

### การออกแบบระบบผู้เชี่ยวชาญที่ทำการวิจัย

เนื้อหาในบทนี้จะเกี่ยวกับการออกแบบระบบผู้เชี่ยวชาญที่ทำการวิจัย โดยตั้งชื่อว่า ระบบ SFES (Shrimp Farming Expert System) พร้อมทั้งแสดงรายละเอียดในส่วนประกอบต่างๆ ของระบบซึ่งแบ่งออกเป็น 2 ส่วนคือ การออกแบบส่วนฐานความรู้ของระบบ ประกอบด้วย ฐานความรู้ กฎความรู้ และฐานข้อมูลพลวัตที่เกิดขึ้นในระหว่างการทำงานของระบบ อีกส่วนหนึ่งคือ ส่วนที่ควบคุมการทำงานของระบบประกอบด้วยโปรแกรมย่อย MONITOR โปรแกรมย่อย FINDOUT และโปรแกรมย่อย EXPLANATION โดยจะกล่าวถึงหน้าที่ และความสำคัญของโปรแกรมย่อยเหล่านี้

การออกแบบระบบ SFES ผู้วิจัยได้ศึกษาเทคนิคและรูปแบบการทำงานของระบบผู้เชี่ยวชาญ MYCIN เป็นตัวอย่าง และทำการพัฒนาระบบใหม่เพื่อให้ได้ระบบผู้เชี่ยวชาญ SFES ที่เหมาะสมกับการแก้ปัญหาสำหรับการเพาะเลี้ยงกุ้งแบบพัฒนา รายละเอียดการเปรียบเทียบส่วนประกอบของระบบทั้งสองได้ทำการสรุปเป็นตารางเปรียบเทียบในภาคผนวก ง.

#### 1. การออกแบบวิธีการแทนค่าความรู้ในรูปของกฎความรู้

ในการพัฒนาระบบผู้เชี่ยวชาญ SFES ความรู้ในการเพาะเลี้ยงกุ้งจะ ได้จากการค้นคว้าเอกสารการเพาะเลี้ยงของกรมประมง รายงานวิชาการ รายงานการประชุมต่างๆ ที่เกี่ยวข้อง และมีการสอบถามขอคำแนะนำจากนักวิจัยของสถานีวิจัยวิทยาศาสตร์ทางทะเล และศูนย์ฝึกนิสิต เกาะสีชัง นอกจากนี้มีการศึกษาสภาพการทำงานจริงของนักวิจัยที่สถานีวิจัยฯ ความรู้ที่ได้จะเกี่ยวกับการเพาะพันธุ์ลูกกุ้งจากพ่อแม่พันธุ์กุ้ง เทคนิคการเร่งให้แม่กุ้งมีไข่ การเลี้ยงดูพ่อแม่พันธุ์กุ้ง ความรู้เหล่านี้ผู้วิจัยได้ทำการแยกประเภทของปัญหาเป็นกลุ่มปัญหาย่อย และกำหนดเนื้อหาของแต่ละปัญหาย่อยรวมทั้งพารามิเตอร์ของเนื้อหาความต่าง ๆ โดยกำหนดโครงสร้างข้อมูลเป็นแบบ OBJECT-ATTRIBUTE-VALUE TRIPLETS แล้วสร้างเป็นฐานความรู้แบบสถิตย์ขึ้น

##### 1.1 องค์ประกอบของกฎความรู้

ฐานความรู้เป็นส่วนที่เก็บความรู้ทั้งหมด โดยออกแบบให้เก็บอยู่ในรูปของกฎความรู้ ซึ่งมีส่วนประกอบดังนี้

<rule> ::= <premise><action>

```

<premise> ::= (<condition>...<condition>)
<action> ::= (<conclude><parameter><context><value>);
              (<recommend><parameter><context><value>);
              (<treatment><parameter><context><value>)
<condition> ::= (<func><parameter><context><value>)
<func> ::= <more>!<less>!<same>
<context> ::= <context name>
<parameter> ::= <parameter name>

```

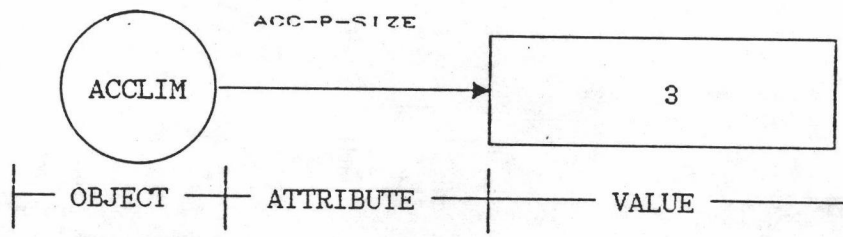
กฎความรู้ทั้งหมดจะเก็บไว้ในตัวแปรชื่อ knowledge\_base โดยเก็บเป็น list ซึ่งมีสมาชิกแต่ละตัวคือกฎความรู้แต่ละกฎ ซึ่งมี RULE### เป็นหมายเลขของกฎความรู้ สามารถแบ่งความรู้เกี่ยวกับการเพาะเลี้ยงกุ้งทั้งหมดมี 3 ส่วนคือ

- การเพาะแม่พันธุ์ลูกกุ้ง และการอนุบาลลูกกุ้ง (larvae culture)
- การเลี้ยงลูกกุ้งวัยรุ่น และกุ้งโตเต็มวัย (postlarvae & mature culture)
- โรคที่เกิดกับกุ้ง (Shrimp disease)

แต่ละประโยคเงื่อนไขของส่วนหลักฐานในกฎความรู้ จะเป็นโครงสร้างข้อมูลพื้นฐานในการแทนค่าความรู้ในรูปของกฎความรู้โดยแทนค่าอยู่ในรูป OBJECT-ATTRIBUTE-VALUE TRIPLETS ซึ่งสามารถแสดงตัวอย่างประโยคเงื่อนไขและลักษณะการแทนค่าความรู้ดังรูปที่ 3.1 เช่น ตัวอย่างประโยคเงื่อนไข (less acc\_p\_size ACCLIM 3)

