบผู้เชี่ยวชาญเป็นผู้ช่วยในการตัดสินใจเกี่ยวกับปัญหา

ปัจจัยที่จะทำให้ระบบผู้เชี่ยวชาญประสบผลสำเร็จ

ระบบผู้เชี่ยวชาญที่สร้างขึ้นจะประสบผลสำเร็จหรือล้มเหลวขึ้นอยู่กับปัจจัยดังนี้

1. องค์กรความรู้ที่นำมาใช้ต้องมีอย่างเพียงพอ
2. ขอบเขตของปัญหาที่จะแก้ไขต้องไม่กว้างเกินไป
3. ปัญหาต้องมีความเหมาะสมต่อการที่จะสร้างระบบผู้เชี่ยวชาญขึ้นมาใช้งานคือ ไม่ยากหรือง่ายเกินไป
4. ส่วนเพิ่มเติมข้อมูล (Expert shell) ต้องมีความยืดหยุ่นต่อการใช้งาน และเก็บความรู้ได้หลากหลาย
5. ผู้พัฒนาระบบต้องเข้าใจองค์ความรู้ของระบบและเห็นความสำคัญของระบบ
6. ระบบแสดงผล ต้องง่ายต่อการเข้าใจของผู้ใช้งาน
7. ผู้บริหารระดับสูงต้องให้การสนับสนุน

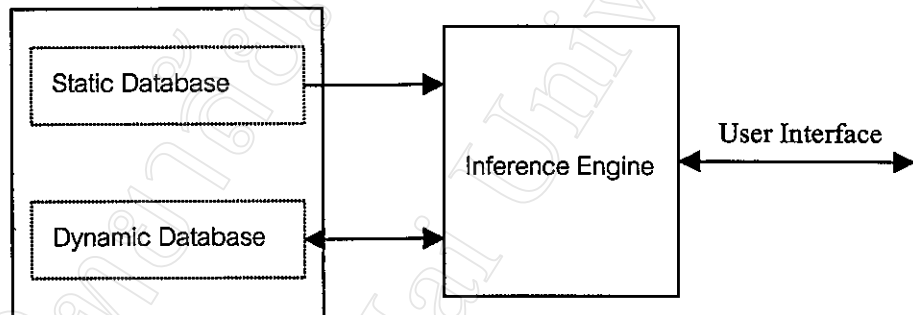


รูป 2.3 ส่วนประกอบของระบบผู้เชี่ยวชาญ(EfraimTurban And Jay E. Aronson,1998:439)

## 2.2 ฐานความรู้ในระบบผู้เชี่ยวชาญ

วริทธิ์ อิงภากรณ์ (2531) กล่าวเพิ่มเติมว่าระบบผู้เชี่ยวชาญจะมีลักษณะดังรูป 2.4 ซึ่งฐานความรู้จะประกอบด้วยข้อเท็จจริงและกฎซึ่งแบ่งเป็นฐานข้อมูลที่เรียกว่า ฐานข้อมูลคงที่(Static Database) และฐานข้อมูลที่เปลี่ยนแปลง (Dynamic Database) สำหรับฐานข้อมูลคงที่นั้นเป็นฐานความรู้ที่ไม่สามารถเปลี่ยนแปลงในขณะที่ใช้ระบบผู้เชี่ยวชาญ ได้แก่ความรู้ที่ผู้สร้างระบบผู้เชี่ยวชาญใส่ไว้เป็นฐานความรู้ สำหรับการตอบคำถาม

ส่วนฐานข้อมูลที่เปลี่ยนแปลงจะเป็นส่วนของฐานความรู้ที่เปลี่ยนแปลงในขณะที่ใช้งานซึ่งถ้าพิจารณาแล้วฐานข้อมูลที่เปลี่ยนแปลงก็คือส่วนที่ใช้สำหรับเก็บข้อมูล YES และ NO ที่ใช้ในระหว่างการสอบถามผู้ใช้ เพื่อที่ระบบจะไม่ต้องถามคำถามเดิมซ้ำอีก



รูป 2.4 ฐานความรู้ในระบบผู้เชี่ยวชาญ(วริทธิ์ อิงภากรณ์,2531:172)

ฐานความรู้นี้จะแปรผันตรงกับกลไกการวินิจฉัยกล่าวคือถ้าฐานความรู้มีจำนวนมาก กลไกการวินิจฉัยก็จะมีมากเช่นกัน และมีการสร้างฐานข้อมูลสำหรับแต่ละกรณีใหม่เสมอ ในการสร้างโปรแกรมและใช้งานระบบผู้เชี่ยวชาญนั้น ส่วนของกลไกวินิจฉัย และส่วนที่อยู่ติดต่อกับผู้ใช้ มักอยู่ร่วมกันจนไม่อาจแยกทั้งสองส่วนออกจากกันได้เด็ดขาด ดังนั้นจึงมีการเรียกชื่อทั้งสองส่วนนี้รวมกันว่า Expert System Shell หรือ Shell อย่างเดียว

ในระหว่างการเขียนโปรแกรม ข้อเท็จจริงที่ได้จากผู้ใช้ในระหว่างการให้คำปรึกษาจะเก็บไว้ในฐานข้อมูลที่เปลี่ยนแปลง ข้อเท็จจริงจะใช้กับเฉพาะกรณีเท่านั้น ดังนั้นเมื่อสิ้นสุดการปรึกษาแต่ละครั้งข้อมูลที่เปลี่ยนแปลงจะถูกลบออกทั้งหมด เพื่อพร้อมที่จะใช้สำหรับการปรึกษาของผู้ใช้คนต่อไป ในการที่จะเก็บข้อมูลเราต้องสร้างฐานข้อมูลแบบทำนาย(Database Predicate) แล้วจึงนำข้อเท็จจริงไปเก็บไว้ในฐานข้อมูลแบบทำนายเหล่านี้ ในระหว่างการทำงานของโปรแกรม โดยอาศัยแบบทำนายของโปรแกรม 2 ประเภทคือ

`asserta(fact)`: ใช้สำหรับเก็บข้อเท็จจริง โดยเรียงเรียงข้อเท็จจริงครั้งหลังสุดไว้ก่อนข้อเท็จจริง

อื่น ๆ ซึ่งเปรียบเทียบได้กับการวางซ้อนกันของหนังสือ นั่นคือข้อเท็จจริงหลังสุดจะอยู่บนสุด ฉะนั้นเวลาตรวจสอบคู่มือโปรล็อกจะพบข้อมูลนี้ก่อนข้อมูลอื่น ๆ

assertz(fact): ใช้สำหรับเก็บข้อเท็จจริงโดยเรียงเรียงข้อมูลครั้งหลังสุดเอาไว้ส่วนล่างสุดของกองหนังสือ ฉะนั้นเวลาตรวจสอบข้อมูลโปรล็อกจะพบข้อมูลนี้หลังสุดเช่นกัน

ในการสร้างฐานข้อมูลแบบทำนายจะต้องเพิ่มส่วนที่เป็นฐานข้อมูลลงในโปรแกรมเสียก่อน พร้อมกับระบบเพรดิคทที่เกี่ยวข้องลงไปด้วย ส่วนฐานของข้อมูลต้องอยู่ในโปรแกรม และอยู่ก่อนส่วนที่เป็นเพรดิคท

### 2.3 ประเภทของระบบผู้เชี่ยวชาญ

Efraim Turban and Jay E. Aronson (1998) ได้กล่าวถึงประเภทของระบบผู้เชี่ยวชาญว่าแบ่งออกได้หลายประเภทแต่ในที่นี้จะให้รายละเอียดเพียง 2 ประเภท

1. Rule based Expert System เป็นระบบที่ปัญหาจะได้รับการแก้ไขแบบ IF..... THEN ซึ่งมีข้อดีคือ

1. กฎเป็นสิ่งง่ายต่อการเข้าใจและการเขียน โปรแกรม
2. จากกฎง่าย ๆ สามารถนำมารวมกันเพื่อสร้างเป็นกฎที่มีความสลับซับซ้อนได้เพื่อแก้ปัญหาที่ยาก ๆ ได้
3. กฎแต่ละกฎเป็นอิสระต่อกัน ไม่มีความจำเป็นต้องเกี่ยวเนื่องกันก็ได้
4. กฎสามารถนำไปเขียนได้ทุกส่วนของโปรแกรม

ตัวอย่างการแก้ปัญหาแบบ Rule based System

IF; ENGINE NOT START	←.....	เงื่อนไข
THEN ; CHECK FUEL	←.....	การแก้ปัญหา 1
ELSE; CHECK BATTERY	←.....	การแก้ปัญหา 2

2. CBR (Case based Reasoning) เป็นการแก้ปัญหาโดยอาศัยความรู้เดิมที่มีอยู่ จากนั้นจึงแปลเป็นคำตอบของการแก้ปัญหาเช่น

ENGINE NOT START	←.....	ปัญหา
CHECK FUEL	←.....	ความรู้เดิม 1
CHECK BATTERY	←.....	ความรู้เดิม 2
CHECK CABLE	←.....	ความรู้เดิม 3

ถ้าผลการตรวจสอบพบว่า ความรู้เดิมที่ 3 เป็นจริง เครื่องอาจจะแปลคำตอบมาเป็นสายไฟขาด

ต้องแก้ไขโดยการเปลี่ยนใหม่ สำหรับงานตรวจสอบเครื่องอัตรูปเหมาะสมกับการใช้ระบบผู้เชี่ยวชาญประเภท Case based Reasoning เพราะ ต้องอาศัยองค์ความรู้เดิมมาช่วยตัดสินใจ

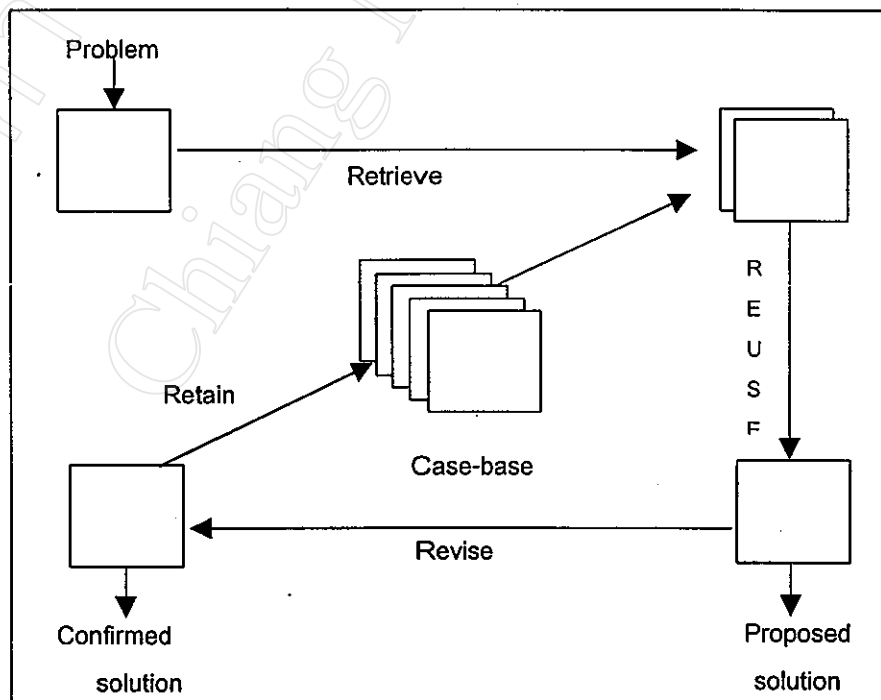
Riesbeck and Schank (1989) ได้ให้คำจำกัดความของ CBR ว่า “คือการแก้ปัญหาใหม่โดยอาศัยการแก้ปัญหาเดิมที่เคยเกิดขึ้นในอดีต”

## 2.4 ขั้นตอนการทำงานของ CBR

Ian Watson (1997) ได้แสดงการทำงานของ CBR ไว้ 4 ขั้นตอน(four REs) ดังต่อไปนี้

1. ค้นหาความรู้เดิมที่ใกล้เคียงปัญหาใหม่(REtrieve )
2. ใช้ฐานความรู้เดิมที่มีอยู่(REuse)
3. หาวิธีแก้ปัญหาใหม่(REvise)
4. เก็บวิธีแก้ปัญหาใหม่เข้าในฐานความรู้(REtain)

ถ้าปัญหาใหม่ที่ป้อนเข้าไปในระบบมีวิธีแก้ปัญหาที่ใกล้เคียงกับที่มีอยู่ในฐานความรู้เดิมก็จะใช้ฐานความรู้ที่มีอยู่ที่เรียกว่า “REtrieve” หากมีวิธีแก้ปัญหาที่ตรงกัน ก็จะนำมารายงานผลให้ทราบเรียกว่า “REuse” แต่หากปัญหาใหม่ไม่มีในระบบและไม่ใกล้เคียงกับฐานข้อมูล ระบบก็จะสร้างวิธีแก้ปัญหาใหม่เรียกว่า “REvise” และในที่สุดเมื่อสร้างวิธีแก้ปัญหาใหม่แล้วระบบก็จะบันทึกไว้ในฐานความรู้ต่อไปเรียกว่า “REtain” ดังรูป 2.5

