บทที่ 2

องค์ประกอบของโครงระบบเชี่ยวชาญ

จากความรู้เบื้องต้นเกี่ยวกับโครงระบบเชี่ยวชาญในบทที่ 1 จะเห็นได้ว่า โครงระบบเชี่ยวชาญมีองค์ประกอบที่สำคัญ 2 ส่วนคือ เครื่องจักรกลวินิจฉัย และฐานความรู้ แต่ละส่วนก็มีวิธีการแตกต่างกันไปในแต่ละระบบ ในบทนี้จะกล่าวถึงวิธีการต่าง ๆ ที่ใช้ในการทำงานของเครื่องจักรกลวินิจฉัย และการแทนค่าความรู้ภายในฐานความรู้อย่างละเอียด รวมทั้งข้อดีข้อเสีย และความเหมาะสมของวิธีการต่าง ๆ เปรียบเทียบกันด้วย นอกจากองค์ประกอบที่สำคัญทั้ง 2 ส่วนนี้แล้ว ส่วนติดต่อกับผู้พัฒนา ส่วนติดต่อกับผู้ใช้ และ ความสามารถในการติดต่อกับซอฟแวร์อื่น ก็จะได้กล่าวไว้ในบทนี้เช่นกัน เพื่อใช้เป็นข้อพิจารณาในการเลือกลักษณะของโครงระบบเชี่ยวชาญที่ต้องการมาใช้ให้เหมาะสม

เครื่องจักรกลวินิจฉัย (Inference engine)

เครื่องจักรกลวินิจฉัยมีหน้าที่ในการควบคุมการนำความรู้จากฐานความรู้และข้อมูลต่าง ๆ จากผู้ใช้ไปใช้ในการแก้ไขปัญหา และทำการแก้ไขปัญหาจากความรู้ที่ได้รับอย่างถูกต้องและมีประสิทธิภาพ เครื่องจักรกลวินิจฉัยยังมีส่วนช่วยปรับปรุงฐานความรู้ในบางระบบอีกด้วย การทำงานของเครื่องจักรกลวินิจฉัยนั้นประกอบด้วยงานหลักที่สำคัญ 3 ประการ คือ (2) (8) (10)

- การควบคุมการทำงาน (Control) ทำหน้าที่ในการกำหนดลำดับและทิศทางของการวินิจฉัย
- การให้เหตุผล (Reasoning) ทำการตรวจสอบข้อเท็จจริงและกฎที่มีอยู่ในระบบ แล้วนำข้อเท็จจริงและกฎที่เหมาะสมไปใช้ในการวินิจฉัยและทำการเพิ่มเติมข้อเท็จจริงใหม่ ๆ ที่ได้จากการวินิจฉัยเข้าไปในระบบ รวมทั้งการสร้าง หรือปรับปรุงฐานความรู้ในโครงระบบเชี่ยวชาญบางระบบอีกด้วย
- การไกล่เกลี่ยความขัดแย้ง (Conflict resolution) ทำการคัดเลือกกฎที่มีความเหมาะสมมากที่สุดไปทำการประมวลผล เมื่อเกิดความขัดแย้งขึ้น คือมีกฎที่มีความเหมาะสมพร้อมกันมากกว่า 1 กฎ ในรอบของการทำงานรอบหนึ่ง ๆ

1. <u>กลวิธีควบคุมการทำงาน (Control strategy)</u> (2) (7)

วิธีการต่าง ๆ ที่ใช้ในการควบคุมการทำงานนั้นได้แก่

- การวินิจฉัยแบบย้อนหลัง (Backward chaining inference)
- การวินิจฉัยแบบไปข้างหน้า (Forward chaining inference)

ก. <u>การวินิจฉัยแบบย้อนหลัง (Backward chaining inference)</u>

การวินิจฉัยแบบย้อนหลังนี้ เครื่องจักรกลวินิจฉัยจะต้องมีการกำหนดเป้าหมายหลักหรือสมมติฐานหลักเอาไว้จำนวนหนึ่งก่อน แล้วเริ่มต้นทำงานจากกฎที่มีส่วนผลสรุปเป็นเป้าหมายหลักหรือสมมติฐานหลักที่วางไว้ จากนั้นกฎอื่น ๆ ในระบบก็จะถูกค้นหา เพื่อให้ได้กฎที่มีส่วนผลสรุปสามารถยืนยันความถูกต้องของส่วนเงื่อนไขของกฎแรกนั้นได้ ในกรณีที่ส่วนเงื่อนไขของกฎที่ค้นหามาได้นั้น มีบางส่วนที่ยังไม่ทราบค่า ส่วนผลสรุปของกฎนี้ก็จะกลายมาเป็นเป้าหมายย่อย ระบบก็จะทำการค้นหาต่อไปอีก เพื่อให้ได้กฎที่สามารถยืนยันความถูกต้องของกฎเป้าหมายย่อยได้ ในบางครั้งระบบจะถามผู้ใช้ถึงค่าที่ต้องการโดยตรงเลยก็ได้ กระบวนการเช่นนี้จะถูกกระทำซ้ำแล้วซ้ำอีก เกิดเป้าหมายย่อยขึ้นอีกมากมาย จนกระทั่งส่วนเงื่อนไขของกฎสมมติฐานหลัก ซึ่งมีได้หลายประโยคนั้นจะได้รับการยืนยันจากกฎอื่น ๆ ว่า มีความถูกต้องสมบูรณ์ครบทุกส่วน สมมติฐานหลักนั้นก็เป็นจริง หรือจนกระทั่งพบข้อสรุปใหม่ว่า สมมติฐานหลักที่วางไว้นั้นไม่เป็นจริง หรือพบจุดสิ้นสุดของการวินิจฉัยที่ไม่สามารถจะทำการวินิจฉัยต่อไปได้ สมมติฐานหลักนั้นก็ไม่เป็นจริง ในกรณีเช่นนี้ เครื่องจักรกลวินิจฉัยอาจจะเริ่มต้นใหม่จากสมมติฐานหลักอันใหม่ไปจนกระทั่งได้สมมติฐานที่เป็นจริง หรืออาจจะเลือกที่จะหยุดทำการวินิจฉัยเลยก็ได้ การวินิจฉัยแบบย้อนหลังนี้ ในบางครั้งจิงเรียกว่า hypothesis-driven inference หรือ goal-driven inference หรือ top-down inference

การวินิจฉัยแบบย้อนหลังมีความเหมาะสมกับปัญหาในลักษณะที่สามารถทราบได้ก่อนล่วงหน้าว่า คำตอบสำหรับปัญหาที่เกิดขึ้นสามารถจะเป็นอะไรได้บ้าง และมีจำนวนไม่มากนัก เนื่องจากการทำงานจะเริ่มต้นโดยการพิสูจน์คำตอบหรือสมมติฐานที่เป็นไปได้มากที่สุด ทำให้สามารถพบคำตอบที่ถูกต้องได้อย่างรวดเร็ว และมีประสิทธิภาพมากที่สุด

ข. <u>การวินิจฉัยแบบไปข้างหน้า (Forward chaining inference)</u>

สำหรับการวินิจฉัยแบบไปข้างหน้านี้ ระบบไม่ได้เริ่มต้นที่เป้าหมายใดเป้าหมายหนึ่งโดยเฉพาะ นั่นคือไม่มีกฎใดในระบบที่เป็นจุดตั้งต้นของการวินิจฉัย การวินิจฉัยแบบไปข้างหน้านี้จะเริ่มต้นจากกลุ่มของหลักฐานหรือข้อมูลที่ผู้ใช้ป้อนเข้าไปในระบบ หรือจากสภาพการณ์ปัจจุบันของระบบซึ่งเก็บไว้ในฐานข้อมูลส่วนกลางอันหนึ่ง คือหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบ (working memory) ข้อมูลหรือสภาพการณ์ต่าง ๆ เหล่านี้จะถูกนำไปจับคู่กับส่วนเงื่อนไขของกฎ เพื่อให้ได้กฎที่สอดคล้องกับสภาพการณ์ปัจจุบันของระบบ ซึ่งโดยปกติแล้วจะมีกฎมากกว่า 1 กฎ ที่สามารถจับคู่ได้พอดีกับข้อมูล หรือสภาพการณ์ต่าง ๆ เหล่านี้ กฎควบคุม (meta rule) หรือกลวิธีไกล่เกลี่ยความขัดแย้ง จะทำการคัดเลือกกฎที่มีความเหมาะสมมากที่สุดไปทำการประมวลผล ส่วนผลสรุปหรือส่วนกระทำของกฎนั้นก็จะทำการเพิ่มเติมข้อมูลใหม่ ๆ ลงในหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบ หรือทำการเรียกกระบวนการหรือวิธีการใด ๆ ให้ทำงาน ซึ่งก็มีผลให้สภาพการณ์ของระบบเปลี่ยนไปเช่นกัน การวินิจฉัยแบบไปข้างหน้านี้จะกระทำกระบวนการเช่นนี้ซ้ำไปเรื่อย ๆ จนกระทั่งได้ผลหรือข้อสรุปที่ต้องการ หรือจนกระทั่งไม่มีกฎใดในระบบที่สามารถจับคู่ได้พอดีกับสภาพการณ์ในขณะนั้นของระบบ ระบบก็จะหยุดทำการวินิจฉัย การวินิจฉัยแบบไปข้างหน้านี้บางครั้งจึงเรียกว่า data-driven inference หรือ event-driven inference หรือ bottom-up inference

ตรงกันข้ามกับการวินิจฉัยแบบย้อนหลัง การวินิจฉัยแบบไปข้างหน้ามีความเหมาะสมกับปัญหาในลักษณะที่ ไม่สามารถทราบได้ล่วงหน้าเลยว่า คำตอบที่เป็นไปได้มีอะไรบ้าง ซึ่งทำให้ไม่สามารถวินิจฉัยแบบย้อนหลังได้ เพราะการวินิจฉัยแบบย้อนหลังจะต้องเริ่มต้นโดยการเลือกเอาคำตอบที่เป็นไปได้มากที่สุดมาเป็นสมมติฐาน แล้วทำการพิสูจน์สมมติฐานนี้ ในกรณีที่สามารถทราบได้ก่อนว่า คำตอบจะเป็นอะไรได้บ้าง แต่จำนวนคำตอบที่เป็นไปได้นั้นมีจำนวนมาก การวินิจฉัยแบบย้อนหลังก็จะไม่มีประสิทธิภาพเท่าการวินิจฉัยแบบไปข้างหน้า เพราะจะทำให้เสียเวลามากก่อนที่จะพิสูจน์ไปถึงสมมติฐานที่ถูกต้อง การแก้ไขปัญหาในลักษณะนี้จะมีประสิทธิภาพมากที่สุด ถ้าใช้การวินิจฉัยทั้งแบบไปข้างหน้าและย้อนหลังร่วมกันทั้ง 2 วิธี การวินิจฉัยแบบไปข้างหน้าจะช่วยทำให้ปริมาณการค้นหาลดน้อยลง ในขณะที่การวินิจฉัยแบบย้อนหลังจะช่วยให้ไปถึงคำตอบที่ถูกต้องได้เร็วขึ้น (8)

2. วิธีการให้เหตุผล (Method of reasoning) (5) (8)

วิธีการให้เหตุผลที่มีใช้ในโครงระบบเชี่ยวชาญทั่ว ๆ ไปนั้นได้แก่

- การให้เหตุผลในระบบโพรดักชัน (Production)
- Object-oriented programming
- ระบบกระดานดำ (Blackboard)
- ระบบ induction
- การให้เหตุผลแบบไม่แน่นอน (Inexact reasoning)

ก. การให้เหตุผลในระบบโพรดักชัน (Production)

ระบบโพรดักชันประกอบด้วยกลุ่มของกฎที่เรียกว่าโพรดักชัน (production) ซึ่งอยู่ในรูปแบบของกฎทั่ว ๆ ไปคือ

ส่วนเงื่อนไข --> ส่วนกระทำ

หน่วยความจำของระบบโพรดักชัน จะแบ่งออกเป็น 2 ส่วน คือ หน่วยความจำโพรดักชัน (production memory) หรือ PM และหน่วยความจำชั่วคราว (working memory) หรือ WM หน่วยความจำชั่วคราวจะเก็บข้อมูลหรือข้อเท็จจริงต่าง ๆ ของระบบในแต่ละขณะไว้ ส่วนหน่วยความจำโพรดักชันจะเก็บโพรดักชันทั้งหมดในระบบ ซึ่งแต่ละโพรดักชันก็จะประกอบด้วยส่วนเงื่อนไขและส่วนปฏิบัติ แต่ละส่วนก็อาจจะประกอบด้วยประโยคย่อยอีกหลายประโยค โพรดักชันอาจแทนได้ด้วยสัญญลักษณ์ดังต่อไปนี้

$$P_j \ (C_1 \ C_2 \ \ldots \ C_n \ \text{--->} \ A_1 \ A_2 \ \ldots \ A_m)$$

ส่วนกระทำคือ A_1 A_2 จนถึง A_m ส่วนใหญ่จะเป็นการแก้ไขปรับปรุงหน่วยความจำชั่วคราว โดยการเพิ่ม การลบ และการเปลี่ยนแปลงข้อมูลเดิมที่มีอยู่ การทำงานของระบบโพรดักชันนี้ โดยหลักการพื้นฐานจะทำการจับคู่ระหว่างข้อเท็จจริงในหน่วยความจำชั่วคราว และโพรดักชันในหน่วยความจำโพรดักชัน โพรดักชันใดสามารถจับคู่ได้พอดีกับข้อมูลในหน่วยความจำชั่วคราวจะเรียกการจับคู่ระหว่างข้อเท็จจริงและโพรดักชันนั้นว่า Instantiation ส่วนกระทำของโพรดักชันนั้นก็จะถูกนำมาทำการประมวลผล ซึ่งจะทำให้หน่วยความจำชั่วคราวเปลี่ยนไปและเกิด Instantiation ใหม่ ๆ ขึ้นตามมาอีก การทำ

016795

งานในลักษณะวนรอบซ้ำเช่นนี้เรียกว่าวงจรค้นหาและกระทำ (recognise-act cycle) การค้นหาจะหา Instantiation ที่เหมาะสมมาประมวลผล ส่วนการกระทำก็จะทำการประมวลผลส่วนกระทำของ Instantiaiton นั้น การค้นหายังสามารถแบ่งออกได้เป็น 2 ขั้นคือ การจับคู่ระหว่างข้อเท็จจริงในหน่วยความจำชั่วคราวและโพรดักชันในหน่วยความจำโพรดักชัน (matching) และการไกล่เกลี่ยความขัดแย้ง (conflict resolution) ที่ต้องใช้เมื่อทีหลาย ๆ Instantiation เกิดขึ้นในรอบของการทำงานรอบเดียวกัน

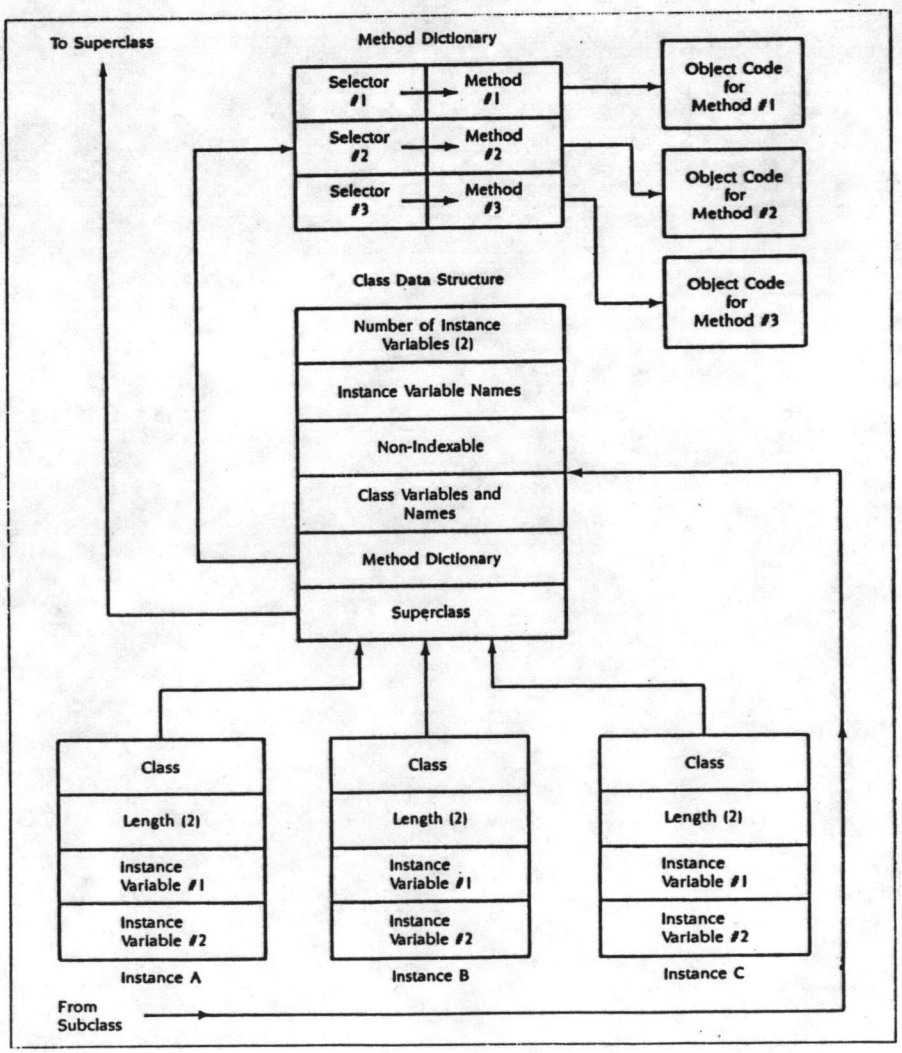
ระบบโพรดักชันนี้เหมาะสำหรับการแทนค่าความรู้โดยใช้กฎที่มีจำนวนกฎความรู้ไม่มากนักคือไม่เกิน 1000 กฎ ซึ่งอาจจะใช้การจัดหมวดหมู่ของกฎอย่างมีโครงสร้างร่วมด้วยก็ได้

ข. Object-oriented programming (11)

