บทที่ 3

ระบบผู้เชี่ยวชาญ

จากความรู้ที่ได้ในระบบหม้อไอน้ำ เราสามารถนำระบบผู้เชี่ยวชาญมาแก้ไขปัญหาที่เกิดขึ้น ในการปฏิบัติการหม้อไอน้ำแบบท่อไฟ ขนาดไม่เกิน 10 ตัน ในบทนี้จะเป็นการแนะนำให้รู้จักระบบผู้เชี่ยวชาญ โดยจะกล่าวถึงหลักการของระบบผู้เชี่ยวชาญ การนำวิศวกรรมความรู้มาใช้ในระบบผู้เชี่ยวชาญ การพัฒนาเทคโนโลยีระบบผู้เชี่ยวชาญตั้งแต่อดีตถึงปัจจุบัน และการนำระบบผู้เชี่ยวชาญไปใช้งานในด้านต่าง ๆ โดยในการศึกษาวิจัยนี้เป็นการนำระบบผู้เชี่ยวชาญไปใช้งานด้านวิศวกรรมอุตสาหการ

1. หลักการของระบบผู้เชี่ยวชาญ

1.1 ระบบผู้เชี่ยวชาญคืออะไร (6)

ระบบผู้เชี่ยวชาญเป็นศาสตร์สาขาหนึ่งของปัญญาประดิษฐ์ (Artificial Intelligence-AI) ปัญญาประดิษฐ์ คือวิชาที่ว่าด้วยการจำลองเครื่องจักรหรือคอมพิวเตอร์ มีความสามารถคล้ายมนุษย์ในด้านการแก้ปัญหาการคิดอย่างมีเหตุผล รวมทั้งการสังเกตจดจำสิ่งต่าง ๆ เช่นการจดจำและเข้าใจภาพ การจดจำและเข้าใจเสียง เป็นต้น

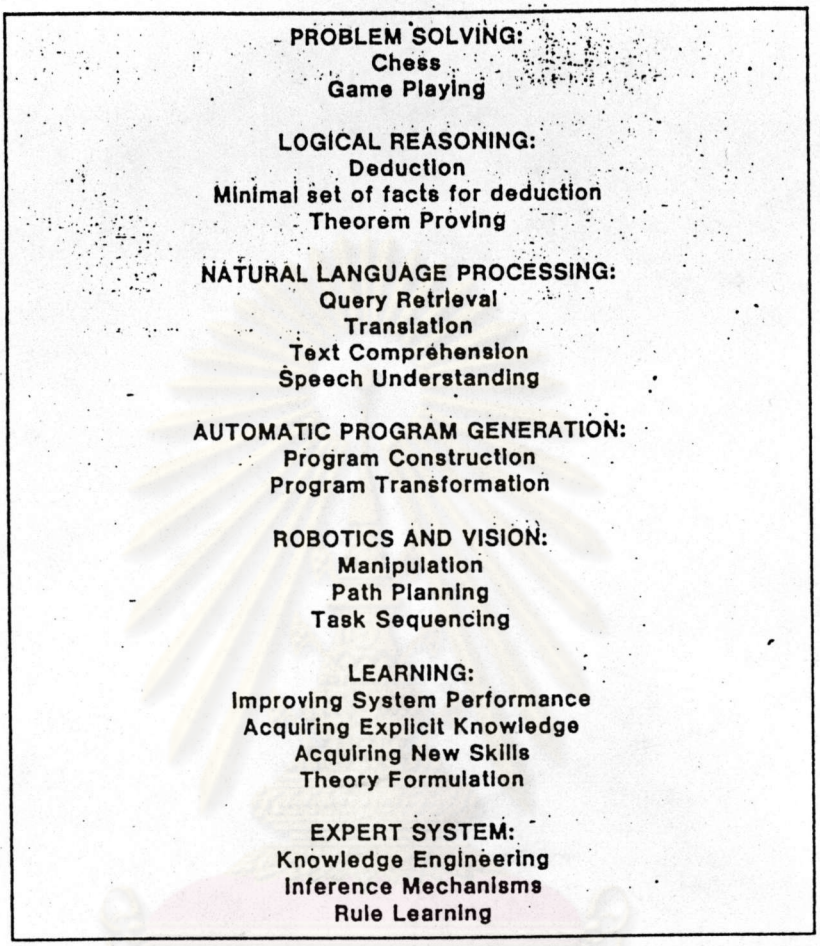
การประยุกต์ใช้ปัญญาประดิษฐ์ในสาขาต่าง ๆ แสดงได้ดังรูปที่ 3.1

AI แบ่งออกเป็น 3 สาขาใหญ่ ๆ คือ

1. Natural Language Processing คือการพัฒนาโปรแกรมที่สามารถอ่านพูดและเข้าใจภาษาคนได้

2. Robotics คือการสร้างหุ่นยนต์ที่ฉลาด(Smart robot)โปรแกรมที่หุ่นยนต์สามารถมีประสาทสัมผัสมองเห็นและทำบางสิ่งบางอย่างคล้ายคนได้ มักจะให้หุ่นยนต์ทำงานในสภาวะแวดล้อมที่อันตราย (Harzardous) หรืองานที่ซ้ำซ้อนน่าเบื่อ(Repetitive jobs)

3. Expert Systems คือโปรแกรมที่ใช้ Symbolic knowledge ในการจำลองความคิดการทำงานของผู้เชี่ยวชาญ



รูปที่ 3.1 การประยุกต์ใช้ปัญญาประดิษฐ์ในสาขาต่าง ๆ

จะเห็นว่าระบบผู้เชี่ยวชาญ(Expert System) คือระบบซอฟต์แวร์ หรือโปรแกรมคอมพิวเตอร์ ที่มีความสามารถในการลอกเลียนแบบพฤติกรรมการคิดหรือตัดสินใจของมนุษย์ในการวิเคราะห์ปัญหา สามารถตอบปัญหาหรือเลียนแบบการแก้ปัญหาของผู้เชี่ยวชาญได้ ระบบผู้เชี่ยวชาญแตกต่างจากซอฟต์แวร์อันอื่นตรงที่ว่ามันมีความฉลาดอยู่ในตัว เพราะสามารถใช้ความรู้ และกระบวนการอนุมานหรือวินิจฉัย การอนุมานในที่นี้หมายถึงการหาข้อสรุปโดยหาเหตุผลจากที่ต่าง ๆ ที่เกิดขึ้น