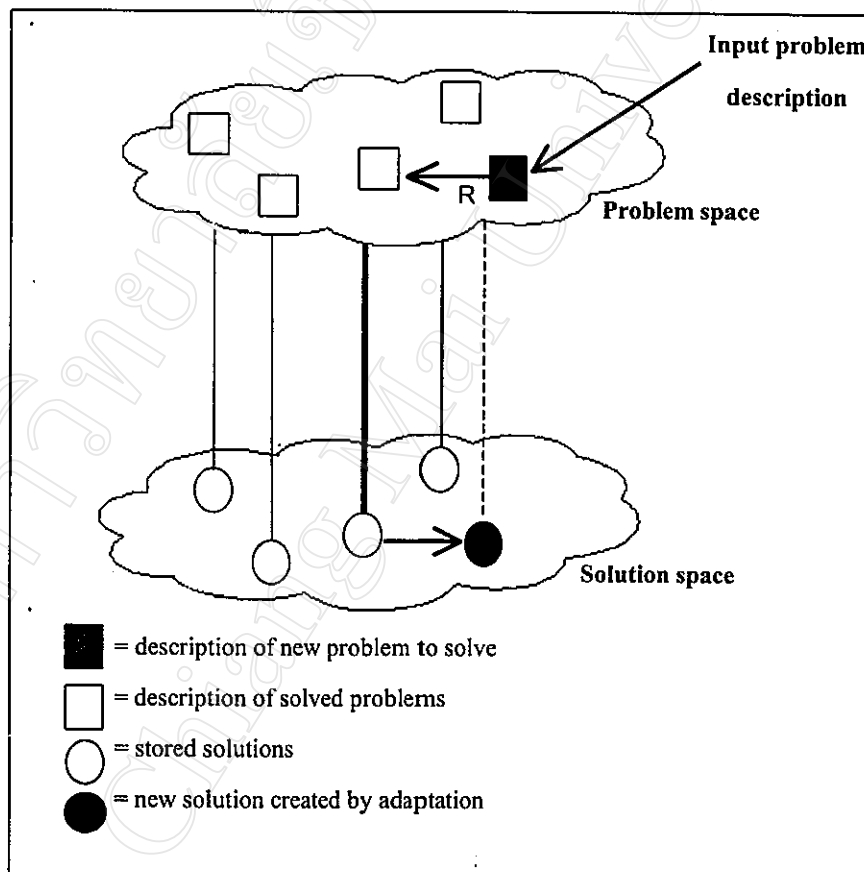


รูป 2.5 ขั้นตอนการทำงานของ CBR(Ian Watson,1997:198)

## 2.5 การหาเป้าหมายของ CBR

Case คือส่วนประกอบของคำอธิบายหรือเหตุผล ที่เป็นส่วนเล็ก ๆ หลาย ๆ ส่วน ของระบบฐานข้อมูลเพื่อให้ได้ซึ่งวิธีการแก้ปัญหา เราสามารถเห็นปัญหาเฉพาะของแต่ละ Case ซึ่งประกอบด้วย 2 ส่วนคือ

- 1 คำอธิบายหรือคำจำกัดความของปัญหา (Problem space) อาจจะเป็นทั้งปัญหาเก่าและปัญหาใหม่
- 3 ส่วนที่เก็บคำอธิบายของปัญหา (Solution space) ที่เรียกว่าข้อมูล(Data) ข้อมูลที่เก็บมีหลายประเภท เช่น ฐานข้อมูล รูปภาพ และเสียง



รูป 2.6 การหาคำตอบของCBR(Ian Watson,1997:20)

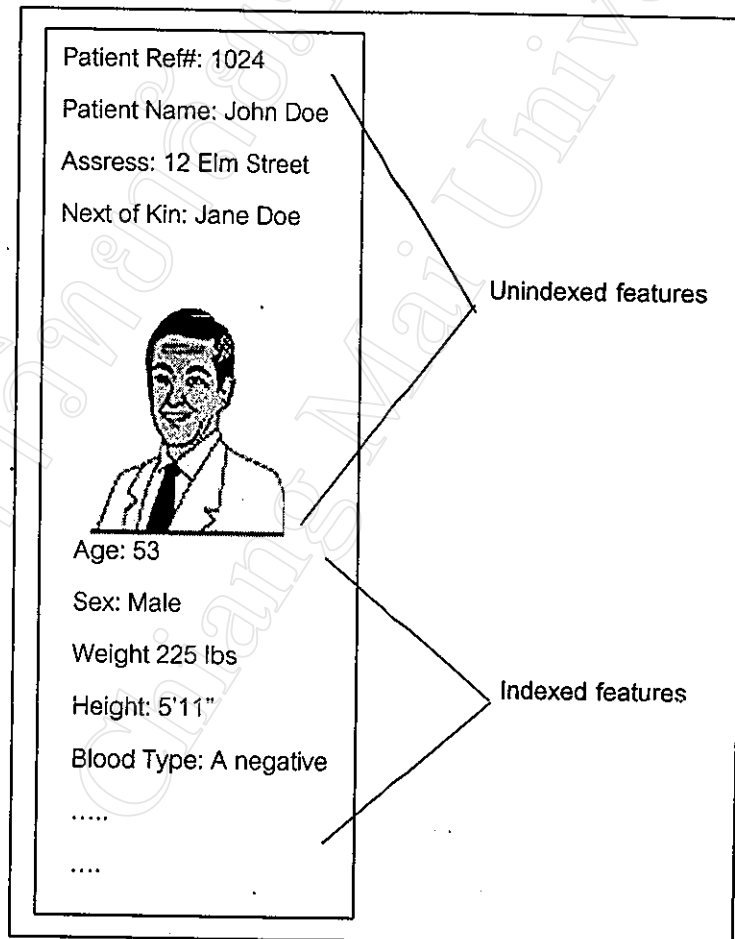
### การค้นหาข้อมูลแบบมีสารบัญ (Indexing)

ระบบฐานข้อมูลส่วนใหญ่จะใช้สารบัญเป็นตัวชี้ถึงข้อมูลแล้วดึงข้อมูลออกมาเพื่อนำเสนอ โดยสารบัญจะคำนวณโครงสร้างของข้อมูลและหาข้อมูลที่ต้องการซึ่งทำได้รวดเร็ว สำหรับ CBR ก็ใช้สารบัญทำหน้าที่ค้นหาข้อมูลเช่นกัน การหาข้อมูลแบบสารบัญแบ่งเป็น 2 ประเภทคือ

1. ข้อมูลแบบสารบัญ(Indexed Information) ใช้สำหรับการดึงข้อมูล
2. ข้อมูลแบบไม่มีสารบัญ(Unindexed Information) เป็นข้อมูลประกอบ ภายหลังจากดึงข้อมูลแบบมีสารบัญ

จากตัวอย่างระบบการรักษาของแพทย์ แพทย์ต้องมีข้อมูลของคนไข้ เช่น อายุ เพศ น้ำหนัก ส่วนสูง ซึ่งเป็นข้อมูลที่จำเป็นที่จะต้องนำเสนอแสดงตัวตน และอาจมีรูปภาพของคนไข้ประกอบด้วย แต่รูปภาพไม่สามารถทำเป็นสารบัญ เพื่อดึงออกมาแล้วทราบว่าคนไข้เป็นใคร ชื่ออะไร ข้อมูลแบบสารบัญควรมีคุณสมบัติดังนี้

1. สามารถทำนาย หรือคาดคะเนได้
2. สามารถใช้ประโยชน์
3. มีความละเอียดพอที่จะขยายไปสู่ข้อมูลอื่น ๆ



รูป 2.7 ข้อมูลแบบมีสารบัญและแบบไม่มีสารบัญ(Ian Watson,1997:21)

CBR ที่ใช้อยู่ส่วนมากที่มีใช้อยู่ในด้านธุรกิจ จะสามารถกำหนดสารบัญ ได้ทั้งแบบกำหนดอัตโนมัติ(Automatic) และแบบกำหนดด้วยตนเอง(Manually) หรือมีทั้งสองแบบในซอฟต์แวร์ CBR



## 2.6 การนำเสนอข้อมูลจากฐานความรู้

Ian Watson (1997) กล่าวถึงการนำเสนอความรู้หรือการอธิบายเพื่อแก้ไขปัญหาในแต่ละกรณี นั้นขึ้นอยู่กับวิธีค้นหาข้อมูลแบบสารบัญ โดยทั่วไปในระบบ CBR แบ่งวิธีการนำเสนอข้อมูลจากฐาน ข้อมูลเป็น ได้ 2 วิธีคือ

1. การนำเสนอแบบใช้ข้อมูลใกล้เคียง (nearest neighbor Retrieval)
2. การนำเสนอแบบกรณีเฉพาะราย (Inductive Retrieval)

วิธีการนำเสนอข้อมูลจากฐานความรู้ทั้ง 2 วิธีมีรายละเอียดดังนี้

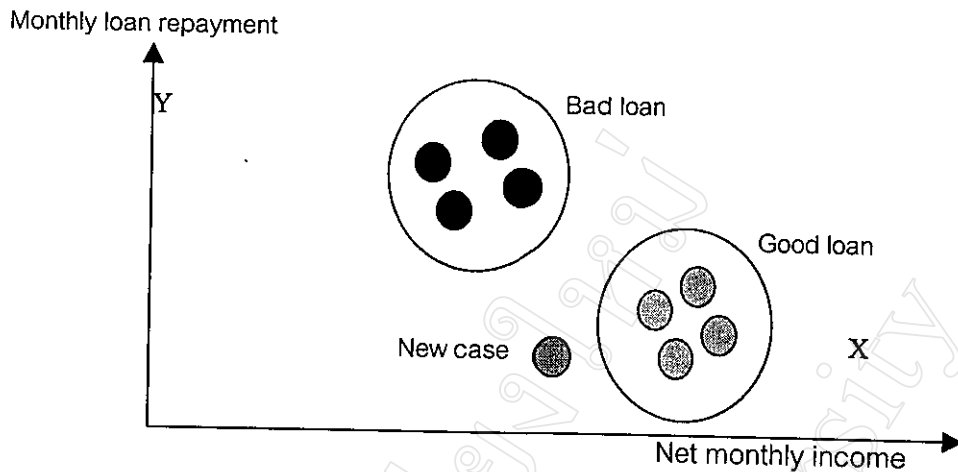
1. การนำเสนอแบบใช้ข้อมูลใกล้เคียง เป็นวิธีที่ง่ายต่อการทำความเข้าใจ ดังตัวอย่างของการ กู้เงินธนาคารดังนี้
  - 1.1 ในการกู้เงินกับธนาคาร สิ่งที่ธนาคารจะพิจารณาเป็นลำดับแรก คือ ราย ได้ของผู้ขอกู้ เช่น ได้รับเงินเดือนเท่าไร หรือมีรายได้จากธุรกิจอะไร บ้าง
  - 1.2 ผู้ขอกู้มี รายจ่ายอะไรบ้าง เช่น ต้องผ่อนบ้านทุกเดือน หรือ จ่ายให้กับ แหล่งกู้เงินจากแหล่งอื่น ๆ

จากนั้นธนาคารจะนำข้อมูลจากข้อพิจารณาทั้ง 2 ข้อมาทำ Case indexes และ Case result ของ การพิจารณานี้คือ สถานะของลูกค้ำหรือไม่

ตาราง 2.1 แสดงกรณีตัวอย่างทั่ว ๆ ไป

<b>Case indexes</b>
Net monthly income in dollars
Monthly loan repayment in dollars
<b>Case result</b>
Good or bad loan

ลำดับต่อมาให้นำข้อมูลที่ได้อามาแปลงเป็นกราฟ เพื่อเป็นเครื่องมือประกอบการตัดสินใจ โดย พิจารณาที่แกน X ซึ่งเป็นรายได้รวมตลอดเดือน(Net monthly income) และ Y ซึ่งเป็นรายจ่ายรวม ตลอดเดือน(Monthly loan repayment) ถ้าแกน X มากกว่า แกน Y ให้ถือว่าเป็นลูกค้ำที่ดีสามารถให้ กู้เงินได้ แต่ถ้าแกน Y มากกว่าแกน X จัดว่าเป็นลูกค้ำที่ไม่ดี ธนาคารมีความเสี่ยงต่อการที่จะให้กู้เงิน ดัง รูป 2.8



รูป 2.8 การหาคำตอบแบบ Nearest Neighbor Retrieval(Ian Watson,1997:25)

จากรูป 2.8 จะเห็นได้ว่า new case อยู่ใกล้กับ Good loan แต่เพื่อความถูกต้องสามารถทำการคำนวณเพื่อหาตำแหน่งที่แท้จริง ของ new case ได้ ด้วยการหาความสัมพันธ์ระหว่างระยะทางของแกน X และ Y เพื่อเทียบกับ case อื่น (source case) เป็นการหาระยะทางที่น้อยที่สุด โดยกำหนดให้ case A คือ Good loan , case B คือ Bad loan และ case T คือ new case