ในระบบ object-oriented นี้ ความรู้ต่าง ๆ ในระบบจะถูกแทนค่าให้อยู่ในรูปของวัตถุทั้งหมด วัตถุเหล่านี้จะสามารถติดต่อสื่อสารถึงกันได้โดยอาศัยการส่งข่าวสาร (message) ถึงกัน แต่ละวัตถุจะมีกระบวนการ (procedure) หรือวิธีการ (method) ต่าง ๆ เก็บไว้ กระบวนการเหล่านี้จะถูกกระตุ้นให้ทำงานต่อเมื่อวัตถุนั้นได้รับข่าวสารที่เหมาะสม ซึ่งการทำงานของแต่ละกระบวนการนี้จะรวมถึงการส่งข่าวสารไปยังวัตถุอื่น ๆ ในระบบต่อไปอีกด้วย ลักษณะโครงสร้างข้อมูลของวัตถุในระบบ object-oriented นี้สามารถแสดงได้ดังรูปที่ 2.1

ประเภทของวัตถุแต่ละประเภท รวมทั้งคุณสมบัติในด้านต่าง ๆ และกระบวนการ หรือวิธีการต่าง ๆ สำหรับวัตถุประเภทนั้น จะถูกเก็บไว้ในโครงสร้างข้อมูลชนิดหนึ่ง ซึ่งเรียกว่าคลาส (class) คลาสประกอบด้วยข้อมูลที่จำเป็นต้องใช้ในการสร้างและการเรียกใช้วัตถุทุกตัวที่ถูกจัดอยู่ภายใต้ประเภทนี้ ข้อมูล 2 อย่างแรกที่จำเป็นต้องมีคือ จำนวนตัวแปรของวัตถุ และชื่อของตัวแปรเหล่านั้น วัตถุทุกตัวที่ถูกจัดอยู่ในประเภทเดียวกันนี้ จะต้องมีจำนวนตัวแปรเท่ากัน และชื่อของตัวแปรทุกตัวเหมือนกัน สิ่งที่จะแตกต่างกันไปในแต่ละวัตถุก็คือ ค่าของตัวแปรต่าง ๆ เหล่านี้ ซึ่งจะต้องถูกเก็บไว้ที่แต่ละวัตถุ การที่ต้องเก็บชื่อของตัวแปรวัตถุไว้ในโครงสร้างข้อมูลคลาสก็เพื่อให้กระบวนการต่าง ๆ ของวัตถุประเภทนี้ สามารถเรียกใช้ชื่อตัวแปรต่าง ๆ เหล่านี้ได้ เมื่อต้องการอ้างถึงตัวแปรวัตถุ กระบวนการดังกล่าวก็จะถูกเก็บไว้ที่โครงสร้างข้อมูลคลาสนี้ด้วยเช่นกัน โดยจะเก็บเป็นตัวชี้ของพจนานุกรมวิธีการ (method dictionary) ไว้ภายในคลาสอีกทีหนึ่ง ซึ่งพจนานุกรมวิธีการนี้จะ

มีลักษณะเป็นการจับคู่ระหว่างตัวรับข่าวสาร (message receiver) หรือตัวเลือก (selector) กับวิธีการต่าง ๆ เมื่อข่าวสารถูกส่งมายังวัตถุใดวัตถุหนึ่ง ระบบก็จะค้นหาภายในพจนานุกรมวิธีการของคลาสนั้น เพื่อนำวิธีการที่เหมาะสมมาทำการประมวลผล นอกเหนือจากตัวแปรวัตถุแล้ว ยังมีตัวแปรอีกจำพวกหนึ่งคือตัวแปรคลาส ซึ่งชื่อของตัวแปรคลาสและค่าของมันก็จะถูกเก็บไว้ที่คลาสนี้ด้วยเช่นกัน ตัวแปรคลาสเป็นตัวแปรที่ใช้ร่วมกันทุก ๆ วัตถุที่ถูกจัดอยู่ภายใต้คลาสนี้ ซึ่งนอกจากจะเป็นการประหยัดหน่วยความจำแล้ว ยังช่วยให้ง่ายต่อการเปลี่ยนแปลงในภายหลังอีกด้วย



รูปที่ 2.1 โครงสร้างข้อมูลคลาส

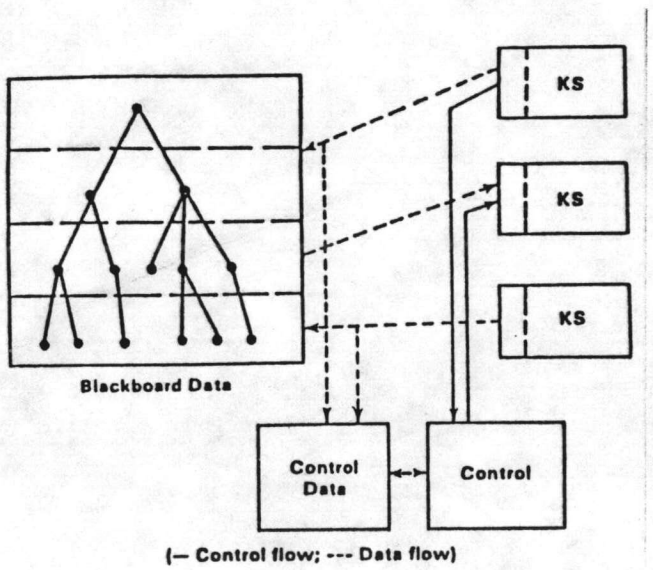
ข้อมูลอีกประเภทหนึ่งที่เก็บไว้ในโครงสร้างข้อมูลคลาสเป็นข้อมูลที่เกี่ยวข้องกับการจัดระดับชั้นของคลาสต่าง ๆ ภายในระบบ นั่นก็คือตัวชี้ไปยังคลาสใหญ่ (superclass) ของคลาสนี้อีกทีหนึ่ง คลาสใหญ่เป็นคลาสอีกคลาสหนึ่งซึ่งมีลักษณะโครงสร้างข้อมูลทุกอย่างเหมือนกับคลาส นั่นคือมีตัวแปรวัตถุ ตัวแปรคลาส พจนานุกรมวิธีการและตัวชี้ไปยังคลาสใหญ่อีกคลาสหนึ่งเช่นเดียวกัน การถ่ายทอดคุณสมบัติต่าง ๆ ก็สามารถเกิดขึ้นได้จากคลาสในระดับสูงไปยังคลาสในระดับต่ำ ยกตัวอย่างเช่น ในระหว่างการส่งข่าวสารนั้น เมื่อข่าวสารหนึ่งถูกส่งมายังวัตถุหนึ่งของคลาส ก พจนานุกรมวิธีการของคลาส ก ก็จะถูกค้นหา ถ้าพจนานุกรมวิธีการของคลาส ก ไม่มีตัวรับข่าวสารหรือตัวเลือกใดที่สามารถจับคู่ได้พอดีกับข่าวสารที่ถูกส่งมา พจนานุกรมวิธีการของคลาส ข ซึ่งเป็นคลาสใหญ่ของคลาส ก ก็จะถูกค้นหาเป็นลำดับต่อไป และถ้ายังไม่พบวิธีการที่เหมาะสม พจนานุกรมวิธีการของคลาสใหญ่ของคลาส ข ก็จะถูกค้นหาต่อไปอีก ดังนี้ไปเรื่อย ๆ จนกระทั่งพบวิธีการที่ต้องการ หรือจนกว่าจะไปถึงคลาสที่อยู่สูงสุดซึ่งไม่มีคลาสใหญ่อีกแล้ว ในกรณีนี้จะถือว่าเป็นข้อผิดพลาดประการหนึ่ง วัตถุแต่ละวัตถุก็จะมีตัวชี้ไปยังคลาสของมัน และมีข้อมูลอีกพวกหนึ่งที่เก็บไว้ที่วัตถุนั่นคือค่าของตัวแปรวัตถุ ดังที่ได้กล่าวไปแล้ว

ข้อดีของระบบ object-oriented ก็คือ สามารถรวบรวมข้อมูล และวิธีการต่าง ๆ ไว้ภายในหน่วยเดียวกันได้ (encapsulation) สามารถถ่ายทอดคุณสมบัติถึงกันได้ (inheritance) มีความสามารถที่จะใช้ข่าวสารเดียวกัน สำหรับวัตถุต่างชนิดกัน และทำให้เกิดผลเช่นเดียวกันได้ แต่วิธีการที่ใช้ต่างกันในแต่ละวัตถุ (polymorphism) สำหรับข้อเสียของระบบ object-oriented นี้ก็มีอยู่บ้างกล่าวคือ การส่งข่าวสารมักจะกินเวลานานกว่าการเรียกใช้โปรแกรมย่อยโดยตรง แต่หากโปรแกรมมีขนาดใหญ่และมีความซับซ้อนพอสมควร เวลาที่ใช้จะไม่ต่างกันมากนัก แต่ระบบ object-oriented จะมีประสิทธิภาพมากกว่า

ค. <u>ระบบกระดานดำ (Blackboard)</u> (12) (13)

กระดานดำ (blackboard) เป็นโครงสร้างที่ได้รับการออกแบบขึ้นมา เพื่อให้ใช้กับระบบเชี่ยวชาญที่ต้องการความรู้เป็นจำนวนมาก จากหลาย ๆ แขนงรวมกันในการแก้ปัญหา ในระบบกระดานดำนี้ หน่วยความจำที่ใช้เก็บสภาพการณ์ชั่วคราวของระบบ ซึ่งเรียกว่ากระดานดำนี้ จะถูกแบ่งออกเป็นระดับชั้นต่าง ๆ แต่ละระดับก็จะเก็บข้อมูลต่าง ๆ เกี่ยวกับเรื่องใดเรื่องหนึ่งโดยเฉพาะ และสามารถเชื่อมโยงถึงกันได้ในระดับชั้นที่อยู่ติดกัน ความรู้ใน

ฐานความรู้ก็จะถูกแบ่งออกเป็นส่วน ๆ เช่นเดียวกัน โดยที่แต่ละส่วนนั้นจะเรียกว่าแหล่งความรู้ (knowledge source) แหล่งความรู้จะถูกเก็บแยกจากกันและเป็นอิสระจากกัน แหล่งความรู้แต่ละแหล่งจะเก็บความรู้ในแต่ละแขนงไว้ภายใน เป็นความรู้ความเชี่ยวชาญพิเศษเฉพาะทางแตกต่างกันไป และจะตอบสนองต่อการเปลี่ยนแปลงภายในกระดานดำที่เหมาะสม โดยการทำงานของกระบวนการ หรือวิธีการต่าง ๆ ที่เก็บไว้ภายในแหล่งความรู้นั้น ซึ่งการทำงานส่วนหนึ่งนั้นจะเกี่ยวข้องกับการเปลี่ยนแปลงข้อมูลภายในกระดานดำ แหล่งความรู้แต่ละแหล่ง จะสามารถเปลี่ยนแปลงค่าภายในกระดานดำได้ที่ระดับชั้นใดระดับชั้นหนึ่งเท่านั้น แต่อาจจะอาศัยข้อมูลจากระดับชั้นอื่น ๆ ได้ทุกระดับชั้น การติดต่อสื่อสารและปฏิกิริยาต่าง ๆ ระหว่างกันของแต่ละแหล่งความรู้จะกระทำโดยใช้กระดานดำเป็นตัวกลาง



รูปที่ 2.2 โครงสร้างของระบบกระดานดำ

การควบคุมการทำงานของระบบกระดานดำ เป็นหน้าที่ของส่วนควบคุม (control) ซึ่งเป็นส่วนประกอบที่สำคัญอีกส่วนหนึ่งของระบบกระดานดำนอกเหนือจากแหล่งความรู้และกระดานดำ ดังที่ได้กล่าวไปแล้วในตอนต้น แหล่งความรู้จะตอบสนองต่อการเปลี่ยนแปลงภายในกระดานดำที่เหมาะสม ซึ่งจะอยู่ภายใต้ความควบคุมของส่วนควบคุมการทำงานอีกทีหนึ่ง ส่วนควบคุมการทำงานจะคอยติดตามการเปลี่ยนแปลงของกระดานดำและตัดสินใจว่า แหล่งความรู้แหล่งไหนควรจะทำงาน หรือกระดานดำในระดับชั้นใดควรจะได้รับการ

เปลี่ยนแปลงแก้ไข โดยใช้ข้อมูลที่เกี่ยวข้องกับการควบคุมการทำงาน (control information) ซึ่งอาจจะเก็บไว้ภายในกระดานดำ หรือเก็บแยกจากกระดานดำก็ได้ ตัวอย่างในรูปที่ 2.2 จะเห็นว่า ข้อมูลที่เกี่ยวข้องกับการควบคุมการทำงานนี้ถูกเก็บไว้ในหน่วยความจำอีกส่วนหนึ่ง ซึ่งแยกออกมาจากกระดานดำ และถูกเรียกใช้โดยส่วนควบคุมการทำงาน การเปลี่ยนแปลงใด ๆ ที่แหล่งความรู้กระทำต่อกระดานดำ ก็จะถูกเก็บบันทึกเอาไว้ในหน่วยความจำส่วนนี้ด้วยเช่นกัน เพื่อเป็นข้อมูลช่วยในการตัดสินใจของส่วนควบคุม ส่วนควบคุมจะต้องตัดสินใจเลือกจุดรวมของความสนใจ (focus of attention) และเลือกการทำงานที่เหมาะสม จุดรวมของความสนใจอาจจะเป็นได้ทั้งแหล่งความรู้ (แหล่งความรู้แหล่งไหนควรจะทำงาน) หรือกระดานดำ (วัตถุใดในกระดานดำที่ระดับชั้นใดควรจะได้รับการเปลี่ยนแปลงแก้ไข) หรือทั้งแหล่งความรู้และกระดานดำ (แหล่งความรู้แหล่งไหนควรจะทำงาน และควรจะกระทำกับวัตถุใดในกระดานดำ) ตัวอย่างของระบบกระดานดำได้แก่ ระบบ HEARSAY-II และระบบ HASP HEARSAY-II เป็นระบบเข้าใจความหมายของคำพูดที่สามารถเข้าใจคำถามต่าง ๆ ในทางวิทยาศาสตร์คอมพิวเตอร์ และตอบคำถามนั้นโดยใช้ข้อมูลที่เก็บไว้ในฐานข้อมูล สำหรับ HASP นั้นเป็นระบบแปลความหมายข้อมูลทางสมุทรศาสตร์ที่ซับซ้อน

กระดานดำมีความเหมาะสมกับปัญหาที่มีลักษณะซับซ้อนและมีโครงสร้างที่ไม่ดี (complex and ill-structured problem) ปัญหาที่มีลักษณะซับซ้อนก็คือปัญหาที่ประกอบด้วยปัญหาย่อย ๆ อีกจำนวนมาก ซึ่งมีปฏิกิริยาระหว่างกันแตกต่างกันไปในแต่ละปัญหา ระบบกระดานดำจะช่วยแยกความรู้สำหรับใช้ในการแก้ไขปัญหาแต่ละปัญหาออกเป็นแหล่งความรู้ต่าง ๆ และใช้กระดานดำเป็นตัวกลางสำหรับปฏิกิริยาระหว่างกันของแต่ละปัญหา สำหรับปัญหาที่มีโครงสร้างไม่ดีนั้น ได้แก่ปัญหาที่ไม่สามารถกำหนดลำดับขั้นตอนของการแก้ไขปัญหาของแต่ละปัญหาย่อยไว้ล่วงหน้าได้ เนื่องจากต้องขึ้นอยู่กับปัญหาย่อยปัญหาอื่นด้วย การทำงานของระบบกระดานดำจะตัดสินใจเลือกลำดับของการทำงานที่เหมาะสมในขณะทำการประมวลผล จึงสามารถแก้ไขปัญหาในลักษณะนี้ได้อย่างถูกต้องเหมาะสมที่สุด

ถึงแม้ว่าระบบกระดานดำจะมีประโยชน์ และเหมาะสมกับปัญหาที่มีลักษณะซับซ้อนและมีโครงสร้างที่ไม่ดี แต่ระบบกระดานดำก็มีค่าใช้จ่ายสูงในการสร้าง และการทำงานในแต่ละครั้ง ดังนั้นปัญหาอื่น ๆ ที่สามารถใช้วิธีการอื่นที่มีค่าใช้จ่ายต่ำกว่าในการแก้ปัญหาได้ จึงควรหลีกเลี่ยงการใช้ระบบกระดานดำ

ง. ระบบ induction (14) (15)

ระบบการวินิจฉัยแบบ induction เป็นการดึงเอาความรู้จากข้อมูลจำนวนหนึ่งที่ป้อนเข้าไปในระบบ ออกมาสร้างเป็นกฎความรู้ที่สามารถใช้ได้กับข้อมูลทั่ว ๆ ไป เก็บไว้ภายในฐานความรู้ เป็นกระบวนการที่ทำงานจากกรณีเฉพาะ (specialize) เพื่อไปสู่กรณีทั่ว ๆ ไป (generalize) ที่สามารถใช้กับกรณีเฉพาะอื่น ๆ ได้ด้วย ระบบการวินิจฉัยแบบ induction นี้จะอนุญาตให้วิศวกรความรู้ใส่ตัวอย่างจำนวนหนึ่งเข้าไปในระบบ ระบบจะทำการวิเคราะห์ตัวอย่างเหล่านี้ แล้วสร้างเป็นกฎชั้นมาชุดหนึ่ง ซึ่งสอดคล้องกับตัวอย่างที่ป้อนเข้าไป นั่นคือเมื่อใช้กฎชุดนี้ในการวินิจฉัยจะได้ผลลัพธ์เช่นเดียวกับตัวอย่างที่ป้อนเข้าไปในครั้งแรกนั้น และยังใช้ในการวินิจฉัยกรณีอื่น ๆ ได้ด้วย ยกตัวอย่างเช่น จากตัวอย่างที่ป้อนเข้าไปในระบบ 2 ตัวอย่างดังต่อไปนี้คือ

ตัวอย่างที่ 1

Jack เป็นพ่อของ Bill
Bill เป็นลูกของ Jack
Tom เป็นพี่ชายของ Jack
Tom เป็นลุง Tom ของ Bill

ตัวอย่างที่ 2

Mary เป็นแม่ของ Jim
Jim เป็นลูกของ Mary
Tom เป็นพี่ชายของ Mary
Tom เป็นลุง Tom ของ Jim

ระบบจะทำการวิเคราะห์ตัวอย่างดังกล่าว แล้วสรุปออกมาเป็นกฎได้ว่า

กฎข้อที่ 1

ถ้า x เป็นแม่ของ y หรือ
x เป็นพ่อของ y
ดังนั้น y จะเป็นลูกของ x

กฎข้อที่ 2

ถ้า z เป็นพี่ชายของ x และ
y เป็นลูกของ x และ
z = Tom
ดังนั้น z จะเป็นลุง Tom ของ y