สำหรับปัญหาในทางวิศวกรรมแบ่งออกเป็น 2 ประเภท คือ

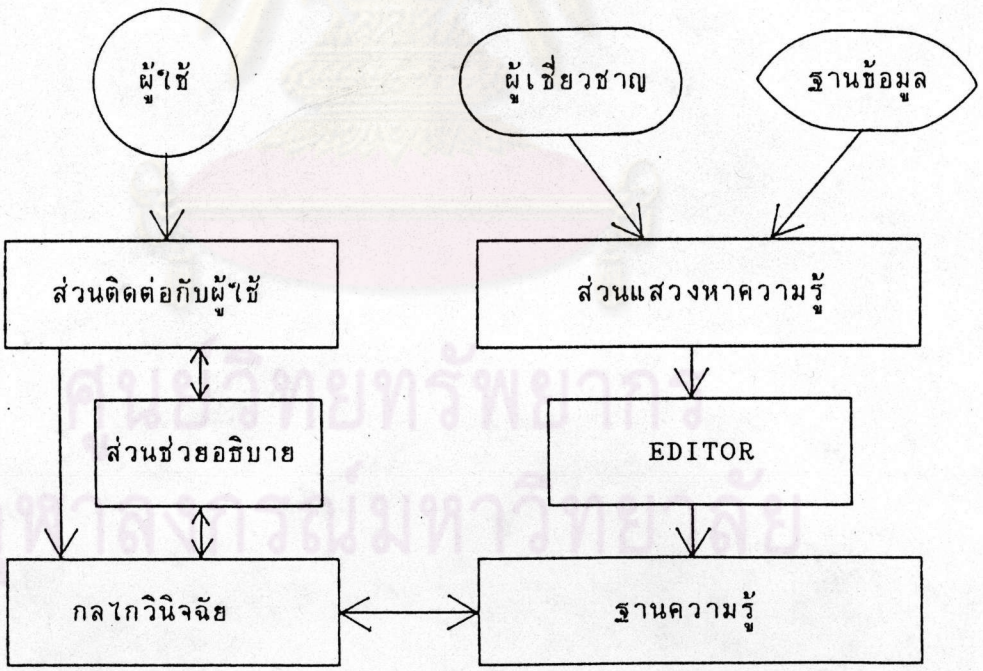
1. ปัญหาที่มีโครงสร้างที่ดี (Well - structure problem) ได้แก่ ปัญหาที่มีจุดมุ่งหมายที่แน่นอนสามารถสร้างโมเดลในการแก้ปัญหาได้ และมีวิธีการแก้ปัญหาอย่างเป็นขั้นเป็นตอนโดยเทคนิคการแก้ปัญหาแบบดั้งเดิม (Conventional techniques)

เช่น ปัญหาด้าน Operations research เป็นต้น

2. ปัญหาที่มีโครงสร้างไม่ดี (Ill-Structure Problem) ได้แก่ปัญหาที่มีสภาพการเกิดปัญหาและวิธีการแก้ไขที่ยุ่งยากซับซ้อน ซึ่งไม่สามารถแก้ไขด้วยเทคนิคแบบดั้งเดิมได้ จะมีวิธีการแก้ปัญหาแบบหยั่งรู้(Heuristic) โดยอาศัยความรู้และประสบการณ์ จากผู้เชี่ยวชาญที่มีความรู้เฉพาะสาขา(Domain Knowledge) ที่จะแก้ไข เป้าหมายหมายของระบบผู้เชี่ยวชาญ คือ การแก้ไขปัญหาลักษณะดังกล่าวนี้

1.2 โครงสร้างของระบบผู้เชี่ยวชาญ (7)

ปัญหาที่มีโครงสร้างไม่ดี (Illstructured Problem) จำเป็นต้องใช้ประสบการณ์ในรูปของความรู้ และกฎเกณฑ์พร้อมทั้งการวินิจฉัยที่สมเหตุสมผล โครงสร้างของระบบ ดังแสดงในรูปที่ 3.2



รูปที่ 3.2 โครงสร้างพื้นฐานของระบบผู้เชี่ยวชาญ

องค์ประกอบของระบบผู้เชี่ยวชาญมีดังนี้

1.2.1 ฐานความรู้ (Knowledge base) เป็นที่เก็บความรู้ที่ได้จากความจริงต่าง ๆ หรือได้มาจากประสบการณ์

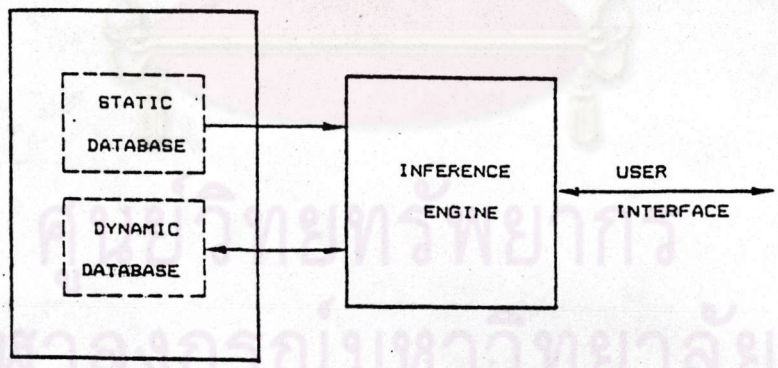
1.2.2 กลไกวินิจฉัย (Inference engine) หรือการอนุมานเป็นส่วนควบคุมการใช้ความรู้ในฐานความรู้เพื่อแก้ไขปัญหาอย่างมีประสิทธิภาพ

1.2.3 ส่วนแสวงหาความรู้ (Knowledge acquisition subsystem) หรือส่วนที่ทำหน้าที่ดึงเอาความรู้จากผู้เชี่ยวชาญหรือจากเอกสารต่าง ๆ มาเก็บไว้ในฐานความรู้

1.2.4 ส่วนช่วยอธิบาย (Explanation subsystem) ทำหน้าที่อธิบายรายละเอียดของขั้นตอนวินิจฉัยให้กับผู้ใช้ (User) ตามคำเรียกร้อง

1.2.5 ส่วนติดต่อกับผู้ใช้ (User interface) หรือส่วนปฏิภาคกับผู้ใช้ทำหน้าที่เป็นตัวกลางระหว่างผู้ใช้กับระบบ

จากพื้นฐานของระบบผู้เชี่ยวชาญที่ได้กล่าวมาแล้วจะเห็นว่าการอนุมานระบบผู้เชี่ยวชาญมีลักษณะดังรูปที่ 3.3



รูปที่ 3.3 ลักษณะการอนุมาน

ในรูปนี้ฐานความรู้จะประกอบด้วยข้อเท็จจริง และกฎรวมกันเรียกว่า Static database และ Dynamic database สำหรับ Static Database นั้น จะเป็นฐานความรู้ที่ไม่เปลี่ยนแปลงในขณะที่เราใช้ระบบผู้เชี่ยวชาญ ส่วน Dynamic database จะเป็นฐานความรู้ที่เปลี่ยนแปลงไปในขณะของการใช้งาน

2. วิศวกรรมความรู้

ในการศึกษาวิจัยเกี่ยวกับเรื่องระบบผู้เชี่ยวชาญจะต้องเกี่ยวข้องกับระบบการจัดเก็บและแสดงความรู้ในรูปแบบต่าง ๆ เพื่อให้สามารถนำความรู้มาใช้งานได้อย่างเหมาะสมและมีประสิทธิภาพสูงสุด ดังนั้นจึงควรที่จะเข้าใจในหลักการของวิศวกรรมความรู้ในเบื้องต้นเสียก่อน

2.1 วิศวกรรมความรู้คืออะไร (5)