จะได้ระยะทางจาก T ถึง A เท่ากับ  $D_A = X_A + Y_A$

ระยะทางจาก T ถึง B เท่ากับ  $D_B = X_B + Y_B$

เมื่อ  $D_A$  คือระยะทางทั้งหมดในแนวแกน X จาก case T ไปยัง case A

เมื่อ  $D_B$  คือระยะทางทั้งหมดในแนวแกน Y จาก case T ไปยัง case B

และ  $X_A$  คือระยะทางในแนวแกน X จาก case T ไปยัง case A

$Y_A$  คือระยะทางในแนวแกน Y จาก case T ไปยัง case A

และ  $X_B$  คือระยะทางในแนวแกน X จาก case T ไปยัง case B

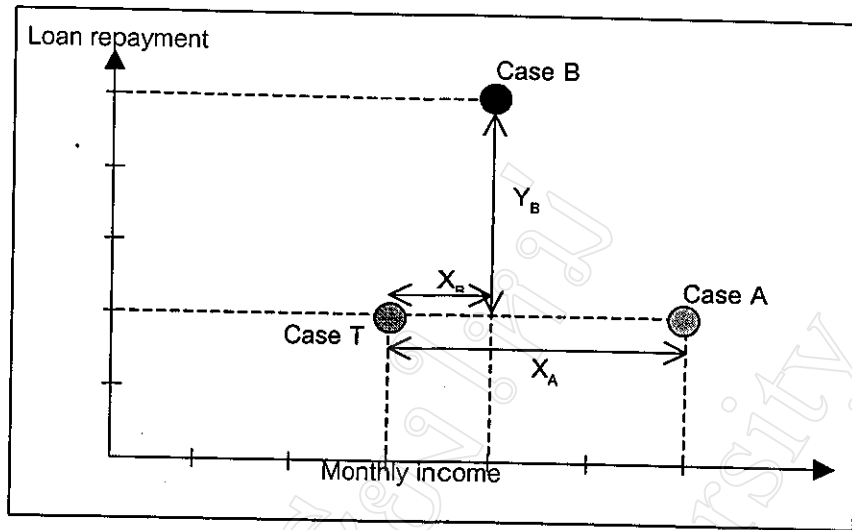
$Y_B$  คือระยะทางในแนวแกน Y จาก case T ไปยัง case B

แทนค่าในสมการจะได้  $D_A = 3 + 0$  ,  $D_B = 1 + 3$

ดังนั้น ค่าของ  $D_A = 3$

และ  $D_B = 4$

สรุปได้ว่า case T เมื่อเทียบระยะห่างระหว่าง  $A=3$  และ  $B=4$  แล้ว case T อยู่ใกล้ A มากกว่า B ดังนั้น จึงยังอยู่ในกลุ่ม เข้าสู่ Good loan เพราะ case index ใน X (Monthly income) และ Y (Loan-repayment) ต่ำ ดังรูป 2.9



รูป 2.9 การหาค่าตอบแบบการให้น้ำหนัก(Ian Watson,1997:26)

การหาตำแหน่ง case T อีกวิธีหนึ่ง คือ การให้น้ำหนักกับ case index ว่า case index ใดมีความสำคัญที่สุด ซึ่งในตัวอย่างให้ case index ที่มาอันดับหนึ่ง (Monthly income) มีค่าเท่ากับ 2 และที่มาอันดับสอง(Loan repayment) มีค่าเท่ากับ 1

จะได้ระยะทางจาก T ถึง A เท่ากับ  $D_A = (X_A \times W_X) + (Y_A \times W_Y)$

ระยะทางจาก T ถึง B เท่ากับ  $D_B = (X_B \times W_X) + (Y_B \times W_Y)$

และ  $X_A$  คือระยะทางในแนวแกน X จาก case T ไปยัง case A

$Y_A$  คือระยะทางในแนวแกน Y จาก case T ไปยัง case A

และ  $X_B$  คือระยะทางในแนวแกน X จาก case T ไปยัง case B

$Y_B$  คือระยะทางในแนวแกน Y จาก case T ไปยัง case B

และ  $W_X$  คือค่าน้ำหนักในแนวแกน X

$W_Y$  คือค่าน้ำหนักในแนวแกน Y

แทนค่าในสมการจะได้  $D_A = (3 \times 2) + (0 \times 1)$

$$D_B = (1 \times 2) + (3 \times 1)$$

สรุปได้ว่าการหาค่า case T โดยการให้ค่าน้ำหนักหากค่าใดมีน้ำหนักมากก็ใกล้เคียง case index นั้น ซึ่งให้ผลเช่นเดียวกับการหาระยะทางของ A และ B

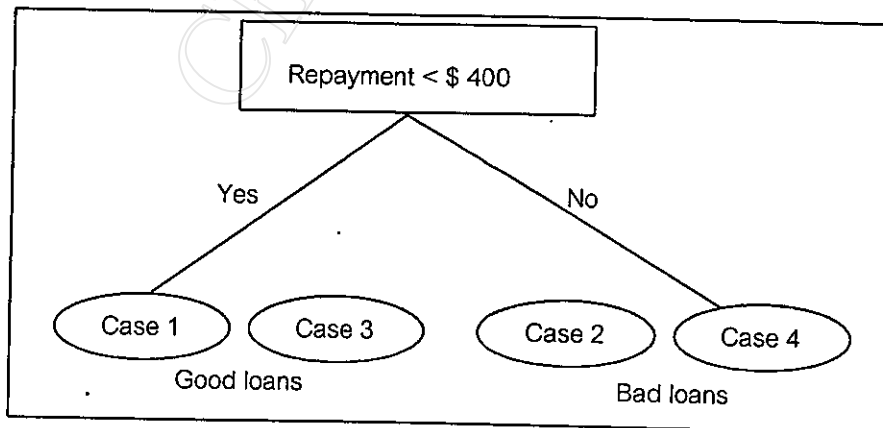
2. การนำเสนอแบบกรณีเฉพาะราย เป็นวิธีที่ CBR จำนวนมากนำไปใช้งาน ซึ่งมีการพัฒนาจากระบบ การเรียนรู้โดยใช้ตารางช่วยตัดสินใจ

ผู้ที่ทำตารางต้องมีประสบการณ์การพิจารณางานที่ทำ โดยจะใช้ตัวอย่างการกู้เงินจากธนาคารอีกครั้ง กล่าวคือตารางจะแบ่งแยกให้เห็นชัดเจนว่า Good, Very good, Bad และ Very bad ถ้าลูกค้ามีเงินเดือนหรือรายได้เท่าไร หรือมีรายจ่ายการกู้ยืมเท่าไรแล้วจัดว่าลูกค้าคนนั้นจะอยู่ในเกณฑ์ใด

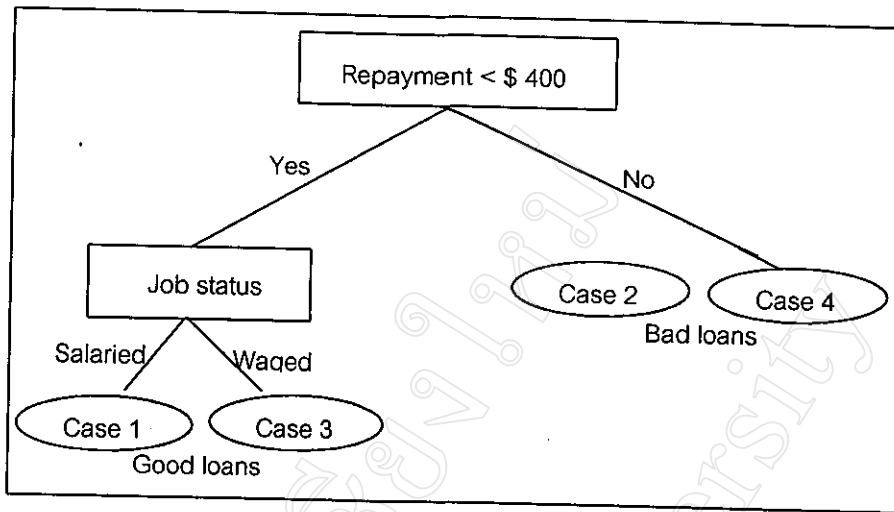
ตาราง 2.2 ตัวอย่างการพิจารณาเงินกู้

	Loan Status	Monthly income	Job status	Repayment
Case1	good	\$2,000	salaried	\$200
Case2	very bad	\$4,000	salaried	\$600
Case3	very good	\$3,000	waged	\$300
Case4	bad	\$1,500	salaried	\$400

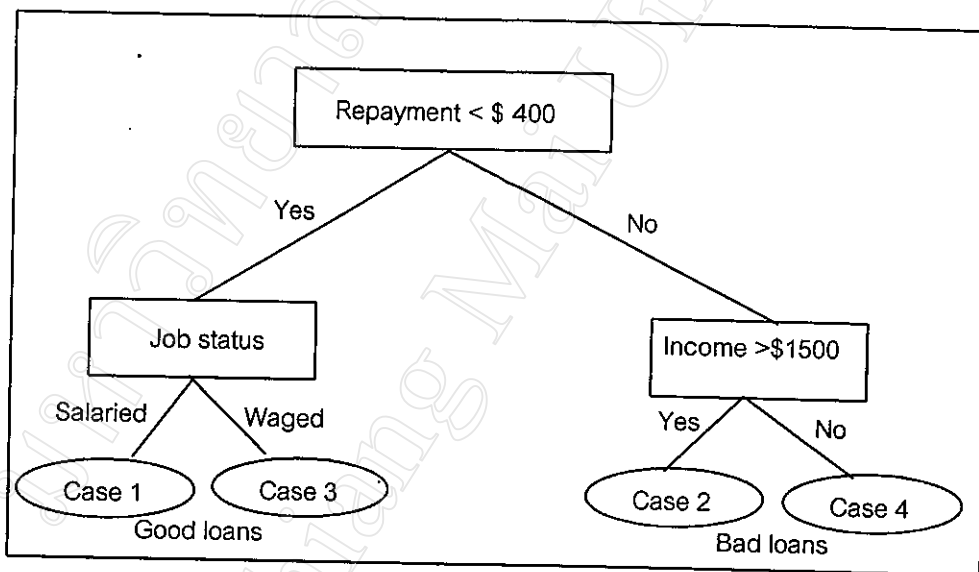
จากตาราง 2.2 ไม่สามารถนำรายได้รายเดือน(Monthly income) มาทำนายการพิจารณาให้กู้ เพราะถึงแม้มีรายได้มากที่สุด 4,000 ดอลลาร์ และต่ำสุด 1,500 ดอลลาร์ จะได้ผลลัพธ์ที่เป็น Bad และ Very bad เท่านั้น เช่นเดียวกันไม่สามารถใช้สถานะการทำงาน(Job status) และการชำระหนี้(Repayment) มาทำนายได้ เพราะถึงแม้การชำระหนี้จะสูงถึง 600 ดอลลาร์แต่ผลลัพธ์ยังเป็น very bad ดังนั้น จึงนำเงื่อนไขของ การชำระหนี้มาเขียนเป็นแผนผังการตัดสินใจขั้นแรกดังรูป 2.10 เพื่อแยกหาเงื่อนไข(Case) ที่เป็น Good Loans และ Bad Loan ขั้นที่สองเขียนแผนผังการตัดสินใจในแบบ Good Loans ดังรูป 2.11 ซึ่งได้ เงื่อนไขที่ 1 และ เงื่อนไขที่ 3 โดยเงื่อนไขของการพิจารณาอยู่ที่สถานะการทำงาน ขั้นตอนที่สาม นำการชำระหนี้ที่มากกว่า 4,000 ดอลลาร์มาพิจารณาจะได้เงื่อนไขที่ 2 และเงื่อนไขที่ 4 ซึ่งมีสถานะการกู้ยืม(Loan status) เป็น bad Loans ได้แผนผังการตัดสินใจที่สมบูรณ์ดังรูป 2.12



รูป 2.10 แผนผังต้นไม้แสดงการตัดสินใจขั้นที่ 1(Ian Watson,1997:30)



รูป 2.11 แผนผังต้นไม้แสดงการตัดสินใจขั้นที่ 2 (Ian Watson, 1997:30)



รูป 2.12 แผนผังต้นไม้แสดงการตัดสินใจขั้นสมบูรณ์ (Ian Watson, 1997:31)

จากแผนผังการตัดสินใจสามารถนำมาพิจารณาเงื่อนไขการตัดสินใจได้ เช่น จากข้อมูลในตาราง 2.3 เมื่อนำมาหาเงื่อนไขที่สัมพันธ์กันที่สุดโดยใช้แผนผังการตัดสินใจ ทำได้ดังนี้(รูป 2.12)

1. การชำระหนี้เงินกู้ น้อยกว่า 400 ดอลลาร์ ใช่หรือไม่? คำตอบเป็นใช่(yes), ดังนั้นให้พิจารณาทางด้านซ้ายมือของแผนผังการตัดสินใจ
2. บุคคลนี้ได้รับเงินเดือน หรือ ค่าจ้าง? คำตอบเป็น เงินเดือน(salary) ดังนั้นเงื่อนไขที่ 1 สัมพันธ์มากที่สุด

ตาราง 2.3 แสดงเงื่อนไขเป้าหมาย

	Loan Status	Monthly income	Job status	Repayment
Case1	?	\$2,500	salaried	\$250

ตาราง 2.4 แสดงคำตอบที่ได้จากการทำนาย

	Loan Status	Monthly income	Job status	Repayment
Case1	good	\$2,500	salaried	\$250