ในที่นี้จะขอยกตัวอย่างการออกแบบฐานความรู้ เนื้อความ และพารามิเตอร์ที่สำคัญโดยละเอียดในฐานความรู้ชื่อ knowledge\_base1 เท่านั้น เนื่องจากลักษณะการออกแบบและคุณสมบัติต่างๆ ของพารามิเตอร์จะมีโครงสร้างที่คล้ายกันจะแตกต่างกันที่ และการทำงานของแต่ละพารามิเตอร์ตามความเกี่ยวข้องของพารามิเตอร์กับความรู้ในเรื่องกุ้งเท่านั้น กฎต่าง ๆ ในฐานความรู้ชื่อ knowledge\_base1 จะแบ่งออกเป็นกลุ่มต่างๆ ตามประเภทของเนื้อความที่เกี่ยวข้องกับกฎความรู้นั้น ๆ ดังนี้

Problem-rule	เป็นกลุ่มของกฎที่เกี่ยวข้องกับเนื้อความของปัญหาทั้งหมดที่เกิดขึ้นในการเพาะเลี้ยง
Ecause-rule	เป็นกลุ่มของกฎที่เกี่ยวข้องกับเนื้อความของปัญหาย่อยที่เกิดจากไขที่ฟักได้จากแม่พันธุ์กุ้ง
Fcause-rule	เป็นกลุ่มของกฎที่เกี่ยวข้องกับเนื้อความของปัญหาย่อยที่เกิด



รูปที่ 3.1 แสดงตัวอย่างประโยคเงื่อนไขของกฎความรู้  
และโครงสร้างข้อมูลที่เก็บอยู่ภายในระบบ

ขึ้นจากพ่อพันธุ์แม่พันธุ์กึ่ง

Acclim-rule เป็นกลุ่มของกฎที่เกี่ยวกับเนื้อหาของปัญหาย่อยที่เกิด

ขึ้นจากการปรับสภาพพ่อพันธุ์แม่พันธุ์กึ่ง

ตัวอย่างกฎความรู้ของระบบ SFES ที่แสดงในภาษา LISP เช่น

กฎที่ 001 เป็นกฎประเภท Problem-rule ซึ่งเป็นตัวบอกสาเหตุที่ทำให้เกิดจุดเนื้อความปัญหาขึ้น

```
(RULE001 ((less spawned_fm FCAUSE 70))
```

```
((conclude problem_status PROBLEM yes)))
```

กฎนี้เป็นกฎหมายเลข 001 ประกอบด้วยส่วนหลักฐานที่มีประโยคเงื่อนไข 1 ประโยค และส่วนกระทำซึ่งมีประโยคกระทำ 1 ประโยคเช่นกัน ประโยคเงื่อนไขประกอบด้วยฟังก์ชัน less ใช้ในการประเมินค่าพารามิเตอร์ชื่อ spawned\_fm สำหรับเนื้อความ FCAUSE ในต้นไม้เนื้อความโดยดูจากค่าความจริงในหน่วยความจำที่เก็บเหตุการณ์ ประโยคกระทำในส่วนกระทำจะประกอบด้วยฟังก์ชัน conclude ใช้สรุปค่าของพารามิเตอร์ problem\_status สำหรับเนื้อความ PROBLEM ว่าเป็นจริงและทำการเก็บในหน่วยความจำที่เก็บเหตุการณ์ นั่นคือถ้าพบว่าแม่พันธุ์กึ่งที่สามารถเร่งให้ไข่แก่ได้มีน้อยกว่า 70% แล้ว แสดงว่าสถานะของการเกิดปัญหาเนื่องจากแม่พันธุ์กึ่งเป็นจริง ลักษณะการออกแบบส่วนหลักฐานจะออกแบบให้ประโยคเงื่อนไขเชื่อมกันด้วย and เช่น กฎเลขที่ 203

กฎที่ช่วยสรุปว่าลักษณะการปรับสภาพกึ่งก่อนการบีบตาไม่เหมาะสม

```
(RULE203 ((less acc_p_size ACCLIM 3)
```

```
(more acc_p_gravid_density ACCLIM 80))
```

```
((conclude acc_p_feature ACCLIM unsuitable)))
```

หมายความว่า ถ้าขนาดความจุของบ่อปรับสภาพพ่อพันธุ์แม่พันธุ์กึ่ง (acc\_p\_size) น้อยกว่า 3 ลูกบาศก์เมตร และความหนาแน่นของพ่อพันธุ์แม่พันธุ์กึ่งที่อยู่ในบ่อมากกว่า 80 ตัวจะสามารถสรุปได้ว่า ลักษณะการปรับสภาพกึ่งก่อนทำการบีบตาไม่เหมาะสมเนื่องจากขนาดความจุของบ่อปรับสภาพน้อยเกินไป และความหนาแน่นของพ่อพันธุ์แม่พันธุ์กึ่งในบ่อปรับสภาพมากเกินไป



ประโยคเงื่อนไขที่เชื่อมกันด้วย OR จะไม่ปรากฏอยู่ในส่วนหลักฐานของกฎแต่จะแยกออกเป็นกฎ ความรู้กฎหนึ่ง โดยที่ทั้ง 2 กฎนี้จะให้ข้อสรุปที่เหมือนกันนั่นคือ ไม่ว่ากฎหนึ่งกฎใดหรือทั้ง 2 กฎ เป็นจริง ก็จะทำให้ข้อสรุปเดียวกัน เช่น

```
(RULE100 (OR (less spawned_fm FCAUSE 70)
              (less fert_egg ECAUSE 200000))
          ((conclude problem_status yes)))
```

จะสามารถแยกออกเป็นกฎ 2 กฎได้ดังนี้คือ

```
(RULE001 ((less spawned_fm FCAUSE 70))
          ((conclude problem_status PROBLEM yes)))
(RULE002 ((less fert_egg ECAUSE 200000))
          ((conclude problem_status PROBLEM yes)))
```