วิศวกรรมความรู้ (Knowledge engineering) เป็นแขนงย่อยอันหนึ่งของปัญญาประดิษฐ์ (Artificial intelligence) ที่เกี่ยวข้องกับการรับ การแสดงและการใช้ความรู้ของมนุษย์ในรูปของสัญญลักษณ์ ความรู้ของมนุษย์นั้นนอกจากจะอยู่ในรูปของความสามารถในการให้ข้อเท็จจริงต่าง ๆ แล้ว ยังรวมถึงความสามารถในการคิดตามเหตุผล (Reasoning) หรือการอนุมาน (Inference) ด้วย ถ้าการปฏิวัติทางอุตสาหกรรมเป็นการแทนแรงด้วยเครื่องจักร และการปฏิวัติคอมพิวเตอร์เป็นการแทนการคำนวณด้วยเครื่องจักรแล้ววิศวกรรมความรู้ก็ถือได้ว่าเป็นการแทนการอนุมานด้วยเครื่องจักร

คนที่ใช้คำว่า "วิศวกรรมความรู้" เป็นคนแรกเชื่อกันว่าเป็นศาสตราจารย์ E.A. Feigenbaum แห่งมหาวิทยาลัย Stanford ซึ่งใช้ในการประชุมทางวิชาการสาขาปัญญาประดิษฐ์ เมื่อปี ค.ศ. 1977 เป้าหมายหลักของวิศวกรรมความรู้คือความต้องการสร้างระบบ (ซอฟต์แวร์) คอมพิวเตอร์ที่ใช้งานได้จริง ๆ สำหรับแก้ปัญหาที่ยุ่งยากซับซ้อนซึ่งวิธีการประมวลผลแบบที่มีมาไม่สามารถใช้แก้ได้ ปัญหาที่จะแก้นั้นก็ไม่ใช่ปัญหาเด็กเล่นเช่น เกมต่าง ๆ ที่นักค้นคว้าปัญญาประดิษฐ์ในยุคต้น ๆ ชอบแก้กัน นอกจากนั้นก็ไม่ใช่ปัญหาที่มีขอบข่ายกว้าง แต่เป็นปัญหาเฉพาะในโลกจริงที่มีขอบข่ายแคบ อย่างเช่น การตรวจโรคติดเชื้อในเลือด หรือการออกแบบระบบคอมพิวเตอร์ให้เข้ากับความต้องการของลูกค้า ปัญหาเหล่านี้ถึงแม้จะมีขอบเขตแคบ แต่ถ้าหากเราต้องการสร้างโปรแกรมคอมพิวเตอร์ที่สามารถเหล่านี้ได้อย่างเท่าเทียบกับผู้เชี่ยวชาญ (Expert) แล้ว จะต้องสร้างโปรแกรมที่บรรจุความรู้เฉพาะสำหรับปัญหานั้นซึ่งมีจำนวนไม่น้อยด้วย

มนุษย์เราเมื่อพบกับปัญหาที่ไม่เคยพบมาก่อน หรือปัญหาที่มีลักษณะพิเศษส่วนใหญ่แล้วจะสามารถปรับตัวตอบสนองปัญหาเหล่านั้นได้ สาเหตุที่ทำให้มนุษย์เรามีความสามารถเช่นนี้คงจะเป็นเพราะมนุษย์มีความรู้พื้นฐานที่เกี่ยวข้องทั้งโดยทางตรงและทางอ้อม

กับปัญหานั้นอยู่ ในทำนองคล้ายกันถ้าหากจะสร้างโปรแกรมที่มีความยืดหยุ่นสามารถแก้ปัญหาที่ยุ่งยากซับซ้อนได้ นอกจากโปรแกรมนั้นจะต้องประกอบด้วยอัลกอรีธึมที่ดีและมีประสิทธิภาพแล้ว ยังจำเป็นต้องมี "ความรู้" ด้วย "ความรู้" นี้ใช้เป็นทั้งข้อมูลในการแก้ปัญหาและเป็นตัวชี้นำสำหรับอัลกอรีธึมต่าง ๆ

ความรู้ในวิศวกรรมความรู้นั้น แตกต่างกับข้อมูลในจุดหลักสองประการ คือ

ประการแรก ข้อมูลแสดงข่าวสาร (Information) อย่างชัดแจ้ง ส่วนความรู้นั้นแสดงข่าวสารอย่างทั้งชัดแจ้งและซ่อนรูป ตัวอย่างเช่น ประโยค "ทุเรียนลูกนี้หนัก 3 กิโลกรัม" อันนี้เป็นข้อมูลเพราะว่าผู้ฟังสามารถรับรู้ข่าวสารได้อย่างชัดแจ้งและทันที แต่ประโยคที่ว่า "ถ้าหากเป็นหวัด พยายามอย่าให้เปียกฝน" ถึงแม้ว่าจะเป็นความรู้สามัญที่เรารู้กันอยู่ แต่ส่วนของประโยคที่ว่า"อย่าให้เปียกฝน"นั้นไม่ได้บอกอะไรอย่างชัดเจนเลยว่าทำอย่างไรจึงไม่เปียกฝน แน่นอนมนุษย์เรารู้ข่าวสารที่ซ่อนรูปอยู่ กล่าวคือ รู้ว่าทำอย่างไรจึงไม่เปียกฝน (เช่น อยู่แต่ในบ้าน หรือถ้าต้องออกไปข้างนอกต้องถือร่ม) แต่สำหรับคอมพิวเตอร์การบรรจุความรู้ง่าย ๆ เช่นนี้ทำไม่ได้ง่าย ปัญหาอยู่ที่ว่าบันทึกส่วนของความรู้ที่ซ่อนรูปอยู่ได้อย่างไร

ประการที่สอง ความรู้แตกต่างจากข้อมูลตรงที่ว่าความรู้ส่วนใหญ่แสดงออกในรูปของภาษาธรรมชาติ (Natural language) เช่น ภาษาไทย ภาษาอังกฤษ ดังนั้นก่อนที่จะสามารถสร้างระบบคอมพิวเตอร์ที่บันทึกและประมวลผลความรู้ได้ จำเป็นจะต้องพัฒนาระบบคอมพิวเตอร์ที่เข้าใจธรรมชาติได้ก่อน ปัญหานี้ไม่ใช่แก้ได้ง่าย ๆ เพราะภาษาธรรมชาติมีลักษณะกำกวมอยู่มากซึ่งเป็นสิ่งที่คอมพิวเตอร์จัดการได้ยาก

จากที่ได้กล่าวมาทั้งหมด จะเห็นได้ว่า "ความรู้" ในวิศวกรรมความรู้นั้นไม่ใช่จะจัดการได้ง่าย ๆ จำเป็นต้องมีการค้นคว้าพัฒนาวิธีที่ดี ๆ ในวิศวกรรมความรู้หัวข้อการค้นคว้าที่เป็นหลัก ๆ ได้แก่