กฎ 1 ที่ได้จะเป็นกฎที่สามารถใช้ได้ทั่วไป แต่กฎ 2 ยังมีลักษณะเฉพาะอยู่บ้าง เป็นเพราะตัวอย่างที่ป้อนเข้าไปในระบบยังไม่มากพอ ทำให้กฎที่ได้ยังไม่สามารถใช้กับกรณีอื่น ๆ ได้ ถ้าเพิ่มตัวอย่างอีกตัวอย่างหนึ่งเข้าไปในระบบคือ

ตัวอย่างที่ 3 Sue เป็นลูกของ Jane
Jane มีพี่ชายชื่อ Fred
Fred เป็นลุง Fred ของ Sue

ระบบจะทำการวิเคราะห์ใหม่ และปรับปรุงกฎเดิมที่ได้ให้มีลักษณะทั่วไปมากขึ้น

กฎข้อที่ 2 ถ้า z เป็นพี่ชายของ x และ
y เป็นลูกของ x
ดังนั้น z จะเป็นลุงของ y

ซึ่งกฎใหม่นี้จะเป็นกฎที่สามารถใช้ได้ทั่ว ๆ ไป ดังนั้นระบบ induction นี้จะใช้การได้ก็ต่อเมื่อมีตัวอย่างที่สมบูรณ์มากพอและตัวอย่างทุกตัวอย่างได้ถูกป้อนเข้าไปในระบบเรียบร้อยแล้วเท่านั้น

วิธีการสร้างกฎความรู้ขึ้นมาจากตัวอย่างที่ป้อนเข้าไปในระบบนั้นมีอยู่ 2 วิธีคือ

1) ป้อนทีละตัวอย่าง การป้อนตัวอย่างเข้าไปในระบบจะต้องป้อนเข้าไปทีละตัวอย่างและลำดับของตัวอย่างที่ป้อนเข้าไปทีละตัวอย่างนั้นมีความสำคัญ ระบบจะรับเอาข้อมูลจากตัวอย่างเหล่านั้นเก็บไว้เป็นความรู้ในรูปแบบต่าง ๆ ในระบบแล้วทิ้งตัวอย่างนั้นไป เมื่อตัวอย่างอันใหม่ถูกป้อนเข้ามาระบบก็จะทำการปรับปรุง หรือเพิ่มเติมความรู้ในฐานความรู้นี้ให้ถูกต้องสมบูรณ์ยิ่งขึ้น จนในที่สุดก็จะได้ฐานความรู้ที่สมบูรณ์เมื่อตัวอย่างสุดท้ายได้ถูกป้อนเข้ามาในระบบ

2) ตัวอย่างทุก ๆ ตัวอย่างจะถูกป้อนเข้ามาในระบบพร้อมกันทีเดียวทั้งหมดโดยไม่ต้องมีลำดับ และจะไม่มีการทิ้งตัวอย่างใด ๆ ออกไปจากระบบเลย ระบบจะจัดการค้นหา จัดกลุ่ม และหาลักษณะที่มีร่วมกันภายในกลุ่ม เพื่อสร้างเป็นกฎความรู้ที่เข้ากับทุก ๆ ตัวอย่าง และสามารถใช้กับกรณีทั่วไปกรณีอื่น ๆ ได้ทุกกรณี

ระบบ induction นี้ วิศวกรความรู้จะต้องใช้เวลามากในการปรับปรุงและเพิ่มเติมตัวอย่างต่าง ๆ เข้าไปในระบบ เพื่อที่จะให้ระบบนั้นมีกฎความรู้ที่สมบูรณ์ที่สุด ระบบ induction จึงเหมาะกับงานที่ไม่มีความซับซ้อน และขนาดไม่ใหญ่มากนัก

จ. <u>การให้เหตุผลแบบไม่แน่นอน (Inexact reasoning)</u> (7)

ความรู้ต่าง ๆ ในความเป็นจริงที่ใช้กันอยู่ในชีวิตประจำวันนั้น มักจะมีความไม่แน่นอนหรือความไม่แน่ใจเกี่ยวข้องอยู่ด้วยเสมอ ความรู้ในฐานความรู้ของระบบเชี่ยวชาญก็เช่นกัน ดังนั้นการให้เหตุผลของเครื่องจักรกลวินิจฉัยจะต้องสามารถให้เหตุผลได้ จากความไม่แน่นอนที่มีอยู่ ทฤษฎีต่าง ๆ สำหรับใช้ในการให้เหตุผลแบบไม่แน่นอนได้ถูกพัฒนาขึ้น ซึ่งมีอยู่หลายทฤษฎี แต่ที่ใช้กันมากได้แก่ทฤษฎีความน่าจะเป็นของเบย์เซียน (Bayesian probability theory) และทฤษฎีความเชื่อมั่น (certainty theory)

1) ทฤษฎีความน่าจะเป็นของเบย์เซียน (Bayesian probability theory) ทฤษฎีนี้เป็นการนำเอาทฤษฎีความน่าจะเป็นพื้นฐานมาใช้กับการหาค่าความน่าจะเป็นของสมมติฐานหนึ่ง ๆ ซึ่งได้กำหนดหลักฐานบางประการที่เกี่ยวข้องกับสมมติฐานอันนั้นมาให้ ทฤษฎีความน่าจะเป็นพื้นฐานประกอบด้วยกฎต่าง ๆ ดังต่อไปนี้คือ

<u>กฎข้อที่ 1</u> ถ้า ความน่าจะเป็นของ A คือ pA และ
ความน่าจะเป็นของ B คือ pB และ
A และ B เป็นเหตุการณ์ที่เป็นอิสระต่อกัน
ดังนั้น ความน่าจะเป็นของ [A and B] จะเท่ากับ pA * pB

<u>กฎข้อที่ 2</u> ถ้า ความน่าจะเป็นของ A คือ pA
ดังนั้น ความน่าจะเป็นของ [not A] จะเท่ากับ 1.0 - pA

<u>กฎข้อที่ 3</u> ถ้า ความน่าจะเป็นของ A คือ pA และ
ความน่าจะเป็นของ B คือ pB และ
ความน่าจะเป็นของ [A and B] คือ p[A and B]
ดังนั้น ความน่าจะเป็นของ [A or B] จะเท่ากับ
pA + pB - p[A and B] ซึ่งเท่ากับ
pA + pB - (pA * pB)

สำหรับทฤษฎีของ Bayes นั้น ได้กำหนดไว้ว่าความน่าจะเป็นที่สมมติฐาน Hi จะเป็นจริง เมื่อมีหลักฐาน E นั้น จะสามารถคำนวณได้ดังนี้

$$p(Hi/E) = \frac{p(E/Hi) * p(Hi)}{\sum_{n=1->k} p(E/Hn) * p(Hn)}$$

โดยที่ p(E/Hi) คือความน่าจะเป็นที่จะปรากฏหลักฐาน E เมื่อสมมติฐาน Hi เป็นจริง ตัวอย่างเช่น E อาจจะเป็นอาการอย่างหนึ่งของโรค และ Hi ก็คือโรคชนิดหนึ่ง

p(Hi) คือความน่าจะเป็นที่สมมติฐาน Hi จะเป็นจริงโดยไม่ขึ้นอยู่กับหลักฐานใด ๆ เลย เช่น ถ้า Hi เป็นโรคชนิดหนึ่ง p(Hi) ก็คือความน่าจะเป็นหรือโอกาสที่ใครก็ตามจะเป็นโรคชนิดนี้ได้

k คือจำนวนของสมมติฐานที่เป็นไปได้ที่จะปรากฏหลักฐาน E ตัวอย่างเช่น จำนวนของโรคที่แสดงอาการ E

ทฤษฎีความน่าจะเป็นของเบย์เซียน มีข้อเสียอยู่บ้างตรงที่ การหาค่า p(Hi) นั้น ทำได้ยากในทางปฏิบัติ ค่าที่ได้ส่วนใหญ่เป็นเพียงค่าโดยประมาณเท่านั้น นอกจากนี้ค่าความน่าจะเป็นของสมมติฐานต่าง ๆ ในระบบจะมีความเกี่ยวข้องกันอยู่มากไม่เป็นอิสระต่อกัน ทำให้ยากต่อการเปลี่ยนแปลงแก้ไขฐานความรู้ ยกตัวอย่างเช่น ถ้ามีการเพิ่มโรคชนิดใหม่เข้าไปในระบบ และโรคใหม่นี้มีอาการอย่างหนึ่งซึ่งไปพ้องกับโรคอื่น ๆ ที่มีอยู่แล้วในระบบ ความน่าจะเป็นของโรคต่าง ๆ จากอาการดังกล่าว จะต้องถูกคำนวณใหม่ทั้งหมด เนื่องจากผลรวมของทุก ๆ สมมติฐานที่เป็นไปได้จากอาการดังกล่าว ซึ่งเป็นตัวหารในการคำนวณข้างต้นเปลี่ยนไป

2) ทฤษฎีความเชื่อมั่น (Certainty theory) ทฤษฎีความเชื่อมั่นใช้ค่าปัจจัยความเชื่อมั่น (certainty factor CF) ในการแสดงความแน่นอนของข้อสรุปของกฎหรือข้อเท็จจริงต่าง ๆ ปัจจัยความเชื่อมั่นหรือ CF นี้ จะมีค่าอยู่ระหว่าง -1 และ 1 CF มีค่าเท่ากับ 1 หมายถึงสามารถเชื่อมั่นได้ 100 % และ CF มีค่าเท่ากับ -1 ก็จะหมายถึง ไม่สามารถเชื่อได้ 100% พิจารณากฎต่อไปนี้, ถ้า A ดังนั้น X ปัจจัยความเชื่อมั่นของกฎ

นี้สามารถแทนได้ด้วยสัญญลักษณ์ CF(X/A) ปัจจัยความเชื่อมั่นของข้อสรุป X ซึ่งแทนด้วยสัญญลักษณ์ CF(X) ก็จะสามารถคำนวณได้ดังนี้

$$CF(X) = CF(X/A) * CF(A)$$

ในกรณีที่มีกฎหลาย ๆ กฎที่มีผลสรุปอย่างเดียวกันเช่น

<u>กฎข้อที่ 1</u> ถ้า A_1 ดังนั้น X ปัจจัยความเชื่อมั่น $CF(X/A_1)$
<u>กฎข้อที่ 2</u> ถ้า A_2 ดังนั้น X ปัจจัยความเชื่อมั่น $CF(X/A_2)$

จากกฎข้อที่ 1 จะได้ว่า $CF(X)_1 = CF(X/A_1) * CF(A_1)$
จากกฎข้อที่ 2 จะได้ว่า $CF(X)_2 = CF(X/A_2) * CF(A_2)$

ปัจจัยความเชื่อมั่นของข้อสรุป X จะต้องหามาจากกฎทั้ง 2 ข้อ ซึ่งสามารถหาค่าปัจจัยความเชื่อมั่นของข้อสรุป X ได้จากกฎต่อไปนี้

ก) ถ้า $CF(X)_1 > 0$ และ $CF(X)_2 > 0$

$$CF(X/A_1A_2) = CF(X)_1 + CF(X)_2 - (CF(X)_1 * CF(X)_2)$$

ข) ถ้า $CF(X)_1 < 0$ และ $CF(X)_2 < 0$

$$CF(X/A_1A_2) = CF(X)_1 + CF(X)_2 + (CF(X)_1 * CF(X)_2)$$

ค) ถ้า $CF(X)_1$ และ $CF(X)_2$ มีเครื่องหมายตรงข้ามกัน หรือผลลัพธ์ของ $CF(X)_1 * CF(X)_2$ มีค่า < 0

$$CF(X/A_1A_2) = \frac{CF(X)_1 + CF(X)_2}{1.0 - \text{minimum of } (|CF(X)_1|, |CF(X)_2|)}$$

ทั้งสองทฤษฎีที่กล่าวมานั้น ยังสามารถนำมาประยุกต์ใช้ร่วมกันได้ โดยการหาปัจจัยความเชื่อมั่นจากความน่าจะเป็นของกฎ ซึ่งมีวิธีการดังต่อไปนี้คือ

- ถ้า CF มีค่าเป็นบวก จะเป็นการวัดระดับความน่าเชื่อถือ โดยที่ CF จะมีค่าเท่ากับ 1 ถ้า P(X) = 1 นอกนั้นจะคำนวณได้จาก

$$CF(X/A) = \frac{\text{maximum of } (P(X/A), P(X)) - P(X)}{1 - P(X)}$$

- ถ้า CF มีค่าเป็นลบจะเป็นการวัดระดับความไม่น่าเชื่อถือ โดยที่ CF จะมีค่าเท่ากับ -1 ถ้า P(X) = 0 นอกนั้นจะคำนวณได้จาก

$$CF(X/A) = \frac{\text{minimum of } (P(X/A), P(X)) - P(X)}{P(X)}$$

ทฤษฎีความเชื่อมั่นมีข้อได้เปรียบกว่าทฤษฎีความน่าจะเป็นในหลาย ๆ ด้านเช่น ถึงแม้ว่าความน่าจะเป็นบางอย่างจะไม่สามารถหาค่าที่แท้จริงได้ แต่ทฤษฎีความเชื่อมั่นก็มีวิธีการคำนวณหาค่าปัจจัยความเชื่อมั่นที่มีหลักเกณฑ์และสามารถเปรียบเทียบกันได้ ปัจจัยความเชื่อมั่นมีค่าอยู่ระหว่าง -1 0 และ -1 เสมอ ซึ่งสามารถแสดงความหมายและแยกกลุ่มได้อย่างชัดเจน

อย่างไรก็ตาม ถึงแม้ว่าการให้เหตุผลแบบไม่แน่นอนจะมีความจำเป็นสำหรับปัญหาส่วนใหญ่เช่นระบบวินิจฉัยทั่ว ๆ ไป แต่ปัญหาบางปัญหาก็อาจจะไม่มีความจำเป็นต้องใช้การให้เหตุผลแบบไม่แน่นอนเลยก็ได้ ยกตัวอย่างเช่น ระบบคำนวณภาษี เป็นต้น เพราะฉะนั้นการพิจารณาวิธีการให้เหตุผลของเครื่องจักรกลวินิจฉัยจึงต้องคำนึงถึงลักษณะของงานประกอบด้วย

3. <u>กลวิธีไกล่เกลี่ยความขัดแย้ง (Conflict resolution strategy)</u> (9) (10)

ลักษณะการทำงานของเครื่องจักรกลวินิจฉัยส่วนมาก จะเป็นไปในลักษณะของการวนรอบคัดเลือกกฎ ที่มีส่วนเงื่อนไขสอดคล้องกับสภาพการณ์ของระบบ แล้วนำกฎที่คัดเลือกได้นี้ไปใช้งาน (recognise-act cycle) ซึ่งอาจมีกฎที่มีส่วนเงื่อนไขสอดคล้องกับสภาพการณ์ของระบบในขณะนั้น หลาย ๆ กฎในคราวเดียวกันได้ เรียกกฎทั้งหลายที่สอดคล้องกับสภาพการณ์ของระบบนี้รวมกันว่า conflict set กลวิธีไกล่เกลี่ยความขัดแย้งกฎจะทำการคัดเลือกกฎที่มีความเหมาะสมมากที่สุดมาทำการประมวลผล เพื่อให้ได้ผลสรุปที่ดีภายในระยะ

เวลาที่สั้นที่สุด โครงระบบเชี่ยวชาญแต่ละระบบก็มีกฎเกณฑ์ในการคัดเลือกกฎแตกต่างกันไป กฎเกณฑ์ต่าง ๆ ที่ใช้ในการคัดเลือกกฎที่มีความเหมาะสมมากที่สุดจาก conflict set มาทำการประมวลผลที่มีใช้อยู่ในโครงระบบเชี่ยวชาญทั่ว ๆ ไปนั้น ได้แก่

- คัดเลือกกฎโดยใช้ลำดับของการใส่กฎแต่ละกฎเข้าไปในฐานความรู้เป็นเกณฑ์ (Rule order)
- คัดเลือกกฎโดยใช้กรณีพิเศษเป็นเกณฑ์ (Special case)
- คัดเลือกกฎโดยใช้ระยะเวลาที่ข้อมูลแต่ละตัวถูกใส่เข้าไปในหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบเป็นเกณฑ์ (Recency)
- คัดเลือกกฎโดยใช้ประวัติของการนำกฎแต่ละกฎมาทำการประมวลผลเป็นเกณฑ์ (Rule firing history)
- คัดเลือกกฎโดยใช้ลำดับความสำคัญของกฎแต่ละกฎเป็นเกณฑ์ (Priority)
- คัดเลือกกฎโดยใช้ความซับซ้อนของกฎแต่ละกฎเป็นเกณฑ์ (Complexity)

ก. <u>คัดเลือกกฎโดยใช้ลำดับของการใส่กฎแต่ละกฎเข้าไปในฐานความรู้เป็นเกณฑ์ (Rule order)</u>

วิธีที่ใช้กันมากที่สุดก็คือ เลือกกฎที่พบเป็นกฎแรกมาทำการประมวลผล (first found) การคัดเลือกกฎแบบนี้เป็นวิธีที่ง่าย และรวดเร็วที่สุดสำหรับการทำงานในขั้นตอนของการวินิจฉัย แต่ในขั้นตอนของการสร้างฐานความรู้นั้น วิศวกรความรู้จะต้องให้ความสนใจเป็นพิเศษกับลำดับของกฎทุก ๆ กฎที่จะใส่เข้าไปในฐานความรู้ และสำหรับการปรับปรุงแก้ไขฐานความรู้ในภายหลังโดยการใส่กฎความรู้เพิ่มเติมเข้าไปนั้น ไม่สามารถกระทำได้ตามปกติโดยการใส่ต่อท้ายเข้าไปในฐานความรู้ แต่จะต้องมีการกำหนดแบบแผนสำหรับการใส่กฎใหม่ ๆ เพิ่มเติมเข้าไปในฐานความรู้ไว้ล่วงหน้า วิธีการหนึ่งได้ถูกเสนอขึ้นโดย Waterman ซึ่งจะทำการใส่กฎใหม่ ๆ เข้าไปข้างหน้ากฎแรกในฐานความรู้ที่มีส่วนเงื่อนไข หรือส่วนกระทำเหมือนกับกฎใหม่นั้น โดยอาศัยสมมติฐานที่ว่า กฎสองกฎที่มีส่วนเงื่อนไขหรือส่วนกระทำเหมือนกัน กฎสองกฎนั้นมีหน้าที่คล้ายคลึงกัน การใส่กฎใหม่โดยวิธีนี้จะไม่ทำให้ลำดับของกฎโดยส่วนรวม และลำดับของกฎเดิมที่มีอยู่เสียไป

ข. <u>คัดเลือกกฎโดยใช้กรณีพิเศษเป็นเกณฑ์ (Special case)</u>

หลักการของการคัดเลือกกฎโดยวิธีนี้ก็คือ ถ้ากฎ ก ได้รับการพิจารณาว่าเป็นกรณีพิเศษของกฎ ข กฎ ก จะได้รับการคัดเลือก และถ้ามีกฎ ค ที่ไม่ได้เป็นกรณีพิเศษของกฎใดและไม่มีกฎใดที่เป็นกรณีพิเศษของกฎ ค กฎ ค ก็จะได้รับการคัดเลือกเช่นกัน การคัดเลือกกฎโดยวิธีนี้จึงเป็นเพียงการป้องกันไม่ให้นำกฎที่เป็นกรณีทั่ว ๆ ไป ไปทำการประมวลผล ในขณะที่กรณีพิเศษของกฎนั้นก็ปรากฏอยู่ใน conflict set ด้วยเช่นกัน

ตัวอย่างในรูปที่ 2.3 จะใช้ช่วยในการพิจารณากรณีพิเศษและวิธีการคัดเลือกกฎแบบต่าง ๆ ซึ่งจะได้กล่าวต่อไป เครื่องหมาย '(' และ ')' แต่ละคู่แทนประโยคของส่วนเงื่อนไขของกฎ ซึ่งแต่ละประโยคเชื่อมต่อกันด้วยเครื่องหมาย ',' ซึ่งหมายถึง 'และ' อักษรตัวใหญ่ภายในวงเล็บหมายถึงค่าคงที่ และตัวอักษรตัวเล็กหมายถึงตัวแปร

Working Memory [(P S) (Q T) (P T) (R V) (Q S) (P V) .....
..... (W V) (W T)]

กฎ 1 IF (Q x),(P x) THEN ....
กฎ 2 IF (P S),(P x),(W x) THEN ....
กฎ 3 IF (x S),(x y),(W y),(R y),(Q S) THEN ....
กฎ 4 IF (Q S),NOT(U S),(P x),NOT(U V),NOT(U T) THEN ....