ดังนั้นผลการทำนายให้ผลลัพธ์ดี (good) เนื่องจากเงื่อนไขที่ 1 อยู่ทางด้านดี (ตาราง 2.4) ในทำนองเดียวกัน ถ้าพิจารณาข้อมูลในตาราง 2.5 โดยใช้แผนผังการตัดสินใจทำนาย (รูป 2.12) จะให้ผลลัพธ์เป็น very bad ดังนั้นจึงปฏิเสธการให้กู้ยืม เป็นต้น

ตาราง 2.5 เงื่อนไขเป้าหมาย

	Loan Status	Monthly income	Job status	Repayment
Case1	?	\$3,500	salaried	\$500

## 2.7 การพิจารณาเลือกเครื่องมือและโปรแกรม

วิลาส ววงค์ และ บุญเจริญ ศิริเนาวกุล (2535) ได้กล่าวถึงการเลือกเครื่องมือที่ใช้พัฒนาระบบผู้เชี่ยวชาญว่า ปัจจุบันมีการเปลี่ยนแปลงอย่างรวดเร็วทั้งด้านความง่ายและความเร็วในการพัฒนาระบบสาเหตุของการพัฒนาระบบอย่างรวดเร็วก็เพราะมีเครื่องมือที่ใช้พัฒนาระบบผู้เชี่ยวชาญ (Expert System Building Tools; ESBTs) นั่นเอง เครื่องมือเหล่านี้ส่วนใหญ่จะสร้างมาเพื่อการค้า และมีลักษณะพิเศษเพื่อการใช้งานที่ต่างกัน

ในที่นี้จะกล่าวถึงรายละเอียดของเทคโนโลยีต่าง ๆ ที่ใช้ในการสร้างเครื่องมือ เช่น การแสดงความรู้ เครื่องอนุมาน การติดต่อกับผู้พัฒนาระบบ การติดต่อกับผู้ใช้งาน และภาษาที่ใช้ในการพัฒนาเครื่องมือ

โครงสร้างหลักของระบบผู้เชี่ยวชาญจะประกอบด้วยฐานความรู้ เครื่องมือที่ทำหน้าที่อนุมานความรู้ในฐานความรู้และเป็นส่วนที่ติดต่อกับผู้พัฒนาระบบ ที่ทำหน้าที่อำนวยความสะดวกให้ผู้ที่จะพัฒนาระบบ และถ้าพิจารณาในส่วนของการติดต่อระหว่างเครื่องมือกับผู้พัฒนาระบบจะเห็นความจำเป็นของการมีระบบผู้เชี่ยวชาญดังนี้

1. เพื่อให้มีการพัฒนาฐานความรู้ที่จะต้องการมีความสอดคล้องกับระบบ
2. เพื่อให้สามารถสร้างส่วนที่ทำหน้าที่ติดต่อกับผู้ใช้ ให้เป็นไปตามที่ผู้พัฒนาระบบต้องการ
3. เพื่อสามารถเพิ่มส่วนที่ทำหน้าที่ในการควบคุมการอนุมานเข้าไปได้
4. ความสามารถในการติดต่อกับซอฟต์แวร์ และฐานข้อมูลชนิดอื่น
5. ความสามารถในการใช้ประโยชน์จากระบบคอมพิวเตอร์ เช่นความสามารถในการติดต่อสื่อสารแบบเครือข่าย

สำหรับลักษณะและวิธีการต่าง ๆ ของการแสดงความรู้ในฐานความรู้ ซึ่งเป็นสิ่งจำเป็นเบื้องต้นต่อการรู้จักระบบมีดังนี้

1. การอนุมาน เป็นการจำแนกชนิดของการติดต่อกันระหว่างระบบผู้เชี่ยวชาญและผู้ขอคำปรึกษา การแสดงผลคำปรึกษา และการรับความรู้เพิ่มเติมจากผู้ใช้
2. ลักษณะซอฟต์แวร์ของระบบ เป็นการกล่าวถึงความสามารถต่าง ๆ ทางซอฟต์แวร์ ภาษาที่ใช้ในการสร้างระบบและการใช้งานกับระบบอื่น
3. ความสามารถในการติดต่อกับผู้พัฒนาระบบ กล่าวถึงวิธีการสร้างกฎ และความจริง การแก้ไขฐานความรู้ และการสร้างหน่วยติดต่อกับผู้ใช้

## 2.8 การแสดงความรู้

การแสดงความรู้เป็นเงื่อนไขประการแรกที่ใช้พิจารณาเลือกเครื่องมือที่ใช้สร้างระบบผู้เชี่ยวชาญ วิธีการแสดงรู้นั้นจะต้องง่ายและสามารถใส่เรื่องที่ต้องการได้ครบถ้วน อย่างน้อยการแสดงความรู้ต้องสามารถจัดการกับเรื่องต่อไปนี้ได้คือ การบรรยาย การกำหนดค่าความมั่นใจ การแสดงการกระทำ ซึ่งทั้งสามส่วนนี้จะต้องประกอบกันเป็นฐานความรู้ ในฐานความรู้แบบกฎนั้น จะมีการแสดงความรู้อีกแบบหนึ่งที่ใช้กันอยู่ทั่วไปคือเฟรม การแสดงความรู้แบบนี้โดยปกติจะสามารถครอบคลุมการกำหนดวัตถุประสงค์ เช่น ตรรกศาสตร์ และ กฎ ได้ในกรณีของเครื่องมือที่แบ่งเป็นหลาย เฟรม

และอาศัยการกระทำเป็นส่วนเชื่อมต่อสถานการณ์และ ฐานข้อมูลที่เกี่ยวข้อง ในฐานะความรู้ นั้นสามารถแสดงข้อมูลได้หลายวิธี แต่โดยส่วนใหญ่จะแสดงในรูปแบบกฎ ซึ่งกฎเหล่านี้อาจนำมาจัดเป็นกลุ่ม ให้อยู่ในรูปของเฟรมเพื่อความสะดวกในการแก้ไขปรับปรุง การแสดงความรู้อีกแบบหนึ่งที่ใช้กัน ในเฟรมก็คือการใช้ตัวอย่าง ซึ่งการแสดงควมแบบนี้จะง่ายกว่าแบบกฎมาก

ในการแสดงความรู้ ผู้ที่พัฒนาระบบผู้เชี่ยวชาญจะต้องแทนค่าของความประสงค์และการกระทำประกอบกันเป็นฐานความรู้ และในการแทนค่านั้น บางครั้งผู้พัฒนาระบบมีความจำเป็นต้องพิจารณาระดับความมั่นใจที่มีต่อความรู้นั้นด้วย ถ้าหากมั่นใจในความรู้ไม่ถึง 100 เปอร์เซ็นต์ ผู้พัฒนาระบบจะต้องระบุจำนวนเปอร์เซ็นต์ของความมั่นใจถัดจากความประสงค์ และลักษณะการแสดงความรู้ดังกล่าวจะเรียกว่า การกำหนดค่าความมั่นใจ

### 2.8.1 เครื่องอนุมาน

เครื่องอนุมานเป็นอีกส่วนหนึ่งที่สามารถกำหนดความสามารถของเครื่องมือได้ โดยปกติแล้ว เครื่องอนุมานมีส่วนสำคัญที่ผู้พัฒนาระบบต้องคำนึงถึงนั่นคือ วิธีการอนุมาน การค้นหากฎที่เหมาะสม การคำนวณทางคณิตศาสตร์ และการเชื่อมโยงกับ โปรแกรมอื่น

สำหรับการจำแนกจะมีการจำแนกลักษณะต่าง ๆ ตามวิธีการแสดงความรู้ ซึ่งโดยปกติแล้วการแสดงความรู้ของระบบผู้เชี่ยวชาญจะเป็นการผลิต ซึ่งอาศัยกฎพื้นฐานของการแสดงความรู้ ลักษณะของการอนุมานสำหรับการแสดงความรู้เช่นนี้จะอาศัยวิธีการหาเหตุผลแบบเดินหน้า และแบบย้อนหลังเป็นหลัก โดยที่บางระบบอาจจะมีเพียงแบบใดแบบหนึ่ง แต่บางระบบจะมีทั้ง 2 แบบผสมกัน โดยอาศัยฟังก์ชันพิเศษในการแสดงความรู้เป็นตัวกำหนดค่าว่า เมื่อไรให้หาเหตุผลแบบเดินหน้าหรือย้อนหลัง นอกจากวิธีการกำหนดทิศทางในการหาเหตุผลดังกล่าวแล้ว วิธีการอนุมานแบบต่าง ๆ เช่นการอนุมานแบบออบเจกต์ การอนุมานแบบตรรกศาสตร์ และการอนุมานภายใต้ความไม่แน่นอน ก็เป็นสิ่งจำเป็นสำหรับการพิจารณาเลือกเครื่องมือ เฉพาะการอนุมานแบบออบเจกต์และการอนุมานแบบตรรกศาสตร์ จะเป็นวิธีการที่ใช้ในเครื่องมือบางระบบเท่านั้น ระบบเหล่านี้จะมีการควบคุมการอนุมานและการแสดงความรู้ชนิดพิเศษ สำหรับการหาเหตุผลภายใต้ความไม่แน่นอนเท่านั้น โดยปกติแล้ว เครื่องมือทั่วไปจะสามารถจัดการเรื่องของความไม่แน่นอนและจะมีเครื่องมือพิเศษบางแบบเท่านั้นที่สามารถหาเหตุผลแบบการตั้งสมมุติฐานได้

เนื่องจากวิธีการอนุมานนั้นจะเป็นตัวกำหนดความเร็วและความเที่ยงตรงของการให้คำปรึกษา ดังนั้น การแสดงความรู้ของเครื่องมือที่มีการอนุมานได้หลาย ๆ แบบนั้น ผู้แสดงความรู้จะต้องเป็นผู้กำหนดว่าจะให้ความรู้ใดมีวิธีอนุมานเช่นไร ในขั้นตอนไหน การที่ผู้แสดงความรู้จะกำหนดวิธีของการอนุมานได้นั้น เครื่องมือเหล่านี้จะต้องมีฟังก์ชันสำหรับควบคุมการอนุมานด้วย



ในการค้นหากฎที่เหมาะสมและการคำนวณทางคณิตศาสตร์สำหรับเครื่องมือนั้นเป็นส่วนประกอบภายในของเครื่องอนุมาน ผู้แสดงความรู้ไม่ต้องควบคุมการทำงานในส่วนนี้ตัวเครื่องจะเป็นผู้จัดการให้โดยอัตโนมัติ แต่สำหรับการเลือกเครื่องมือ ผู้เลือกจะต้องดูว่า การหากฎข้อที่เหมาะสมของเครื่องนั้นมีวิธีการเช่นไร เช่นการเปรียบเทียบแพทเทิร์นของตัวแปร หรือการใช้ดัชนี สำหรับการคำนวณทางคณิตศาสตร์จะต้องดูว่า สามารถคำนวณอะไรได้บ้าง

การเชื่อมโยงกับโปรแกรมอื่นนั้นเป็นการดูว่าเครื่องมือนั้นสามารถใช้งานร่วมกับโปรแกรมอื่นได้หรือไม่ บางครั้งเครื่องมือส่วนใหญ่จะสามารถติดต่อกับโปรแกรมฐานข้อมูลที่สำคัญได้เช่น Dbase, Foxpro และ Lotus เป็นต้น

การอนุมานของเครื่องมือที่ใช้ในการพัฒนาระบบผู้เชี่ยวชาญที่ดีต้องมีคุณสมบัติดังที่กล่าวมาเกือบทั้งหมดหรือทั้งหมด เพื่อให้ผู้พัฒนาระบบสามารถเลือกวิธีการอนุมานได้ด้วยตัวเอง แต่การมีเครื่องมืออนุมานมากเกินไป ก็สร้างความสับสนให้ผู้พัฒนาระบบได้เช่นกันหากผู้พัฒนาระบบมีความชำนาญในเรื่องการอนุมานไม่เพียงพอ

## 2.9 การติดต่อกับผู้ใช้

การติดต่อกับผู้ใช้ของระบบผู้เชี่ยวชาญ เป็นส่วนที่ระบบจะติดต่อกับผู้ที่มาขอคำปรึกษา ระบบที่ดีต้องมีหน้าที่ทำให้ผู้ใช้สะดวกต่อการขอคำปรึกษา มีการแสดงการตอบโต้ที่ง่ายและชัดเจน การติดต่อกับผู้ใช้ประกอบด้วย

1. วิธีการโต้ตอบระหว่างผู้ใช้และระบบ
2. การแสดงภาพในระหว่างขอคำปรึกษา
3. การเก็บความรู้จากคำตอบของผู้ใช้