- การแสดงความรู้ (Knowledge representation) : จะบันทึกความรู้อย่างไร

- วิธีการใช้ความรู้หรือกลไกการอนุมาน (Inference mechanism) : จะผสมผสานความรู้แล้วใช้ให้บรรลุจุดประสงค์ได้อย่างไร

- การรับเอาและการจัดการความรู้ (Knowledge acquisition and management): ทำอย่างไรจึงจะทำให้ฐานความรู้สามารถเพิ่มพูนความรู้ได้โดยอัตโนมัติหรือกึ่งอัตโนมัติและจัดการฐานความรู้อย่างไรจึงจะไม่เกิดความขัดแย้งในระหว่างความรู้

ที่อยู่ในฐานความรู้ หรือความรู้ที่มีอยู่กับความรู้ที่รับเข้ามาใหม่

- การติดต่อกับผู้ใช้ (User interface): ทำอย่างไรจึงจะสร้างส่วนติดต่อกับผู้ใช้ที่ทำให้ผู้ใช้รู้สึกใช้ง่ายและยอมรับระบบหัวข้อนี้รวมวิธีการประมวลผลกราฟฟิกและภาษาธรรมชาติด้วย

หัวข้อค้นคว้าดังกล่าวข้างต้น โดยความเป็นจริงแล้วไม่ได้เป็นอิสระต่อกัน แต่มีความสัมพันธ์กัน โดยเฉพาะอย่างยิ่งการแสดงความรู้นั้นมีความสัมพันธ์กับหัวข้ออื่นอย่างลึกซึ้ง

2.2 เงื่อนไขการแสดงความรู้ที่ดี

ความรู้ที่มนุษย์เรามีอยู่นั้นมีหลายรูปแบบ ถึงแม้เราจะไม่รู้ว่าความรู้เหล่านี้ถูกเก็บอยู่ในสมองในรูปโครงสร้างแบบใด แต่การที่จะทำให้คอมพิวเตอร์สามารถใช้ความรู้ได้ เราจะเป็นจะต้องบันทึกความรู้ในรูปแบบโครงสร้างใดโครงสร้างหนึ่งเข้าไปในคอมพิวเตอร์ ปัญหานี้เราเรียกว่า การแสดงความรู้ การแสดงความรู้นี้เป็นหัวใจสำคัญของการสร้างระบบความรู้ และมีความสัมพันธ์อย่างลึกซึ้งกับการอนุมาน การรับเอาและการจัดการความรู้

ความรู้ถึงแม้จะมีอยู่หลายแบบ แต่พอจะแยกออกเป็นสี่ประเภทใหญ่ ๆ ได้ดังนี้

1. ความรู้ที่บอกความจริง ลักษณะคือคุณสมบัติ เช่น ที่ดินผืนนี้กว้าง 100 ตารางวา

2. ความรู้ที่บอกความสัมพันธ์เช่น นกเขาเป็นสัตว์ปีกชนิดหนึ่ง หรือแขนเป็นอวัยวะส่วนหนึ่งของมนุษย์

3. ความรู้ที่บอกขั้นตอนหรือวิธีการ เช่น ถ้าหากอุณหภูมิห้องสูงกว่า 40 องศาเซนติเกรด ให้ปิดเครื่อง

4. ความรู้ที่เกี่ยวกับความรู้ (Meta knowledge) เช่นความรู้เกี่ยวกับคุณลักษณะของความรู้อื่น หรือเกี่ยวกับวิธีการใช้ความรู้อื่น

ความรู้ประเภท 1 และ 2 นั้นได้จากความจริงที่เป็นรูปธรรมความรู้ประเภท 3 เป็นความรู้ประเภทที่อาจจะให้ชื่อได้ว่า "อัลกอริธึม" ส่วนความรู้ประเภท 4 นั้นอาจจะเข้าใจยาก ดังนั้นจะขอยกตัวอย่างประกอบสองตัวอย่างเพื่อช่วยให้เข้าใจง่ายขึ้น ตัวอย่างแรกเป็นตัวอย่างของความรู้เกี่ยวกับลักษณะของความรู้อื่น สมมติว่ามีฐานความรู้หนึ่งมีแต่ความรู้เกี่ยวกับเงินเดือนของพนักงานในบริษัท A (ความรู้ประเภท 1) และ

ความสัมพันธ์ระหว่างพนักงานในบริษัทนั้น(ความรู้ประเภท2) ว่าใครเป็นหัวหน้าใคร ความรู้ที่ว่าเงินเดือนของพนักงานที่เป็นหัวหน้าจะสูงกว่าพนักงานที่เป็นลูกน้อง ถือได้ว่าเป็น Meta knowledge สำหรับฐานความรู้นี้ เพราะบอกความสัมพันธ์ระหว่างความรู้เกี่ยวกับเงินเดือน ส่วนตัวอย่างที่สองเป็นตัวอย่างของความรู้เกี่ยวกับวิธีการใช้ความรู้อื่นสมมติว่ามีฐานความรู้เกี่ยวกับวิธีการแบบต่าง ๆ ในการผลิตสินค้าชนิดหนึ่ง ความรู้ที่ว่าให้เริ่มลองใช้วิธีการผลิตที่ใช้วัตถุดิบราคาถูกก่อนถือได้ว่าเป็น Meta knowledge สำหรับฐานความรู้นี้เพราะบอกวิธีการใช้ความรู้

Meta knowledge ที่บอกคุณลักษณะของความรู้อื่นนั้นช่วยในการจัดการ (management) ฐานความรู้ได้ อย่างเช่น ในตัวอย่างแรกข้างต้น ถ้าหากมีการพิมพ์เข้าไปในฐานความรู้ว่า เงินเดือนของนาย ก. (หัวหน้า) เท่ากับ 10,000 บาท และเงินเดือนของนาย ข.(ลูกน้อง)เท่ากับ 15,000 บาท โดยการใช้ Meta knowledge เราจะรู้ได้ทันทีเลยว่ามีอะไรผิดในความรู้ที่พิมพ์เข้าไปนี้เพราะเป็นไปไม่ได้ที่เงินเดือนของพนักงานหัวหน้าจะต่ำกว่าพนักงานลูกน้อง การจัดการความรู้เพื่อไม่ให้มีความขัดแย้งกันเองระหว่างความรู้หรือความผิดพลาดเป็นสิ่งจำเป็นและเป็นปัญหาสำคัญมาก โดยเฉพาะอย่างยิ่งเมื่อฐานความรู้มีขนาดใหญ่ขึ้น

Meta knowledge ที่บอกวิธีการใช้ความรู้อื่นนั้น ช่วยในการแก้ปัญหาให้มีประสิทธิภาพมากยิ่งขึ้น อย่างเช่น ถูกขึ้น ง่ายขึ้น หรือเร็วขึ้น Meta knowledge ประเภทนี้เปรียบเสมือนประสบการณ์ หรือ sense ในการแก้ปัญหาที่มนุษย์เรามีอยู่ ความรู้ประเภทนี้เป็นสิ่งที่ ES (Expert System) ที่ดีจะขาดเสียมิได้ ES จะดีหรือไม่ดีขึ้นอยู่กับ Meta knowledge นี้มาก เปรียบได้กับในคนเรา ถึงแม้จะจบการศึกษาจากที่เดียวกันมีคุณวุฒิเท่ากัน แต่ถ้าหากประสบการณ์ หรือ sense ในการแก้ปัญหาต่างกัน ความสามารถในการแก้ปัญหาจะต่างกันไปด้วย