## 1.2 เนื้อความและต้นไม้เนื้อความ

เนื้อความหมายถึงปัญหาย่อยชนิดต่างๆ ที่แบ่งมาจากปัญหาทั้งหมด เนื้อความย่อยเหล่านี้จะสามารถแบ่งย่อยเป็นเนื้อความย่อยๆ ลงไปได้อีก ในแต่ละชนิดเนื้อความย่อย จะมีพารามิเตอร์สำคัญต่าง ๆ ที่เกี่ยวข้อง ชื่อเนื้อความในระบบ SFES จะใช้ตัวอักษรตัวใหญ่ เพื่อให้แตกต่างกับชื่อฟังก์ชันในโปรแกรมและชื่อพารามิเตอร์ซึ่งใช้ตัวอักษรตัวเล็ก ตัวอย่างชนิดเนื้อความย่อยในฐานความรู้ knowledge\_base1 เช่น

PROBLEM เป็นเนื้อความที่อยู่ในระดับสูงสุดของต้นไม้เนื้อความ (root node) จะหมายถึงปัญหาทั้งหมดที่เกิดขึ้นในการเพาะเลี้ยงกุ้ง

FCAUSE เป็นเนื้อความที่แทนปัญหาที่เกิดกับพ่อแม่พันธุ์แม่พันธุ์กุ้งที่เร่งให้ไข่แก่ได้น้อยกว่าปริมาณเฉลี่ย

ECAUSE เป็นเนื้อความที่แทนปัญหาที่เกิดเนื่องจากไข่ที่ได้จากการเพาะฟักโดยพ่อแม่พันธุ์แม่พันธุ์กุ้งมีน้อยกว่าค่าเฉลี่ยทั่วไป

ACCLIM เป็นเนื้อความที่แทนปัญหาที่เกิดจากสาเหตุในการปรับสภาพของพ่อแม่พันธุ์แม่พันธุ์กุ้ง

คุณสมบัติที่ใช้ในการทำงานของเนื้อความจะมีทั้งหมด 7 ประเภทคือ

ก. PROMPT1 เป็นรูปประโยคที่กำหนดให้ระบบใช้ถามผู้ใช้ระบบว่าจุดแรกของเนื้อความย่อชนิดนั้น ๆ จำเป็นต้องเพิ่มลงในต้นไม้เนื้อความหรือไม่ ใช้ในกรณีที่เริ่มทำการสร้างเนื้อความชนิดนั้นเป็นจุดแรกในต้นไม้เนื้อความ

ข. PROMPT2 เป็นรูปประโยคที่ใช้ถามผู้ใช้ระบบว่ามีจุดเนื้อความชนิดนั้น จุดอื่นที่สร้างเพิ่มในต้นไม้เนื้อความอีกหรือไม่ ใช้ในกรณีที่มีเนื้อความชนิดนั้นเกี่ยวข้องกับปัญหาในต้นไม้เนื้อความมากกว่า 1 จุด

ค. TRANS เป็นวลีในภาษาเขียนตามปกติใช้แทนชื่อชนิดของเนื้อความชนิดนั้นเมื่อถูกอ้างอิง

ง. TYPE เป็นชื่อชนิดของเนื้อความชนิดนั้นๆ ที่ใช้ในการสร้างจุดในต้นไม้เนื้อความเช่น "FCAUSE-" ซึ่งเมื่อระบบเชี่ยวชาญ SFES พบว่าเนื้อความชนิดนี้เกี่ยวข้องกับการทำงานจะทำการกำหนดหมายเลขชื่อให้กับเนื้อความนั้นตามลำดับโดยไม่ซ้ำกันเช่น FCAUSE-1 FCAUSE-2 เป็นต้น

จ. MAINPROPS เป็นรายการของพารามิเตอร์ที่เป็นตัวกำหนดลักษณะสำหรับเนื้อความย่อชนิดนั้นๆ โดยระบบจำเป็นต้องรู้ค่าของพารามิเตอร์เหล่านี้เพื่อใช้สรุปเนื้อความย่อชนิดนั้น

ฉ. ASSOCWITH เป็นคุณสมบัติที่บอกชนิดเนื้อความที่อยู่ในระดับสูงกว่าเนื้อความชนิดนั้น (parent node) ในต้นไม้เนื้อความ

ช. SUBJECT เป็นชื่อประเภทของกฎที่เนื้อความชนิดนั้น ๆ มีการอ้างอิงอยู่

คุณสมบัติต่าง ๆ ของเนื้อความทั้งหมดนั้น อาจจะมีไม่ครบทุกคุณสมบัติก็ได้ แต่คุณสมบัติที่จำเป็นต้องมีคือ PROMPT1 SUBJECT TYPE และ TRANS

ตัวอย่างเนื้อความและคุณสมบัติต่างๆ ที่กำหนดลักษณะในฐานความรู้ knowledge base1 เช่น

#### PROBLEM

PROMPT1 : (Is there any problem occurring in farming ?)

PROMPT2 : (Is there any more problem in farming session ?)

TRANS : (Problem that occurred)

TYPE : PROBLEM-

MAINPROPS : (problem status stage problem farm name)

ASSOCWITH : ( )

SUBJECT : (problem\_rule)

แสดงตัวอย่างเนื้อความชนิด PROBLEM ซึ่งเมื่อมีการสร้างเนื้อความชนิด PROBLEM ขึ้นจะกำหนดเป็น PROBLEM-# (ตามคุณสมบัติ TYPE) โดย # เป็นหมายเลขต่อไปที่ยังไม่มีการกำหนดชื่อให้ เนื้อความชนิดนั้น ค่าของพารามิเตอร์ problem\_status stage\_problem และ farm\_name เป็นตัวที่กำหนดลักษณะเนื้อความชนิดนั้น ซึ่งพารามิเตอร์เหล่านี้จะถูกหาค่าโดยโปรแกรมย่อย FINDOUT (คุณสมบัติ MAINPROPS) จุดเนื้อความเกี่ยวกับปัญหาจะถูกสร้างขึ้นเป็นจุดแรกในต้นไม้เนื้อความ (คุณสมบัติ ASSOCWITH เป็นเซตว่าง) จุดเนื้อความปัญหา ก่อนจะมีการสร้างระบบจะถามผู้ใช้ระบบโดยใช้คำถามว่า "Were there any problem happened in farming sessions ?" (คุณสมบัติ PROMPT1) หรือตัวอย่างชนิดเนื้อความชื่อ FCAUSE จะมีคุณสมบัติดังนี้

#### FCAUSE

PROMPT1 : (Were the cause of problem concerning  
with gravid female problem?)