ในส่วนของวิธีการตอบโต้ระหว่างผู้ใช้และระบบนี้ โดยปกติแล้วในส่วนของเครื่องมือจะเป็นแบบระบบป้อนคำถามออกมาเป็นตัวหนังสือที่หน้าจอ พร้อมกับมีเมนูของคำตอบที่ผู้ใช้สามารถเลือกได้ด้วยการเลื่อนแถบสว่าง หรือมีบรรทัดว่างให้ผู้ใช้ใส่คำตอบด้วยการพิมพ์คำตอบเข้าทางแป้นพิมพ์ สำหรับการใส่คำตอบของผู้ใช้นั้นจะสามารถใส่ค่าความมั่นใจต่อคำตอบนั้นได้ด้วย ระบบที่ดีของการแสดงคำตอบของเครื่องมือจะแสดงภาพได้ด้วย ในการแสดงภาพของระบบมีด้วยกันหลายแบบคือ การแสดงเฉพาะภาพนิ่ง การแสดงภาพเคลื่อนที่ การจำลองแบบ และการแสดงกราฟ เป็นต้น

สำหรับวิธีการติดต่อกับผู้ใช้ มีเรื่องที่จะต้องพิจารณาเพิ่มเติมอีกคือ ในระหว่างการให้คำปรึกษานั้น ผู้ใช้สามารถบอกให้เครื่องมืออนุมานแสดงสิ่งต่อไปนี้ให้ปรากฏบนจอภาพ คือขั้นตอนของการอนุมานหน้าจอ เหตุผลของการถามคำถามที่กำลังถาม และคำตอบที่เป็นไปได้ของคำถามสำหรับเครื่องมือที่มีอยู่ในท้องตลาดส่วนใหญ่สามารถทำเรื่องดังกล่าวอยู่แล้ว

การให้คำตอบของผู้ใช้ต่อความไม่มั่นใจในคำตอบ โดยทั่วไป นอกจากการกำหนดค่าความมั่นใจดังกล่าวข้างต้น ระบบผู้เชี่ยวชาญที่คิดจะต้องอนุญาตให้ผู้ใช้สามารถใส่คำตอบได้มากกว่า 1 คำตอบหรือให้ผู้ตอบตอบว่า “ไม่ทราบ” ก็ได้ เพราะเรื่องเหล่านี้เป็นเรื่องของการอนุมานภาพที่ให้ความไม่แน่นอนที่ระบบที่คิดต้องมีเสมอ ระบบผู้เชี่ยวชาญที่คิด การติดต่อกับผู้ใช้จะต้องสะดวกและสามารถทำเรื่องดังกล่าวได้ทั้งหมดหรือเกือบทั้งหมด

### 2.9.1 ลักษณะซอฟต์แวร์

ลักษณะซอฟต์แวร์จะบ่งบอกถึงธรรมชาติของระบบนั้น ๆ ว่ามีข้อดีและข้อจำกัดอย่างไร ลักษณะที่น่าสนใจอย่างหนึ่งคือ ภาษาคอมพิวเตอร์ที่ใช้สร้างเครื่องมือ ความสามารถในการคอมไพล์รวมทั้งการใช้งานร่วมกับภาษาคอมพิวเตอร์อื่น ๆ และระบบจัดการที่ใช้งานร่วมกับเครื่องมือดังกล่าว

สำหรับภาษาคอมพิวเตอร์ที่ใช้สร้างเครื่องมือและใช้งานร่วมกับภาษาอื่นนั้น โดยปกติแล้วจะมีความเกี่ยวข้องกันโดยตรง เช่น เครื่องมือสร้างด้วยภาษาซี(C) มักจะใช้งานร่วมกับภาษาซีได้ เป็นต้น ภาษาที่มักจะนำมาใช้เครื่องมือนี้ได้แก่ ภาษาลิสป(Lisp) ภาษาซี ภาษาปาสคาล(Pascal) และภาษาโปรล็อก(Prolog) เป็นต้น สำหรับผู้ทำหน้าที่สร้างฐานความรู้จะพิจารณาภาษาที่ใช้ในการสร้างเครื่องมือเป็นองค์ประกอบหนึ่งด้วย เพราะภาษาที่ตนถนัดก็จะทำให้ผู้พัฒนามีความคล่องตัวมากขึ้นในการแสดงความรู้ เนื่องจากผู้พัฒนาระบบสามารถใช้ประโยชน์จากการใช้ภาษานั้นในการทำงานร่วมกับเครื่องมือที่ใช้อยู่ได้

ความสามารถในการคอมไพล์เป็นอีกเรื่องหนึ่งที่ต้องพิจารณาเพราะเมื่อผู้พัฒนาระบบพัฒนาระบบผู้เชี่ยวชาญมาแล้ว ก็ต้องการทำให้ความรู้ที่พัฒนาขึ้นมีความสะดวกต่อการนำไปใช้งาน การคอมไพล์ความรู้ให้อยู่เป็นแอสเซมบลีไพล์ นอกจากจะทำให้เกิดความสะดวกในการใช้แล้ว ยังเป็นประโยชน์การการรักษาความลับของฐานข้อมูลด้วย

นอกจากความสามารถในการคอมไพล์ เรื่องที่ต้องพิจารณาประกอบกันคือการใช้ร่วมกันกับระบบจัดการ(Operating System) และความสามารถในการขยายระบบ เครื่องมือแต่ละชนิด จะมีความยุ่งยากในการขยายระบบแตกต่างกันและมีวิธีการที่แตกต่างกัน ทั้งนี้ขึ้นอยู่กับ การแสดงความรู้และบรรณาธิกรของการสร้างฐานความรู้ด้วย

### 2.9.2 ความสามารถในการติดต่อกับผู้พัฒนาระบบ

ความสามารถในการติดต่อกับผู้พัฒนาระบบ เป็นส่วนที่เกี่ยวข้องกับการพัฒนาระบบโดยตรง สิ่งที่จะต้องพิจารณาในส่วนนี้คือ การสร้างฐานความรู้ และวิธีการสร้างฐานความรู้ การแก้ไขความรู้ของเครื่องมือเหล่านี้มีอยู่ 2 แบบ คือ แบบที่อาศัยกรรมบรรณาธิกร(Word processor) จากภายนอกและแบบที่อาศัยกรรมบรรณาธิกรสำเร็จซึ่งทั้ง 2 แบบมีข้อดีและเสียต่างกัน โดยปกติแล้วโปรแกรมที่อาศัย

กรมบรรณาธิการจากภายนอกจะเหมาะกับผู้พัฒนาระบบที่มีความชำนาญในการสร้างฐานความรู้ เพราะทำให้สร้างและแก้ไขความรู้สะดวก แต่สำหรับผู้เริ่มต้นควรใช้บรรณาธิการฐานความรู้ (Knowledge-base) ที่มีมากับระบบ ซึ่งโดยปกติแล้วบรรณาธิการเหล่านี้จะออกมาให้ใช้งานอยู่แล้ว บรรณาธิการเหล่านี้มีหลายรูปแบบ เช่น แบบเมนูดึงลง (pull-down menu) แบบเติมคำลงในช่องว่าง ในระบบบรรณาธิการใด ๆ โปรแกรมจะตรวจสอบขบวนการเติมกฎให้กับผู้พัฒนาระบบได้

ของการสร้างและการแก้ไขความรู้ ยังมีเรื่องที่เกี่ยวข้องอีก 2 ส่วนคือ ความสามารถในการสร้างส่วนที่ติดต่อกับผู้ใช้และส่วนควบคุมการอนุมาน ส่วนแรกหมายถึงผู้พัฒนาสามารถสร้างหน้าจอ กำหนดรูปแบบการถามตอบ การสร้างภาพ สำหรับส่วนหลังหมายถึงวิธีการที่กำหนดขบวนการควบคุมการอนุมาน เช่น การกำหนดความมั่นใจ การกำหนดการควบคุมตามคำสั่ง “How” และ “Why” ซึ่งเป็นการกำหนดวิธีการอนุมาน

### 2.9.3 กระบวนการถอดความรู้

กระบวนการถอดความรู้หมายถึง กระบวนการที่ผู้พัฒนาระบบเรียนรู้และทำความเข้าใจกับความรู้นำมาพัฒนาเป็นระบบผู้เชี่ยวชาญ ในการเรียนรู้และทำความเข้าใจกับความรู้ นั้น ผู้พัฒนาสามารถศึกษาได้จากแหล่งความรู้ต่าง ๆ เช่น ผู้เชี่ยวชาญ หนังสือ เป็นต้น นอกจากนั้น ผู้พัฒนาระบบยังต้องกำหนดขอบเขตที่แน่นอนของการสร้างความรู้และขอบเขตนั้นต้องสามารถใช้ได้ดีกับระบบผู้เชี่ยวชาญ

เมื่อผู้พัฒนาระบบสามารถหาแหล่งและกำหนดขอบเขตของความรู้ได้แล้ว ขั้นตอนต่อไปคือ จำแนกโดเมนของปัญหา และปัญหาเฉพาะของงาน ในการเลือกปัญหาที่ถูกต้องบางครั้งอาจเป็นส่วนที่สำคัญที่สุดของการพัฒนาระบบ ขั้นตอนนี้เข้าใจว่าเทคโนโลยีทางด้านนี้ยังมีข้อจำกัดอยู่มาก ถ้าหากเลือกปัญหาไม่ถูกต้อง บางครั้งการพัฒนาระบบผู้เชี่ยวชาญจะเกิดปัญหาขึ้นและไม่มีใครสามารถแก้ไขได้ ดังนั้น ในการเลือกปัญหาจึงจำเป็นต้องพิจารณาความสามารถของระบบผู้เชี่ยวชาญ เพราะในระบบใหญ่ ถ้าหากเลือกปัญหาผิด อาจทำให้ระบบทั้งระบบล้มเหลวได้ เรื่องที่มีความสำคัญมากอีกเรื่องหนึ่งคือ กระบวนการถอดความรู้หมายความว่า การพัฒนาระบบต้องมีผู้เชี่ยวชาญที่พร้อมให้ความช่วยเหลือ ดังที่กล่าวมาแล้วว่าระบบผู้เชี่ยวชาญเป็นระบบที่สร้างขึ้นมาเพื่อพยายามเลียนแบบการทำงานของผู้เชี่ยวชาญ การให้ความช่วยเหลือในการพัฒนาระบบใหญ่นี้จะแตกต่างจากระบบเล็กซึ่งผู้เชี่ยวชาญสามารถประมวลความรู้ได้ด้วยตนเอง และหาความรู้เพิ่มเติมจากแหล่งภายนอกได้ แต่สำหรับระบบใหญ่แล้ว ความรู้ที่ได้จะต้องเป็นแหล่งที่มาจากผู้เชี่ยวชาญจำนวนมาก ที่มีประสบการณ์ในความรู้ นั้น ๆ เป็นอย่างดี สามารถเข้าใจและทราบวิธีแก้ปัญหาในเรื่องดังกล่าว ได้เช่นกัน

#### 2.9.4 การสร้างต้นแบบ

ผู้พัฒนาระบบควรเริ่มต้นการสร้างต้นแบบจากการเขียนแนวความคิดของความรู้ทั้งหมดที่ต้องการสร้าง โดยเริ่มต้นตามลำดับดังนี้

1. กำหนดเป้าหมาย(Goal) ที่เด่นชัด เป้าหมายในที่นี้คือ จุดหมายปลายทางของระบบการให้คำปรึกษาจะเป็นเช่นไร หรือคำตอบของการให้คำปรึกษานั้นเอง และคำตอบนี้จะมีอยู่หลาย ๆ คำตอบ ซึ่งระบบผู้เชี่ยวชาญจะเป็นผู้เลือกให้สอดคล้องกับลักษณะของปัญหา
2. กำหนดการไหลของโคอะแกรม(Flow diagram) ของปัญหาทั้งหมด ขั้นตอนนี้เป็นขั้นตอนการแสดงความรู้ การแสดงความรู้ต้องจัดลำดับของขั้นตอนที่จำเป็นออกมาแยกเป็น Case แต่ละ Case

การสร้างต้นแบบเป็นการแสดงความรู้เฉพาะส่วนขึ้นมา โดยการจำกัดโดเมนของความรู้ให้แคบลง การสร้างระบบต้นแบบนั้นมีวัตถุประสงค์เพื่อหาความเป็นไปได้ของการสร้างระบบและหาหนทางแก้ปัญหาก่อนที่จะสร้างระบบจริง ระบบต้นแบบที่สร้างขึ้นนี้ จะต้องมีลักษณะการทำงานที่เหมือนระบบจริงที่จะพัฒนาต่อ แต่กำหนดให้ขอบเขตของการแก้ปัญหาทำได้น้อยกว่า ระบบต้นแบบนี้จะเป็นการทดสอบที่ใช้ในการทดสอบว่าการแก้ปัญหาที่ได้ทำการออกแบบมานั้นถูกต้องหรือไม่และเพื่อเป็นแนวทางการขยายระบบต่อไป