วิธีการแสดงความรู้ที่ดีนั้น จะต้องมีคุณสมบัติดังต่อไปนี้

1. มีสมรรถภาพในการแสดงความรู้ชนิดต่าง ๆ ได้ กล่าวคือ จะต้องสามารถบันทึกความรู้ทั้งที่มีโครงสร้าง ความรู้ที่มีความไม่แน่นอน และความรู้ Meta knowledge เป็นต้น โดยการใช้โครงสร้างชนิดเดียวกัน และถ้าหากเป็นไปได้ โครงสร้างที่ใช้ในการแสดงความรู้จะต้องเป็นโครงสร้างที่ง่าย แต่มีสมรรถภาพในการแสดงออกสูง

2. มี modularity กล่าวคือ ความสามารถในการแยกออกเป็นส่วนย่อย (module) ทั้งนี้เพื่อให้สามารถเพิ่มหรือแก้ไขฐานความรู้ได้ง่าย คุณสมบัติอันนี้จำเป็นเพราะทำให้เกิดความยืดหยุ่นในการใช้ฐานความรู้

3. ง่ายต่อการจัดการ กล่าวคือ เป็นคุณสมบัติที่ช่วยในการตรวจสอบฐานความรู้ อย่างเช่น ช่วยในการตรวจดูความขัดแย้งระหว่างความรู้ การซ้ำกัน หรือความผิดพลาดในความรู้

4. ง่ายต่อการเข้าใจมนุษย์ การแสดงความรู้ที่ดีนอกจากจะให้เข้ากับคอมพิวเตอร์แล้วยังจะต้องให้เข้ากับมนุษย์ได้ดีด้วย คุณสมบัติอันนี้ช่วยทำให้การสร้าง ส่วนอธิบายใน ES ง่ายขึ้น ทั้งยังช่วยในการตรวจความผิดพลาดในการพิมพ์ความรู้เข้าไปในฐานความรู้ด้วย

5. เข้ากันได้ดีกับการอนุมาน ทั้งนี้เนื่องจากการอนุมานต้องใช้ความรู้ในฐานความรู้เป็นข้อมูล ดังนั้นเพื่อที่จะให้การอนุมานนี้ประสิทธิภาพดี การแสดงความรู้จะต้องเข้ากันได้ดีกับการอนุมาน

จากคุณสมบัติของการแสดงความรู้ที่ดีห้าประการดังกล่าวข้างต้น บางอย่างอาจจะขัดแย้งกัน ดังนั้นจึงเป็นการยากที่จะหาการแสดงความรู้ที่มีคุณสมบัติครบห้าประการได้

2.3 การแสดงความรู้ในรูปของกฎ (rule)

2.3.1 ประวัติและโครงสร้าง

การแสดงความรู้ในรูปของกฎมีชื่อเรียกอีกอย่างว่า การแสดงความรู้ในรูป Production System (PS) PS ถือได้ว่าเป็นโมเดลคอมพิวเตอร์ชนิดหนึ่งที่มีรากฐานทางทฤษฎีอยู่บน Post machine ที่เสนอโดย E.L.Post ในปี ค.ศ. 1943 อย่างที่เราทราบกันอยู่ เครื่องคอมพิวเตอร์ปัจจุบันมีพื้นฐานทางทฤษฎีอยู่บน Turing machine ใน Turing machine เราจะบันทึกขั้นตอนการควบคุมไว้ทั้งหมด การปฏิบัติหรือการประมวลผลของเครื่องจะเป็นไปตามขั้นตอนการควบคุมนั้น แต่ใน Post machine ขั้นตอนการปฏิบัติการหรือการประมวลผลจะถูกบันทึกอยู่ในรูปของเซ็ตของกฎ อันไหนจะถูกใช้ก่อนหลังนั้นไม่ได้ขึ้นอยู่กับลำดับการบันทึกกฎ แต่ขึ้นอยู่กับว่าเงื่อนไขของกฎนั้นครบสมบูรณ์หรือไม่ ถ้าครบก็จะมีการปฏิบัติการตามกฎนั้นมีคนพิสูจน์แล้วว่าทั้ง Turing

และ Post machines มีความสามารถในการประมวลผลเท่าเทียมกัน ดังนั้น ความสามารถในการประมวลผลของ PS จึงเท่ากับภาษาโปรแกรมมิงอื่น ๆ เช่น ฟอร์แทรน หรือปาสคาล กล่าวคือ โปรแกรมทุกโปรแกรมที่เขียนด้วยฟอร์แทรน หรือปาสคาลจะแทนได้ด้วยการใช้ PS

กฎใน PS จะอยู่ในรูป

IF .....

THEN .....

ส่วนของ IF เรียกว่า ส่วนเงื่อนไข และส่วนของ THEN เรียกว่า ส่วนข้อสรุปหรือส่วนการปฏิบัติ

. แสดงความรู้โดยใช้กฎ (IF....THEN....)
. modularity ของการแสดงความรู้ดี
. เข้าใจง่าย

รูปที่ 3.4 โครงสร้างของ production system และคุณสมบัติเฉพาะ

โครงสร้างของ PS จะประกอบด้วยส่วนย่อยหลักสามส่วนด้วยกัน ตามรูปที่ 3.4 คือ

1. ฐานกฎ (Rule base) หรือ Production memory
2. ส่วนตีความ (Interpreter) หรือส่วนอนุมาน
3. Working Memory (WM) หรือ Global database

ฐานกฎหรือ RB เป็นฐานความรู้ที่เก็บความรู้ที่อยู่ในรูปกฎ Working memory หรือ WM นั้น เป็นที่เก็บข้อมูลและสถานะของ PS ข้อมูลและสถานะใน WM เป็นอินพุตของส่วน IF ของกฎ และจะถูกอ้างอิงและเปลี่ยนแปลงโดยกฎใน RB ส่วน I นั้นจะ

ตรวจดูเนื้อหาใน RB และ WM แล้วก็จะเลือกกฎใดกฎหนึ่งจากเซ็ตของกฎที่มีเงื่อนไขครบขึ้นมาปฏิบัติการ