Instantiation ที่ได้จากกฎ 1 I_{11} [(Q T) (P T)]
I_{12} [(Q S) (P S)]
Instantiation ที่ได้จากกฎ 2 I_{21} [(P S) (P T) (W T)]
I_{22} [(P S) (P V) (W V)]
Instantiation ที่ได้จากกฎ 3 I_{31} [(P S) (P V) (W V) (R V) (Q S)]
Instantiation ที่ได้จากกฎ 4 I_{41} [(Q S) (P S)]
I_{42} [(Q S) (P T)]
I_{43} [(Q S) (P V)]

รูปที่ 2.3 ตัวอย่างของกฎ และ Instantiation

การพิจารณาว่า กฎใดจะเป็นกรณีพิเศษของกฎใดหรือไม่นั้น มีวิธีการพิจารณาอยู่หลายวิธีด้วยกัน ได้แก่

1) พิจารณาจากฐานความรู้ โดยอาศัยลักษณะโครงสร้างของส่วนเงื่อนไขของกฎในการพิจารณา ซึ่งกฎ ก จะเป็นกรณีพิเศษของกฎ ข ถ้า

ก) กฎ ก มีจำนวนประโยคของส่วนเงื่อนไขไม่น้อยกว่ากฎ ข

ข) ในกรณีที่ส่วนเงื่อนไขประโยคใด ๆ ของกฎ ข นั้นมีค่าคงที่ ส่วนเงื่อนไขของกฎ ก ที่สอดคล้องกันก็จะต้องมีค่าคงที่นั้นประกอบอยู่ด้วย

ค) กฎ ก และกฎ ข จะต้องไม่เหมือนกันทุกประการ

จากตัวอย่างในรูปที่ 2.3 กฎ 4 จะเป็นกรณีพิเศษของกฎ 1 ดังนั้น Instantiation ที่ได้จากกฎ 1 คือ I_{11} และ I_{12} จะถูกตัดออกไปจาก conflict set

2) พิจารณาจากหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบ คือสนใจข้อมูลในหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบที่ทำให้ส่วนเงื่อนไขของกฎเป็นจริง โดยที่ Instantiation I_s จะเป็นกรณีพิเศษของ Instantiation I_g ถ้า I_s มีข้อมูลทุก ๆ ตัวที่ปรากฏอยู่ใน I_g จากตัวอย่างในรูปที่ 2.3 I_{31} จึงเป็นกรณีพิเศษของ I_{12} I_{22} I_{41} และ I_{43} ดังนั้นโดยวิธีการนี้ I_{31} ซึ่งเป็นกรณีพิเศษ และ Instantiation อื่น ๆ ที่เหลือคือ I_{11} I_{21} และ I_{42} จึงได้รับการคัดเลือก

3) พิจารณาทั้งจากหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบ และฐานความรู้ วิธีการพิจารณาในข้อนี้จะคล้ายคลึงกับในข้อ 2) แต่จะพิจารณาจากลักษณะโครงสร้างของส่วนเงื่อนไขของกฎเพิ่มเติม โดยที่ Instantiation I_s จะเป็นกรณีพิเศษของ Instantiation I_g ถ้า

ก) I_s มีข้อมูลทุก ๆ ตัวที่ปรากฏอยู่ใน I_g

ข) และกฎที่ทำให้เกิด I_s นั้น จะต้องมีจำนวนของประโยคเงื่อนไขมากกว่ากฎที่ทำให้เกิด I_g

จากตัวอย่างในรูปที่ 2.3 I_{31} ซึ่งเป็นกรณีพิเศษของ I_{41} และ I_{43} ถ้าพิจารณาตามเงื่อนไขในข้อ 2) จะไม่เป็นกรณีพิเศษของ I_{41} และ I_{43} อีกต่อไป ถ้าพิจารณาตามเงื่อนไขในข้อ 3) นี้ เพราะกฎ 4 มีจำนวนประโยคเงื่อนไขเท่ากับกฎ 3 (ไม่มากกว่า) ดังนั้นนอกเหนือจาก I_{31} I_{11} I_{21} และ I_{42} ซึ่งได้รับการคัดเลือกจากข้อ 2) แล้ว I_{41} และ I_{43} ก็จะได้รับการคัดเลือกด้วยเช่นกันในข้อนี้

การคัดเลือกกฎโดยใช้กรณีพิเศษนี้มักใช้ร่วมกับการคัดเลือกกฎโดยวิธีอื่น และไม่ค่อยได้รับความนิยมเท่าไรในระบบเชี่ยวชาญทั่ว ๆ ไป เพราะจำนวนกฎที่จะมีกรณีพิเศษในระบบเชี่ยวชาญหนึ่ง ๆ มีน้อย และการคัดเลือกกฎโดยวิธีนี้เป็นเพียงการตัดกรณีทั่ว ๆ ไปออกจาก conflict set เท่านั้น

ค. <u>คัดเลือกกฎโดยใช้ระยะเวลาที่ข้อมูลแต่ละตัวถูกใส่เข้าไปในหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบเป็นเกณฑ์ (Recency)</u>

ส่วนใหญ่จะเลือกข้อมูลที่เพิ่งถูกใส่เข้าไปในหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบ (more recent) มากกว่าข้อมูลที่ถูกใส่เข้าไปนานแล้ว (less recent) วิธีการคัดเลือกกฎโดยใช้หลักเกณฑ์นี้มีหลายวิธีด้วยกันคือ

1) พิจารณาข้อมูลแต่ละตัวในแต่ละ Instantiation แล้วเลือกข้อมูลที่ใหม่ที่สุดของแต่ละ Instantiation มาทำการเปรียบเทียบกัน Instantiation ที่มีข้อมูลที่ใหม่ที่สุดจะได้รับการคัดเลือก จากตัวอย่างในรูปที่ 2.3 ถ้ากำหนดให้ข้อมูลในหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบเรียงลำดับจากข้อมูลที่ใหม่ที่สุดคือ (P S) ไปหาข้อมูลที่เก่าที่สุดคือ (W T) จะมีบาง Instantiation ที่ประกอบด้วย (P S) ซึ่งเป็นข้อมูลที่ใหม่ที่สุดในหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบ Instantiation เหล่านั้นซึ่งได้แก่ I_{12} I_{21} I_{22} I_{31} และ I_{41} ก็จะได้รับการคัดเลือก

2) พิจารณาข้อมูลแต่ละตัวในแต่ละ Instantiation คล้ายกับในหัวข้อ 1) แต่จะเลือกข้อมูลที่เก่าที่สุดของแต่ละ Instantiation มาทำการเปรียบเทียบกัน ข้อมูลที่เก่าที่สุดของ Instantiation ไหนที่ใหม่ที่สุด Instantiation นั้นก็จะได้รับการคัดเลือก จากตัวอย่างข้อมูลที่เก่าที่สุดของ I_{11} คือ (P T) ใหม่กว่าข้อมูลที่เก่าที่สุดของ Instantiation อื่น ๆ ทุก Instantiation ดังนั้น I_{11} จึงได้รับการคัดเลือก

3) พิจารณา และจัดลำดับของข้อมูลทุกตัวในแต่ละ Instantiation จากใหม่ไปเก่าแล้วนำข้อมูลแต่ละตัวของแต่ละ Instantiation มาทำการเปรียบเทียบกันดังนี้คือ นำข้อมูลที่ใหม่ที่สุดของแต่ละ Instantiation มาทำการเปรียบเทียบกันก่อน ถ้าข้อมูลที่ใหม่ที่สุดของแต่ละ Instantiation ถูกใส่เข้ามาในหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบพร้อมกัน จะนำข้อมูลตัวถัดไปของแต่ละ Instantiation มาทำการเปรียบเทียบกันต่อไปอีกดังนี้ไปเรื่อย ๆ จนกว่าจะได้ Instantiation ใด Instantiation หนึ่งที่มีข้อมูลที่ใหม่กว่า Instantiation อื่น ๆ ในกรณีที่ Instantiation ใด

Instantiation หนึ่ง หมดข้อมูลที่จะนำไปเปรียบเทียบก่อน Instantiation อื่น ๆ Instantiation นั้น ก็จะถูกตัดออกไปจาก conflict set ทันที จากตัวอย่างในรูปที่ 2.3 (P S) เป็นข้อมูลที่ใหม่ที่สุดในหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบ ดังนั้น Instantiation อื่นที่นอกเหนือจาก I_{12} I_{21} I_{22} I_{31} และ I_{41} ซึ่งมี (P S) เป็นตัวประกอบจะถูกตัดออกจาก conflict set ก่อนในขั้นตอนแรก ในขั้นตอนที่สองข้อมูลตัวถัดมาของแต่ละ Instantiation จะถูกนำมาเปรียบเทียบกัน (P T) ซึ่งเป็นข้อมูลหนึ่งใน Instantiation I_{21} ใหม่กว่าข้อมูลตัวถัดมาของ Instantiation อื่น ๆ ทุก Instantiation ดังนั้น Instantiation I_{21} จึงได้รับการคัดเลือกภายใต้การพิจารณาโดยวิธีนี้

ง. <u>คัดเลือกกฎโดยใช้ประวัติของการนำกฎแต่ละกฎมาทำการประมวลผลเป็นเกณฑ์ (Rule Firing History)</u>

การคัดเลือกกฎโดยวิธีนี้ จะต้องมีการเก็บประวัติของการนำกฎแต่ละกฎไปใช้ไว้ในหน่วยความจำอีกส่วนหนึ่ง การคัดเลือกกฎในลักษณะนี้วิธีการหนึ่งได้แก่ ถ้ากฎใดถูกนำไปทำการประมวลผลในรอบการทำงานที่แล้ว กฎนั้นก็จะไม่ได้รับการคัดเลือกให้ทำการประมวลผลในรอบนี้อีก คือเป็นการป้องกันไม่ให้กฎใดกฎหนึ่ง ถูกนำไปประมวลผลในรอบการทำงานที่ติด ๆ กันสองรอบ อีกวิธีการหนึ่งซึ่งจะใช้ประวัติของการนำกฎไปใช้ทั้งหมดตั้งแต่เริ่มต้นจนถึงปัจจุบันมาพิจารณาก็คือ ถ้ากฎใดเคยถูกนำไปทำการประมวลผลมาแล้ว กฎนั้นก็จะไม่ได้รับการคัดเลือกอีกเลย

จ. <u>คัดเลือกกฎโดยใช้ลำดับความสำคัญของกฎแต่ละกฎเป็นเกณฑ์ (Priority)</u>

การคัดเลือกกฎโดยวิธีการนี้ จะต้องมีการเก็บลำดับความสำคัญของกฎทุก ๆ กฎในฐานความรู้เก็บไว้ ซึ่งอาจจะเป็นลำดับความสำคัญของกฎทั้งกฎ หรือเป็นลำดับความสำคัญเฉพาะของส่วนเงื่อนไขแต่ละประโยค หรือของส่วนกระทำของกฎก็ได้ การใช้ลำดับความสำคัญของกฎนี้อาจจะเทียบได้กับการใช้ลำดับของการใส่กฎเข้าไปในฐานความรู้ ซึ่งอาจจะพิจารณาได้ว่าเป็นการกำหนดลำดับความสำคัญของกฎโดยดูจากตำแหน่งของกฎในฐานความรู้ แต่การกำหนดลำดับความสำคัญให้กฎแต่ละกฎไปเลยนั้นจะสะดวกกว่าตรงที่ ไม่ต้องคำนึงถึงตำแหน่งของกฎ กฎใหม่จะใส่เข้าไปตรงไหนก็ได้เพียงแต่ต้องกำหนดลำดับความสำคัญให้ถูกต้องเท่านั้น

ฉ. <u>คัดเลือกกฎโดยใช้ความซับซ้อนของกฎแต่ละกฎเป็นเกณฑ์ (Complexity)</u>

ความซับซ้อนของกฎในที่นี้จะพิจารณาจากจำนวนของประโยคในส่วนต่าง ๆ ของกฎคือส่วนเงื่อนไขและส่วนผลสรุปหรือส่วนกระทำ การคัดเลือกกฎแบบนี้มีอยู่ 2 วิธีคือ เลือกกฎที่มีจำนวนประโยคในส่วนเงื่อนไขมากที่สุด ถ้าเป็นการวินิจฉัยแบบย้อนหลัง หรือเลือกกฎที่มีจำนวนประโยคในส่วนกระทำมากที่สุด ถ้าเป็นการวินิจฉัยแบบเดินหน้า (Most Complex First) อีกวิธีการหนึ่งซึ่งตรงข้ามกับวิธีการแรกก็คือ เลือกกฎที่มีจำนวนประโยคในส่วนเงื่อนไขหรือส่วนกระทำน้อยที่สุด (Simplest First)

การคัดเลือกกฎโดยใช้ลำดับของการใส่กฎเข้าไปในฐานความรู้เป็นเกณฑ์นั้น เป็นวิธีที่ง่ายและใช้กันมากที่สุด การทำงานในขั้นตอนของการวินิจฉัยจะถูกต้อง และมีประสิทธิภาพมาก ถ้าลำดับของกฎที่ใส่เข้าไปนั้นถูกต้อง ความยุ่งยากจะอยู่ในขั้นตอนของการแก้ไขฐานความรู้ ดังที่ได้กล่าวแล้วในตอนต้น การคัดเลือกกฎโดยใช้ลำดับความสำคัญจะไม่มีปัญหาเรื่องการเพิ่มกฎใหม่ ๆ เข้าไปในระบบเหมือนการใช้ลำดับของการใส่กฎเข้าไปในฐานความรู้ แต่ต้องมีวิธีการเพิ่มเติมสำหรับจัดเก็บ และนำลำดับความสำคัญของกฎไปใช้ การคัดเลือกกฎโดยใช้ความซับซ้อนของกฎเป็นเกณฑ์นั้น ก็เป็นวิธีที่ง่าย และรวดเร็วเช่นกัน แต่ความซับซ้อนของกฎแต่ละกฎนั้นก็ไม่สามารถที่จะเปรียบเทียบกันได้เสมอไป การใช้กรณีพิเศษเป็นเกณฑ์จะมีความถูกต้องมากกว่า เพราะพิจารณาจากฐานความรู้ และ/หรือ หน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบอย่างละเอียด แต่การพิจารณาอย่างละเอียดนี้ก็มีผลทำให้ประสิทธิภาพในการวินิจฉัยตกลงและยังตัดกฎออกไปจาก conflict set ได้เพียงบางส่วนเท่านั้น การคัดเลือกกฎโดยใช้ระยะเวลาที่ข้อมูลแต่ละตัวถูกใส่เข้าไปในหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบเป็นเกณฑ์นั้น เป็นวิธีการที่มีความไวต่อการเปลี่ยนแปลงของสภาพการณ์ของระบบเป็นพิเศษ และใช้ข้อมูลที่ทำให้เกิดความขัดแย้งซึ่งอยู่ในหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบได้อย่างมีประโยชน์มากที่สุด จึงเหมาะสมกับระบบที่ต้องการความไวต่อการเปลี่ยนแปลง เช่น ระบบตรวจจับ และระบบควบคุม เป็นต้น การคัดเลือกกฎอีกวิธีหนึ่งคือใช้ประวัติของการนำกฎไปใช้มีใช้กันน้อยมาก เนื่องจากต้องมีการเก็บประวัติของการนำกฎไปใช้ไว้ในหน่วยความจำอีกส่วนหนึ่ง ทำให้เสียเวลาในการจัดเก็บและการนำไปใช้

การแทนค่าความรู้ (Knowledge representation)

ความรู้ในระบบเชี่ยวชาญจะอยู่ในรูปของข้อเท็จจริง (fact) และความสัมพันธ์ (relationship) ซึ่งสามารถแทนค่าได้หลายวิธีด้วยกันได้แก่ (2)

- การแทนค่าความรู้โดยใช้กฎ (Rule)
- การแทนค่าความรู้โดยใช้กรอบ (Frame)
- การแทนค่าความรู้โดยใช้ข่ายความหมาย (Semantic network)
- การแทนค่าความรู้โดยใช้ Object-attribute-value triplet (O-A-V triplet) หรือ Attribute-value pair (A-V pair)
- การแทนค่าความรู้โดยใช้การแสดงความหมายทางตรรก (Logical expression)

1. การแทนค่าความรู้โดยใช้กฎ (Rule) (7) (16)