PROMPT2 : (Is there any more gravid female problem  
in this session ?)

TRANS : (Problem about gravid female that happened )

TYPE : FCAUSE-

MAINPROPS : (spawned\_fm)

ASSOCWITH : (PROBLEM)

SUBJECT : (FCAUSE-RULE)

แสดงตัวอย่างเนื้อความชนิด FCAUSE ซึ่งเมื่อมีการสร้างเนื้อความชนิด FCAUSE ขึ้นจะกำหนดชื่อเป็น FCAUSE-# (ตามคุณสมบัติ TYPE) โดย # เป็นหมายเลขต่อไปที่ยังไม่มีการกำหนดให้ เนื้อความชนิด FCAUSE นั้น ค่าของพารามิเตอร์ spawned\_fm จะใช้เป็นตัวกำหนดลักษณะเนื้อความชนิดนั้น ซึ่งจะหาค่าได้โดยโปรแกรมย่อย FINDOUT (คุณสมบัติ MAINPROPS) จุดเนื้อความเกี่ยวกับพ่อพันธุ์แม่พันธุ์จะถูกสร้างขึ้นเป็นจุดต่อ (child node) จากจุดเนื้อความชนิด PROBLEM ในต้นไม้เนื้อความ (คุณสมบัติ ASSOCWITH) ในการสร้างจุด

เนื้อความเกี่ยวกับพ่อพันธุ์แม่พันธุ์กึ่ง เป็นจุดแรก ระบบจะถามคำถามเพื่อให้ผู้ใช้ระบบเลือกว่าจะสร้างจุดนี้หรือไม่ โดยถามว่า "Were the cause of problem concerning with gravid female problem ?" (ตามคุณสมบัติ PROMPT1) ถ้าจำเป็นจะต้องมีการสร้างเนื้อความชนิดนี้ขึ้นอีกในต้นไม้เนื้อความ ระบบจะถามคำถามว่า "Is there any more gravid female problem in this session ?" (ตามคุณสมบัติ PROMPT2)

ส่วนต้นไม้เนื้อความ จะเป็นโครงสร้างข้อมูลแบบต้น ไม้แสดงความสัมพันธ์ระหว่างเนื้อความย่อยต่าง ๆ ที่เกี่ยวข้องกับปัญหาที่กำลังทำการหาสาเหตุในขณะนั้น ตัวอย่างต้นไม้เนื้อความสำหรับการทำงานดังรูปที่ 3.2

ชื่อที่กำหนดให้แต่ละจุดเนื้อความระหว่างการทำงานเป็นชื่อของจุดต่างๆ ที่อยู่บนต้นไม้เนื้อความ ชื่อในวงเล็บหมายถึงชนิดของเนื้อความสำหรับชื่อที่กำหนดให้แต่ละชื่อ จากตัวอย่างต้นไม้เนื้อความในรูปที่ 3.2 สามารถอธิบายได้ดังนี้ การเพาะเลี้ยงกึ่งมีปัญหาเกิดขึ้น (PROBLEM-1) สาเหตุของปัญหาที่เกิดขึ้นเกิดจาก 2 กรณี กรณีแรกเกิดจากตัวพ่อพันธุ์แม่พันธุ์กึ่ง (FCAUSE-1) ซึ่งมีสาเหตุจากการปรับสภาพที่ไม่เหมาะสมก่อนการมีบุตร (ACCLIMATIZATION-1) อีกกรณีหนึ่งเป็นปัญหาเกิดจากการที่ไซซึ่งนำจากแม่พันธุ์กึ่งมีจำนวนน้อยกว่าค่าเฉลี่ยที่ควรจะเป็น (ECAUSE-1) เป็นต้น

### 1.3 พารามิเตอร์ที่เกี่ยวข้องกับการเลี้ยงกึ่ง

พารามิเตอร์จะเป็นค่าตัวแปรที่กำหนดลักษณะของเนื้อความต่างๆ โดยมีค่าของพารามิเตอร์สำหรับแต่ละเนื้อความต่างๆ กันตามคุณสมบัติของพารามิเตอร์นั้นๆ ค่าของพารามิเตอร์จะหาได้โดยการทำงานของโปรแกรมย่อย MONITOR และโปรแกรมย่อย FINDOUT ซึ่งจะทำการถามค่านั้นจากผู้ที่ใช้ระบบหรือสรุปเองจากกฎความรู้อื่น โดยอาศัยค่าความจริงที่มีอยู่ในหน่วยความจำที่เก็บเหตุการณ์ คุณสมบัติที่กำหนดลักษณะและใช้ในการหาค่าพารามิเตอร์มีดังนี้คือ

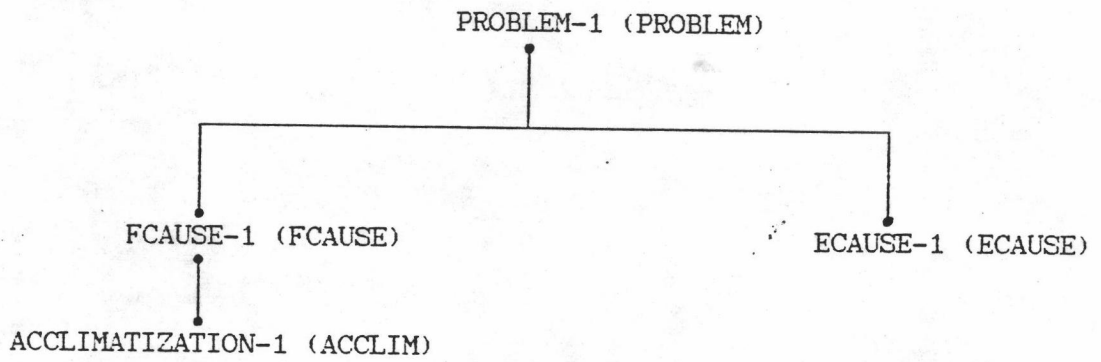
ก. EXPECTED เป็นคุณสมบัติที่กำหนดขอบเขตหรือกำหนดค่าคำตอบที่ควรจะเป็นสำหรับพารามิเตอร์นั้น ๆ

ถ้าเป็น (YN) แสดงว่าคำตอบสำหรับพารามิเตอร์นั้นจะมีค่าเป็น YES หรือ NO เท่านั้น

ถ้าเป็น (NUMB) แสดงว่าคำตอบสำหรับพารามิเตอร์นั้นจะมีค่าเป็นตัวเลข

ถ้าเป็น (CHAR) แสดงว่าคำตอบสำหรับพารามิเตอร์นั้นจะเป็นข้อความหรือค่าที่เป็นตัวอักษร

ข. PROMPT เป็นประโยคที่ระบบ SFES ใช้ถามผู้ใช้ระบบเมื่อต้องการทราบค่าของพารามิเตอร์นั้น ถ้าในวลีมีเครื่องหมายดอกจัน จะถูกแทนค่าโดยใช้ชื่อเนื้อความที่คำถามนั้นกำลังถามอยู่แทนค่าลงไป



รูปที่ 3.2 แสดงต้นไม้เนื้อความของการปรึกษาปัญหาเลี้ยงกุ้ง



ค. LABDATA เป็นคุณสมบัติที่มีค่า T หรือ NIL ถ้าเป็น T แสดงว่า พารามิเตอร์นั้นเป็นค่าพื้นฐานซึ่งผู้ใช้ระบบจะต้องรู้ค่าพารามิเตอร์ ระบบจะถามค่าจากผู้ใช้ระบบทันที ถ้าเป็น NIL ระบบจะพยายามทำการสรุปค่าจากกฎความรู้อื่นโดยอาศัยค่าความจริงที่อยู่ในหน่วยความจำที่เก็บเหตุการณ์

ง. LOOKAHEAD เป็นรายการของกฎความรู้ทั้งหมดในระบบที่มีการอ้างอิงถึงพารามิเตอร์นั้นในส่วนหลักฐาน

จ. UPDATED\_BY เป็นรายการของกฎความรู้ทั้งหมดในระบบที่ส่วนกระทำมีการสรุปค่าความจริงเข้าสู่หน่วยความจำที่เก็บเหตุการณ์โดยอาศัยค่าของพารามิเตอร์นั้นในส่วนหลักฐาน

ฉ. TRANS เป็นวิธีที่ใช้แปลความหมายของพารามิเตอร์จากชื่อตัวแปร เป็นความหมายในภาษาอังกฤษ

ช. DEFAULT เป็นหน่วยสำหรับพารามิเตอร์ที่มีค่าคำตอบเป็นตัวเลข (NUMB) เช่น tons, days เป็นต้น

ตัวอย่างพารามิเตอร์บางตัวที่เกี่ยวกับการเร่งไข่แม่กุ้ง เช่น

spawned\_fm : <spawned\_fm is an attribute of a problem  
context>

EXPECTED : (YN)

PROMPT : (Is \* less than 70 %)

UPDATED\_BY : (RULE101)

TRANS : (The number of spawned gravid female)

DEFAULT : (shrimps)

acc\_period : <acc\_period is an attribute of acclimatization>

EXPECTED : (NUMB)

PROMPT : (How many days is the acclimatization  
period ?)

LABDATA : T

LOOKAHEAD : (RULE201)

TRANS : (the acclimatization period of gravid female &  
male)

DEFAULT : (days)

#### 1.4 ฐานข้อมูลพลวัต (Dynamic database)

ฐานข้อมูลพลวัตเป็นส่วนที่เก็บความรู้ที่ได้ทั้งจากผู้ใช้ระบบและจากการสรุปเองจากกฎความรู้ที่ประกอบด้วย 2 ส่วนคือ ประโยคสมมติฐาน (hypothesis clause) และหน่วยความจำที่เก็บเหตุการณ์ (working memory) โดยแต่ละส่วนจะเก็บเป็นโครงสร้างแบบ OBJECT-ATTRIBUTE-VALUE TRIPLET รวมกันเป็น property list ในภาษา Lisp โดยมี OBJECT คือ ชื่อของเนื้อความในต้นไม้เนื้อความที่กำลังทำการถามหาค่าอยู่ และ ATTRIBUTE คือค่าพารามิเตอร์ที่เป็นตัวกำหนดลักษณะของเนื้อความนั้น ส่วนค่าของพารามิเตอร์คือ ค่าที่ผู้ใช้ระบบตอบเข้ามาเขียนเป็นรูปประโยคได้ดังตัวอย่างเช่น ถ้าระบบ SFES ถามผู้ใช้ระบบถึงข้อมูลเกี่ยวกับความจุของบ่อปรับสภาพพ่อแม่พันธุ์แม่ม้าพันธุ์กึ่งที่ว่า

12. How many cubic metre are the volume of acclimatization pond ? (NUMB-cubic metre)

\*\* 3

ซึ่งหมายถึงว่าผู้ใช้ระบบให้ข้อมูลขนาดปริมาตรของบ่อปรับสภาพพ่อแม่พันธุ์กึ่งเท่ากับ 3 ลูกบาศก์เมตร

ระบบจะเก็บคำตอบของผู้ใช้ระบบไว้เป็นส่วนของประโยคสมมติฐานในรูปของ

```
Val [ACCLIMATIZATION-1,acc_p_size] = 3
```

สำหรับค่าความจริงที่ระบบสรุปได้จากส่วนกระทำของกฎ เช่น กฎความรู้ที่สรุปเกี่ยวกับการปรับสภาพพ่อแม่พันธุ์แม่ม้าพันธุ์กึ่งที่ว่า

IF : 1) The volume of acclimatization pond is less than 3 cubic metre ,and

2) The density of gravid in acclimatization pond is more than 80

THEN : I can conclude that the cause of problem is acclimatization feature that unsuitable which the

volume of acclimatization pond and density of gravid in that pond.