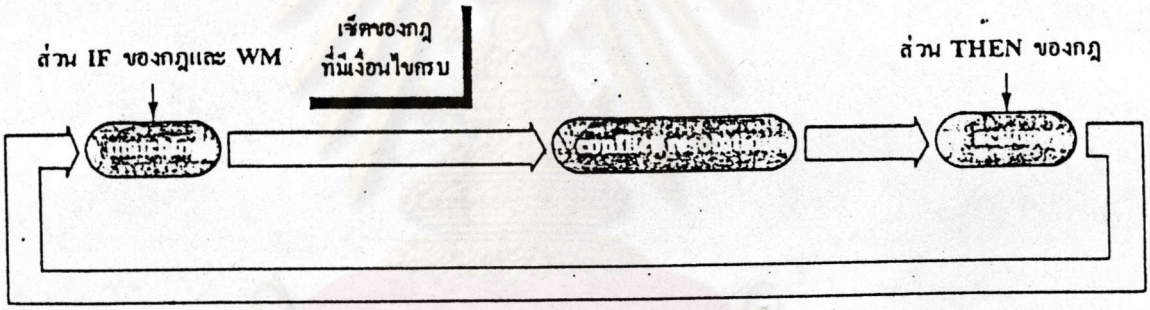
ในการปฏิบัติการแต่ละครั้งจะประกอบด้วยวงจรดังต่อไปนี้

1. Matching ทำการตรวจดูเนื้อหาของ WM และ RB เพื่อหากฎทั้งหมดที่มีเงื่อนไขพร้อม

2. Conflict resolution จากกฎที่หาได้จากการ matching จะมีการเลือกกฎที่เหมาะสมขึ้นมา 1 กฎ

3. Action ปฏิบัติการตามส่วน THEN ของกฎที่ได้รับการคัดเลือกในข้อ 2 ซึ่งบางการปฏิบัติการอาจจะเป็นการเปลี่ยนแปลงเนื้อหาของ WM

จากรูปที่ 3.5 จะแสดงวงจรดังกล่าวข้างต้น



รูปที่ 3.5 วงจรปฏิบัติการของ PS

2.3.2 ทิศทางการอนุมาน

ใน PS มีวิธีการอนุมานสองแบบใหญ่ ๆ ด้วยกัน คือ

1. การอนุมานแบบเดินหน้า (Forward chaining) การอนุมานแบบนี้เป็นแบบดั้งเดิมของ PS โดยความเป็นจริงแล้วรูปที่ 3.5 แสดงวงจรการปฏิบัติการของ PS ภายใต้การอนุมานแบบเดินหน้ากล่าวคือ PS จะเริ่มต้นจาก WM ---> หากฎ ----> ปฏิบัติการ ตามกฎ PS จะปฏิบัติการซ้ำ ๆ กันเช่นนี้จนกว่าจะได้คำตอบหรือบรรลุเป้าหมาย การอนุมานแบบนี้มีชื่ออื่นเรียกอีกว่า Data-driven inference หรือ Bottom-up inference

2.การอนุมานแบบย้อนหลัง (Backward chaining) การอนุมานแบบนี้เริ่มต้นจาก เป้าหมาย -----> กฎที่ทำให้เป้าหมายบรรลุผล ------>

เป้าหมายย่อย ---------> กฎที่ทำให้เป้าหมายย่อยบรรลุผล ทำซ้ำกันเช่นนี้จนกว่าจะพบว่าข้อมูลที่จำเป็นที่ทำให้เป้าหมายย่อยทั้งหมดบรรลุผลนั้นมีอยู่ใน WM ในกรณีที่ค้นไม่พบข้อมูลดังกล่าวอาจจะต้องทำการย้อนรอย (Backtracking) และลองเปลี่ยนเป้าหมายย่อยระหว่างทางเสียใหม่ การอนุมานแบบนี้มีชื่อเรียกอย่างอื่นว่า Goal driven inference หรือ Top-down inference

2.3.3 การเลือกกฎ

หลังจากทำการ matching แล้ว อาจจะได้กฎหลายกฎที่มีเงื่อนไขพร้อม ดังนั้นจึงจำเป็นจะต้องเลือกกฎใดกฎหนึ่งขึ้นมา วิธีการเลือกกฎอาจจะทำได้หลายวิธีดังนี้

1. เลือกตามลำดับความสำคัญของกฎ วิธีการนี้จะนำกฎมาทำการ matching ตามลำดับความสำคัญของกฎ กฎแรกที่การ matching ประสบความสำเร็จจะได้รับการปฏิบัติการทันที

2. เลือกตามลำดับความละเอียดของส่วน IF ของกฎ กฎไหนที่ส่วน IF มีการบันทึกเงื่อนไขไว้ละเอียดหรือเฉพาะที่สุดจะได้รับเลือกก่อน เช่น ถ้ามี 2 กฎ

- กฎ ก.IF X เป็นนักเรียน THEN X อ่านข้อความนี้เข้าใจ
- กฎ ข.IF X เป็นนักเรียนชั้น ม.6 THEN X อ่านข้อความนี้เข้าใจ

3. เลือกตามความใหม่ของกฎ กฎที่ได้รับการปฏิบัติการล่าสุดจะได้รับการเลือกก่อน

4. เลือกตามความใหม่ของข้อมูลใน WM เลือกกฎที่ match กับข้อมูลล่าสุดใน WM ไว้ก่อน

5. แบบขนาน : ให้กฎทุกกฎที่ match สำเร็จ (ส่วนเงื่อนไขพร้อม) ปฏิบัติการพร้อมกัน

2.3.4 ข้อดีข้อเสียของ PS

PS มีข้อดีดังต่อไปนี้

1. มี modularity สูง เนื่องจากกฎแต่ละกฎมีความสมบูรณ์ในตัวเอง และตัวแปรที่ใช้ในกฎหนึ่งก็มีขอบเขตแค่ภายในกฎนั้นเท่านั้น เช่น ตัวแปร x ตัวเดียวกันที่ปรากฏอยู่ในกฎสองกฎจะแทนสิ่งที่ต่างกัน

2. ง่ายต่อการเข้าใจ เนื่องจากการแสดงความรู้ในกฎ IF.... THEN.... นั้นมนุษย์เราคุ้นเคยอยู่แล้ว จึงเป็นการง่ายต่อการเข้าใจ

3. มีโครงสร้างที่ง่าย ทำให้การสร้างอธิบายใน ES ทำได้ไม่ยาก

ส่วนข้อเสียมีดังนี้

1. ในการ matching ส่วนใหญ่จะใช้วิธีลองผิดลองถูก ดังนั้นถ้ากฎหรือข้อมูลใน WM มีจำนวนมาก เวลาที่ใช้ในการ matching จะมากจนยอมรับไม่ได้ในทางปฏิบัติ

2. ยากต่อการบันทึกความสัมพันธ์ระหว่างความรู้ และผลกระทบซึ่งกันและกันของความรู้ที่มีความสัมพันธ์กัน อันนี้เนื่องมาจาก PS มี modularity สูงมากจนยากจะบันทึกโครงสร้างของความรู้ได้

3. <u>การพัฒนาเทคโนโลยีระบบผู้เชี่ยวชาญ</u>

การพัฒนาเทคโนโลยีระบบผู้เชี่ยวชาญ (12) แสดงได้ดังตารางที่ 3-1 ซึ่งเป็นการสรุปการพัฒนาที่สำคัญในช่วงปี 1950 กับ 1980

ในช่วงทศวรรษ ค.ศ.1950 ระบบผู้เชี่ยวชาญที่มีชื่อเสียงมากที่สุดคือ General Problem Solver สร้างขึ้นโดย Newell และ Simon ซึ่งเป็นการแสดงการทำงานในลักษณะ Human Problem Solving โดยสร้างเป็นกฎที่เรียกว่า Chunk แทนความรู้ของมนุษย์ เมื่อกฎหลาย ๆ กฎมารวมกันเรียกว่า Long-term memory เช่น

IF there is flame THEN there is a fire.