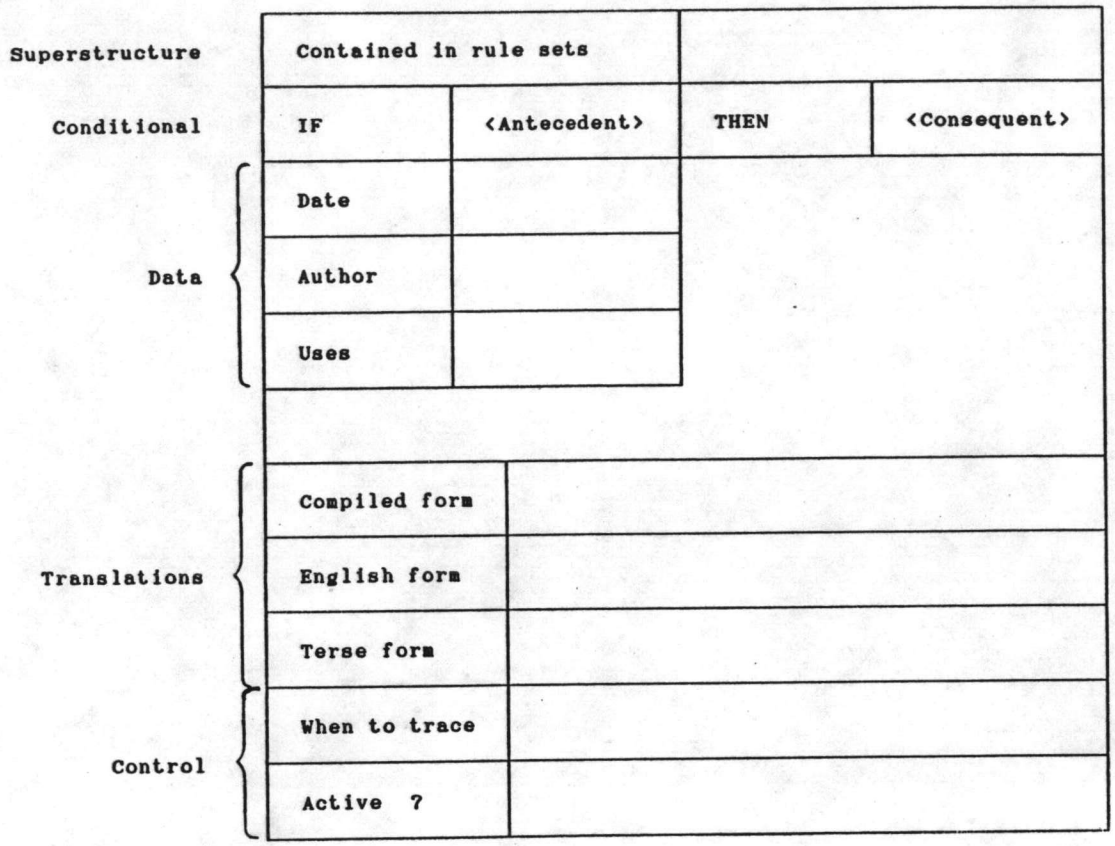
ความรู้จะถูกแทนค่าภายในฐานความรู้ในรูปของกฎ ซึ่งประกอบด้วยส่วนนำ (antecedent) และส่วนตาม (consequent) ส่วนนำจะอยู่ในรูปของประโยคเงื่อนไขซึ่งอาจจะประกอบด้วยประโยคเงื่อนไขหลาย ๆ ประโยคเชื่อมต่อกันด้วยตรรก AND หรือตรรก OR ก็ได้ ในบางครั้งจึงเรียกส่วนนำนี้ว่าส่วนเงื่อนไข หรือส่วนหลักฐาน (premise) สำหรับส่วนตามนั้นอาจเป็นได้ทั้งส่วนผลสรุป (conclusion) หรือส่วนกระทำ (action) ซึ่งจะถูกนำมาใช้ได้ก็ต่อเมื่อส่วนเงื่อนไขของกฎนั้นเป็นจริงเท่านั้น

```
If   : 1) the gram stain of the organism is gram negative, and
       2) the morphology of the organism is rod, and
       3) the aerobicity of the organism is anaerobic,
Then: There is suggestive evidence (0.6) that the identity
      of the organism is Bacteroides.
PREMISE :  ($AND (SAME CNTXT GRAM GRAMNEG)
                 (SAME CNTXT MORPHO ROD)
                 (SAME CNTXT AIR ANAEROBIC))
ACTION  :  (CONCLUDE CNTXT IDENT BACTEROIDES TALLY 600)
```

รูปที่ 2.4 ตัวอย่างของกฎความรู้ในรูปแบบภาษาอังกฤษ และ LISP

ระบบเชี่ยวชาญที่ใช้การแทนค่าความรู้ในรูปของกฎความรู้นี้เรียกว่าระบบฐานกฎ (rule-based system) ระบบฐานกฎนี้ประกอบด้วยส่วนพื้นฐานที่สำคัญ 2 ส่วนคือ ฐานความรู้และเครื่องจักรกลวินิจฉัย ฐานความรู้จะประกอบด้วยกฎ และข้อเท็จจริง (fact) เก็บไว้ภายในหน่วยความจำสถิตย์ (long term static) ของระบบ หน่วยความจำอีกส่วนหนึ่งของระบบก็คือหน่วยความจำจลน์ (short term dynamic) ซึ่งใช้เก็บค่าต่าง ๆ ที่บ่งบอกถึงสภาพการณ์ของระบบในขณะใดขณะหนึ่ง รวมทั้งใช้เก็บการยืนยัน (assertion) ต่าง ๆ ที่ได้เป็นการชั่วคราว หน่วยความจำส่วนนี้อาจเรียกว่า หน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบก็ได้ ลักษณะการทำงานของเครื่องจักรกลวินิจฉัยในระบบฐานกฎนี้จะเป็นการทำงานในลักษณะของการวนรอบคัดเลือกกฎแล้วนำกฎที่คัดเลือกได้ไปใช้ (recognise-act cycle) กฎใด ๆ จะได้รับการคัดเลือกไปใช้งานก็ต่อเมื่อส่วนเงื่อนไขของกฎนั้น สามารถจับคู่ในลักษณะของ pattern matching ได้พอดีกับข้อมูลที่อยู่ในหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบ ทำให้ส่วนเงื่อนไขนั้นของกฎนั้นเป็นจริง ส่วนกระทำ หรือส่วนผลสรุปของกฎนั้นก็จะถูกนำไปประมวลผล ซึ่งการประมวลผลนั้น ส่วนหนึ่งจะเกี่ยวข้องกับการเปลี่ยนแปลงค่าภายในหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบเป็นผลให้สภาพการณ์ของระบบเปลี่ยนไป และกฎอื่น ๆ ก็จะได้รับการคัดเลือกมาทำการประมวลผลต่อไปอีก ในช่วงของการทำการคัดเลือกกฎนั้นอาจมีกฎมากกว่า 1 กฎ ที่มีส่วนเงื่อนไขสอดคล้องกับสภาพการณ์ของระบบในขณะนั้นพร้อมกันได้ กลวิธีไกล่เกลี่ยความขัดแย้งจึงต้องเข้ามามีบทบาทในการทำการคัดเลือกกฎที่มีความเหมาะสมมากที่สุดไปทำการประมวลผล สำหรับทิศทางในการวินิจฉัยในระบบฐานกฎนั้น อาจเป็นได้ทั้งแบบวินิจฉัยไปข้างหน้า (forward chaining inference) หรือวินิจฉัยแบบย้อนหลัง (backward chaining inference) ก็ได้

กฎนอกจากจะประกอบไปด้วยส่วนนำ และส่วนตามดังที่ได้กล่าวไปแล้วนั้น ยังอาจจะมีข้อมูลเพิ่มเติมอื่น ๆ ได้อีก ยกตัวอย่างเช่น ในระบบฐานกฎขนาดใหญ่ที่มีกฎความรู้เป็นจำนวนมากนั้น จะมีการจัดกลุ่มของกฎออกเป็นส่วน ๆ แต่ละส่วนก็อาจจะแบ่งย่อยลงไปได้อีกหลายกลุ่ม ทำให้ได้โครงสร้างของกฎทั้งระบบเป็นแบบระดับชั้น นั่นคือกฎใดกฎหนึ่งจะเป็นสมาชิกของกลุ่มกฎหนึ่งเสมอ ซึ่งกลุ่มกฎนี้ก็อาจจะเป็นสมาชิกของกลุ่มกฎอื่นที่อยู่ในระดับชั้นที่สูงกว่า กฎแต่ละกฎจึงต้องระบุกลุ่มของกฎที่มันเป็นสมาชิกอยู่ไว้ภายใน นอกจากนี้กฎยังอาจจะมีข้อมูลเกี่ยวกับการแปรรูปกฎให้อยู่ในรูปแบบต่าง ๆ ข้อมูลที่ใช้ในการควบคุมการวินิจฉัย และข้อมูลทั่ว ๆ ไปของกฎนั้นเช่น ผู้เขียน และวันที่ เป็นต้น



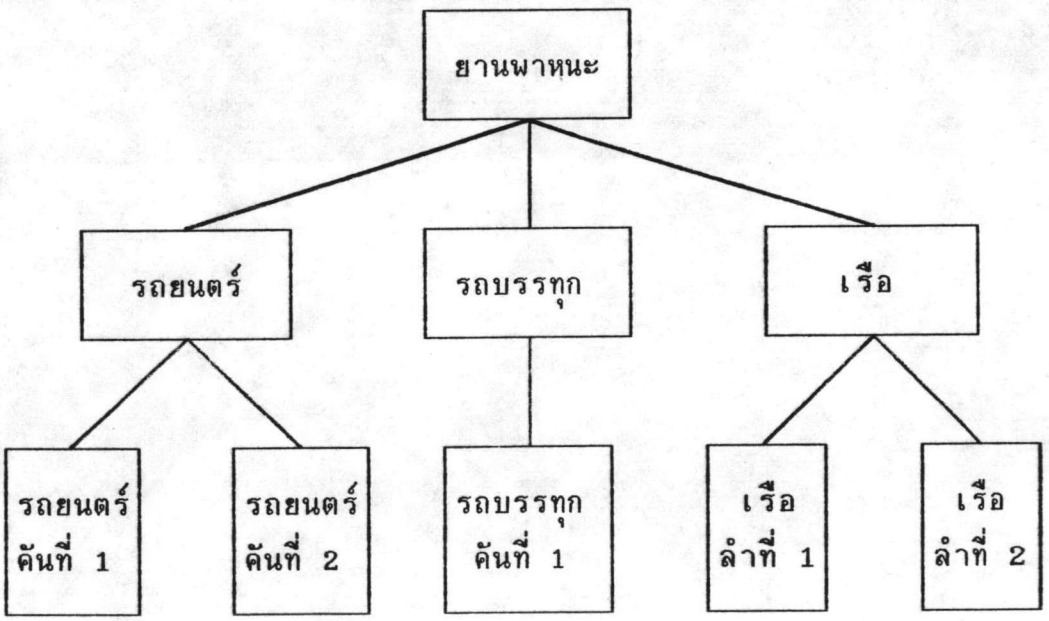
รูปที่ 2.5 กฎและส่วนประกอบของกฎ

2. การแทนค่าความรู้โดยใช้กรอบ (Frame) (17)

กรอบเป็นรูปแบบของการแทนค่าความรู้ ที่เสนอขึ้นครั้งแรกโดย M. Minsky ในปีค.ศ.1974 (6) เป็นการแทนค่าความรู้แบบโครงสร้างชนิดหนึ่ง ซึ่งใช้ในการสร้างแบบจำลองของกระบวนการเรียนรู้และความจำของมนุษย์ กรอบหนึ่งกรอบอาจใช้แทนวัตถุ กลุ่มของวัตถุ การกระทำ ความคิด สถานะการณ์ หรือเหตุการณ์ก็ได้ โดยที่ภายในกรอบก็จะเก็บข้อมูลต่าง ๆ เกี่ยวกับสิ่งนั้นไว้ และมีการเก็บบันทึกความสัมพันธ์ระหว่างข้อมูลต่าง ๆ เหล่านั้นด้วย กรอบยังสามารถแบ่งออกได้เป็น กรอบที่มีระดับชั้น (hierarchical frame) ซึ่งมีความสามารถในการถ่ายทอดคุณสมบัติ (inheritance) และกรอบที่ไม่มีระดับชั้น ในระบบกรอบที่มีระดับชั้นนั้น กรอบหนึ่งกรอบสามารถใช้แทนกลุ่มของวัตถุชนิดใดชนิดหนึ่ง ในขณะที่มีกรอบอื่น ๆ ในระบบใช้แทนวัตถุหนึ่ง ๆ หรือกลุ่ม

I1031506

ของวัตถุอีกกลุ่มหนึ่งที่เป็นสมาชิกของกลุ่มแรกนั้นได้ โดยที่คุณสมบัติต่าง ๆ จะได้มาจากคุณสมบัติของกรอบนั้นเอง บวกกับคุณสมบัติของกรอบที่มันเป็นสมาชิกอยู่ ซึ่งได้มาจากการถ่ายทอดคุณสมบัติ ตัวอย่างเช่น ในฐานความรู้หนึ่ง อาจมีกรอบยานพาหนะเป็นกรอบที่อยู่ในระดับชั้นที่หนึ่ง มีกรอบรถยนตร์ รถบรรทุก และเรือเป็นกรอบที่อยู่ในระดับชั้นที่สองและเป็นสมาชิกของกรอบยานพาหนะ และมีกรอบรถยนตร์คันที่หนึ่ง รถยนตร์คันที่สอง รถบรรทุกคันที่หนึ่ง เรือลำที่หนึ่ง และเรือลำที่สองเป็นกรอบที่อยู่ในระดับชั้นที่สามและเป็นสมาชิกของกรอบต่าง ๆ ในระดับชั้นที่สอง ความสัมพันธ์ระหว่างกรอบต่าง ๆ เหล่านี้สามารถแสดงได้ดังรูปที่ 2.6 คุณสมบัติต่าง ๆ ของรถยนตร์คันที่หนึ่งก็จะได้มาจากคุณสมบัติของกรอบรถยนตร์คันที่หนึ่งบวกกับคุณสมบัติของกรอบรถยนตร์และกรอบยานพาหนะซึ่งได้มาจากการถ่ายทอดคุณสมบัติ



รูปที่ 2.6 ระบบกรอบแบบมีระดับชั้น

ลักษณะโครงสร้างของกรอบจะประกอบด้วย ชื่อกรอบ และกลุ่มของสล็อต (slot) โดยที่แต่ละสล็อตจะใช้แทนค่าคุณสมบัติในด้านต่าง ๆ ของกรอบ และภายในสล็อตก็จะเก็บค่าของคุณสมบัตินั้น ๆ ของกรอบ นอกจากนี้สล็อตยังอาจจะใช้เก็บค่า default ตัวชี้ไปยังกรอบอื่น ๆ (pointer) กลุ่มของกฎ หรือกระบวนการ (procedure) ที่จะทำงานเมื่อถูกเรียกเพื่อให้ได้มาซึ่งค่าของสล็อตนั้นก็ได้

ชื่อกรอบ	
ชื่อสล็อต	ค่าของสล็อต
ชื่อสล็อต	ค่าของสล็อต
ชื่อสล็อต	ค่าของสล็อต

รูปที่ 2.7 ลักษณะโครงสร้างของกรอบ

จากการที่ค่าภายในสล็อตอาจเป็นได้หลายชนิด ทั้งตัวเลข ตัวอักษร ค่า default ตัวชี้ กลุ่มของกฎ และกระบวนการ ในบางกรณีจึงมีความจำเป็นที่จะต้องอธิบายเพิ่มเติมถึงลักษณะต่าง ๆ ของค่าที่จะเก็บไว้ภายในสล็อตเช่น ชนิด ค่าสูงสุด และต่ำสุดที่สามารถเป็นไปได้ และค่า default เป็นต้น ในระบบกรอบส่วนมากจะใช้สล็อตย่อย ๆ ที่เรียกว่าฟาเซ็ต (facet) ในการอธิบายเพิ่มเติมถึงลักษณะต่าง ๆ เหล่านี้ สล็อตที่ประกอบไปด้วยฟาเซ็ตจะมีลักษณะดังต่อไปนี้

ชื่อสล็อต	
ฟาเซ็ต	ค่าฟาเซ็ต
ฟาเซ็ต	ค่าฟาเซ็ต
ฟาเซ็ต	ค่าฟาเซ็ต

รูปที่ 2.8 ลักษณะโครงสร้างของสล็อตที่มีฟาเซ็ต

ฟาเซ็ตมีหลายชนิด แต่ที่ใช้กันทั่วไปได้แก่

ก. Value facet เก็บค่าของสล็อตนั้น

ข. Data-type facet กำหนดชนิดของค่าที่จะเก็บไว้ภายในสล็อตนี้

ค. Range facet กำหนดช่วงที่เป็นไปได้ของค่าที่จะเก็บไว้ภายในสล็อต ทั้งค่าสูงสุดและต่ำสุด

ง. Default facet กำหนดค่า default ของสล็อต

Data-type facet และ Range facet นั้น สามารถใช้สำหรับตรวจสอบความถูกต้องของค่าที่จะนำไปเก็บไว้ภายในสล็อตได้

กระบวนการ หรือวิธีการใด ๆ จะสามารถติดอยู่กับกรอบ และถูกเรียกให้ทำงานได้หลายวิธี ในโครงระบบเชี่ยวชาญ KEE (2) นั้นมีอยู่ 2 วิธีด้วยกันคือ method และ active value

- Method เป็นกระบวนการในภาษา LISP ที่ติดอยู่กับกรอบและจะตอบสนองต่อข่าวสารที่ส่งเข้ามายังกรอบนั้น method จะถูกเก็บไว้ในลักษณะที่เป็นค่าของสล็อตที่เรียกว่า สล็อตตอบสนองต่อข่าวสาร (message responder) ข่าวสารที่ถูกส่งเข้ามายังกรอบนั้นจะระบุสล็อตที่ต้องการ พร้อมทั้งข้อมูลต่าง ๆ ที่จำเป็นต่อการประมวลผลของ method ในสล็อตนั้น

- Active value เป็นกระบวนการ หรือกลุ่มของกฎที่ติดอยู่กับสล็อตของกรอบ ซึ่งกระบวนการหรือกลุ่มของกฎนี้จะถูกเรียกใช้ก็ต่อเมื่อมีการเปลี่ยนแปลงของค่าภายในสล็อต ค่าภายในสล็อตถูกเรียกใช้ หรือมีการเก็บค่าใหม่ลงในสล็อต

ทั้ง method และ active value สามารถใช้ได้กับกรอบทุกกรอบในระบบไม่ว่าจะอยู่ในระดับชั้นใดก็ตาม

ถึงแม้ว่าการเสนอรูปแบบของกรอบขึ้นมาในครั้งแรกโดย Minsky นั้นจะเป็นการนำกรอบมาใช้ในแง่ของการควบคุมการให้เหตุผลในระบบวิเคราะห์ภาพ (scene analysis system) ก็ตาม แต่ผลงานเกี่ยวกับกรอบที่ปรากฏตามกันออกมาในภายหลังนั้น ส่วนใหญ่จะให้ความสนใจในด้านของการแทนค่าความรู้แบบโครงสร้างมากกว่าในด้านของการควบคุมการให้เหตุผล นั่นคือสิ่งที่เก็บไว้ภายในกรอบจะเป็นเพียงฐานข้อมูลสำหรับระบบความรู้เท่านั้น การควบคุมการให้เหตุผลจะอยู่ในรูปแบบอื่น ในส่วนอื่นของระบบ ลักษณะเช่นนี้เป็นการใช้ประโยชน์ของกรอบในการจัดระเบียบและการจัดเก็บความรู้เท่านั้น ตัวอย่างของระบบเชี่ยวชาญที่ใช้กรอบในลักษณะนี้ได้แก่ KRL UNITS และ KL-ONE เป็นต้น

อย่างไรก็ตาม กรอบสามารถมีบทบาทที่สำคัญในส่วนของการควบคุมการให้เหตุผลได้อีกด้วย ยกตัวอย่างเช่น ในระบบ production ที่มี production rule อยู่เป็นจำนวนมากนั้น สามารถนำกรอบไปใช้ในการจัดระเบียบ และควบคุมการทำงานของกฎเหล่านั้นได้อย่างมีประสิทธิภาพ กรอบอาจนำไปใช้ในการแทนค่าของกฎให้อยู่ในรูปของกรอบเลยก็ได้ จะทำให้จัดกลุ่มของกฎได้ง่ายขึ้นโดยใช้ลักษณะทางโครงสร้างของกรอบ กรอบกฎดังกล่าวจะประกอบด้วย สล็อตรูปแบบภายนอก (external form slot) สล็อตวิธีการกระจายกฎ (parse method slot) สล็อตเงื่อนไข (condition slot) สล็อตผลสรุป (conclusion slot) และสล็อตปฏิบัติ (action slot)

ก. สล็อตรูปแบบภายนอก (external form slot) เก็บกฎในรูปแบบที่ผู้ใช้ป้อนเข้าไปในระบบ

ข. สล็อตวิธีการกระจายกฎ (parse method slot) วิธีการกระจายกฎ (parse method) ที่เก็บไว้ภายในสล็อตนี้ เป็นวิธีการที่ระบบจะใช้ในการเปลี่ยนรูปแบบของกฎที่ผู้ใช้ป้อนเข้ามา (external form) ให้อยู่ในรูปแบบที่สามารถนำไปประมวลผลได้ (internal form) ซึ่งจะกระจายอยู่ในสล็อตต่าง ๆ อีก 3 สล็อตคือ สล็อตเงื่อนไข (condition slot) สล็อตผลสรุป (conclusion slot) และสล็อตปฏิบัติ (action slot)

ค. สล็อตเงื่อนไข (condition slot) เป็นสล็อตที่ใช้เก็บส่วนเงื่อนไขของกฎ

ง. สล็อตผลสรุป (conclusion slot) เป็นสล็อตที่ใช้เก็บส่วนผลสรุปของกฎ

จ. สล็อตปฏิบัติ (action slot) เป็นสล็อตที่ใช้เก็บกระบวนการหรือวิธีการใด ๆ ที่จะทำงานก็ต่อเมื่อส่วนเงื่อนไขของกฎนั้นเป็นจริง

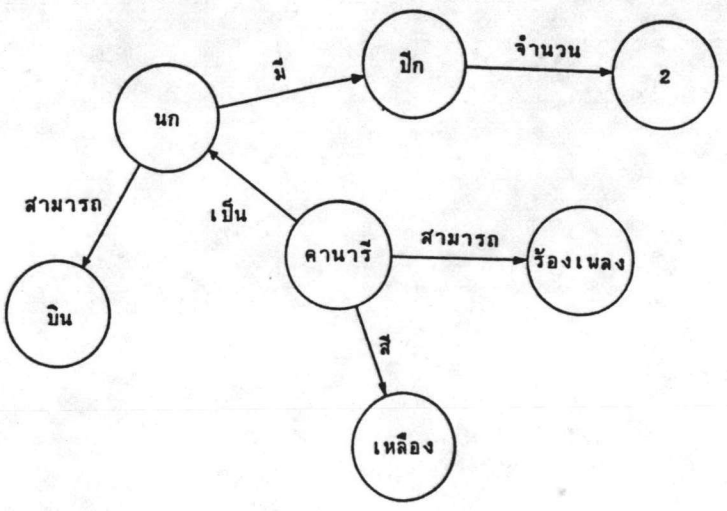
นอกจากการแทนค่าของกฎให้อยู่ในรูปของกรอบแล้ว อีกวิธีการหนึ่งที่เป็นการนำเอากรอบมาใช้ประโยชน์ในแง่ของการควบคุมการให้เหตุผลก็คือ ใช้กรอบในการจัดหมวดหมู่ของกฎให้เป็นไปตามการใช้งาน เพื่อความสะดวกและรวดเร็วในการเรียกใช้ ยกตัวอย่างเช่น กฎต่าง ๆ ที่เกี่ยวข้องกับการหาสาเหตุ และวิธีการแก้ไขปัญหาของระบบไฟฟ้าของรถบรรทุกนั้น สามารถจัดรวมเป็นกลุ่มเดียวกันได้ และเชื่อมโยงกลุ่มของกฎนี้เข้ากับกรอบรถบรรทุก กฎต่าง ๆ นี้จะถูกเรียกใช้ผ่านกรอบรถบรรทุก ในขณะที่ระบบกำลังทำการหาสาเหตุและวิธีการแก้ไขปัญหาของระบบไฟฟ้าของรถบรรทุกคันใดก็ตามในระบบ การเชื่อมโยงกลุ่มของกฎดังกล่าวเข้ากับกรอบรถบรรทุกกระทำได้หลายวิธี กรอบรถบรรทุกอาจมีสล็อตชื่อ DIAGNOSE ซึ่งภายในบรรจุวิธีการเรียกใช้กลุ่มกฎ รถ

บรรทุกทุกคันในระบบจะสามารถเรียกใช้สล็อต DIAGNOSE นี้ได้โดยการถ่ายทอดคุณสมบัติ ดังนั้น ทุกครั้งที่มีการส่งข่าวสาร DIAGNOSE ไปยังกรอบรถบรรทุกคันใดก็ตามในระบบ กลุ่มกฎดังกล่าวจะถูกเรียกใช้ กลุ่มกฎนี้อาจจะเชื่อมโยงอยู่กับกรอบรถบรรทุกในลักษณะที่เป็น active value ของสล็อต ELECTRICAL. FAULTS ก็ได้ และถูกเรียกใช้เมื่อระบบต้องการค่าของสล็อต ELECTRICAL. FAULTS นี้สำหรับรถบรรทุกทุกคันในระบบ

3. <u>การแทนค่าความรู้โดยใช้ข่ายความหมาย (Semantic network)</u>

(2)

ข่ายความหมาย เป็นรูปแบบของการแทนค่าความรู้ที่เสนอขึ้นโดย M.R.Quillian ในปีค.ศ.1968 เป็นการแทนค่าความรู้ในรูปของความสัมพันธ์ระหว่างวัตถุสองอย่าง ซึ่งอาจจะเป็นสิ่งของ เหตุการณ์ การกระทำ หรือสิ่งที่เป็นนามธรรมอื่น ๆ ก็ได้ ข่ายความหมายนี้สามารถแสดงได้โดยใช้โครงสร้างข้อมูลแบบกราฟ โดยให้โหนดแทนวัตถุและให้ลูกศรแทนความสัมพันธ์ระหว่างวัตถุ ลูกศรแต่ละเส้นจะมีชื่อบอกลักษณะของความสัมพันธ์กำกับอยู่ และหัวลูกศรจะแสดงทิศทางของความสัมพันธ์นั้น ในลักษณะเช่นนี้ ข่ายความหมายก็จะเป็นกลุ่มของวัตถุที่เชื่อมต่อกันด้วยความสัมพันธ์จำนวนหนึ่ง จากประโยคต่อไปนี้ "คานารีเป็นนกชนิดหนึ่งที่มีสีเหลืองและสามารถร้องเพลง" เราสามารถสร้างข่ายความหมาย เพื่อใช้แทนประโยคดังกล่าวและความรู้สามัญอื่น ๆ ที่เกี่ยวข้องได้ดังนี้



รูปที่ 2.9 แสดงตัวอย่างของข่ายความหมาย

ความสัมพันธ์ที่มีปรากฏอยู่ในรูปที่ 2.9 ได้แก่

- สามารถ (นก,บิน)
- มี (นก,ปีก)
- จำนวน (ปีก,2)
- เป็น (คานารี,นก)
- สามารถ (คานารี,ร้องเพลง)
- สี (คานารี,เหลือง)

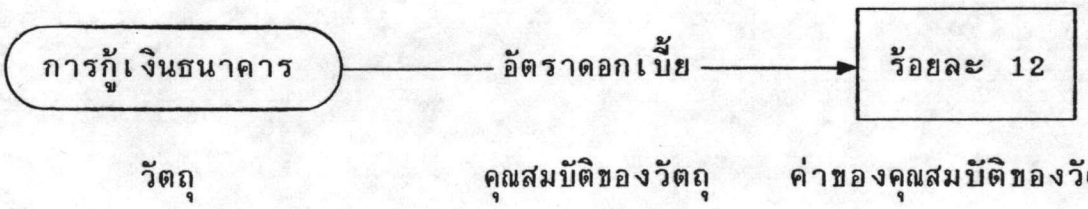
โดยทั่ว ๆ ไปโหนดมักใช้แทนวัตถุ และข้อมูลอื่น ๆ ที่ใช้บรรยายคุณสมบัติต่าง ๆ ของวัตถุ ตัวอย่างเช่นข้อความต่อไปนี้ "เสื้อสีแดง" "เสื้อ" และ "แดง" จะเป็นโหนดสองโหนดที่เชื่อมต่อกันด้วยความสัมพันธ์ "สี" โหนด "เสื้อ" เป็นวัตถุ ส่วนโหนด "แดง" เป็นข้อมูลบรรยายคุณสมบัติของวัตถุ

ความสัมพันธ์ที่ใช้เชื่อมระหว่างโหนดสองโหนดนั้น มีอยู่สองอย่างที่พบอยู่เสมอ ๆ คือความสัมพันธ์ "เป็น" และความสัมพันธ์ "มี" ความสัมพันธ์ "เป็น" ใช้แทนการเป็นสมาชิกหนึ่งของวัตถุที่อยู่ในระดับของการจัดจำพวกที่สูงกว่า นั่นคือ ความสัมพันธ์ระหว่างเซ็ตกับสมาชิกของเซ็ต หรือความสัมพันธ์ระหว่างเซ็ตกับเซ็ตย่อยก็ได้ ตัวอย่างเช่น "สมชายเป็นคน" และ "คนเป็นสิ่งมีชีวิต" เป็นต้น สำหรับความสัมพันธ์ "มี" นั้นใช้แทนการแสดงลักษณะต่าง ๆ ของวัตถุ ตัวอย่างเช่น "นกมีปีก" และ "รถมีล้อ" เป็นต้น ความสัมพันธ์ทั้งสองแบบนี้เป็นความสัมพันธ์ชนิดที่สามารถถ่ายทอดคุณสมบัติได้ กล่าวคือ คุณสมบัติต่าง ๆ ของวัตถุหนึ่ง สามารถถ่ายทอดไปให้อีกวัตถุหนึ่ง ที่มีความสัมพันธ์กันโดยตรง และอยู่ในระดับที่ต่ำกว่าได้ ซึ่งจากตัวอย่างในรูปที่ 2.9 นั้นจะได้จากการถ่ายทอดคุณสมบัติว่า "นกคานารีมีปีก 2 ปีกและบินได้" ความสามารถในการถ่ายทอดคุณสมบัตินี้ช่วยทำให้ลดความซ้ำซ้อนของข้อมูล และประหยัดหน่วยความจำที่ใช้ในการเก็บความรู้ไปได้มาก แต่ละวัตถุไม่จำเป็นต้องกำหนดความสัมพันธ์ทุกอย่างที่มีไว้กับวัตถุนั้น ความสัมพันธ์ใด ๆ ที่มีร่วมกับวัตถุอื่นในจำพวกเดียวกัน ก็สามารถไปกำหนดรวมกันไว้ที่วัตถุหนึ่งได้ จากตัวอย่างในรูปที่ 2.9 จะเห็นว่าไม่จำเป็นที่จะต้องมีความสัมพันธ์ระหว่างโหนด "คานารี" กับโหนด "ปีก" และโหนด "คานารี" กับโหนด "บิน" ก็ยังสามารถวินิจฉัยได้ว่า "นกคานารีมีปีก 2 ปีกและบินได้" จากการถ่ายทอดคุณสมบัติ ในกรณีที่วัตถุใดวัตถุหนึ่งมีคุณสมบัติพิเศษบางอย่างแตกต่างออกไปจากวัตถุอื่น ๆ ในจำพวกเดียวกัน ก็จะต้องมีการกำหนดความสัมพันธ์ชนิดนั้นไว้ที่วัตถุนั้นด้วย ดังนั้นการถ่ายทอดคุณสมบัติจะถูกนำไปใช้ก็ต่อเมื่อไม่มีความสัมพันธ์ชนิดเดียวกันนั้นอยู่กับวัตถุที่กำลังพิจารณา

ข้อดีประการหนึ่งของการแทนค่าความรู้โดยใช้ข่ายความหมายนี้คือ มีความยืดหยุ่นสูงในการเพิ่มเติมความรู้ใหม่ ๆ เข้าไปในฐานความรู้ วัตถุ หรือ ความสัมพันธ์อันใหม่สามารถถูกใส่เข้าไปในฐานความรู้ได้ตามต้องการ โดยจะมีผลกระทบต่อวัตถุอื่น ๆ ในระบบน้อยที่สุด

4. <u>การแทนค่าความรู้โดยใช้ Object-attribute-value triplet (O-A-V triplet) หรือ Attribute-value pair (A-V pair)</u> (2)

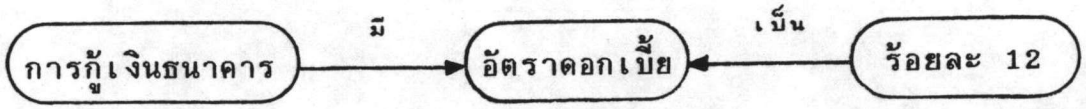
วิธีการแทนค่าความรู้แบบนี้เป็นวิธีที่ใช้ในระบบเชี่ยวชาญ MYCIN ซึ่งเป็นระบบเชี่ยวชาญขนาดใหญ่ระบบแรก และเป็นแม่แบบสำหรับการพัฒนาระบบเชี่ยวชาญและโครงระบบเชี่ยวชาญต่อมาอีกหลายระบบ วัตถุในวิธีการนี้สามารถเป็นได้ทั้งวัตถุทางกายภาพ เช่น ประตู และสิ่งที่เป็นมโนภาพ เช่น การกู้เงินธนาคาร เป็นต้น คุณสมบัติ (attribute) คือลักษณะทั่ว ๆ ไปในด้านต่าง ๆ ของวัตถุ เช่น ขนาด รูปร่าง และสี ก็เป็นคุณสมบัติต่าง ๆ ของวัตถุทางกายภาพ ส่วนอัตราดอกเบี้ยก็เป็นคุณสมบัติหนึ่งของการกู้เงินธนาคารซึ่งเป็นวัตถุทางมโนภาพ ค่าของคุณสมบัติ (value) ซึ่งเป็นส่วนสุดท้ายที่มาประกอบขึ้นเป็น O-A-V triplet เป็นตัวกำหนดลักษณะเฉพาะของแต่ละวัตถุในสถานการณ์หนึ่ง ๆ เช่น ค่าของอัตราดอกเบี้ยเงินกู้อาจมีค่าเท่ากับร้อยละ 12 เป็นต้น



รูปที่ 2.10 O-A-V triplet

O-A-V triplet อาจเรียกได้ว่าเป็นข่ายความหมายชนิดพิเศษที่ใช้ความสัมพันธ์พื้นฐานเพียง 2 อย่างคือ ความสัมพันธ์ "มี" และความสัมพันธ์ "เป็น" โดยใช้คุณสมบัติเป็นตัวกลาง กล่าวคือ ความสัมพันธ์ของวัตถุไปยังคุณสมบัติ (วัตถุ --> คุณสมบัติ) จะใช้ความสัมพันธ์ "มี" และความสัมพันธ์ของค่าของคุณสมบัติไปยังคุณสมบัติ (คุณสมบัติ <-- ค่าของคุณสมบัติ) จะใช้ความสัมพันธ์ "เป็น" ยกตัวอย่างเช่น การกู้เงินธนาคาร มี อัตราดอกเบี้ย และร้อยละ 12 เป็น อัตราดอกเบี้ยของการกู้เงินธนาคาร เป็นต้น ดังนั้นโหนด

ในข่ายความหมายชนิดพิเศษนี้ สามารถแบ่งออกได้เป็น 3 พวกคือ โหนดของวัตถุ โหนดของคุณสมบัติ และโหนดของค่าของคุณสมบัติ



รูปที่ 2.11 O-A-V triplet ในรูปของข่ายความหมาย

การทำงานของระบบที่ใช้การแทนค่าความรู้แบบ O-A-V triplet นี้ จะเกี่ยวข้องกับความรู้ 2 ประเภทคือความรู้สถิตย์ (static knowledge) และค่าเชิงจลน์ (dynamic instance) ความรู้สถิตย์จะประกอบด้วยส่วนประกอบ 2 ส่วนแรกของ O-A-V triplet คือ วัตถุ และคุณสมบัติของวัตถุ ซึ่งส่วนประกอบทั้ง 2 ส่วนนี้จะเป็นส่วนที่ไม่มีการเปลี่ยนแปลง วัตถุก็จะหมายถึงวัตถุทั่ว ๆ ไป ไม่มีการเฉพาะเจาะจงไปที่วัตถุใดวัตถุหนึ่ง เช่น ถ้าเป็นรถยนตร์ ก็จะหมายถึงรถยนตร์ทั่ว ๆ ไป สำหรับค่าเชิงจลน์นั้นเป็นค่าของคุณสมบัติของวัตถุ ซึ่งจะมีการเปลี่ยนแปลงอยู่ตลอดเวลาจากกรณีหนึ่งไปอีกกรณีหนึ่ง ยกตัวอย่างเช่น วัตถุรถยนตร์สามารถมีคุณสมบัติอันหนึ่งคือ "เจ้าของ" หรือ "เลขทะเบียน" เพื่อใช้ชี้เฉพาะลงไปว่าเป็นรถยนตร์คันไหน โดยใช้ค่าของคุณสมบัตินั้นเป็นตัวกำหนด เมื่อมีการอ้างถึงรถยนตร์อีกคันหนึ่งค่าของคุณสมบัตินี้ก็จะเปลี่ยนไปอีก ความรู้สถิตย์จะถูกเก็บไว้ในฐานความรู้ ส่วนค่าเชิงจลน์นั้นเป็นค่าที่ได้จากการทำงานของระบบ โดยจะเก็บค่าต่าง ๆ เหล่านี้เป็นค่าเชิงจลน์ไว้ในหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบ ซึ่งจะมีการเปลี่ยนแปลงอยู่ตลอดเวลา และอาจจะมีการระบุค่าปัจจัยความเชื่อมั่น (certainty factor) ไปควบคู่กับค่าเชิงจลน์แต่ละค่าด้วย วิธีการกำหนดค่าเฉพาะให้กับคุณสมบัติต่าง ๆ ของวัตถุที่เก็บไว้ในฐานความรู้สถิตย์ แล้วเก็บค่าที่ได้ไว้ในหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบนี้เรียกว่า instantiation