เมื่อโปรแกรมย่อย MONITOR และ โปรแกรมย่อย FINDOUT ทำการตรวจสอบค่าพารามิเตอร์เกี่ยวกับความจุและความหนาแน่นของการปรับสภาพพ่อพันธุ์แม่พันธุ์กึ่งว่าเป็นจริงแล้ว จะทำงานในส่วนกระทำของกฎซึ่งจะสรุปค่าความจริงว่า

Val [ACCLIMATIZATION-1,acc\_p\_feature] = (unsuitable)

ค่าความจริงที่ได้จากผู้ใช้ระบบตอบคำถามเข้ามานั้น จะเก็บไว้ในประโยคสมมติฐานซึ่งจะถูกฟังกัซันที่กำหนดในประโยคเงื่อนไขแต่ละประโยคทำการประเมินผลว่าประโยคสมมติฐานนั้นๆ เป็นจริงหรือไม่ ฟังกัซันที่ใช้ในการประเมินผลที่ใช้ได้แก่ LESS MORE และ SAME

LESS คือ ฟังกัซันที่ใช้ในการตรวจสอบว่าค่าพารามิเตอร์ของเนื้อความที่กำหนดในประโยคสมมติฐานมีค่าน้อยกว่าค่าของพารามิเตอร์นั้นในประโยคเงื่อนไขของกฎความรู้

MORE คือ ฟังกัซันที่ใช้ในการตรวจสอบว่าค่าพารามิเตอร์ของเนื้อความที่กำหนดในประโยคสมมติฐานมีค่ามากกว่าค่าของพารามิเตอร์นั้นในประโยคเงื่อนไขของกฎความรู้

SAME คือ ฟังกัซันที่ใช้ในการตรวจสอบว่าค่าพารามิเตอร์ของเนื้อความที่กำหนดในประโยคสมมติฐานมีค่าเท่ากับค่าของพารามิเตอร์นั้นในประโยคเงื่อนไขของกฎความรู้

ฟังกัซันเหล่านี้จะให้ค่า T หรือ F ซึ่งแสดงว่าประโยคเงื่อนไขนั้นเป็นจริงหรือเป็นเท็จ คำตอบที่ได้จากผู้ใช้ระบบ SFES จะเป็นคำตอบที่มีความแน่นอนซึ่งในปัญหาของการเลี้ยงกึ่งพารามิเตอร์ต่างๆ จะเป็นลักษณะที่มีความแน่นอน เช่น ความจุของบ่อ ช่วงเวลาที่ทำการปรับสภาพกึ่ง หรือความหนาแน่นของพ่อพันธุ์แม่พันธุ์กึ่ง จึงไม่มีการทำงานในส่วนของการหาเหตุผลจากความไม่แน่นอนของเหตุการณ์ซึ่งต่างจากลักษณะปัญหาทางการแพทย์ที่ใช้งานกับระบบเชี่ยวชาญ MYCIN ซึ่งแพทย์ผู้ใช้ระบบจะตอบคำถามด้วยความไม่แน่นอนได้

ฟังกัซันในส่วนกระทำที่ใช้ในการทำงานเกี่ยวกับหน่วยความจำที่เก็บเหตุการณ์และทำงานให้คำแนะนำแก่ผู้ใช้ระบบคือ CONCLUDE RECOMMEND และ TREATMENT แต่ละฟังกัซันจะมีหน้าที่ดังนี้

CONCLUDE เป็นฟังกัซันในส่วนกระทำที่ใช้ทำการเพิ่มค่าความจริงสำหรับเนื้อความย่อยนั้นๆ เก็บไว้ในหน่วยความจำที่เก็บเหตุการณ์

TREATMENT เป็นฟังกัซันที่ใช้กำหนดข้อแก้ไขสำหรับปัญหาที่เกิดขึ้น โดย

เก็บรวมไว้ในรายการวิธีการแก้ไข (treatment list) สำหรับเนื้อความปัญหานั้น

RECOMMEND เป็นฟังก์ชันที่ใช้ที่เป็นการแสดงข้อความวิเคราะห์ปัญหาที่เกิดขึ้น โดยการแสดงข้อความในรูปแบบของรายการสรุปผลการวิเคราะห์ (conclude list) และรายการวิธีแก้ไขปัญหานั้น

## 2. โครงสร้างควบคุมการทำงานของระบบ SFES

การทำงานของ SFES จะเป็นการให้คำปรึกษาโดยเมื่อผู้ดำเนินการเลี้ยงกุ้งเกิดปัญหาขึ้น เช่นแม่กุ้งที่นำมาบิบาดตายหรือหลังจากบิบาดแล้วมีแม่กุ้งที่เร่งให้ไข่สุกน้อยกว่าร้อยละ 70 หรือไข่ที่ได้จากการบิบาดแม่กุ้งมีจำนวนน้อย ระบบ SFES จะช่วยให้ผู้ใช้ระบบหาสาเหตุของปัญหา โดยถามสภาพความจริงต่างๆ ที่เกี่ยวข้องกับปัญหานั้น ส่วนการหาเหตุผลจะใช้วิธีหาเหตุผลแบบย้อนหลัง โดยเริ่มจากการกำหนดกฎเป้าหมายซึ่งเป็นกฎแรกที่ระบบจะเริ่มทำการประเมินผลหลังจากที่สร้างจุดเนื้อความเกี่ยวกับปัญหาชื่อ PROBLEM-1 ขึ้นในต้นไม้เนื้อความ กฎเป้าหมายคือกฎหมายเลข 000 ซึ่งมีข้อความดังนี้

RULE000

IF : 1) There is problem happened in Shrimp farming

THEN : 1) Identify cause of Shrimp farming problem, and

2) Recommend solving method to user.

OTHERWISE: Indicate that no problem need to be solved.

ขั้นตอนในการให้คำปรึกษาจะมี 2 ขั้นตอน คือ

1. สร้างจุดเนื้อความเกี่ยวกับปัญหาที่เกิดขึ้นเป็นจุดแรกในต้นไม้เนื้อความ
2. พยายามใช้งานกฎเป้าหมายกับจุดเนื้อความปัญหาที่สร้างขึ้นใหม่นั้น

จากขั้นตอนที่ 2 จะมีการเรียกใช้โปรแกรมย่อย MONITOR และโปรแกรมย่อย

FINDOUT เพื่อทำการหาเหตุผลอย่างต่อเนื่องตามกฎความรู้ต่างๆ โดยสร้างเป็นข่ายการหาเหตุผลตามกฎต่างๆ

### 2.1 หน้าทีของโปรแกรมย่อย MONITOR

หลังจากที่เริ่มทำการประเมินผลกฎเป้าหมายจากส่วนกระทำของกฎ โปรแกรมย่อย MONITOR จะถูกเรียกใช้เพื่อทำการสรุปหาสาเหตุของปัญหานั้น โดยจะทำการเลือกกฎที่เกี่ยวข้องกับเนื้อความ PROBLEM-1 โดยจะเลือกกฎในกลุ่มของ PROBLEM-RULE จากนั้นจะทำการ



พิจารณาประโยคเงื่อนไขแต่ละประโยคในส่วนหลักฐานของกฎ

โปรแกรมย่อย MONITOR มีการทำงานที่สำคัญคือ

ก. เรียกใช้โปรแกรมย่อย MONITOR-PREMISE เพื่อทำการตรวจสอบ ส่วนหลักฐานของกฎโดยละเอียด เริ่มตรวจสอบตั้งแต่ประโยคเงื่อนไขประโยคแรก

ข. ตรวจสอบว่า ชนิดของเนื้อความในประโยคเงื่อนไขนั้นมีการสร้าง อยู่ในต้นไม่เนื้อความแล้วหรือไม่ โดยการเรียกใช้โปรแกรมย่อย FINDOUT เพื่อสร้างเนื้อความ ชนิดนั้นลงบนต้นไม่เนื้อความ

ค. ทำการตรวจสอบค่าพารามิเตอร์ของเนื้อความนั้นในประโยคเงื่อนไข จากรายการประโยคสมมติฐานที่มีอยู่ว่าระบบ SFES รู้ค่าแล้วหรือไม่ ถ้าทราบค่าพารามิเตอร์แล้ว ก็จะทำการประเมินผลโดยเรียกโปรแกรมย่อย EVALUATE เพื่อทำการประเมินผลประโยคสมมติฐาน ที่ทราบค่ากับค่าของประโยคเงื่อนไขในกฎความรู้โดยใช้ฟังก์ชันที่กำหนดในประโยคเงื่อนไขของกฎ

ง. ถ้ายังไม่ทราบค่าของพารามิเตอร์ โปรแกรมย่อย MONITOR จะเรียก โปรแกรมย่อย FINDOUT เพื่อทำการหาค่าของพารามิเตอร์นั้น

จ. ทำการตรวจสอบผลการประเมินผลประโยคเงื่อนไขที่ทำในข้อ ค. ถ้า ประโยคเงื่อนไขเป็นจริงเมื่อเทียบกับประโยคสมมติฐาน โปรแกรมย่อย MONITOR จะทำการเรียก ใช้ตัวเอง (recursive call) เพื่อตรวจสอบประโยคเงื่อนไขต่อไปในส่วนหลักฐาน ถ้าไม่มี ประโยคเงื่อนไขแล้วแสดงว่า ส่วนหลักฐานของกฎนั้นเป็นจริง แต่ถ้าพบว่าประโยคเงื่อนไขเป็น เท็จ โปรแกรมย่อย MONITOR จะทำการเลือกกฎความรู้อื่นที่เกี่ยวข้องกับเนื้อความชนิดเดียวกันนั้น และทำการเรียกใช้โปรแกรมย่อย MONITOR เองเพื่อตรวจสอบส่วนหลักฐานของกฎที่เรียกใช้ใหม่ นั้นอีกครั้ง

ฉ. ในกรณีที่พบว่าส่วนหลักฐานของกฎเป็นจริง โปรแกรมย่อย MONITOR จะทำการกำหนดหมายเลขกฎนั้นเป็นกฎหนึ่งในเส้นทางการวินิจฉัย และทำการเรียกใช้โปรแกรมย่อย MONITOR-ACTION เพื่อทำงานในส่วนกระทำของกฎความรู้นั้น โดยจะเลือกทำงานประโยคกระทำ ที่ละประโยคตามฟังก์ชันที่กำหนด

ถ้าเป็นฟังก์ชัน CONCLUDE ก็จะใช้โปรแกรมย่อย CONCLUDE-WORK เพื่อทำการเก็บค่าความจริงที่กฎสรุปได้เข้าสู่หน่วยความจำที่เก็บเหตุการณ์

ถ้าเป็นฟังก์ชัน TREATMENT ก็จะใช้โปรแกรมย่อย TREATMENT-WORK เพื่อทำการเก็บข้อมูลที่จะใช้ในการแก้ปัญหาไว้ในรายการวิธีแก้ไข

ถ้าเป็นฟังก์ชัน RECOMMEND ก็จะใช้โปรแกรมย่อย RECOMMEND-WORK ทำการแสดงรายการวินิจฉัย และรายการวิธีการแก้ไขปัญหาที่เก็บไว้เพื่อแสดงผลการทำงานของ



ระบบแก่ผู้ใช้ มีการเรียกใช้โปรแกรมย่อย PRT-DIAG เพื่อทำการพิมพ์แสดงผลแก่ผู้ใช้ระบบ แบ่งออกเป็น ส่วนผลการวินิจฉัย (diagnosis part) และส่วนวิธีการแก้ไขปัญหา (treatment part)

## 2.2 หน้าทีของโปรแกรมย่อย FINDOUT

จะรับชื่อเนื้อความเข้าเป็นข้อมูล และทำการตรวจสอบสภาพขณะนั้น ว่ากำลังหาค่าของพารามิเตอร์ หรือเนื้อความ

ถ้าสภาพเป็นเนื้อความ โปรแกรมย่อย FINDOUT จะตรวจสอบในต้นไม้เนื้อความว่า เนื้อความชนิดนี้มีการสร้างแล้วหรือไม่ถ้ามีการสร้างแล้วจะทำการดึงคุณสมบัติ PROMPT2 ของเนื้อความชนิดนั้นมาทำการถามผู้ใช้ว่าจะต้องสร้างเนื้อความชนิดนี้อีกหรือไม่ ถ้าต้องสร้างอีกจะทำการเรียกโปรแกรม MAKE-CNTXT-NODE เพื่อกำหนดเนื้อความชนิดนี้ลงในต้นไม้อีก เนื้อความ ถ้าไม่ต้องการสร้างก็จะส่งการทำงานกลับไปยังโปรแกรมย่อย MONITOR ในกรณีที่เนื้อความชนิดนั้นยังไม่มีการสร้างในต้นไม้อีกเนื้อความก็จะทำการดึงคุณสมบัติ PROMPT1 ของเนื้อความชนิดนั้นมาถามผู้ใช้ระบบ และทำการตรวจสอบว่าจะสร้างจุดเนื้อความชนิดนี้ในต้นไม้อีกหรือไม่

ถ้าสภาพเป็นพารามิเตอร์ ก่อนอื่นโปรแกรมย่อย FINDOUT จะทำการตรวจสอบว่ารู้ค่าพารามิเตอร์สำหรับเนื้อความชนิดนั้นหรือไม่ ถ้ารู้แล้วก็จะส่งการทำงานกลับไปยังโปรแกรมย่อย MONITOR ถ้ายังไม่ทราบค่าของพารามิเตอร์โปรแกรมย่อย FINDOUT จะตรวจสอบคุณสมบัติ LABDATA ของพารามิเตอร์เป็นค่า T หรือ NIL

ถ้าเป็น T จะทำการดึงคุณสมบัติ PROMPT ของพารามิเตอร์นั้นมาถามหาค่าของพารามิเตอร์กับผู้ใช้ระบบ โดยเรียกโปรแกรมย่อย ASK ซึ่งจะรับค่าคำตอบของผู้ใช้ระบบ พร้อมกับทำการตรวจสอบความถูกต้องของคำตอบ จากนั้นโปรแกรม FINDOUT เก็บค่าที่ได้ไว้ในรายการประวัติสมมติฐานที่ผู้ใช้ตอบกับระบบ พร้อมทั้งส่งค่ากลับไปยังโปรแกรมย่อย MONITOR เพื่อทำการประเมินผลโดยฟังก์ชันที่กำหนดในประวัติเงื่อนไขต่อไป

ถ้าเป็น NIL โปรแกรมย่อย FINDOUT จะดึงคุณสมบัติ UPDATED-BY ของพารามิเตอร์มาสร้างเป็นรายการของกฎที่จะใช้หาค่าพารามิเตอร์นั้น แล้วทำการประเมินผลรายการกฎเหล่านั้น โดยเรียกโปรแกรมย่อย MONITOR ในลักษณะ recursive

## 2.3 ระบบช่วยอธิบายการทำงานของระบบ SFES

เป็นระบบย่อยที่ทำหน้าที่ช่วยอธิบายเหตุผลแก่ผู้ใช้ระบบใน 2 ลักษณะคือ

ก. เหตุใดระบบ SFES จึงต้องถามค่าของพารามิเตอร์นั้นกับผู้ใช้ระบบ โดยจะแสดงว่า กฎความรู้ที่ระบบใช้งานอยู่ในขณะนั้นมีการอ้างอิงถึงพารามิเตอร์ของเนื้อความนั้นในประวัติเงื่อนไขอย่างไร ในกรณีที่ผู้ใช้ต้องการถามย้อนกลับไปอีกชั้นหนึ่ง ระบบจะสามารถย้อนเส้นทางการวินิจฉัยเพื่อแสดงถึงกฎที่เกี่ยวข้องกับพารามิเตอร์นั้นจนถึงกฎเป้าหมายได้

ข. เหตุใดระบบ SFES จึงสรุปผลการวินิจฉัยปัญหา และวิธีการแก้ไข เช่นนั้น โปรแกรมย่อย EXPLAIN จะแสดงค่าความจริงที่อยู่ในหน่วยความจำที่เก็บเหตุการณ์ และกฎความรู้ทั้งหมดที่สรุปค่าความจริงเหล่านั้น โดยบอกว่าค่าความจริงใดที่ได้จากผู้ใช้ระบบ ตอบให้แก่ระบบ และค่าความจริงใดที่ระบบทำการสรุปเองจากกฎความรู้อื่น

ส่วนประกอบย่อยของโปรแกรมย่อย EXPLAIN มีดังต่อไปนี้

โปรแกรมย่อย EXPLANATION-NEED ทำการตรวจสอบความต้องการของผู้ใช้ระบบว่า ต้องการคำอธิบายในขณะนั้นหรือไม่ ถ้าต้องการผู้ใช้ระบบจะตอบคำถามด้วยคำว่า "WHY"

โปรแกรมย่อย EXPN-DISP ทำการสอบถามผู้ใช้ระบบว่าต้องการให้แสดงเส้นทางการวินิจฉัยย้อนหลังไปอีกหรือไม่

โปรแกรมย่อย PRT-EXPN-RULE จะพิมพ์กฎความรู้ที่อ้างถึงพารามิเตอร์ หรือเนื้อความในขณะนั้นออกมา โดยแยกออกเป็นกฎเป้าหมาย และกฎความรู้ทั่วไป พร้อมทั้งมีการระบุส่วนหลักฐานและส่วนกระทำของกฎ และแสดงชื่อเนื้อความ พารามิเตอร์ หมายเลขกฎอย่างละเอียด

โปรแกรมย่อย PRT-EXPN-CLAUSE ใช้พิมพ์ประโยคที่อยู่ในกฎความรู้ ว่าจะเป็นประโยคเงื่อนไขหรือประโยคกระทำ โดยแสดงอยู่ในรูปภาษาเขียนตามธรรมชาติจากคุณสมบัติ TRANS ของพารามิเตอร์และเนื้อความ

### สรุป

จากการออกแบบส่วนฐานความรู้ และส่วนควบคุมการทำงานของระบบ SFES ซึ่งมีหน้าที่ และการทำงานของโปรแกรมต่างๆ ที่อยู่ภายในระบบ ในบทความนี้จะกล่าวถึงการพัฒนาโปรแกรมต่างๆ ที่กล่าวมาแล้วในลักษณะผังการทำงานรวมของระบบ SFES ผังงานย่อยของแต่ละโปรแกรมย่อย ส่วนโปรแกรมจริงได้ทำการพัฒนาขึ้นด้วยภาษา Franz Lisp และรายละเอียดต่างๆ เกี่ยวกับชื่อตัวแปรและฟังก์ชันที่เกี่ยวข้อง ได้รวบรวมตัวโปรแกรมไว้ในภาคผนวก ข.