IF there is smoke THEN there may be a fire.

IF there is siren THEN there may be a fire.

ส่วนการเก็บความรู้ไว้ชั่วคราวในระหว่างการแก้ไขปัญหา เรียกว่า Short-term memory หรือในปัจจุบันเรียกว่า Working memory อีกส่วนหนึ่งตัวกระตุ้นให้เกิดการตัดสินใจจะเรียกว่า Cognitive processor หรือปัจจุบันเรียกว่า Inference engine

ในช่วงทศวรรษ ค.ศ. 1960 มีระบบผู้เชี่ยวชาญที่มีประสิทธิภาพสูงและค้นพบหลักการของ Domain Knowledge นั้นจะมีประสิทธิภาพสูงกว่าในลักษณะของ General Problem Solver ตัวอย่างการสร้างระบบผู้เชี่ยวชาญสำหรับแก้ไขปัญหาอย่างหนึ่งอย่างใดโดยเฉพาะ เช่นการเล่นหมากรุก ก็จะแก้ไขปัญหาการเล่นหมากรุกอย่างเดียวเท่านั้น ไม่สามารถแก้ปัญหาทางด้านคณิตศาสตร์ได้

ในช่วงทศวรรษ ค.ศ. 1970 จะใช้หลักการของ Domain Knowledge เป็นหลักในการสร้างระบบผู้เชี่ยวชาญ มีระบบผู้เชี่ยวชาญที่สำคัญ ได้แก่ DENDRAL-สำหรับวิเคราะห์องค์ประกอบของสารเคมี MYCIN-สำหรับวินิจฉัยรักษาโรค DIPMETER-สำหรับวิเคราะห์ข้อมูลทางธรณีวิทยาเพื่อหาแหล่งน้ำมัน PROSPECTOR-สำหรับวิเคราะห์ข้อมูลทางธรณีวิทยาเพื่อหาแร่ธาตุ XCON/R1 ของบริษัท DEC-สำหรับวิเคราะห์องค์ประกอบของระบบคอมพิวเตอร์ตามความต้องการของลูกค้าที่สั่งซื้อ เป็นต้น

ระบบผู้เชี่ยวชาญ MYCIN มีจุดเด่นหลายประการคือ

1. เป็นโปรแกรมที่สามารถนำ AI เข้ามาแก้ไขปัญหาในทางปฏิบัติได้อย่างเหมาะสม

2. MYCIN มีลักษณะการให้คำอธิบาย การสอบถามความรู้อัตโนมัติและการเป็นผู้ฝึกสอนที่ดีสำหรับผู้เชี่ยวชาญในการทดสอบความรู้

3. MYCIN เป็นผู้เริ่มต้นแนะนำการใช้ Shell ในการสร้างระบบผู้เชี่ยวชาญ

ในปี ค.ศ. 1980 ได้เริ่มมีการนำเอาระบบผู้เชี่ยวชาญจากห้องทดลองในมหาวิทยาลัยมาผลิตทางการค้า ระบบผู้เชี่ยวชาญที่มีประสิทธิภาพในขณะนั้นได้แก่ ART (Automated Reasoning Tool) โดย Intelli Corp. และ Rule Master โดย Radian Corp. นอกจากนั้นยังพัฒนา Hardware ในลักษณะ Work station ให้ทำงานกับ Software ให้มีประสิทธิภาพสูงขึ้นกว่าเดิม เช่น บริษัท Symbolic & LMI ได้พัฒนา Symbolics machines และ LISP machines เพื่อใช้กับภาษา LISP บริษัท Tentronix ได้พัฒนา Textronix work stations เพื่อใช้ภาษา SMALLTALK,LISP, MPROLOG เป็นต้น

ตารางที่ 3.1 ประวัติความเป็นมาของระบบผู้เชี่ยวชาญ

Year	*Events*
1943	Post production rules; McCulloch and Pitts Neuron Model
1954	Markov Algorithm for controlling rule execution
1956	Dartmouth Conference; Logic Theorist; Heuristic Search; "AI" term coined
1957	Perceptron invented by Rosenblatt; GPS (General Problem Solver) started (Newell, Shaw and Simon)
1958	LISP AI language (McCarthy)
1962	Rosenblatt's *Principles of Neurodynamics* on Perceptions
1965	Resolution Method of automatic theorem proving (Robinson) Fuzzy Logic for reasoning about fuzzy objects (Zadeh) Work begun on DENDRAL, the first expert system (Feigenbaum, Buchanan, et.al.)
1968	Semantic nets, associative memory model (Quillian)
1969	MACSYMA math expert system (Martin and Moses)
1970	Work begins on PROLOG (Colmerauer, Roussell, *et. al.*)
1971	HEARSAY I for speech recognition *Human Problem Solving* popularizes rules (Newell and Simon)
1973	MYCIN expert system for medical diagnosis(Shortliffe, *et. al.*) leading to GUIDON, intelligent tutoring (Clancey) and TEIRESIAS, explanation facility concept (Davis) and EMYCIN, first shell (Van Melle, Shortliffe and Buchanan) HEARSAY II, blackboard model of multiple cooperating experts
1975	Frames, knowledge representation (Minsky)
1976	AM (Artificial Mathematician) creative discovery of math concepts (Lenat) Dempster-Shafer Theory of Evidence for reasoning under uncertainty Work begun on PROSPECTOR expert system for mineral exploration (Duda, Hart, *et. al.*)
1977	OPS expert system shell (Forgy), used in XCON/R1
1978	Work started on XCON/R1 (McDermott, DEC) to configure DEC computer systems Meta-DENDRAL, metarules and rule induction (Buchanan)
1979	Rete Algorithm for fast pattern matching (Forgy) Commercialization of AI begins Inference Corp. formed (releases ART expert system tool in 1985)
1980	Symbolics, LMI founded to manufacture LISP machines
1982	SMP math expert system; Hopfield Neural Net; Japanese Fifth Generation Project to develop intelligent computers
1983	KEE expert system tool (IntelliCorp)
1985	CLIPS expert system tool (NASA)

4. การนำระบบผู้เชี่ยวชาญไปใช้งาน

4.1 เครื่องมือในการสร้างระบบผู้เชี่ยวชาญ

4.1.1 ภาษาคอมพิวเตอร์ (Programming Language) เช่น โฟร์แทรน ปาสคาล โคบอล ดิเบสทู ดีเบสทรี โปรล็อก เป็นต้น จุดดีที่เราใช้ภาษาคอมพิวเตอร์ก็คือ มีความยืดหยุ่น และสร้างได้ในรูปแบบที่ต้องการ แต่จะสิ้นเปลืองเวลาเพราะเราต้องเริ่มจากศูนย์