#### 2.9.5 การขยายระบบ ทดสอบระบบ และปรับปรุงระบบ

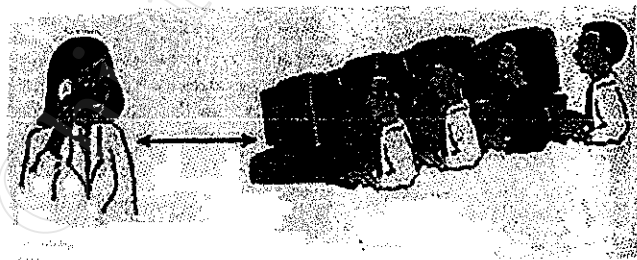
การขยายระบบโดยการนำต้นแบบที่แน่ใจว่าถูกต้องแล้วมาเพิ่มองค์ประกอบต่าง ๆ จนกระทั่งเป็นระบบที่สมบูรณ์ตามที่ได้มีการวางแผนไว้ โดยเดิมความรู้ในส่วนที่ยังขาดและตกแต่งระบบให้ดูประณีต เพิ่มเติมส่วนที่ใช้ในการอธิบายส่วนต่าง ๆ การพัฒนาระบบผู้เชี่ยวชาญขนาดใหญ่ก่อนที่จะมีการขยายระบบต้นแบบนี้จะต้องตรวจสอบอย่างละเอียดโดยผู้เชี่ยวชาญและวิศวกรความรู้ด้วยการนำเงื่อนไขต่าง ๆ ที่ได้กำหนดไว้ในการสร้างระบบต้นแบบมาทำการทดสอบ และตรวจโดยผู้เชี่ยวชาญเพื่อดูเงื่อนไขต่าง ๆ ที่ทดสอบนั้นว่าถูกต้องหรือไม่ ถ้าหากระบบต้นแบบมีความคลาดเคลื่อนจากที่กำหนดไว้ ก็จะต้องกลับไปออกแบบระบบใหม่ สำหรับการทดสอบระบบต้นแบบ มีสิ่งที่จะต้องคำนึงถึงเสมอว่า ระบบนี้ได้จำลองระบบให้มีขอบเขตของการแก้ปัญหาที่เล็กกว่าระบบจริง ดังนั้นเงื่อนไขการทดสอบบางอย่างที่ไม่ได้กำหนดไว้ในการสร้างระบบต้นแบบก็จะนำมาทดสอบไม่ได้ และเมื่อสร้างระบบเรียบร้อยแล้วยังต้องมีการประเมินผลด้วยว่าระบบที่จัดทำขึ้นเป็นไปตามความต้องการของผู้ออกแบบระบบหรือไม่ ในการตรวจสอบ ผู้ตรวจสอบจะต้องมีความเชี่ยวชาญและให้คำปรึกษาอย่างใกล้ชิด วิศวกรความรู้จะต้องตรวจสอบเงื่อนไขต่าง ๆ ของการอนุมานให้ครบถ้วน และผู้เชี่ยวชาญจะต้องตรวจพิจารณาความรู้ทุกอย่างที่มีอยู่ในระบบว่า ตรงกับความเป็นจริงหรือหากเกิดความผิดพลาดขึ้น วิศวกรความรู้จะต้องเป็นผู้แก้ไขกฎหรือข้อมูลต่าง ๆ ในฐานความรู้

การบำรุงรักษาผู้สร้างระบบผู้เชี่ยวชาญต้องเข้าใจว่าความรู้ที่ใส่เข้าไปในระบบนั้นมีวันล้าสมัย สามารถเปลี่ยนแปลงเพิ่มเติมความรู้ต่อไปในอนาคต ดังนั้นระบบผู้เชี่ยวชาญที่สร้างขึ้นจึงจำเป็นต้องอย่างยิ่งที่จะต้องมีการเพิ่มเติมความรู้ และต้องปรับปรุงแก้ไขอย่างสม่ำเสมอ เพื่อให้ระบบผู้เชี่ยวชาญนั้น มีความทันสมัยตลอดเวลา

## 2.10 CBR กับงานบริการลูกค้า

Ian Watson (1997) กล่าวถึงการนำ CBR มาใช้ในปัจจุบันว่า งานบริการลูกค้า(Customer Service) มีส่วนสำคัญที่ทำให้ได้เปรียบคู่แข่งด้านการค้ามาก ซึ่งในปี 1990 มีมูลค่าสูงถึงล้านดอลลาร์ โดยนำมาสร้างความสำเร็จเปรียบคู่แข่งทางการตลาดในการเพิ่มความพึงพอใจให้ลูกค้า นั่นคือวัตถุประสงค์หลักของบริษัท รวมทั้งนำมาเพิ่มประสิทธิภาพในการทำงานเพื่อให้ได้ผลผลิตที่มีคุณภาพและมีจำนวนมากตามต้องการ ที่เป็นเช่นนี้ก็เนื่องจากความต้องการของลูกค้าที่มากขึ้น และคาดหวังว่าจะได้รับสิ่งของที่มีคุณภาพดี มีผลตอบแทนคุ้มค่ามากที่สุด เช่น ราคาอุปกรณ์คอมพิวเตอร์ถูกลง อันเนื่องมาจากการเจริญเติบโตของบริษัทคอมพิวเตอร์ที่เกิดขึ้นมากมาย ทำให้มีการแข่งขันสูงขึ้น ผู้บริหารระดับสูงหลายบริษัทจึงใช้วิธีการลดราคาอุปกรณ์คอมพิวเตอร์ซึ่งเป็นทางหนึ่งที่ทำให้บริษัทอยู่ได้ แต่หลายบริษัทเน้นบริการลูกค้าหลังการขาย โดยมุ่งเน้นการตอบปัญหาที่ลูกค้าติดต่อเข้าสู่ศูนย์บริการ (Service centers หรือ Help desks,HD) เพื่อขอคำแนะนำซึ่งมีมากในแต่ละวัน และศูนย์นี้ยังทำหน้าที่อื่น ๆ ในงานบริการดังนี้

1. การฝึกอบรม ก่อนที่ผู้เชี่ยวชาญตอบคำถาม(Help desk operators,HDOs) ต้องมีความรู้ ซึ่งเป็นสิ่งสำคัญมากที่สุด โดยให้ HDOs เข้าใหม่ใช้เวลาไม่มากเพื่อพัฒนาตนเอง



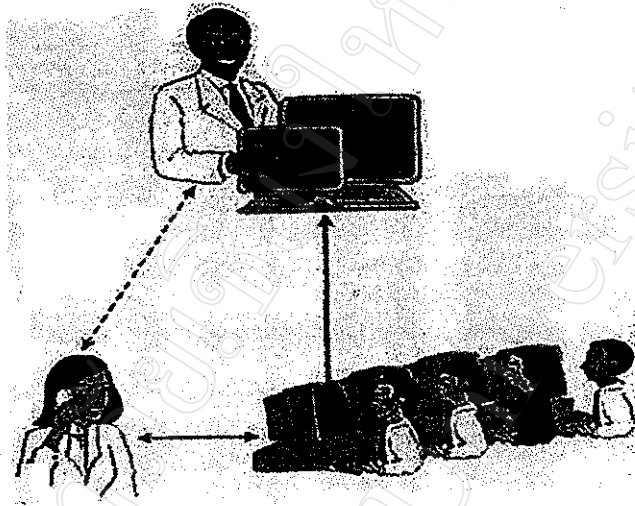
รูป 2.13 Help desk ตอบคำถามลูกค้า(Ian Watson,1997:90)

2. ในกรณีที่มีปัญหากับผลิตภัณฑ์ใหม่ก็สามารถตอบคำถามได้เพราะข้อมูลต่าง ๆ ของผลิตภัณฑ์นั้นได้บันทึกไว้ในระบบแล้ว

โครงสร้าง Help Desk

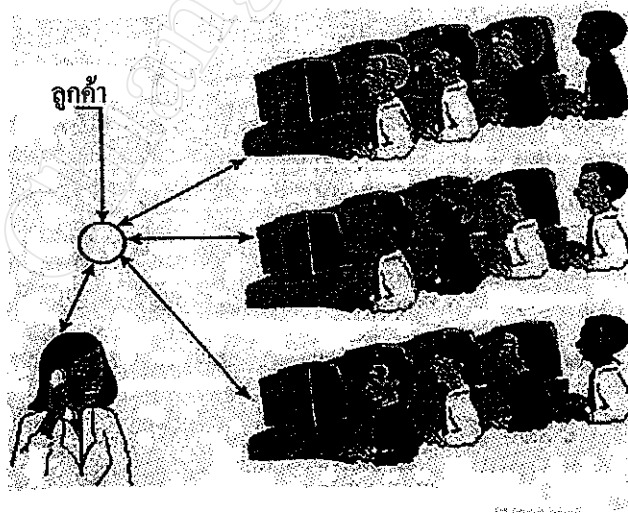
Help Desk มีด้วยกัน 2 รูปแบบคือ

1. Help Desk แบบเดี่ยว (Single point of contact) คือ ลูกค้าทุกคนที่โทรศัพท์เข้ามายัง Help Desk Operators(HDOs) HDO คนเดียวจะตอบปัญหาลูกค้า(เรียกว่า Frontline HDOs) แต่ในกรณีที่ HDOs ตอบปัญหาไม่ได้ ก็จะโอนสายไปยัง HDOs ที่มีความชำนาญกว่าก็ตอบปัญหาแทน ดังรูป 2.14



รูป 2.14 Help Desk แบบเดี่ยว(Ian Watson,1997:92)

2. Help Desk แบบหลากหลาย(Multiple point of contact) คือ ลูกค้าที่โทรศัพท์เข้ามาสามารถเลือก HDOs ที่ชำนาญเรื่องที่ลูกค้าต้องการถาม ดังรูป 2.15



รูป 2.15 Help Desk แบบหลากหลาย(Ian Watson,1997:93)

ในปัจจุบันได้มีแนวโน้มที่จะทำให้เป็น Help Desk แบบเดี่ยวทั้งหมดแล้วด้วยเหตุผลต่อไปนี้

1. Help Desk แบบเดี่ยว สามารถจัดรูปแบบขององค์กรได้ง่าย
2. มีการคัด HDOs ที่มีความชำนาญมากมาปฏิบัติหน้าที่เพื่อให้สามารถตอบปัญหาได้ทุกปัญหา
3. ลูกค้าไม่พอใจที่ต้องรอสายเป็นเวลานาน ๆ

CBR จึงเป็นแนวคิดที่จะนำมาสนับสนุน Help Desk แบบเดี่ยวที่เพิ่มประสิทธิภาพในการตอบปัญหาได้ดียิ่งขึ้น

### 2.11 ระบบผู้เชี่ยวชาญ

30 กว่าปีที่ผ่านมามีระบบผู้เชี่ยวชาญได้ถูกพัฒนามาเรื่อย ๆ มีรูปแบบของปัญหาที่ชัดเจน รูปแบบที่เดิมเป็น Rules ซึ่งบางครั้งถูกคัดค้าน มาในปัจจุบันเทคโนโลยีในระบบ CBR ได้พัฒนาการประยุกต์ใช้ระบบให้มีเหตุผล มีหลักสำคัญและมีขอบเขตระบบผู้เชี่ยวชาญสามารถสำเร็จได้ อย่างไรก็ตามยังมีปัญหาของแบบแผนระบบผู้เชี่ยวชาญ 5 ประการดังนี้

1. การพยายามนำความรู้จริง ๆ ออกมาทำได้ยากมากและต้องใช้เวลา
2. ระบบผู้เชี่ยวชาญเป็นระบบที่ซับซ้อนมาก และสามารถใช้เวลาหลายปีในการพัฒนา
3. ระบบผู้เชี่ยวชาญทำงานอย่างช้า เพราะเป็นขบวนการทำงานที่ซับซ้อนและมักล้มเหลวในตัวระบบเอง
4. ระบบผู้เชี่ยวชาญมักจะมีปัญหาในเรื่องของการจัดการขนาดบรรจุข้อมูล
5. ในการพัฒนาแต่ละครั้ง สิ่งที่ยากที่สุดคือการบำรุงรักษาเครื่องมือ CBR

ปัญหาแรก อยู่ที่หน่วยความจำของระบบผู้เชี่ยวชาญ แต่ได้พัฒนาวิธีการค้นหาคำตอบใหม่ขึ้นมาเรียกว่า การจำกัดความรู้ให้แคบลง(knowledge elicitation bottleneck) ปัญหาที่ 2 เป็นปัญหาคุ้นเคยที่ระบบผู้เชี่ยวชาญต้องพัฒนาและเพิ่มการตอบสนองให้มากขึ้นในระบบผู้เชี่ยวชาญ พัฒนาการออกแบบรูปแบบภาษาที่ใช้สร้างและภาษาที่ใช้ติดต่อกับระบบ ปัญหาที่ 3 แก้ไขได้โดยการอาศัยพลังจากขบวนการ ขณะที่การแก้ปัญหาค 4 ได้รับการแก้ไขโดยการผสมผสานกันระหว่างเทคโนโลยีของเอไอ (AI) กับเทคโนโลยีด้านข้อมูล

อย่างไรก็ตาม หลายปีที่ผ่านมา ผู้ใช้เชื่อว่าการดูแลบำรุงรักษาระบบผู้เชี่ยวชาญนั้นทำได้ง่าย นอกจากนี้มีข้อได้เปรียบสำหรับ Case base expert systems เนื่องจาก CBR มีประโยชน์หลาย ๆ อย่างมากกว่า Rule based reasoning

1. ระบบCBR ต้องใช้เวลาพอสมควรในการสร้าง Knowledge elicitation bottleneck เนื่องจากการดึงเอาความรู้ออกมา จากการทำงานจริงใน Case ที่ผ่าน มา ซึ่งสิ่งนี้ถูกพิสูจน์โดยระบบ SMART

2. ระบบ CBR สามารถจะสร้างรูปในแบบใดก็ได้ เช่น ระบบฐานข้อมูลที่แตกต่างกัน ก็ไม่เป็นอุปสรรค ดังตัวอย่างของ SMART
3. การดำเนินการเป็นการทำงานจริง ของการบ่งชี้ลักษณะพิเศษของ Case
4. ระบบ CBR สามารถเปลี่ยนแปลงไปตามสภาพการณ์ เฉพาะบางส่วนของ Case-base เช่น เหตุการณ์ที่เกิดขึ้นกับ SMART, CASELINE และ SWIFT การใช้ CBR ระบบไม่จำเป็นต้องสมบูรณ์เนื่องจากระบบสามารถเติบโตได้ ซึ่งทำให้ลืมนความวิตกกังวลของ Rule based systems ที่จะบอกได้อย่างไรว่าระบบที่สร้างขึ้นมาสมบูรณ์แล้ว
5. ระบบ CBR สามารถเสนอวิธีแก้ไขปัญหาได้อย่างรวดเร็วโดยไม่จำเป็นว่าการแก้ปัญหที่เกิดขึ้นมาเคยปรากฏก่อนหรือหลัง แต่เป็นการได้มาของข้อสรุปคำตอบมาจาก เงื่อนไขของแต่ละช่วงเวลา ความสำคัญ และสถานะ Help desk ในขณะนั้น
6. Case ทั่วไปหรือเฉพาะ สามารถที่จะหาคำอธิบาย ซึ่งบางครั้งให้ความกระจ่างของปัญหาได้ดีกว่า Rule base
7. CBR สามารถเรียนรู้โดยนำ Case ใหม่เข้ามา
8. การเติบโตของ CBR จะแสดงให้เห็นความเป็นไปขององค์กรหรือในหน่วยงานนั้น เปรียบเทียบได้เช่น ถ้าระบบ rule base ออกแบบมาเพื่อใช้งานกับ 6 บริษัทเป็นเวลา 6 เดือน เมื่อมาประเมินผลความสามารถของระบบหรือโปรแกรมจะพบว่ายังเหมือนเดิม แต่หากเป็น CBR จะพบว่าระบบจะมีการเปลี่ยนแปลงไป สามารถตอบคำถามได้มากขึ้น เนื่องจากมีการนำ Case ใหม่ ๆ เข้าไปในระบบการเรียนรู้แบบนี้เกิดขึ้นเพราะ CBR จะพัฒนาตัวเองจากองค์ความรู้ที่ได้เข้ามา นั่นเอง

ตาราง 2.6 เปรียบเทียบการทำงานของระบบ CBR กับ Rule base reasoning

	Rule-base reasoning	CBR
Problem area	แคบ, เข้าใจง่าย	กว้าง, เข้าใจยาก
Knowledge representation	อยู่ในรูป IF-THEN	CASE
System provides	ในรูปแบบคำถาม	เป็นเรื่องราว, คำอธิบาย
Explanation by	อธิบายเป็นกฎ	เป็นเรื่องราว, คำอธิบาย



### อนาคตของ CBR

Case base ถือได้ว่าเป็นวิทยาการที่ใหม่ เกิดขึ้นในสาขาคอมพิวเตอร์ประมาณ 10 ปี จึงยังต้องมีอีกหลายส่วนที่ต้องทำให้ CBR มีหน้าที่การทำงานที่สมบูรณ์กว่าที่เป็นอยู่ในปัจจุบัน ซึ่งแบ่งเป็นข้อ ๆ ได้ดังนี้

1. การนำเสนอปัญหาและการแก้ไขควรเป็นอย่างไร
2. การค้นหาข้อมูลควรเป็นอย่างไรเพื่อประสิทธิภาพสูงสุด
3. โครงสร้างด้านความสัมพันธ์ของข้อมูลควรเป็นอย่างไร
4. ทำอย่างไรให้ CBR รองรับกับระบบมัลติมีเดีย
5. ทำอย่างไรให้ CBR ใสข้อมูลได้มาก ๆ

การค้นคว้านี้เป็นการแนะนำรู้ถึงข้อสรุปของ CBR และวิธีใช้ และกล่าวถึงวิธีที่จะพัฒนาระบบนี้ต่อไป ซึ่งการพัฒนาตรงนี้ทำให้เกิดการพัฒนาเครื่องเครื่องที่สนับสนุน CBR อย่างมาก โดยใช้วิธีการของ Data base และ AI ทำให้ข้อมูลที่ออกมาในคำสั่งเป็นความรู้แบบอัตโนมัติ

### หน่วยความจำที่สร้างขึ้น

จากการเจริญเติบโตทางด้านวิศวกรรมและคอมพิวเตอร์ CBR สามารถใช้ประโยชน์ในเรื่องการจัดการเพื่อช่วยแก้ปัญหาต่าง ๆ ที่เกิดขึ้นได้ โดยอาศัยข้อมูลจากผู้แก้ไขไว้แล้วหลาย ๆ บริษัทได้พยายามสร้างขึ้นมา และตัวอย่างที่ดีของ Case base

#### Case base เป็นประโยชน์ดังนี้

1. ความรู้เกี่ยวกับการแก้ปัญหา ถูกรักษาเอาไว้ในสภาพดีและพนักงานสามารถนำมาใช้ได้
2. ให้คำแนะนำ และความช่วยเหลืออย่างสม่ำเสมอ ทำให้ลูกค้าเชื่อถือคำพูดของพนักงาน
3. Case base เกิดการเรียนรู้จากปัญหาใหม่ ๆ และการที่ระบบพยายาม ในการแก้ปัญหาให้ได้

ข้อความข้างบนเรียกว่า การจัดการทางด้านความรู้ที่มีค่า(Knowledge assetmanagement) และหลาย ๆบริษัท กำลังเริ่มต้นใช้ระบบ CBR เป็นเครื่องมือช่วยเหลือในการรักษาความรู้และกระจายประสบการณ์การทดลองของเขา ตัวอย่างเช่น

1. NEC รวบรวม case base ของปัญหาต่าง ๆ ของพนักงาน โดยสร้าง SQUAD แล้วบรรจุข้อมูลไว้มากกว่า 20,000 case และครอบคลุมกว่า 3,000 ปัญหาใหม่เพิ่มขึ้นในแต่ละปี บางปัญหาสามารถแก้ไขได้แต่ยังมีปัญหาอื่น ๆ ที่ช่วยคนให้หลีกเลี่ยงการทำพลาดซ้ำ

2. Apple computer ใช้ระบบการสร้างหน่วยความจำเรียกว่า NNApple ในหนทางที่คล้ายคลึงกับ NEC case base บรรจุคำบรรยาย case และวิธีการแก้ปัญหาจากประสบการณ์ ซึ่งสนับสนุนให้ Apple ถูกเรียกว่า คนทำงานด้วยความรอบรู้ (Knowledge workers)

Case base จะต้องใหญ่ เพื่อการใช้ประโยชน์

คำถามที่มักพบเสมอจากการประชุมเกี่ยวกับ CBR คือ ระบบของท่านมีกี่ Case จะสามารถให้ข้อมูลออกมาได้เร็วหรือไม่ คำตอบคือมี Case จำนวนมากพอที่จะนำไปใช้ประโยชน์และรวดเร็วพอที่จะใช้ประโยชน์ ดังนั้น Case base ควรจะมีขนาดใหญ่เพื่อที่จะบรรจุ Case ไว้ได้มาก ๆ เพื่อการแก้ปัญหาที่สมบูรณ์

ข้อดีของโปรแกรม CBR Express V2.0

1. โปรแกรม CBR Express ได้ถูกสร้างขึ้นเพื่อสำหรับการตรวจสอบ และวิเคราะห์สาเหตุ เพื่อให้ได้คำตอบ และนำไปใช้ในแผนกบริการลูกค้า จึงมีบริษัทชั้นนำหลายแห่งนำระบบ CBR ไปใช้ แต่จะใช้ชื่อเรียกที่ต่างกันออกไป
2. โปรแกรม CBR Express สามารถทำงานบนระบบปฏิบัติการบนวินโดวส์ 95,98 และ 2000 ซึ่งการทำงานที่ไม่ซับซ้อน ผู้ใช้ระบบหากไม่เคยใช้คอมพิวเตอร์ ก็จะใช้เวลาเพียงเล็กน้อยเพื่อเรียนรู้วิธีการใช้
3. โปรแกรม CBR Express มีการค้นหาคำตอบแบบใช้ข้อมูลใกล้เคียง ซึ่งได้กล่าวถึงรายละเอียดมาแล้ว จึงทำให้ โปรแกรม CBR Express สามารถแสดงปัญหาใกล้เคียงให้ผู้ใช้งานทราบ
4. การสร้างฐานความรู้ให้กับ โปรแกรม CBR Express สามารถทำได้ง่าย แม้ว่าผู้ที่ไม่เคยพัฒนาระบบผู้เชี่ยวชาญก็ตาม

## 2.12 เอกสารและงานวิจัยที่เกี่ยวข้อง

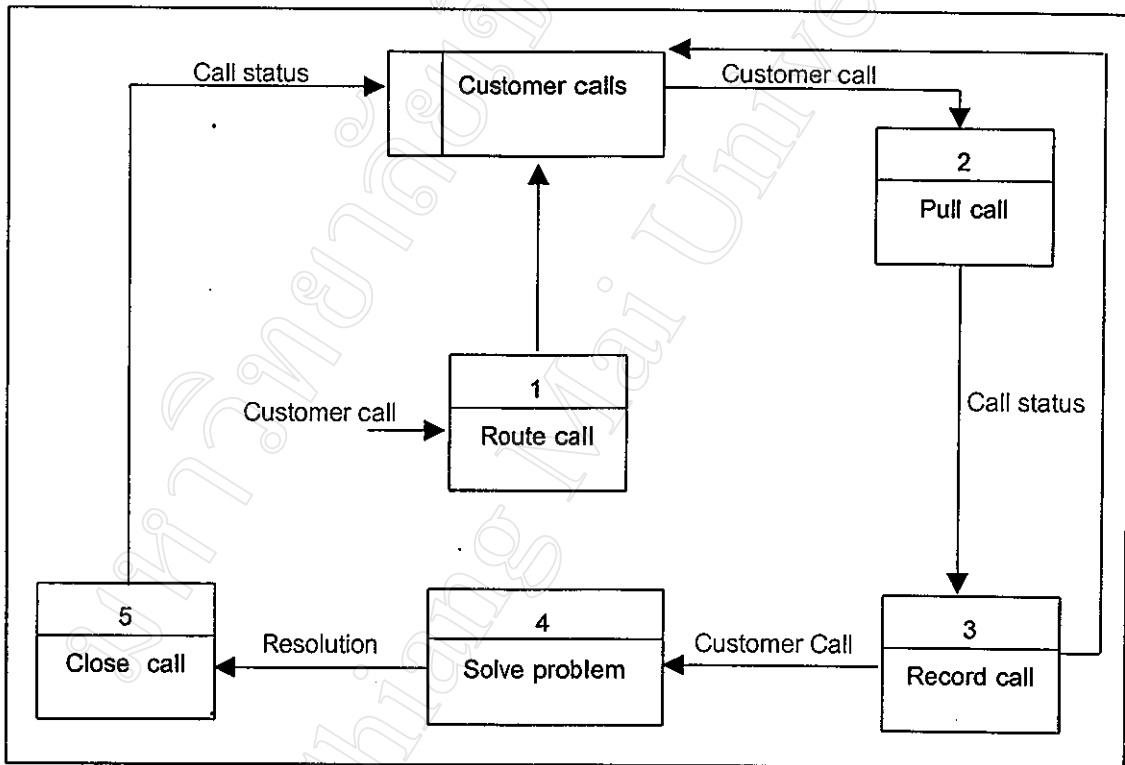
Ian Watson (997):91 บริษัทคอมแพคได้ใช้ระบบ SMART เข้ามาสนับสนุน Help Desk โดยมีรายละเอียดดังนี้

บริษัทคอมแพคเป็นบริษัทที่มีมูลค่าหลายล้านเหรียญในวงการธุรกิจ การดำเนินกิจการจำหน่ายคอมพิวเตอร์ส่วนบุคคล คอมพิวเตอร์พกพา คอมพิวเตอร์เก็บข้อมูลขนาดใหญ่ มีเป้าหมายที่จะเหนือคู่แข่ง โดยใช้กลยุทธ์ที่แตกต่างจากบริษัทอื่น ๆ คือ การบริการหลังการขาย บริษัทมีแผนกบริการที่กำหนดจุดมุ่งหมายว่า “เมื่อลูกค้าทุกคนเข้ามาหาเราแล้ว เราจะกระตือรือร้น สุภาพอ่อนโยน ช่วย

เหลือเต็มความสามารถและใส่ใจต่อท่าน”

ในปี 1990 บริษัทคอมแพคได้นำเครื่องตอบรับโทรศัพท์อัตโนมัติ เพื่อตอบปัญหาให้กับลูกค้าที่ติดต่อเข้ามายังบริษัท ซึ่งวิศวกรจะบันทึกเสียงไว้ที่เครื่องแทนการติดต่อกับลูกค้าโดยตรง โดยมีขั้นตอนการทำงานดังรูป 216

ระบบดังกล่าวจะติดตั้งไว้ที่ตัวแทนของบริษัทคอมแพคตามเมืองต่าง ๆ แต่ไม่สามารถประเมินผลงานได้อย่างชัดเจน และในที่สุดในปี 1991 บริษัทตัดสินใจเปิดศูนย์บริการลูกค้าขึ้นเองทำให้ต้องจ้างพนักงานเพิ่มขึ้นอีกจำนวนมากเนื่องจากมีลูกค้าติดต่อเข้ามามากกว่า 1000 รายต่อวัน เมื่อเป็นเช่นนี้ทางบริษัทจึงเห็นว่าควรมีระบบใหม่ที่มีประสิทธิภาพเข้าสนับสนุนแผนกบริการ



รูป 2.16 เครื่องตอบรับโทรศัพท์อัตโนมัติ(Ian Watson,1997:94)

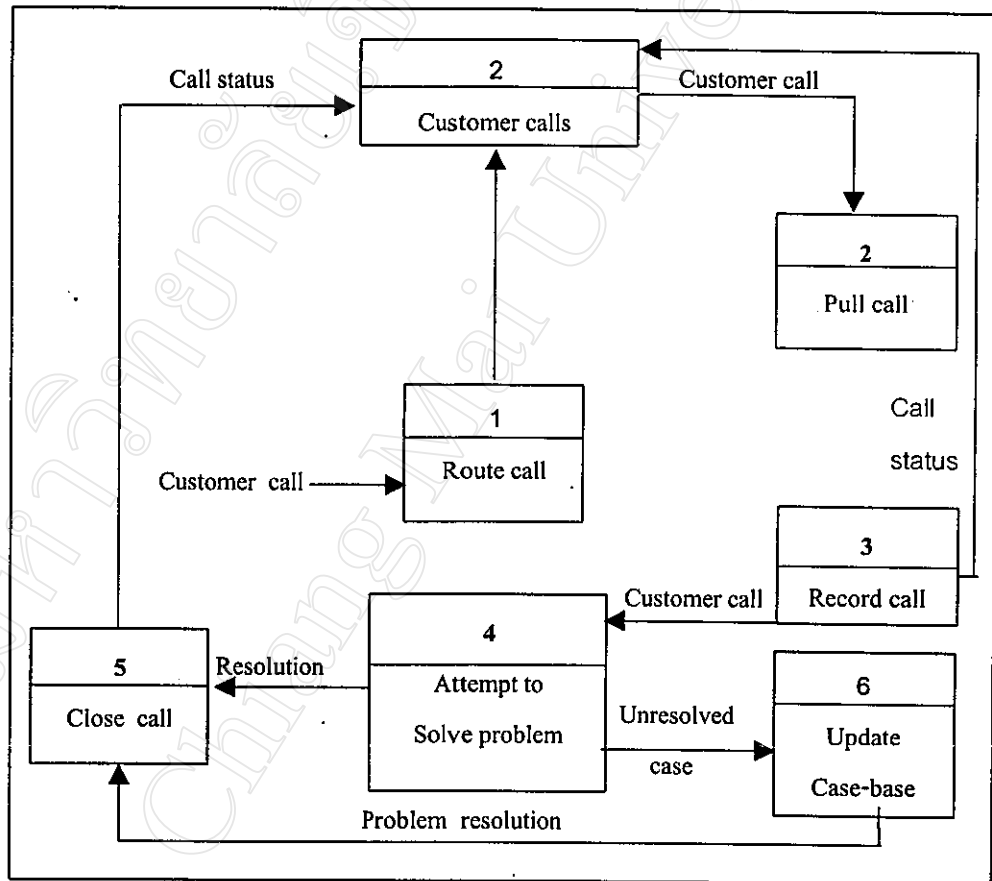
การแก้ปัญหาของบริษัทคอมแพค

เพื่อให้บรรลุวัตถุประสงค์ในการแก้ไขปัญหา บริษัทคอมแพคจึงให้ความรู้แก่แผนกบริการลูกค้าดังนี้

1. ข้อมูลต่าง ๆ ที่จำเป็นต่อการปฏิบัติงาน
2. ผู้เชี่ยวชาญเพื่อให้คำแนะนำลูกค้าตลอด 24 ชั่วโมง และทุกวัน
3. ให้คำตอบที่ถูกต้อง ตรงกัน ทุกครั้งเมื่อมีลูกค้าโทรเข้าขอคำแนะนำ

4. ต้องการลดเวลาในการแก้ปัญหา ไม่ว่าจะปัญหาเก่าหรือใหม่
5. พนักงานใหม่ๆ สามารถเรียนรู้ได้จากข้อมูลที่มีอยู่ และให้เป็นข้อมูลของบริษัท คอมแพคเท่านั้น

หลังจากที่ได้วัตถุประสงค์แล้วบริษัทคอมแพคได้ติดต่อกับบริษัทอินเฟอเรนซ์(Inference Corp.) เพื่อพัฒนาระบบ CBR ที่เรียกว่า SMART (Support Management Automated Reasoning Technology) ซึ่งมีขั้นตอนการทำงานดังรูป 2.17 วิศวกรของบริษัทคอมแพคได้ใส่ข้อมูลที่จำเป็นต่าง ๆ ลงไปในโปรแกรมได้แก่ ตั้งแต่รายละเอียดของลูกค้า เช่น ชื่อ ที่อยู่ เก็บเป็นข้อมูลเพื่อให้สะดวกต่อการค้นหา และหลังจากนั้นก็จะสอบถามลูกค้าถึงปัญหา และพิมพ์ลงในโปรแกรมที่พัฒนาขึ้น โปรแกรมก็จะแสดงรายละเอียดของปัญหาที่ตรงกับอาการที่ได้รับแจ้งมากที่สุดเรียงตามลำดับ



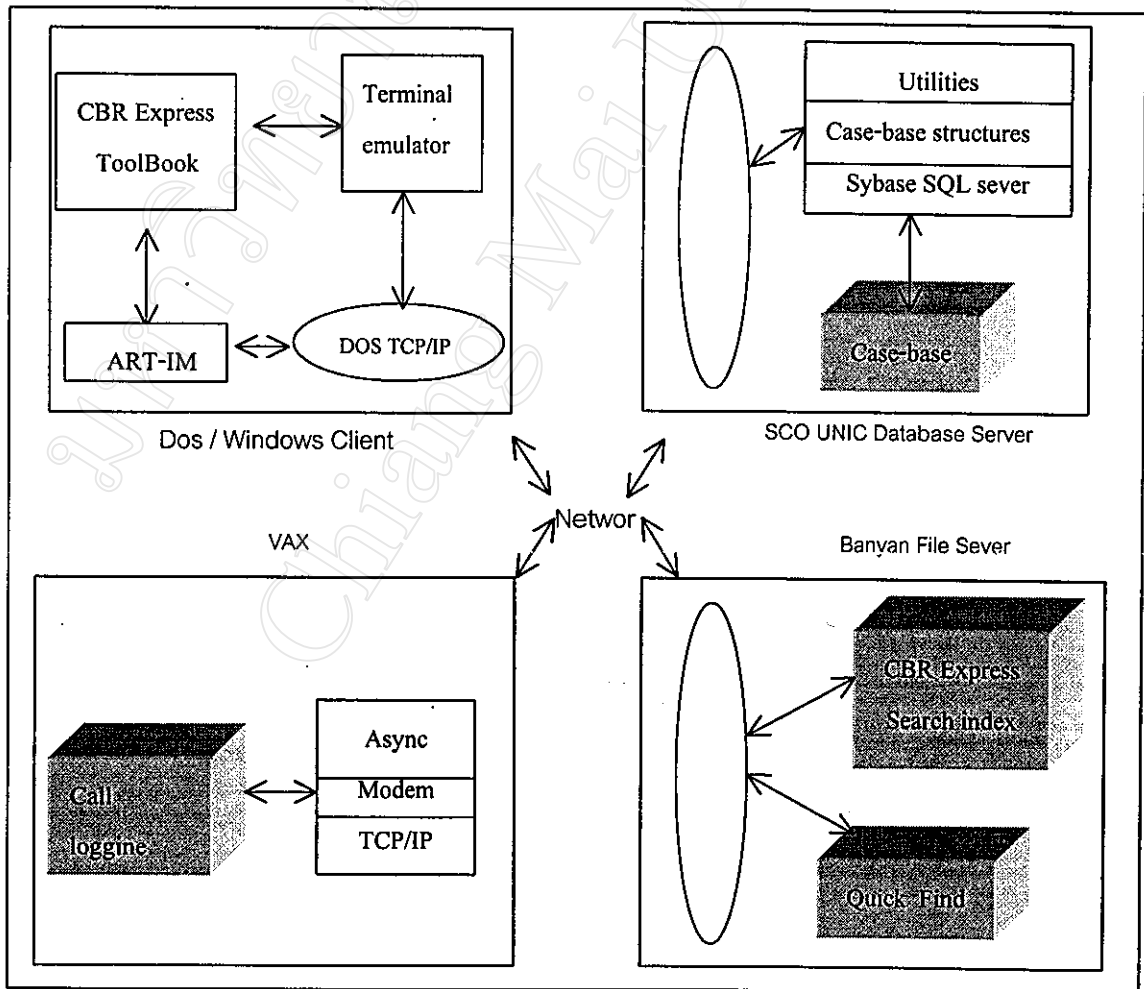
รูป 2.17 ขั้นตอนการทำงาน SMART(Ian Watson,1997:96)

แผนกบริการและวิศวกร ได้ใช้คำถามจากลูกค้ามากำหนดเป็นปัญหา โดยมีการปรับเปลี่ยนให้เหมาะสมเพื่อการหาข้อมูลที่ดีขึ้น SMART ที่จัดทำขึ้นใช้ได้มีการจัดปัญหาแบบ case base และหาคำตอบได้เป็นอย่างดีเมื่อมีการใส่ข้อมูลเข้าไปเรื่อย ๆ นอกจากนั้น SMART ยังช่วยเปลี่ยนแปลงการทำงานของแผนกบริการและให้ผลเป็นที่น่าพอใจอย่างยิ่ง

ตาราง 2.7 แสดงถึงการพัฒนาโปรแกรม SMART

Date	Activity
Aug 1991	Development lab on-line to support case building
Sep 1991	Multi-user access available in lab
Nov 1992	Case-base moved to relational database; 12 case-builders on-line plus 20 additional users
Dec 1991	First LAN completed 50+users
Jan 1992	Second LAN completed additional 50+users

อย่างไรก็ตาม SMART ก็ยังมีปัญหาบ้างในระยะเริ่มแรกกล่าวคือไม่สามารถตอบปัญหาที่ได้จากข้อมูลในโปรแกรม ปัญหาประเภทนี้เรียกว่า ปัญหาที่แก้ไม่ได้(Unresolved case) ซึ่งทางแผนกก็ใช้วิธี โอนสายไปยังวิศวกรที่มีประสบการณ์เพื่อให้ตอบคำถามแทน



รูป 2.18 การจัดระบบฐานข้อมูลของระบบ SMART(Ian Watson,1997:98)

เมื่อให้คำตอบแก่ลูกค้าได้แล้ว ก็จะนำข้อมูลนั้น ๆ มาบันทึกลงใน SMART เสมือนเป็นการเรียนรู้งานเพิ่มขึ้น และระบบก็ฉลาดขึ้นตามลำดับ

#### การประยุกต์ใช้งานของ SMART

SMART มีลักษณะ โครงสร้างแบบ Client-server และเขียนรวมเป็นโปรแกรมของบริษัทคอมพิวเตอร์ใช้ชื่อว่า VAX-base สำหรับระบบติดต่อกับผู้ใช้มีการใช้โปรแกรม ToolBook ช่วยออกแบบในส่วนของหน้าจอ สำหรับ terminal ทำงานบนระบบปฏิบัติการวินโดวส์ VAX system, UNIX SCO, และ Banyan Vines เป็น server ทำหน้าที่เก็บข้อมูลแบบ Case-base ดังรูป 2.18

#### ผลกระทบที่เกิดขึ้น

บริษัทคอมพิวเตอร์ได้ทดสอบระบบ SMART เมื่อเดือนตุลาคม 1991 โดยคาดหวังผลทดสอบการแก้ปัญหาไว้ที่ 50 % แต่เมื่อนำไปใช้งานจริง ๆ ปรากฏว่าสามารถแก้ปัญหาได้ถึง 87 % นอกจากนี้ในปี 1992 SMART ยังได้รับรางวัล IAIA “Innovative Application in Artificial Intelligence Award” ซึ่งเป็นรางวัลสำหรับบริษัทที่มีการพัฒนาระบบผู้เชี่ยวชาญมาใช้ในงานบริการ ปัจจุบัน SMART มีส่วนสำคัญในแผนกบริการลูกค้า และสามารถแก้ปัญหาได้ถึง 95 % และขณะนี้มีการนำ SMART มาใส่ในเครื่องคอมพิวเตอร์ส่วนบุคคล และเปลี่ยนชื่อเป็น EZSearch เพื่อให้ลูกค้า สามารถเรียนรู้และแก้ปัญหาด้วยตนเอง