วัตถุต่าง ๆ ในระบบ O-A-V triplet นี้ ยังสามารถมีความสัมพันธ์กันและเรียงลำดับกันได้ในลักษณะของโครงสร้างต้นไม้ ซึ่งทำให้สามารถถ่ายทอดคุณสมบัติต่าง ๆ ไปยังวัตถุในระดับล่าง ๆ ของระบบได้

ระบบเชี่ยวชาญบางระบบมีขนาดเล็ก ทำการประมวลผลกับวัตถุใดวัตถุหนึ่งเพียงวัตถุเดียวทั้งระบบ ไม่มีวัตถุอื่นมาเกี่ยวข้อง แต่มีคุณสมบัติในด้านต่าง ๆ ของวัตถุนั้นเข้ามาเกี่ยวข้องด้วยเป็นจำนวนมาก ระบบเชี่ยวชาญ

ในลักษณะนี้ จึงไม่มีความจำเป็นต้องใช้ O-A-V triplet ในการแทนค่าข้อเท็จจริงเพราะวัตถุในแต่ละ O-A-V triplet ของระบบเชี่ยวชาญดังกล่าวจะหมายถึงวัตถุตัวเดียวกันเสมอ การใช้เพียง attribute-value pair หรือ A-V pair จะมีความเหมาะสมมากกว่า การแทนค่าข้อเท็จจริงโดยใช้ A-V pair จะทำให้ระบบขาดความสามารถในการถ่ายทอดคุณสมบัติไป แต่เนื่องจากมีวัตถุเพียงตัวเดียว การถ่ายทอดคุณสมบัติจึงไม่มีความจำเป็น

5. <u>การแทนค่าความรู้โดยใช้การแสดงความหมายทางตรรก (Logical expression)</u> (2)

การแทนค่าความรู้โดยวิธีนี้ เป็นที่นิยมใช้กันในประเทศทางแถบยุโรปมากกว่าในอเมริกา จุดเด่นของการแทนค่าความรู้โดยใช้ตรรกวิทยานั้นอยู่ที่ ตรรกวิทยามีรากฐานทางทฤษฎีที่ดี ถูกต้อง และแน่นอน การทำงานของเครื่องจักรกลวินิจฉัยในระบบที่ใช้การแทนค่าความรู้โดยใช้ตรรกวิทยานั้น จะสามารถทำงานได้เร็วและมีประสิทธิภาพมากกว่าระบบที่ใช้การแทนค่าความรู้แบบอื่น ๆ แต่วิธีการในการแทนค่าความรู้ลงในฐานความรู้จะมีความยุ่งยากมากกว่า ตรรกวิทยาที่นำมาใช้ในการแทนค่าความรู้มีอยู่ 2 แขนงคือ

- Propositional logic
- Predicate calculus

ก. <u>Propositional logic</u>

Proposition คือประโยคต่าง ๆ ซึ่งมีค่าทางตรรกได้ 2 ค่าคือ จริง หรือเท็จ proposition หลาย ๆ ประโยคที่เชื่อมต่อกันด้วยตัวเชื่อม (connective) จะเรียกว่าประโยคประกอบ (compound statement) ซึ่งก็จะมีค่าได้เพียง 2 ค่าเช่นกันคือ จริง หรือเท็จ อย่างใดอย่างหนึ่ง ตัวเชื่อมดังกล่าวได้แก่ AND (ใช้สัญญลักษณ์ $\triangle$) OR (ใช้สัญญลักษณ์ ∇) NOT (ใช้สัญญลักษณ์ ~) IMPLIES (ใช้สัญญลักษณ์ ->) และ EQUIVALENT (ใช้สัญญลักษณ์ =) propositional logic จะเกี่ยวข้องกับการหาค่าความจริงของประโยคประกอบ ซึ่งขึ้นอยู่กับค่าความจริงของแต่ละประโยคที่มาเชื่อมต่อกันและตัวเชื่อม กฎต่าง ๆ ที่ใช้ในการหาค่าความจริงของแต่ละตัวเชื่อมนั้น สามารถสรุปเป็นตารางความจริงได้ดังรูปที่ 2.12

P	Q	~P	~Q	P △ Q	P ▽ Q	P -> Q	P = Q
จริง	จริง	เท็จ	เท็จ	จริง	จริง	จริง	จริง
จริง	เท็จ	เท็จ	จริง	เท็จ	จริง	เท็จ	เท็จ
เท็จ	จริง	จริง	เท็จ	เท็จ	จริง	จริง	เท็จ
เท็จ	เท็จ	จริง	จริง	เท็จ	เท็จ	จริง	จริง

รูปที่ 2.12 ตารางความจริง

ยกตัวอย่างเช่น มี proposition P ซึ่งมีค่าความจริงเป็นจริง และ proposition Q ซึ่งมีค่าความจริงเป็นเท็จ จากตารางความจริงจะได้ว่า ประโยคประกอบ P AND Q จะมีค่าความจริงเป็นเท็จ ในขณะที่ประโยคประกอบ P OR Q จะมีค่าความจริงเป็นจริง

ข. Predicate calculus

Predicate calculus เป็นตรรกวิทยาอีกแขนงหนึ่ง ที่ขยายออกมาจาก propositional logic นั่นคือ แต่ละ proposition หรือประโยค จะถูกแบ่งออกเป็นภาคแสดงหรือ predicate ซึ่งเป็นส่วนที่เป็นกริยาของประโยค และมีอาร์กิวเมนท์ (argument) เป็นคำนาม ซึ่งอาจมีได้มากกว่า 1 อาร์กิวเมนท์ รูปที่ 2.13 แสดงตัวอย่างบางส่วนของฐานความรู้ที่ใช้การแทนค่าความรู้โดย predicate calculus

1. คน (ดำ)
2. ลูก (ดำ,มี)
3. ลูก (ดำ,มา)
4. ผู้หญิง (ดำ)
5. ผู้ชาย (มา)
6. ผู้หญิง (มี)

รูปที่ 2.13 การแทนค่าความรู้โดยใช้ predicate calculus

ในการแทนค่าความรู้แบบอื่น ๆ นั้น เราสามารถได้รับค่าต่าง ๆ ที่มีอยู่ในข้อเท็จจริงได้โดยตรง ซึ่งอาจจะเป็นค่าของคุณสมบัติต่าง ๆ ของวัตถุในระบบ O-A-V triplet หรือค่าของสล็อตในระบบกรอบ หรือเป็นชื่อของโหนดใดโหนดหนึ่งในข่ายความหมายก็ได้ แต่ในการแทนค่าความรู้โดยใช้ตรรกวิทยานั้น ค่าของประโยคต่าง ๆ ที่ใช้ในการแทนค่าความรู้มีเพียง 2 ค่า คือ จริงหรือเท็จเท่านั้น ดังนั้นเราจึงไม่สามารถป้อนคำถาม "ดำเป็นลูกของใคร" และได้รับคำตอบว่า มี และ มา ได้โดยตรง แต่เราสามารถป้อนคำถาม "ดำเป็นลูกของมีใช่ไหม" และ "ดำเป็นลูกของมาใช่ไหม" ได้ ถ้าค่าที่ได้รับเป็นจริงทั้งคู่ ก็จะสามารถสรุปได้ว่า ดำเป็นลูกของมีและมา

ตรรกวิทยาได้ถูกนำมาใช้เป็นหลักในการเขียนโปรแกรมในภาษา PROLOG ซึ่งถูกคิดค้นขึ้นมาในปี ค.ศ.1972 โดย Colmerauer และ Roussel ที่ Marseilles ประเทศฝรั่งเศส ปัจจุบัน PROLOG เป็นภาษาที่ประเทศญี่ปุ่นเลือกใช้ในการพัฒนาคอมพิวเตอร์ในยุคที่ 5 (5th generation computer) และได้รับความสนใจมากขึ้นในสหรัฐอเมริกา เนื่องจากมีขีดความสามารถในการแทนค่าความรู้สูงมาก

<u>ส่วนติดต่อกับผู้พัฒนา</u>

ส่วนติดต่อกับผู้พัฒนามีหน้าที่หลักคือ ช่วยผู้พัฒนาในการสร้างฐานความรู้ให้ถูกต้องสมบูรณ์ โดยการจัดหาเครื่องมือเครื่องใช้ต่าง ๆ ให้ผู้พัฒนาสามารถสร้าง ปรับปรุง และทดสอบฐานความรู้ได้อย่างมีประสิทธิภาพ นอกจากหน้าที่หลักในการสร้างฐานความรู้ ในบางระบบส่วนติดต่อกับผู้พัฒนายังมีหน้าที่สำคัญอีก 2 ประการคือ

- ช่วยพัฒนาส่วนติดต่อกับผู้ใช้ให้เหมาะสมกับงานมากยิ่งขึ้น
- ช่วยพัฒนาเครื่องจักรกลวินิจฉัยโดยอนุญาตให้ผู้พัฒนาเพิ่มคำสั่งพิเศษบางคำสั่ง หรือเพิ่มกฎควบคุมเข้าไปในระบบเพื่อให้เครื่องจักรกลวินิจฉัยทำงานตามต้องการในบางกรณี

ส่วนติดต่อกับผู้พัฒนาเป็นส่วนที่มีความสำคัญรองจากเครื่องจักรกลวินิจฉัยและการแทนค่าความรู้ ส่วนติดต่อกับผู้พัฒนาที่แตกต่างกันจะมีผลให้เวลาที่ใช้ในการสร้างระบบเชี่ยวชาญแตกต่างกัน และมีผลต่อความถูกต้องสมบูรณ์ของระบบเชี่ยวชาญที่สร้างขึ้นมาด้วย ส่วนติดต่อกับผู้พัฒนาที่ดีควรจะมีเครื่องมือที่ช่วยผู้พัฒนาในการสร้างฐานความรู้ การปรับปรุงแก้ไขฐานความรู้ และการทดสอบฐานความรู้ที่สร้างขึ้น เครื่องมือต่าง ๆ เหล่านี้ได้แก่ (5) (8)

- เครื่องมือในการสร้างฐานความรู้
- เครื่องมือในการแก้ไขฐานความรู้
- เครื่องมือในการติดตามการวินิจฉัย (Inference tracing)
- เครื่องมืออธิบายการทำงาน
- เครื่องมือค้นหาฐานความรู้ที่ต้องการ (Apropos facility)
- การแสดงฐานความรู้โดยใช้รูปภาพ
- การเก็บบันทึกผลการทำงานของแต่ละปัญหา
- เครื่องมือในการพัฒนาส่วนติดต่อกับผู้ใช้
- เครื่องมือในการพัฒนาเครื่องจักรกลวินิจฉัย

1. <u>เครื่องมือในการสร้างฐานความรู้</u>

เครื่องมือที่ใช้ในการสร้างฐานความรู้จะแตกต่างกันไปในโครงระบบเชี่ยวชาญแต่ละระบบ ซึ่งสามารถแบ่งออกได้เป็น

- ใช้เวิร์ดโปรเซสเซอร์ทั่ว ๆ ไป
- ใช้เอดิเตอร์ของโครงระบบเชี่ยวชาญ

ก. <u>ใช้เวิร์ดโปรเซสเซอร์ทั่ว ๆ ไป</u>

โครงระบบเชี่ยวชาญที่อนุญาติให้ผู้พัฒนาใช้เวิร์ดโปรเซสเซอร์ทั่ว ๆ ไปในการสร้างฐานความรู้ จะกำหนดรูปแบบของกฎ หรือข้อเท็จจริงมาให้อย่างชัดเจน ผู้พัฒนาจะต้องสร้างฐานความรู้โดยใช้รูปแบบที่กำหนดมาให้เท่านั้น การใช้เวิร์ดโปรเซสเซอร์นี้มีข้อดีคือ ผู้พัฒนาสามารถใช้เวิร์ดโปรเซสเซอร์ตัวใดก็ได้ที่ผู้พัฒนามีความถนัดอยู่แล้ว ทำให้สามารถสร้างได้อย่างรวดเร็ว

ข. <u>ใช้เอดิเตอร์ของโครงระบบเชี่ยวชาญ</u>

ซึ่งอาจจะเป็นเอดิเตอร์แบบเต็มจอหรือเอดิเตอร์แบบบรรทัดต่อบรรทัดก็ได้ การใช้เอดิเตอร์ที่จัดหามาให้โดยโครงระบบเชี่ยวชาญนั้น ผู้พัฒนาจะต้องมาเรียนรู้วิธีการใช้เอดิเตอร์นั้นใหม่ ซึ่งอาจจะขาดความชำนาญและเอดิเตอร์ของโครงระบบเชี่ยวชาญก็อาจจะไม่มีประสิทธิภาพเท่าเวิร์ดโปรเซสเซอร์ทั่ว ๆ ไป โครงระบบเชี่ยวชาญที่ทำงานบนเครื่องคอมพิวเตอร์ส่วนบุคคลขนาดเล็กส่วนมากมักอนุญาตให้ผู้พัฒนาใช้เวิร์ดโปรเซสเซอร์ทั่ว ๆ ไปในการสร้างฐานความรู้ เพราะเวิร์ดโปรเซสเซอร์บนเครื่องขนาดเล็กนี้มีหลายชนิด และใช้กันอย่างแพร่หลาย

2. เครื่องมือในการแก้ไขฐานความรู้

เมื่อฐานความรู้ได้ถูกสร้างขึ้นมาแล้วจะด้วยวิธีการใดก็ตาม ฐานความรู้นั้นก็จะถูกนำมาทดสอบกับเครื่องจักรกลวินิจฉัย ซึ่งอาจจะมีการเปลี่ยนแปลงแก้ไขอีกมาก จนกว่าจะได้ฐานความรู้ที่ถูกต้องสมบูรณ์ และมักจะกินเวลานานกว่าช่วงการสร้างฐานความรู้มาก ขั้นตอนนี้จึงนับว่ามีความสำคัญต่อการพัฒนาระบบเชี่ยวชาญมาก เครื่องมือที่ใช้ในการแก้ไขฐานความรู้ที่มีประสิทธิภาพจะช่วยให้การพัฒนาระบบถูกต้อง และรวดเร็วยิ่งขึ้น เครื่องมือแก้ไขฐานความรู้แบบออนไลน์ (Online knowledge base editing) จะช่วยให้ผู้พัฒนาสามารถแก้ไขฐานความรู้ได้ในระหว่างการทดสอบการทำงาน หลังจากแก้ไขเสร็จเรียบร้อยแล้วก็สามารถทำงานต่อจากจุดตรวจสอบเดิมด้วยฐานความรู้ที่ปรับปรุงใหม่แล้ว โครงระบบเชี่ยวชาญที่ไม่มีเครื่องมือเช่นนี้ ผู้พัฒนาจะต้องเลิกจากการทดสอบเพื่อทำการแก้ไขฐานความรู้ แล้วจึงจะเริ่มทำการทดสอบได้ใหม่ โดยต้องเริ่มใหม่ตั้งแต่ต้น ทำให้เสียเวลาและต้องป้อนข้อมูลต่าง ๆ จำนวนมากเข้าไปใหม่ ซึ่งอาจจะไม่ได้สภาพการทำงานเดิมก็ได้

การแปลฐานความรู้เป็นข้อพิจารณาที่สำคัญอีกข้อหนึ่งในเรื่องของการแก้ไขฐานความรู้ โครงระบบเชี่ยวชาญบางระบบจะต้องแปลฐานความรู้ก่อนจะนำมาใช้งาน ดังนั้นหลังจากที่ได้แก้ไขฐานความรู้ก็จะต้องนำไปแปลอีกครั้งหนึ่งก่อนจะนำมาใช้งานได้ โครงระบบเชี่ยวชาญประเภทนี้จึงไม่สามารถแก้ไขฐานความรู้แบบออนไลน์ได้ แต่เมื่อได้ฐานความรู้ที่สมบูรณ์แล้ว การทำงานจะเร็วกว่าระบบที่ไม่มีการแปลฐานความรู้ ตัวอย่างของโครงระบบเชี่ยวชาญที่ต้องแปลฐานความรู้ก่อนนำไปใช้งานเช่น โครงระบบเชี่ยวชาญแบบ induction เป็นต้น

3. เครื่องมือในการติดตามการวินิจฉัย (Inference tracing)

เครื่องมือประเภทนี้จะช่วยผู้พัฒนาในขั้นตอนการทดสอบฐานความรู้ ด้วยการทดสอบการทำงานของเครื่องจักรกลวินิจฉัยที่ใช้ฐานความรู้ที่สร้างขึ้นมา เครื่องมือดังกล่าวได้แก่ การอนุญาตให้ผู้พัฒนาสามารถหยุดการทำงานของระบบชั่วขณะ และสามารถตรวจสอบค่าต่าง ๆ ในหน่วยความจำที่เก็บสภาพการณ์ชั่วคราวของระบบ การแจกแจงสิ่งที่สามารถวินิจฉัยออกมาได้และที่มาของค่าต่าง ๆ เหล่านั้น ซึ่งจะทำให้ผู้พัฒนาค้นพบจุดบกพร่องของการทำงานได้อย่างรวดเร็ว

4. เครื่องมืออธิบายการทำงาน

เครื่องมืออธิบายการทำงานของเครื่องจักรกลวินิจฉัย ได้แก่

- WHY
- HOW
- WHAT-IF

ก. WHY

ผู้พัฒนาสามารถสอบถามระบบได้ด้วยคำถาม WHY เมื่อระบบตั้งคำถามบางคำถามให้ผู้พัฒนาตอบ ซึ่งหมายความว่า ทำไมระบบจึงถามคำถามนี้ออกมา ซึ่งโดยทั่วไปแล้วระบบมักจะตอบคำถาม WHY นี้ด้วยการแสดงกฎความรู้ต่าง ๆ ที่กำลังพิจารณาอยู่ออกมาให้เห็น เพื่อให้ผู้พัฒนาทราบว่าระบบกำลังหาข้อสรุปอะไรอยู่ และใช้อะไรในการหาข้อสรุปอันนั้น คำถามนี้จะช่วยให้ผู้พัฒนามองเห็นการทำงานของเครื่องจักรกลวินิจฉัยได้ชัดเจนยิ่งขึ้น

ข. HOW

ผู้พัฒนาสามารถสอบถามระบบได้ด้วยคำถาม HOW เมื่อระบบได้แสดงข้อสรุป หรือคำแนะนำบางประการออกมา ซึ่งจะหมายความว่าข้อสรุป หรือคำแนะนำนั้นระบบได้มาอย่างไร คำถาม HOW นี้ ระบบจะตอบด้วยการแสดงข้อมูลและกฎความรู้ต่าง ๆ ที่ทำให้ได้ข้อสรุป หรือคำแนะนำนั้นมา

ค. WHAT-IF

เมื่อผู้ใช้ถามระบบด้วยคำถาม WHAT-IF x จะหมายความว่าจะเกิดอะไรขึ้นถ้า x ระบบจะทำการวินิจฉัยโดยใช้วิธีการวินิจฉัยแบบไปข้างหน้าจากข้อมูลที่ได้รับมาในคำถามคือ x แล้วแสดงผลออกมาเป็นข้อสรุปที่สามารถวินิจฉัยได้ คำถาม WHAT-IF นี้ช่วยให้ผู้พัฒนาสามารถทำการเปรียบเทียบการทำงานและผลลัพธ์ที่ได้จากข้อมูลที่ต่างกันได้อย่างชัดเจน

5. เครื่องมือค้นหาฐานความรู้ที่ต้องการ (Apropos facility)

ในบางครั้ง ผู้พัฒนาอาจจะต้องการทราบว่ามีกฎหรือข้อเท็จจริงใดบ้างในฐานความรู้ ที่อ้างถึงวัตถุ (Object) หรือตัวแปรใดตัวแปรหนึ่ง หรือมีกฎใดบ้างที่มีการสรุปค่าของตัวแปรใดตัวแปรหนึ่ง เพื่อที่จะสามารถทำการแก้

ไขกฎที่เกี่ยวข้องกับวัตถุ หรือตัวแปรเดียวกันได้อย่างครบถ้วน เครื่องมือนี้จะช่วยผู้พัฒนาในการค้นหาความรู้ที่ตรงกับความต้องการของผู้พัฒนาดังกล่าว และอาจจะช่วยค้นหาและแทนที่ไปในเวลาเดียวกันด้วย ในโครงระบบเชี่ยวชาญขนาดใหญ่บางระบบที่ใช้การแทนค่าความรู้แบบกรอบ เครื่องมือนี้ก็ยังสามารถใช้ในการค้นหากรอบทั้งหมดที่เป็นกรอบลูกของกรอบที่กำหนดได้อีกด้วย ซึ่งจะช่วยให้ผู้พัฒนาค้นพบความรู้ที่ต้องการได้สะดวกรวดเร็ว เพื่อทำการปรับปรุงแก้ไขได้ตามต้องการ

6. การแสดงฐานความรู้โดยใช้รูปภาพ

เครื่องมือนี้จะมีประโยชน์มากในโครงระบบเชี่ยวชาญที่มีฐานความรู้ขนาดใหญ่และซับซ้อน เช่นที่ใช้กรอบในการแทนค่าความรู้ จะทำให้เห็นความสัมพันธ์ระหว่างกรอบ และโครงสร้างแบบมีระดับชั้นได้อย่างชัดเจน ผู้พัฒนาจะสามารถไล่ไปตามรูปที่ปรากฏบนจอภาพ และเรียกดูหรือแก้ไขรายละเอียดของแต่ละส่วนภายในได้โดยตรงจากจอภาพนี้

7. การเก็บบันทึกผลการทำงานของแต่ละปัญหา

การเก็บบันทึกการทำงานของระบบนี้มีหลายแบบเช่น การเก็บบันทึกการโต้ตอบระหว่างผู้ใช้กับระบบ การเก็บบันทึกสภาพการณ์ของระบบและการนำกฎไปใช้ในแต่ละขั้นตอน และการเก็บบันทึกผลการทำงานที่ได้ในขั้นตอนสุดท้าย การเก็บบันทึกการทำงานนี้ใช้เพื่อนำมาพิจารณาการทำงานในภายหลัง และสามารถนำไปเปรียบเทียบกับการทำงานของปัญหาอื่นได้ด้วย

8. เครื่องมือในการพัฒนาส่วนติดต่อกับผู้ใช้

ดังที่ได้กล่าวแล้วในตอนต้น หน้าที่ของส่วนติดต่อกับผู้พัฒนาที่นอกเหนือไปจากการสร้างฐานความรู้ก็คือ การช่วยพัฒนาส่วนติดต่อกับผู้ใช้และเครื่องจักรกลวินิจฉัย สำหรับการพัฒนาส่วนติดต่อกับผู้ใช้นั้น เครื่องมือที่มีอยู่ได้แก่ เครื่องมือกำหนดคำอธิบายกำกับจอภาพแต่ละจอที่ปรากฏต่อผู้ใช้ เครื่องมือสร้างระบบช่วยเหลือแบบออนไลน์ icon editor และเครื่องมือกำหนดลักษณะของจอภาพ ซึ่งอาจจะเป็นตำแหน่งและขนาดของหน้าต่างแสดงผล และหน้าต่างที่ให้ผู้ใช้ป้อนข้อมูล เป็นต้น

9. เครื่องมือในการพัฒนาเครื่องจักรกลวินิจฉัย

ผู้พัฒนาอาจจะสามารถกำหนดการทำงานของเครื่องจักรกลวินิจฉัยได้บ้างเป็นบางส่วน โดยการใส่คำสั่งพิเศษบางคำสั่งลงไปในระบบ แล้วแต่โครงระบบเชี่ยวชาญแต่ละระบบจะอนุญาตให้ทำได้มากน้อยเพียงใด หรืออาจจะโดยการใส่กฎควบคุม (meta rule) ลงไปในฐานความรู้ก็ได้ นอกจากนี้ การอนุญาตให้ผู้พัฒนาสามารถกำหนดลำดับความสำคัญ (priority) ของกฎแต่ละกฎ หรือกำหนดกลวิธีไกล่เกลี่ยความขัดแย้ง ก็เป็นการควบคุมการทำงานของเครื่องจักรกลวินิจฉัยอีกทางหนึ่ง

ส่วนติดต่อกับผู้ใช้

ภายหลังจากที่ได้สร้างระบบเชี่ยวชาญที่สมบูรณ์ขึ้นมาเรียบร้อยแล้วนั้น การที่ผู้ใช้จะสามารถนำระบบเชี่ยวชาญนั้นไปใช้งานจริง ๆ ได้ดีเพียงใดนั้นจะขึ้นอยู่กับส่วนติดต่อกับผู้ใช้ โครงระบบเชี่ยวชาญบางระบบก็ยอมให้ผู้พัฒนาสามารถปรับปรุงส่วนติดต่อกับผู้ใช้ได้บ้าง แต่ก็ยังคงลักษณะเดิมที่มากับโครงระบบเชี่ยวชาญนั้นอยู่มาก ลักษณะต่าง ๆ ของส่วนติดต่อกับผู้ใช้ได้แก่ (5) (8)

- จอภาพแบบบรรทัดและจอภาพแบบเลือกรายการ (line/menu screen)
- การตอบหลายคำตอบและการตอบในลักษณะที่ไม่แน่ใจ
- Initial pruning หรือ Directed search
- ระบบช่วยเหลือแบบออนไลน์
- ความเร็วในการทำงาน

1. จอภาพแบบบรรทัดและจอภาพแบบเลือกรายการ (line/menu screen)

จอภาพแบบบรรทัดจะเป็นการถามตอบระหว่างระบบกับผู้ใช้เป็นบรรทัดต่อบรรทัด ผู้ใช้ต้องป้อนคำตอบที่ระบบสามารถจะรับได้เข้าไปด้วยตัวเองทีละบรรทัด จอภาพแบบนี้เป็นจอภาพแบบที่ธรรมดาที่สุด จอภาพแบบเลือกรายการจะสะดวกสบายต่อผู้ใช้มากกว่า เพราะระบบจะแสดงคำตอบที่เป็นไปได้มาให้ผู้ใช้ทราบ ผู้ใช้เพียงแต่เลื่อนตัวชี้หรือตัวเลือกไปยังคำตอบที่ต้องการเท่านั้น ลักษณะเช่นนี้จะทำให้ผู้ใช้สามารถโต้ตอบกับระบบได้อย่างรวดเร็ว และ

สะดวกมากขึ้น

2. การตอบหลายคำตอบและการตอบในลักษณะที่ไม่แน่ใจ

ในการทำงานของระบบเชี่ยวชาญนั้น ระบบจะต้องโต้ตอบกับผู้ใช้โดยการป้อนคำถามให้ผู้ใช้ตอบอยู่ตลอดเวลา ผู้ใช้อาจจะไม่สามารถตอบคำถามได้เสมอไป ในกรณีนี้ ระบบจะต้องยอมให้ผู้ใช้ไม่ตอบคำถามดังกล่าว หรือตอบว่าไม่ทราบได้ ไม่เช่นนั้นระบบจะไม่สามารถทำงานต่อได้ สำหรับคำถามบางคำถาม ผู้ใช้อาจจะมีคำตอบหลาย ๆ คำตอบสำหรับคำถามเดียวกันนั้นได้ ระบบก็จะต้องยอมให้ผู้ใช้ตอบได้ทุก ๆ คำตอบที่มี โดยมีค่าความไม่แน่ใจกำกับอยู่สำหรับแต่ละคำตอบด้วย

3. Initial pruning หรือ Directed search

เป็นความสามารถของระบบที่ยอมให้ผู้ใช้สามารถกำหนดขอบเขตของปัญหาที่เกี่ยวข้อง ทำให้ลดปริมาณการค้นหาที่ไม่จำเป็นลงโดยการตัดปัญหาและความรู้ในส่วนอื่น ๆ ที่ไม่เกี่ยวข้องทิ้งไป ระบบจะให้ความสนใจเฉพาะในส่วนที่เกี่ยวข้องกับปัญหาที่เกิดขึ้นจริง ๆ เท่านั้น โครงระบบเชี่ยวชาญที่ไม่มีความสามารถในข้อนี้จะทำให้ผู้ใช้รู้สึกเบื่อหน่าย เนื่องจากต้องคอยตอบคำถามจำนวนมากที่ผู้ใช้ทราบดีว่า ไม่เกี่ยวข้องกับปัญหาที่เกิดขึ้นเลย นอกจากนี้ยังเสียเวลาไปกับการทำงานของระบบที่ไม่จำเป็นอีกด้วย ตัวอย่างของ initial pruning และ directed search ได้แก่คุณสมบัติ PROMPT1ST ในโครงระบบเชี่ยวชาญ PC ของ TI ฐานความรู้ของโครงระบบเชี่ยวชาญนี้จะถูกแบ่งออกเป็นส่วน ๆ เรียกว่าเนื้อความของปัญหา (context) แต่ละส่วนจะเกี่ยวข้องกับปัญหาในด้านต่าง ๆ แตกต่างกัน เช่นถ้าเป็นระบบเชี่ยวชาญให้คำปรึกษาด้านเกษตรกรรมของพืชชนิดหนึ่ง ปัญหาที่เกี่ยวข้องก็อาจจะแบ่งออกได้เป็นปัญหาที่เกิดจากโรคพืช ปัญหาที่เกิดจากแมลง และปัญหาที่เกิดจากวัชพืช เป็นต้น แต่ละเนื้อความของปัญหาก็จะมีคุณสมบัติ PROMPT1ST ซึ่งใช้ในการถามผู้ใช้ว่าต้องการให้ระบบทำการตรวจสอบและวินิจฉัยปัญหาในเนื้อความนั้นหรือไม่ ผู้ใช้สามารถตอบปฏิเสธได้ถ้าแน่ใจว่าปัญหาที่เกิดขึ้นไม่เกี่ยวข้องกับปัญหาในเนื้อความนั้น ระบบก็จะตัดฐานความรู้ในส่วนนั้นทิ้งไป รายละเอียดของ PROMPT1ST ได้กล่าวไว้ในบทที่ 4 ที่จะได้กล่าวต่อไป

4. ระบบช่วยเหลือแบบออนไลน์

ในบางครั้ง ผู้ใช้อาจจะต้องการคำอธิบายเพิ่มเติมเกี่ยวกับศัพท์บางคำที่ระบบใช้ หรือต้องการทราบว่า คำตอบสำหรับคำถามพิเศษบางคำถามจะมีลักษณะเป็นอย่างไร เป็นตัวเลขหรือตัวอักษร ถ้าเป็นตัวอักษรจะเป็นอะไรได้บ้าง และถ้าเป็นตัวเลข ค่าสูงสุดต่ำสุดเป็นเท่าไร ระบบช่วยเหลือแบบออนไลน์จะให้คำอธิบายต่าง ๆ เหล่านี้ ทำให้ผู้ใช้สามารถโต้ตอบกับระบบได้คล่องตัวยิ่งขึ้น ไม่ติดขัดเมื่อเกิดปัญหาขึ้น

5. ความเร็วในการทำงาน

โครงระบบเชี่ยวชาญที่พัฒนาขึ้นโดยใช้ภาษา LISP ที่ไม่สามารถแปลภาษาก่อนนำมาใช้งานได้นั้น จะทำงานค่อนข้างช้า แต่โครงระบบเชี่ยวชาญส่วนใหญ่จะเขียนขึ้นโดยใช้ภาษาดั้งเดิมเช่น FORTRAN C Pascal และ LISP ที่สามารถจะแปลภาษาก่อนนำมาใช้งานได้ ทำให้ทำงานได้รวดเร็วกว่าโครงระบบเชี่ยวชาญบางระบบเช่น ART ซึ่งพัฒนาโดยใช้ภาษา LISP ได้รับการออกแบบเป็นพิเศษให้สามารถทำงานได้อย่างรวดเร็ว เพื่อใช้กับการที่ต้องการการตอบสนองไวเป็นพิเศษเช่น ระบบตรวจจับและเตือนภัย เป็นต้น

ความสามารถในการติดต่อกับซอฟแวร์อื่น

โครงระบบเชี่ยวชาญระบบหนึ่งจะมีความสามารถมากยิ่งขึ้นถ้าสามารถติดต่อกับซอฟแวร์อื่นหรือระบบอื่นที่มีอยู่แล้ว เพื่อใช้ช่วยในการทำงานที่ซับซ้อนซึ่งระบบอื่นสามารถทำได้ดีกว่า หรือติดต่อกับระบบฐานข้อมูลอื่น ๆ เพื่อนำข้อมูลที่มีอยู่แล้วมาใช้ร่วมกับฐานความรู้ของระบบ เป็นการใช้ประโยชน์จากสิ่งที่มีอยู่แล้ว เพื่อลดการทำงานของเครื่องจักรกลวินิจฉัย และลดขนาดของฐานความรู้และความซ้ำซ้อนของข้อมูล ความสามารถในการติดต่อกับซอฟแวร์อื่นนี้แบ่งออกได้เป็น

- Embeddable
- ความสามารถเชื่อมโยงเข้ากับระบบฐานข้อมูลอื่น
- ความสามารถเชื่อมโยงเข้ากับภาษาอื่น

1. Embeddable

โครงระบบเชี่ยวชาญที่เขียนขึ้นด้วยภาษาที่มีเครื่องคอมพิวเตอร์ที่รองรับภาษานั้นโดยเฉพาะ จะทำให้โครงระบบเชี่ยวชาญนั้นสามารถใช้รวมกับเครื่องมืออื่นที่มีอยู่บนเครื่องคอมพิวเตอร์ชนิดนั้นได้ เช่น โครงระบบเชี่ยวชาญที่เขียนขึ้นด้วยภาษา C บนเครื่อง VAX หรือ LISP บน LISP machine เป็นต้น

2. ความสามารถเชื่อมโยงเข้ากับระบบฐานข้อมูลอื่น

ความรู้บางอย่างที่ระบบเชี่ยวชาญต้องการใช้อาจจะมีความเหมาะสมที่จะเก็บไว้ในฐานข้อมูลภายนอก มากกว่าที่จะเก็บไว้ภายในฐานความรู้ หรือมีฐานข้อมูลที่ต้องการอยู่แล้วแต่ต้องการนำมาใช้กับระบบเชี่ยวชาญด้วย โครงระบบเชี่ยวชาญจะต้องมีวิธีการเรียกใช้ฐานข้อมูลภายนอกเข้ามาในระบบ เป็นการประหยัดหน่วยความจำและลดความซ้ำซ้อนของข้อมูล

3. ความสามารถเชื่อมโยงเข้ากับภาษาอื่น

เช่นเดียวกับการเชื่อมโยงเข้ากับฐานข้อมูลอื่น โครงระบบเชี่ยวชาญจะต้องมีวิธีการเรียกใช้กระบวนการหรือโปรแกรมย่อยที่เขียนขึ้นด้วยภาษาอื่นมาใช้งาน เช่นระบบเชี่ยวชาญที่เขียนขึ้นด้วยภาษา LISP อาจจะมีความต้องการเรียกใช้โปรแกรมย่อยที่เขียนขึ้นด้วยภาษา FORTRAN ให้ทำการคำนวณบางอย่างที่ภาษา FORTRAN จะทำได้ดีกว่าภาษา LISP

สรุป

แต่ละองค์ประกอบของโครงระบบเชี่ยวชาญมีวิธีการต่าง ๆ แตกต่างกันไปในแต่ละระบบ แต่ละวิธีก็มีทั้งข้อดีและข้อเสียแตกต่างกัน บริษัทผู้ผลิตโครงระบบเชี่ยวชาญได้เลือกนำเอาวิธีการต่าง ๆ ของแต่ละองค์ประกอบมาใช้ร่วมกันตามความต้องการและความเหมาะสม ทำให้ได้โครงระบบเชี่ยวชาญจำนวนมากที่มีลักษณะแตกต่างกันออกไป ในบทต่อไปจะเป็นการแนะนำโครงระบบเชี่ยวชาญบางระบบที่มีใช้อยู่ในปัจจุบัน และวิธีการที่ใช้ในองค์ประกอบต่าง ๆ