4.1.2 เปลือก (Shell) คือระบบผู้เชี่ยวชาญที่มีแต่เปลือก ไม่มีอะไรอยู่ในฐานข้อมูลเลย แต่ส่วนอื่น ๆ เช่น กลไกวินิจฉัย ส่วนอธิบายจะมีการพิจารณาเอาไว้ให้เราสามารถนำมาใช้ได้ ไม่ว่าจะเป็นปัญหาการวิเคราะห์การวินิจฉัยโรคทางการแพทย์ การสำรวจแหล่งน้ำมัน ประโยชน์ของเปลือกคือ ทำให้เราสามารถพัฒนาระบบผู้เชี่ยวชาญไปได้อย่างรวดเร็ว แต่มีข้อเสียคือ ไม่มีความยืดหยุ่น เนื่องจากมีการกำหนดโครงสร้างการแสดงความรู้และกลไกการวินิจฉัย

4.2 การประยุกต์ใช้ระบบผู้เชี่ยวชาญ

เราสามารถใช้ระบบผู้เชี่ยวชาญได้ในด้าน

4.2.1 ตีความหมายข้อมูล (Interpretation) คือรับข้อมูลจาก Sensor เข้ามาแล้วตีความข้อมูล ซึ่งจะประยุกต์ใช้ในด้าน Remote sensing หรือตีความข้อมูลในการขุดเจาะน้ำมันเพื่อดูว่าหลังจากการส่งข้อมูลไปแล้ว ตัดสินใจว่าบริเวณเหล่านั้นมีแหล่งน้ำมันหรือไม่

4.2.2 วินิจฉัยความบกพร่อง (Diagnosis) เช่น การวินิจฉัยโรคต่าง ๆ การวินิจฉัยความบกพร่องของเครื่องจักร เราจะมีการทดสอบ และบอกขั้นตอนว่าต้องแก้ไขอย่างไร

4.2.3 ดูแลและควบคุม (Monitoring and Control) หมายถึงระบบ Process control ทั้งหลาย สมมติว่าอุณหภูมิสูงขึ้นต้องตัดความร้อน ซึ่งแทนที่จะใช้คนก็ใช้ระบบผู้เชี่ยวชาญทำได้ เหมาะกับใช้ในงานด้านอุตสาหกรรม

4.2.4 การวางแผนและออกแบบ (Planning and Design) เป็นเสมือน Advisor ว่าถ้าข้อมูลเข้ามาแบบนี้ สถานการณ์เป็นแบบนี้ เราควรจะวางแผนอย่างไร

4.2.5 การเรียนการสอน (Learning) เราใช้ Computer - aided

instruction หรือ Computer-aided learning เนื่องจากระบบผู้เชี่ยวชาญมีส่วนอธิบายส่วนให้เหตุผล จะใช้กันมากกับการสอนนักศึกษาแพทย์ วิศวกรรมศาสตร์

4.3 ลักษณะของระบบผู้เชี่ยวชาญที่ดี (12)

4.3.1 มีประสิทธิภาพสูง (High performance) ระบบจะต้องมีความสามารถเทียบเท่าหรือดีกว่าผู้เชี่ยวชาญ ซึ่งหมายความว่าคุณภาพการให้คำแนะนำต้องสูง

4.3.2 ใช้เวลาทำงานรวดเร็ว (Adiquate response time) ระบบจะต้องทำงานด้วยเวลาที่สมเหตุสมผล คือเทียบเท่าหรือเร็วกว่าเวลาที่ผู้เชี่ยวชาญตัดสินใจ

4.3.3 มีความน่าเชื่อถือ (Good reliability) ระบบผู้เชี่ยวชาญจะต้องน่าเชื่อถือและไม่ล่าเอียง

4.3.4 เข้าใจง่าย (Understandable) ระบบสามารถอธิบายขั้นตอนการให้เหตุผลในขณะทำงานให้เข้าใจได้

4.4 ข้อดีของระบบผู้เชี่ยวชาญ

ระบบผู้เชี่ยวชาญมีข้อดีในตัวมันเองหลายประการคือ

4.4.1 ทำให้มีผู้เชี่ยวชาญเพิ่มขึ้นได้ตลอดเวลา

4.4.2 ลดค่าใช้จ่ายให้กับผู้เชี่ยวชาญ น้อยลง

4.4.3 ระบบผู้เชี่ยวชาญสามารถใช้งานในสภาพแวดล้อมที่เป็นอันตรายต่อมนุษย์ได้

4.4.4 เป็นผู้เชี่ยวชาญได้ถาวร ไม่มีกำหนด

4.4.5 สามารถทำงานพร้อม ๆ กัน ในเวลาเดียวกันได้

4.4.6 ระบบผู้เชี่ยวชาญสามารถเพิ่มความน่าเชื่อถือว่าเป็นการตัดสินใจที่ถูกต้องเพราะนำความคิดเห็นจากผู้เชี่ยวชาญหลายคน

4.4.7 ระบบผู้เชี่ยวชาญสามารถอธิบายในรายละเอียดของเหตุผลได้ตลอดเวลา

4.4.8 สามารถโต้ตอบได้ทันทีขณะใช้งาน

4.4.9 สามารถทำงานอย่างต่อเนื่องไม่มีอารมณ์เข้ามาเกี่ยวข้อง ไม่มีความกดดัน ไม่มีความเหนื่อยล้า

4.4.10 ระบบผู้เชี่ยวชาญ สามารถเป็นครูฝึกสอนที่ดีแก่นักศึกษาได้

Name	Reference	Domain	Classification	Knowledge representation	Reasoning	Language	Conflict resolution
GARI	Descotte & Latombe (1981)	m/c process plans	planning	rules	backward	MACLISP	weights
TOM	Matsushima *et al.* (1982)	m/c process plans (holes)	planning	rules	backward	PASCAL	heuristics
SIPP	Nau & Chang (1985)	m/c process selection	planning	rules & frames	backward	PROLOG	weights
AGMPO	Eshel *et al.* (1985)	forming (deep drawing)	planning	rules	forward	PROLOG	rule ordering
EXCAP	Davis & Darbyshire (1984)	m/c process planning	planning	rules	backward	PASCAL	weights
CUTTECH	Barkocy & Zdeblick (1984)	m/c parameter selection	planning	rules	forward	*	rule ordering
PWA Planner	Terwilliger (1985)	printed wiring assembly	planning	rules	forward	PROLOG	rule ordering
ISIS-II	Fox (1983)	job-shop scheduling	planning	rules & frames	forward	SRL LISP	heuristics
KBRS	Ben-Arieh *et al.* (1985)	job-shop scheduling	planning	rules	forward	PROLOG	rule ordering
FADES	Fisher (1984)	Facilities design	planning	rules	forward	PROLOG	*
IFLAPS	Tirupatikumara (1985)	Facilities design	planning	rules	forward	PROLOG	heuristics
DELTA	Bonissone (1983)	locomotive trouble shooting	diagnosis & repair	rules	hybrid	FORTH	*
FOREST	Finin (1984)	fault diagnosis in ATE	diagnosis	rules & frames	forward	PROLOG	weights

* information not available

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย