ภาคผนวก

Java Agent DEvelopment Framework (JADE)

JADE เป็นโครงสร้างการทำงานสำหรับการพัฒนาระบบมัลติเอเจนท์และงานประยุกต์ตามมาตรฐาน FIPA ซึ่งพัฒนาด้วยภาษาจาวา JADE ประกอบด้วย 2 ส่วนหลักคือ JADE Platform เป็นระบบเอเจนท์ตามมาตรฐาน FIPA (FIPA-compliant agent platform) และ JADE Agent เป็นชุดคำสั่งซึ่งเป็นคลาสสำหรับพัฒนาเอเจนท์

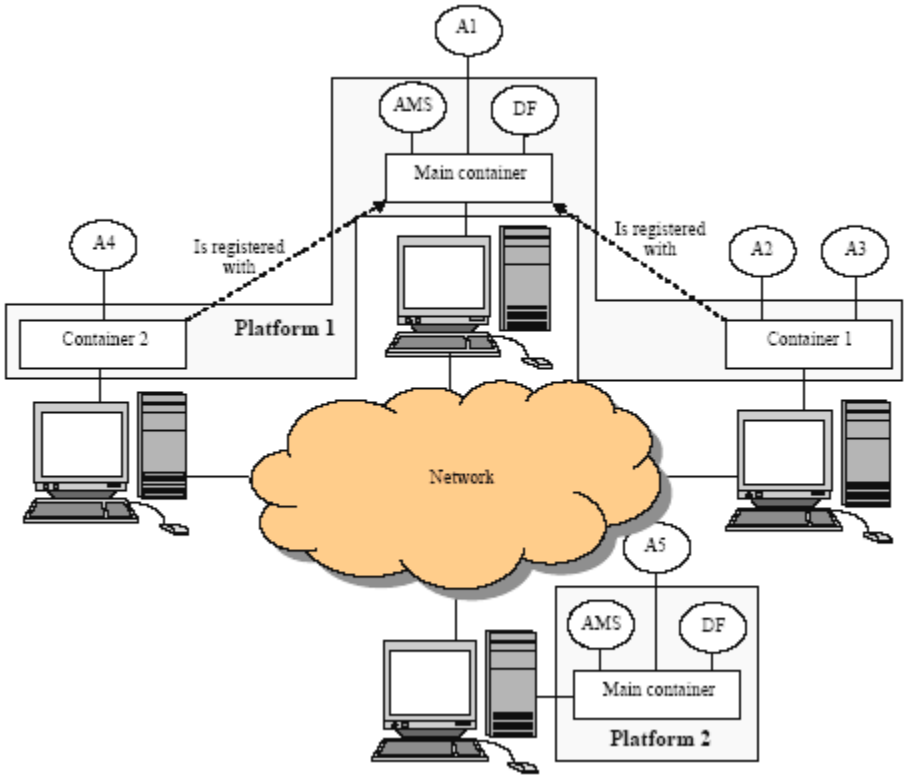
1. JADE Platform

JADE Platform สามารถทำงานได้บนเครื่องคอมพิวเตอร์ที่มีการติดตั้ง Java Virtual Machine (JVM) โดยเมื่อมีการเรียกใช้ JADE Platform ขึ้นมาก็จะมีการสร้างคอนเทนเนอร์ซึ่งจะใช้เป็นพื้นที่หลักเพื่อให้เอเจนท์ทำงานเรียกว่า **main-container** รวมทั้งมีการสร้างเอเจนท์หลักเพื่อควบคุมการทำงานของแพลตฟอร์มและสร้างคอมโพเน็นท์ในการควบคุมการสื่อสารของเอเจนท์ ได้แก่

- **AMS (Agent Management System)** เป็นเอเจนท์ที่ทำหน้าที่ควบคุมการเข้าถึงและการใช้งานแพลตฟอร์มรวมถึงการจัดการเอเจนท์ที่ถูกสร้างขึ้นมาใหม่ แต่ละแพลตฟอร์มจะมีเพียง 1 AMS เพื่อจัดการรายการเอเจนท์ในแพลตฟอร์ม ดูแลวงจรชีวิตของเอเจนท์และสถานะของเอเจนท์ โดยทุกเอเจนท์จะต้องลงทะเบียนกับ AMS เพื่อรับ AID
- **DF (Directory Facilitator)** เป็นเอเจนท์ที่ทำหน้าที่บริการรายชื่อบริการต่าง ๆ ของเอเจนท์ที่อยู่ในแพลตฟอร์ม
- **Message Transport System หรือ Agent Communication Channel (ACC)** เป็นส่วนการทำงานที่ทำหน้าที่ควบคุมการแลกเปลี่ยนข้อความระหว่างเอเจนท์ทั้งภายในแพลตฟอร์มและระหว่างแพลตฟอร์ม

เอเจนท์ที่ถูกสร้างขึ้นมาใช้งานสามารถทำงานอยู่ใน main-container หรือคอนเทนเนอร์อื่นในแพลตฟอร์มก็ได้ และแพลตฟอร์มของ JADE สามารถกระจายอยู่บนคอมพิวเตอร์หลาย

เครื่องได้โดยแต่ละเครื่องต่างก็มี JVM ของตนเอง คอนเทนเนอร์ที่สร้างใหม่จะถูกควบคุมโดย AMS และ DF จาก main-container แสดงดังภาพประกอบ 1



ภาพประกอบ 1 แสดงแพลตฟอร์มของ JADE แบบกระจาย

2. JADE Agent

JADE มีคลาส Agent ซึ่งเป็นคลาสพื้นฐานเพื่อใช้ในการสร้างเอเจนท์ที่สามารถทำงานได้ใน JADE Platform ประกอบด้วยเมธอดต่าง ๆ ที่จำเป็นต่อการทำงานของเอเจนท์ เช่น การการรับส่งข้อความ และการเปลี่ยนสถานะของเอเจนท์ เป็นต้น โดยเมธอดที่สำคัญของ JADE Agent แสดงดังตาราง ก.1

ตาราง 1 เมธอดที่สำคัญของ JADE Agent

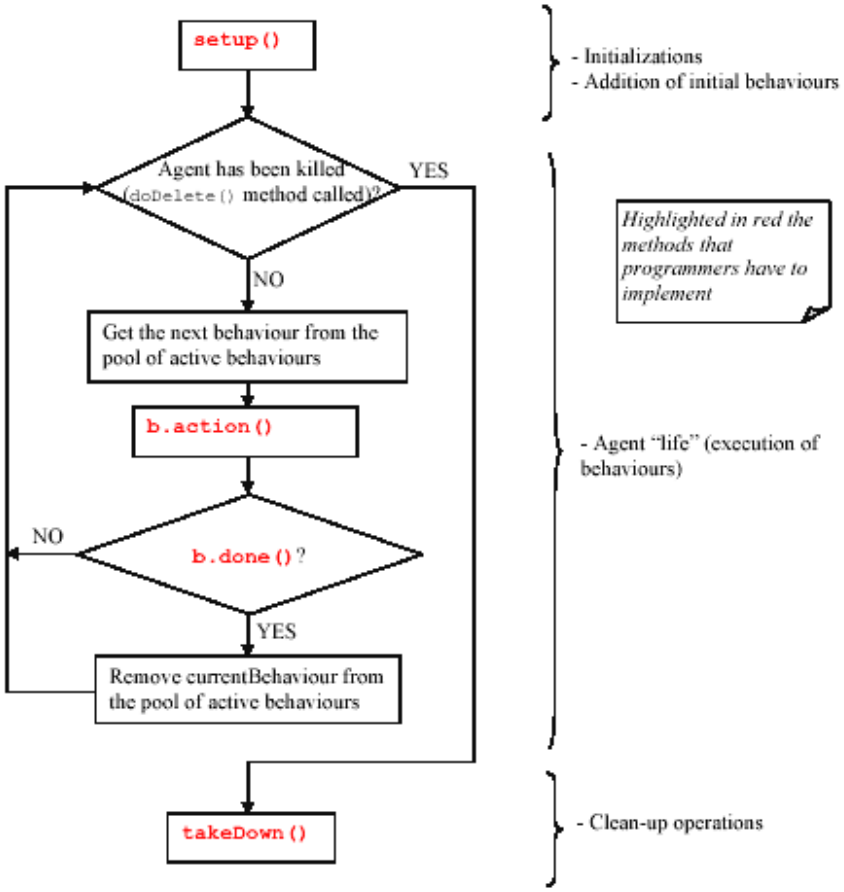
เมธอด	คำอธิบาย
setup()	เป็นเมธอดที่ใช้ในการเริ่มต้นการทำงานของเอเจนท์
takedown()	เป็นเมธอดที่ใช้ในการกำหนดการทำงานก่อนที่เอเจนท์จะถูกทำลาย

ตาราง 1 (ต่อ)

เมธอด	คำอธิบาย
receive()	รับ ACL Message จากคิวของข้อความ (message queue) ของเอเจนท์
send(ACLmessage msg)	ส่ง ACL message ไปยังเอเจนท์อื่น
addBehaviour()	เป็นเมธอดในการเรียกใช้เธรดเพื่อทำงาน
doClone(Location destination, java.lang.String newName)	ให้เอเจนท์นี้สำเนาตัวเองไปยังที่อยู่อื่นในแพลตฟอร์ม
doMove(Location destination)	ให้เอเจนท์นี้เดินทางไปยัง location อื่น
doDelete()	เป็นเมธอดในการทำลายเอเจนท์

3. งานของเอเจนท์

งานที่เอเจนท์ต้องทำจะถูกกำหนดไว้ใน behaviour ซึ่งแต่ละ behaviour จะใช้เธรดในการประมวลผลแยกจากกัน เอเจนท์สามารถจัดการและใช้งาน behaviour ได้โดยการขยายคลาส jade.core.behaviours.Behaviour และกำหนดงานที่ให้แก่ behaviour ภายในเมธอด action() เอเจนท์สามารถเรียกอ็อบเจกของคลาส Behaviour ได้โดยการเรียกเมธอด addBehaviour() ซึ่งอยู่ในคลาส Agents หากและต้องการให้ behaviour สิ้นสุดการทำงานก็เรียกเมธอด removeBehaviour() นอกจากนี้เอเจนท์สามารถประมวลผลได้หลาย behaviours พร้อมกันในลักษณะที่เรียกว่า cooperative ซึ่งหมายความว่าเมื่อ behaviour ถูกจัดอยู่ในลำดับการประมวลผลแล้วก็จะทำงานในเมธอด action() จนกว่าเสร็จ ดังนั้นโปรแกรมเมอร์ต้องเป็นคนกำหนดให้เอเจนท์สลับการทำงานของ behaviour เอง ลำดับการทำงานของเธรดของเอเจนท์แสดงดังภาพประกอบ 2



ภาพประกอบ 2 ลำดับการทำงานของเธรดของเอเจนท์แสดงดัง

3.1 ชนิดของ Behaviour

JADE มี behaviour ให้ใช้งาน 3 ชนิด ได้แก่

1. **One-shot behaviour** เป็น behaviour ที่ประมวลผลทันทีและเมธอด action() จะถูกประมวลผลเพียงครั้งเดียว การเรียกOne-shot behaviour ทำได้โดยการขยายคลาส jade.core.behavious.OneShotBehaviour ตัวอย่างเช่น

```
public class MyOneShotBehaviour extends OneShotBehaviour {
    public void action() {
        // perform operation X
    }
}
```

โดย X จะทำงานเพียงครั้งเดียว

2. **Cyclic behaviour** เป็น behaviour ที่จะทำงานภายในเมธอด action() วนซ้ำไม่สิ้นสุด เรียกใช้ behaviour ได้โดยการขยายคลาส jade.core.behaviours.CyclicBehaviour ตัวอย่างเช่น

```
public class MyCyclicBehaviour extends CyclicBehaviour {
    public void action() {
        // perform operation Y
    }
}
```

Y จะทำงานซ้ำ ๆตลอดไปจนกว่าเอเจนท์ที่เรียก behaviour นี้จะถูกทำลาย

3. **Generic behaviour** เป็น behaviour ที่มีการกำหนดสถานะไว้ และจะทำงานตามสถานะที่กำหนดจนเสร็จตามเงื่อนไข เรียก behviour นี้ได้โดยขยายคลาส jade.core.behaviours.Behaviour ตัวอย่างเช่น

```
public class MyThreeStepBehaviour extends Behaviour {
    private int step = 0;
    public void action() {
        switch (step) {
            case 0:
                // perform operation X
                step++;
                break;
            case 1:
                // perform operation Y
                step++;
                break;
            case 2:
                // perform operation Z
                step++;
```

```
                    break;
                }
            }
            public boolean done() {
                return step == 3;
            }
        }
```

X, Y, Z จะถูกทำงานอย่างละครั้งตามลำดับ แล้ว behaviour จึงทำงานเสร็จ

3.2 การกำหนดการดำเนินการตามเวลาที่กำหนด

JADE มี behaviour สำหรับใช้กำหนดให้มีการดำเนินการได้ตามเวลาที่กำหนด ซึ่งทำได้โดยการเรียกใช้คลาส jade.core.Behaviours. WakerBehaviour และ jade.core.Behaviours. TickerBehaviour package

1. **WakerBehaviour** สำหรับการระบุเวลาที่จะให้ดำเนินการเมื่อหมดเวลาที่ระบุ ตัวอย่างเช่น

```
public class MyAgent extends Agent {
    protected void setup() {
        System.out.println("Adding waker behaviour");
        addBehaviour(new WakerBehaviour(this, 10000) {
            protected void handleElapsedTimeout() {
                // perform operation X
            }
        } );
    }
}
```

X จะทำงานหลังพิมพ์ข้อความแล้ว 10 วินาที

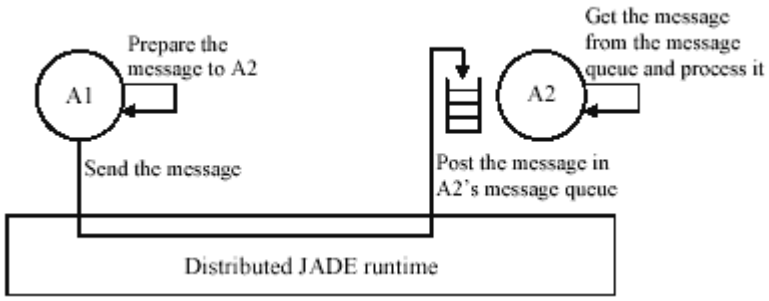
2. **TickerBehaviour** สำหรับการกำหนดให้เกิดการดำเนินการซ้ำ ๆ โดยมีการรอในช่วงเวลาที่ระบุ ตัวอย่างเช่น

```
public class MyAgent extends Agent {
    protected void setup() {
        addBehaviour(new TickerBehaviour(this, 10000) {
            protected void onTick() {
                // perform operation Y
            }
        } );
    }
}
```

Y จะถูกเรียกทำงานทุก 10 วินาที

4. การติดต่อสื่อสารระหว่างเอเจนท์

JADE Agent ติดต่อสื่อสารกันโดยใช้การส่งข้อความ โดยแต่ละเอเจนท์มีคิวสำหรับเก็บข้อความ (message queue) เมื่อมีเอเจนท์อื่นส่งข้อความมาให้ ข้อความจะถูกเก็บอยู่คิวของเอเจนท์ผู้รับก่อนกว่าผู้รับจะดึงข้อความออกจากคิว ภาพประกอบ 3 แสดงการส่งข้อความระหว่างเอเจนท์



ภาพประกอบ 3 การส่งข้อความระหว่างเอเจนท์

4.1 การส่งข้อความ

JADE มีคลาสในการส่งข้อความคือ jade.lang.acl.ACLMessage การส่งข้อความทำได้โดยการระบุฟิลด์ของอ็อบเจก ACLMessage แล้วเรียกเมธอด send() ของคลาส Agent ตัวอย่างรหัสคำสั่งในการส่งข้อความแจ้งผู้รับให้แก่เอเจนท์ที่ชื่อ Peter มีข้อความว่า today it's raining

```
ACLMessage msg = new ACLMessage(ACLMessage.INFORM);
msg.addReceiver(new AID("Peter", AID.ISLOCALNAME));
msg.setLanguage("English");
msg.setOntology("Weather-forecast-ontology");
msg.setContent("Today it's raining");
send(msg);
```

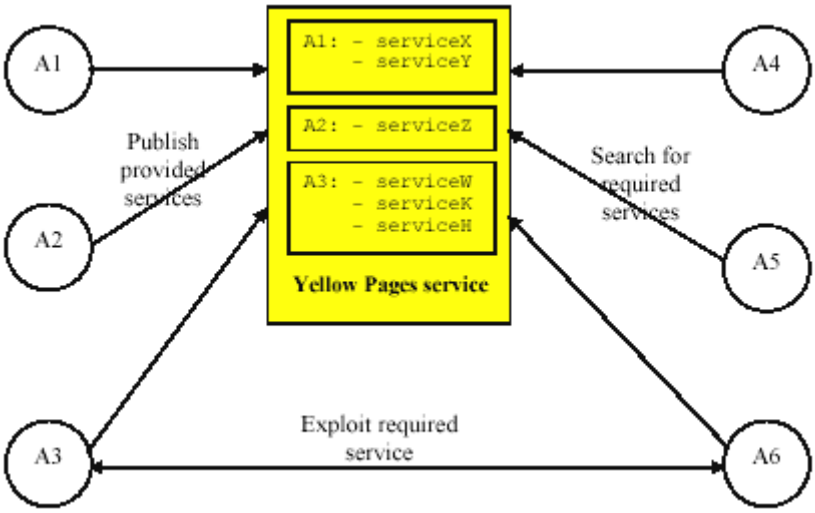
4.2 การรับ message

ข้อความจะถูกส่งไปยังคิวของข้อความของผู้รับอัตโนมัติโดย JADE Platform เมื่อมีข้อความอยู่ในคิวแล้วเอเจนท์สามารถดึงข้อความออกจากคิวโดยใช้เมธอด receive() ซึ่งจะส่งข้อความแรกในคิวออกมา และลบข้อความออกจากคิว ตัวอย่างระหัสคำสั่งในการรับข้อความ

```
ACMessage msg. receive();
if (msg != null) {
// process the message
}
```

5. การลงทะเบียนบริการ

JADE มีเอเจนท์ Directory Facilitator (DF) ทำหน้าที่รับลงทะเบียนบริการและจัดการบริการสืบค้นผู้ให้บริการ เรียกบริการนี้ว่า Yellow Pages Service แสดงดังภาพประกอบ 4



ภาพประกอบ 4 Yellow Pages Service

เอเจนท์ที่ต้องการลงทะเบียนบริการทำได้โดยการเรียกใช้คลาส jade.domain.DFService ตัวอย่างรหัสคำสั่งในการลงทะเบียนบริการ

```
protected void setup() {
    ...// Register the book-selling service in the yellow pages
    DFAgentDescription dfd = new DFAgentDescription();
    dfd.setName(getAID());
    ServiceDescription sd = new ServiceDescription();
    sd.setType("book-selling");
    sd.setName("JADE-book-trading");
    dfd.addServices(sd);
    try {
       DFService.register(this, dfd);
    }
    catch (FIPAException fe) {
       fe.printStackTrace();
    }  ...
}
```

ผู้ที่สนใจสามารถอ่านรายละเอียดการใช้งาน JADE เพิ่มเติมได้ที่เว็บไซต์ http://jade.tilab.com

การประยุกต์เอเจนท์แบบเคลื่อนที่สำหรับการประมูลอิเล็กทรอนิกส์: กรณีศึกษา การประมูลยาง ณ ตลาดกลางยางพาราแห่งประเทศไทย

The Adoption of Mobile Agents for Electronic Auction: Case Study of Electronic Auction at Rubber Brokering Service of Thailand

จามิกร หิรัญรัตน์ และ อำนาจ เปาะทอง
ภาควิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์ มหาวิทยาลัยสงขลานครินทร์
jamikorn.h@psu.ac.th and amnart.p@psu.ac.th

บทคัดย่อ

การประมูลอิเล็กทรอนิกส์เป็นรูปแบบหนึ่งของตลาดอิเล็กทรอนิกส์ที่ได้รับความนิยมในปัจจุบัน การนำเทคโนโลยีเอเจนท์แบบเคลื่อนที่มาใช้ในการประมูลอิเล็กทรอนิกส์จะช่วยเพิ่มศักยภาพในการจัดการประมูลในองค์กร งานวิจัยนี้มีจุดประสงค์ในการนำเสนอการประยุกต์เอเจนท์แบบเคลื่อนที่ในการประมูลอิเล็กทรอนิกส์โดยใช้รูปแบบการประมูลของตลาดกลางยางพาราของประเทศไทยมาเป็นกรณีศึกษา และได้ทำการพัฒนาระบบต้นแบบเพื่อทดสอบการทำงานของระบบเอเจนท์ที่ออกแบบไว้

คำสำคัญ: เอเจนท์แบบเคลื่อนที่ การประมูลอิเล็กทรอนิกส์

Abstract

An electronic auction is a form of electronic markets that become popular nowadays. To adopt mobile-agent technology for electronic auctions will enhance auction management in organizations. The research presented in this paper proposed a prototype of the adoption of mobile-agent technology for electronic auction at Rubber Brokering Service of Thailand and its performance testing.

Keyword: Mobile Agent, Electronic Auction.

1. บทนำ

พาณิชย์อิเล็กทรอนิกส์และตลาดอิเล็กทรอนิกส์ได้เข้ามามีบทบาทต่อการทำธุรกิจในปัจจุบันเป็นอย่างมาก เนื่องจากเป็นช่องทางที่จะทำให้ผู้ซื้อและผู้ขายสามารถติดต่อทำธุรกรรมได้สะดวกรวดเร็วยิ่งขึ้น การประมูลอิเล็กทรอนิกส์เป็นรูปแบบหนึ่งของตลาดอิเล็กทรอนิกส์ที่ได้รับความนิยมมากในปัจจุบัน หากมีการนำการประมูลอิเล็กทรอนิกส์มาประยุกต์ใช้กับตลาดกลางยางพารา ซึ่งเป็นหน่วยงานของรัฐที่ทำหน้าที่เสมือนนายหน้ารับซื้อยางพาราจากเกษตรกรและเปิดให้บริษัทที่ใช้ยางพาราเป็นวัตถุดิบในการผลิตเข้ามาประมูลก็จะเป็นทางเลือกในการปรับเปลี่ยนรูปแบบการประมูลของตลาดกลางยางพาราเพื่ออำนวยความสะดวกในการประมูลแก่ผู้ประกอบการที่ใช้บริการกับตลาดกลางยางพาราและเพื่อรองรับตลาดประมูลใหม่ๆ ที่อาจจะเกิดขึ้นในอนาคตอันเนื่องมาจากการขยายพื้นที่เพาะปลูกยางไปยังภาคอื่น ๆ ของประเทศไทย

งานวิจัยนี้ได้นำเสนอการประยุกต์ใช้เอเจนท์แบบเคลื่อนที่ในการประมูลอิเล็กทรอนิกส์โดยใช้รูปแบบการประมูลยางพาราของตลาดกลางยางพาราของประเทศไทยมาเป็นกรณีศึกษา เนื่องจากเอเจนท์แบบเคลื่อนที่มีความสามารถในการย้ายการทำงานไปยังคอมพิวเตอร์เครื่องอื่นในเครือข่าย และมีข้อดีคือ เครื่องที่เป็นเจ้าของเอเจนท์กับเครื่องที่เอเจนท์เดินทางไปประมวลผลนั้นไม่จำเป็นต้องติดต่อกันตลอดเวลา จึงเหมาะที่จะใช้งานกับเครือข่ายมีเสถียรภาพต่ำหรือเกิดปัญหา

เครือข่ายล่มได้ง่าย [1] สำหรับประเทศไทยแล้วนับว่ามีการกล่าวถึงการประยุกต์ใช้เทคโนโลยีเอเจนท์แบบเคลื่อนที่ในงานประมูลอิเล็กทรอนิกส์ค่อนข้างน้อยมาก

ในหัวข้อถัดไปของเอกสารนี้จะกล่าวถึงทฤษฎีที่เกี่ยวข้อง หัวข้อที่ 3 กล่าวถึงการออกแบบระบบ หัวข้อที่ 4 กล่าวถึงผลการดำเนินงาน และหัวข้อที่ 5 กล่าวถึงบทสรุปและงานที่จะทำในอนาคต

2. ทฤษฎีที่เกี่ยวข้อง

2.1 เอเจนท์แบบเคลื่อนที่

เอเจนท์ คือโปรแกรมคอมพิวเตอร์ซึ่งทำงานอัตโนมัติแทนคนหรือองค์กร [2] โดย [3] ได้กล่าวถึงคุณสมบัติพื้นฐานของเอเจนท์ไว้ได้แก่ การทำงานเองได้ (autonomy) การติดต่อกับเอเจนท์อื่น (social ability) การตอบโต้ (reactivity) และการตอบสนองในเชิงรุก (pro-activeness) ส่วนคุณสมบัติการเคลื่อนที่ (mobility) ของเอเจนท์นั้นเป็นคุณสมบัติเฉพาะเพิ่มขึ้นมาสำหรับการใช้เอเจนท์ในระบบแบบกระจาย โดยเอเจนท์สามารถทำการร้องขอไปยังคอมพิวเตอร์ปลายทางที่ต้องการไปประมวลผล โดยนำสถานะการทำงานรวมถึงรหัสคำสั่งและข้อมูลบางอย่างที่จำเป็นต้องใช้ไปด้วย หรือสามารถย้ายการประมวลผลไปยังเครื่องคอมพิวเตอร์อื่นที่อยู่ในเครือข่ายได้โดยอัตโนมัติ [4]

โดยปกติแล้วเอเจนท์สามารถทำงานอยู่ในสภาวะแวดล้อมที่รองรับการทำงานของเอเจนท์ เรียกว่า Agent Platform (AP) ตามข้อกำหนดของ FIPA [5] AP ประกอบด้วย เครื่องคอมพิวเตอร์ ระบบปฏิบัติการ ซอฟต์แวร์สนับสนุนเอเจนท์ เอเจนท์ และองค์ประกอบในการจัดการเอเจนท์ซึ่งประกอบด้วย (1) Agent Management System (AMS) ทำหน้าที่ในการควบคุมการเข้าถึงและใช้งานเอเจนท์ การลงทะเบียนเอเจนท์และวงจรชีวิตของเอเจนท์ (2) Directory Facilitator (DF) ทำหน้าที่ให้บริการไดเร็กทอรีสำหรับการสืบค้นบริการของเอเจนท์ และ (3) Message Transport Service (MTS) ทำหน้าที่ในการส่งข้อความระหว่างเอเจนท์ โดยข้อความที่ใช้ในการติดต่อสื่อสารระหว่างเอเจนท์นั้นจะต้องเป็นไปตามข้อกำหนดของโครงสร้างภาษาที่เลือกใช้เพื่อให้ทุกเอเจนท์เข้าใจตรงกัน และแม้ว่าจะไม่เข้าใจเนื้อหาแต่ก็ควรเข้าใจโครงสร้างของข้อความที่เป็นมาตรฐาน โครงสร้างนี้เรียกว่า Agent Communication Languages (ACLs)

2.2 การประมูลอิเล็กทรอนิกส์

การประมูลอิเล็กทรอนิกส์ช่วยลดข้อจำกัดของการประมูลปกติโดยผู้ประมูลไม่จำเป็นต้องเดินทางมายังสถานที่ในการประมูลด้วยตนเองแต่จะทำการเสนอราคาประมูลผ่านทางเครือข่ายอินเทอร์เน็ตซึ่งทำได้สะดวกรวดเร็ว วิธีการประมูลมีหลายรูปแบบ [6] เช่น

1. English Auction เป็นการประมูลโดยกำหนดราคาขั้นต่ำไว้ ผู้ที่เข้าร่วมประมูลจะเสนอราคาสูงขึ้นเรื่อยๆ จนกว่าจะไม่มีผู้ใดให้ราคาสูงกว่า หรือตรงตามเงื่อนไขของเจ้าของสินค้า ผู้ที่ให้ราคาสูงสุดจะได้รับสินค้าไป
2. Dutch Auction เป็นการประมูลโดยทำการตั้งราคาสูงสุดไว้ ผู้เข้าร่วมประมูลจะเสนอราคาต่ำลงเรื่อยๆ จนกว่าจะมีผู้ที่พอใจราคานั้น
3. American Auction หรือ Closed Bid Auction เป็นการประมูลที่การเสนอราคาเป็นความลับ โดยเมื่อถึงเวลาเปิดซองประมูล ผู้ที่ให้ราคาสูงสุดจะเป็นผู้ชนะ
4. Vickrey Auction หรือ Uniform Second Auction เป็นการประมูลที่การเสนอราคาเป็นความลับเช่นเดียวกับ American Auction และผู้ให้ราคาสูงสุดเป็นผู้ชนะแต่จะจ่ายเงินในราคาเท่ากับที่ผู้เสนอเป็นอันดับสองเท่านั้น

แต่อย่างไรก็ตาม การประมูลอิเล็กทรอนิกส์มักอาศัยการสร้างโปรแกรมประยุกต์บนระบบอินเทอร์เน็ตเช่นเดียวกับพาณิชย์อิเล็กทรอนิกส์โดยทั่วๆ ไปหรืออาจใช้ในรูปแบบการสื่อสารแบบอิเล็กทรอนิกส์

2.3 การประมูลยางพารา ณ ตลาดกลางยางพารา

สำนักตลาดกลางยางพาราเป็นหน่วยงานภายใต้การบริหารและการจัดการของกรมวิชาการเกษตร กระทรวงเกษตรและสหกรณ์ซึ่งได้มีการจัดตั้งตลาดกลางยางพาราเพื่อให้บริการซื้อขายยางโดยวิธีการประมูลภายใต้กฎและระเบียบที่ตลาดกลางกำหนด ปัจจุบันเปิดให้บริการ 3 แห่งคือ ตลาดกลางยางพาราหาดใหญ่ ตลาดกลางยางพาราสุราษฎร์ธานีและตลาดกลางยางพารานครศรีธรรมราช ขั้นตอนการประมูลยางพารา ณ ตลาดกลางยางพารา อำเภอหาดใหญ่ จังหวัดสงขลา เป็นดังนี้

1. ผู้ซื้อและผู้ขายยางต้องทำการลงทะเบียนก่อนการใช้บริการของตลาดกลาง

2. ผู้ขายนำยางมาคัดคุณภาพยาง โดยเจ้าหน้าที่ของตลาดกลางเป็นผู้คัดคุณภาพยางตามที่สถาบันวิจัยยางกำหนด และนำยางที่ผ่านการคัดคุณภาพแล้วมาชั่งน้ำหนักยาง ซึ่งตลาดกลางจะมีการเก็บข้อมูลว่าผู้ขายแต่ละรายขายยางประเภทใดได้กี่กิโลกรัม

3. ตลาดกลางจะรวบรวมยางที่มีคุณภาพเดียวกันของแต่ละประเภทเพื่อนำเข้าประมูลพร้อมกันโดยมีการกำหนดราคาขั้นต่ำไว้

4. เมื่อถึงเวลาที่กำหนดยื่นประมูล ผู้ซื้อทำการยื่นซองประมูลเป็นความลับด้วยตนเองหรือทางโทรศัพท์

5. เมื่อถึงเวลาเปิดซองประมูล ผู้ซื้อที่ให้ราคาสูงสุดสำหรับยางแต่ละประเภทเป็นผู้ชนะการประมูลและได้รับยางทั้งหมดของชนิดที่ยื่นซองประมูลนั้น กรณีเสนอราคาสูงสุดเท่ากันให้ผู้ยื่นประมูลก่อนเป็นผู้ชนะการประมูล

3. การออกแบบระบบ

3.1 การกำหนดเอเจนท์และความรับผิดชอบของเอเจนท์

ขั้นตอนนี้เป็นการกำหนดประเภทของเอเจนท์และหน้าที่ความรับผิดชอบดังนี้

1. CoordinatorAgent เอเจนท์ที่ทำหน้าที่ประสานงานในการประมูลโดยมีงานที่ต้องทำได้แก่ ลงทะเบียนบริการรับเสนอราคาประมูล รับราคาประมูลจาก BuyerMobileAgent ในช่วงเวลาที่กำหนดและถอดรหัสข้อมูลการเสนอราคา บันทึกการเสนอราคาในฐานข้อมูล รับผลการประมูลจาก SelectionAgent และส่งผลการประมูลให้ BuyerMobileAgent

2. SelectionAgent เอเจนท์ที่ทำหน้าที่เลือกผู้ชนะการประมูลโดยมีงานที่ต้องทำได้แก่ เลือกผู้ชนะการประมูลที่ให้ราคาสูงสุดของยางแต่ละประเภท มีเงื่อนไขว่าราคาประมูลต้องไม่ต่ำกว่าราคากลาง บันทึกผลการประมูลในฐานข้อมูลและส่งผลการประมูลให้ CoordinatorAgent

3. BuyerAgentInterface เอเจนท์ที่ทำหน้าที่ติดต่อกับผู้ใช้ในการรับราคาประมูลของผู้ซื้อโดยมีงานที่ทำได้แก่ ลงทะเบียนเข้าประมูล รับข้อมูลราคากลางและกำหนดการประมูลจากตลาดกลาง รับราคาเสนอประมูลจากผู้ซื้อและเข้ารหัสข้อมูลและสร้าง BuyerMobileAgent ที่มีข้อมูลในการประมูล

4. BuyerMobileAgent เอเจนท์แบบเคลื่อนที่ของผู้ซื้อทำหน้าที่เสนอราคาประมูลโดยมีงานที่ทำได้แก่ เดินทางไปยื่นราคาประมูลแก่ CoordinatorAgent ในตลาดอิเล็กทรอนิกส์ รับผลการประมูลจาก CoordinatorAgent เดินทางกลับไปยังระบบเอเจนท์ของผู้ซื้อ และส่งผลการประมูลให้ BuyerAgentInterface

3.2 การกำหนดขั้นตอนการประมูลด้วยเอเจนท์

ก่อนที่จะเกิดการประมูลนั้นผู้ที่ประมูลต้องทำการขอใช้งานระบบกับตลาดกลางยางพาราก่อนเพื่อที่ตลาดกลางจะทำการจัดเก็บข้อมูลของผู้ที่จะเข้าประมูลพร้อมทั้งให้ข้อมูลที่จำเป็นในการใช้งานระบบเช่น บัญชีผู้ใช้ รหัสผ่าน กุญแจลับ (private key) และกุญแจสาธารณะ (public key) ที่จะใช้ในการเข้ารหัสและถอดรหัสในการประมูล

ขั้นตอนการประมูลด้วยเอเจนท์นั้นต้องมีการทำงานร่วมกันระหว่างเอเจนท์ เจ้าของเอเจนท์ และทรัพยากรที่เกี่ยวข้องดังแสดงในภาพที่ 1 โดยมีขั้นตอนดังนี้

1. เจ้าหน้าที่ตลาดกลางจะเป็นผู้กำหนดราคากลางและกำหนดการประมูลไว้ในฐานข้อมูลผ่านระบบเว็บ

2. ผู้ประมูลสร้าง BuyerAgentInterface เพื่อขอลงทะเบียนและรับข้อมูลราคากลางและกำหนดการประมูล

3. ผู้ประมูลกำหนดราคาประมูลให้ BuyerAgentInterface

4. BuyerAgentInterface เข้ารหัสข้อมูลการเสนอราคาและสร้าง BuyerMobileAgent ที่มีข้อมูลในการเสนอราคาที่ถูกเข้ารหัสแล้ว

5. BuyerMobileAgent เดินทางไปยังตลาดอิเล็กทรอนิกส์เพื่อยื่นประมูลให้แก่ CoordinatorAgent

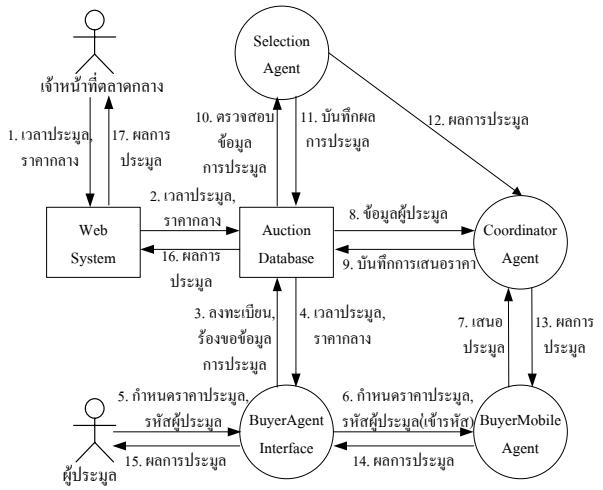
6. CoordinatorAgent รับข้อความการเสนอราคา ทำการถอดรหัสข้อมูลและตรวจสอบข้อมูลของผู้ประมูล และทำการบันทึกข้อมูลการเสนอราคาในฐานข้อมูล

7. เมื่อถึงเวลาเลือกผู้ชนะการประมูลที่กำหนดไว้แล้ว SelectionAgent จะตรวจสอบการเสนอราคาและเลือกผู้ที่ให้ราคาสูงสุดเป็นผู้ชนะการประมูล และส่งผลการประมูลให้แก่ CoordinatorAgent

8. CoordinatorAgent ส่งผลการประมูลให้ BuyerMobileAgent

9. BuyerMobileAgent เดินทางกลับไปยังเซิร์ฟเวอร์ของผู้ประมูลและส่งผลการประมูลให้แก่ BuyerAgentInterface

10. เจ้าหน้าที่ตลาดกลางสามารถดูผลการประมูลผ่านระบบเว็บ



ภาพที่ 1 : ขั้นตอนการประมูลด้วยเอเจนท์และความสัมพันธ์ระหว่างเอเจนท์ ฐานข้อมูลการประมูล ระบบเว็บ และผู้ที่เกี่ยวข้อง

3.3 การออกแบบการประยุกต์เอเจนท์แบบเคลื่อนที่ในการประมูลอิเล็กทรอนิกส์ของตลาดกลางยางพารา

การออกแบบการประยุกต์เอเจนท์แบบเคลื่อนที่ในการประมูลอิเล็กทรอนิกส์ของตลาดกลางยางพารามีองค์ประกอบหลัก ดังนี้

1. เอเจนท์ของตลาดกลาง ได้แก่ CoordinatorAgent และ SelectionAgent

2. เอเจนท์ของผู้ซื้อได้แก่ BuyerAgentInterface และ BuyerMobileAgent

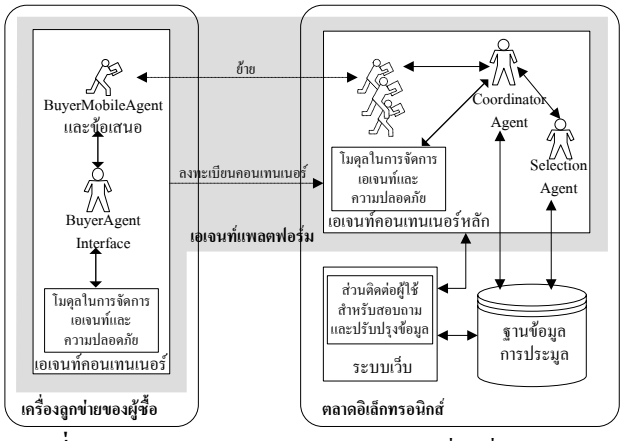
3. ส่วนติดต่อผู้ใช้ในลงทะเบียนและสืบค้นข้อมูลการประมูล

4. โมดุลในการจัดการเอเจนท์และความปลอดภัย (Agent Management and Security Module) เพื่อทำหน้าที่ในการจัดการเข้ารหัสและถอดรหัสข้อมูลการเสนอราคา รวมถึงทำหน้าที่ในการตรวจสอบและรับรองเอเจนท์แบบเคลื่อนที่ที่จะเข้ามาประมูลในตลาดอิเล็กทรอนิกส์

5. ระบบฐานข้อมูลสำหรับการประมูล ซึ่งประกอบด้วยข้อมูลการลงทะเบียนของผู้ใช้บริการและข้อมูลการประมูล

6. ระบบเว็บ (Web System) รองรับดำเนินการในการควบคุมการประมูลโดยเจ้าหน้าที่ของตลาดกลาง ได้แก่การบันทึกข้อมูลการลงทะเบียนเข้าใช้งาน ข้อมูลราคากลาง และการเปิดตลาดประมูล

การออกแบบการประยุกต์เอเจนท์แบบเคลื่อนที่ในการประมูลอิเล็กทรอนิกส์ของตลาดกลางยางพาราแสดงดังภาพที่ 2



ภาพที่ 2: การออกแบบการประยุกต์เอเจนท์แบบเคลื่อนที่ในการประมูลอิเล็กทรอนิกส์ของตลาดกลางยางพารา

3.4 การออกแบบรูปแบบการส่งข้อความในการประมูล

การเสนอราคาประมูลนั้นต้องมีรูปแบบการติดต่อสื่อสารระหว่างเอเจนท์เพื่อให้ทุกเอเจนท์เข้าใจตรงกันและเป็นไปตามกระบวนการในการประมูล ลำดับการส่งข้อความระหว่างเอเจนท์ที่ใช้ในการประมูลเป็นดังนี้

1. CoordinatorAgent ลงทะเบียนบริการรับเสนอราคาประมูลกับ Directory Facilitator

2. BuyerMobileAgent สอบถาม DF ว่ามีใครบริการรับเสนอราคา

3. DF ส่งชื่อเอเจนท์ที่ลงทะเบียนบริการรับเสนอราคาไว้ในที่นี้คือ CoordinatorAgent

4. BuyerMobileAgent ส่งข้อเสนอ (Proposal) ที่เข้ารหัสแล้วให้แก่ CoordinatorAgent

5. CoordinatorAgent ส่งยอมรับหรือปฏิเสธรับเสนอราคาแก่ BuyerMobileAgent

6. SelectionAgent ทำการส่งผลการประมูลให้แก่ CoordinatorAgent

7. CoordinatorAgent ทำการส่งผลการประมูลให้แก่ BuyerMobileAgent

8. BuyerMobileAgent ทำการส่งผลการประมูลให้แก่ BuyerAgentInterface

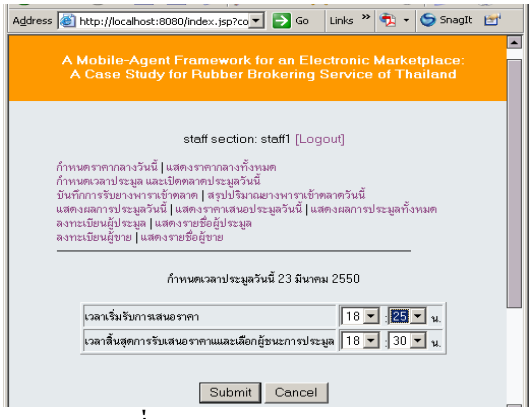
4. ผลการดำเนินงาน

จากการออกแบบในหัวข้อที่ 3 ได้นำมาพัฒนาโปรแกรมประยุกต์ต้นแบบเพื่อทดสอบการประมูลอิเล็กทรอนิกส์โดยใช้เอเจนท์แบบเคลื่อนที่ได้ผลการดำเนินงานและการทดสอบการประมูลดังนี้

4.1 การพัฒนาโปรแกรมประยุกต์ต้นแบบ

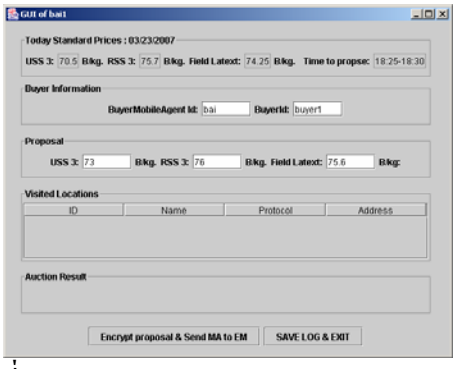
ในการพัฒนาโปรแกรมประยุกต์ได้ทำการติดตั้งเครื่องมือบนระบบซึ่งประกอบด้วยเครื่องเซิร์ฟเวอร์สำหรับตลาดอิเล็กทรอนิกส์และเครื่องไคลเอ็นท์สำหรับผู้ประมูลซึ่งทำการเชื่อมต่อกันผ่านเครือข่ายอินเทอร์เน็ต เครื่องมือหลักในการพัฒนาระบบคือ JADE (Java Agent Development Framework) [7] ซึ่งเป็นเครื่องมือในการพัฒนาเอเจนท์ด้วยภาษาจาวา ประกอบด้วยชุดคำสั่งสำหรับการพัฒนาเอเจนท์และ JADE Platform เป็นสภาวะแวดล้อมในการทำงานของเอเจนท์ ไลบรารี่ Bouncy Castle Crypto APIs [8] สำหรับการจัดการการเข้ารหัส ไลบรารี่ log4j [9] สำหรับการจัดการบันทึกการทำงาน ระบบจัดการฐานข้อมูล MySQL และเซิร์ฟเวอร์ให้บริการเว็บ Apache Tomcat สำหรับเป็นส่วนติดต่อผู้ใช้

ในการประมูลด้วยเอเจนท์นั้น เจ้าหน้าที่ตลาดกลางจะทำการกำหนดราคากลางและกำหนดเวลาการประมูลผ่านเว็บและจัดเก็บข้อมูลในฐานข้อมูล ตัวอย่างส่วนติดต่อผู้ใช้ผ่านเว็บในการกำหนดเวลาการประมูลของเจ้าหน้าที่ตลาดกลางยางพาราแสดงดังภาพที่ 3



ภาพที่ 3: การกำหนดเวลาประมูลผ่านเว็บ

เมื่อกำหนดราคากลางและเวลาประมูลแล้ว ระบบเอเจนท์จะเริ่มทำงานพร้อมกับสร้างเอเจนท์ของระบบคือ AMS และ DF จากนั้นทำการสร้าง CoordinatorAgent และ SelectionAgent ในคอนเทนเนอร์หลักซึ่งจะทำงานตามหน้าที่และเวลาที่กำหนด เมื่อมีผู้ประมูลทำการสร้างลงทะเบียนขอเข้าใช้งานในตลาดอิเล็กทรอนิกส์ก็จะเกิดคอนเทนเนอร์ใหม่ขึ้น จากนั้นผู้ประมูลจะทำการสร้าง BuyerAgentInterface ดังภาพที่ 4



ภาพที่ 4: ส่วนติดต่อผู้ใช้แบบกราฟิกของ BuyerAgentInterface

BuyerAgentInterface จะทำการเข้ารหัสข้อมูลด้วยโมดุลในการเข้ารหัสซึ่งในที่นี้ใช้อัลกอริทึม RSA [10] ในการเข้ารหัสด้วยกุญแจสาธารณะของตลาดกลางเพื่อให้ข้อมูลการเสนอราคาเป็นความลับและเฉพาะเอเจนท์ที่มีกุญแจลับของตลาดกลางเท่านั้นที่จะสามารถถอดรหัสได้ จากนั้นทำการสร้าง BuyerMobileAgent และส่ง BuyerMobileAgent ที่มีข้อมูลการเสนอราคาที่เข้ารหัสไว้แล้วไปยังคอนเทนเนอร์หลัก (main container) เพื่อประมูลตามเวลาที่กำหนด การทำงานที่เกิดขึ้นในระบบเอเจนท์ของการประมูลอิเล็กทรอนิกส์จะถูกจัดเก็บบันทึกการทำงานไว้

4.2 การทดสอบการประมูล

ในการทดสอบการประมูลมีจุดประสงค์เพื่อทดสอบว่าเมื่อมีหลายเอเจนท์ร่วมทำการประมูล แต่ละเอเจนท์ที่เกี่ยวข้องในการประมูลสามารถทำงานได้ถูกต้องตามที่ออกแบบไว้ สภาวะแวดล้อมในการทดสอบประกอบด้วยเครื่องเซิร์ฟเวอร์ของตลาดอิเล็กทรอนิกส์ 1 เครื่อง มีหน่วยประมวลผลกลาง Intel Pentium III 733 MHz หน่วยความจำ 320 MB และเครื่องไคลเอ็นท์ของผู้ประมูล 5 เครื่องเชื่อมต่อกันเพื่อส่งเอเจนท์แบบเคลื่อนที่รวม 40 เอเจนท์พร้อมกันผ่านเครือข่ายของ

ห้องปฏิบัติการภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยสงขลานครินทร์ โดยได้ทำการทดสอบว่าเอเจนท์ทั้งหมดในตลาดอิเล็กทรอนิกส์สามารถทำงานได้ถูกต้องตามเงื่อนไขการประมูลของตลาดกลางหรือไม่ โดยเงื่อนไขการประมูลของตลาดกลางมีดังนี้

1. ผู้ประมูลสามารถยื่นประมูลได้ครั้งเดียว ดังนั้นหากมีเอเจนท์ยื่นประมูลมากกว่า 1 ครั้งจะรับการเสนอราคาเฉพาะครั้งแรกเท่านั้น

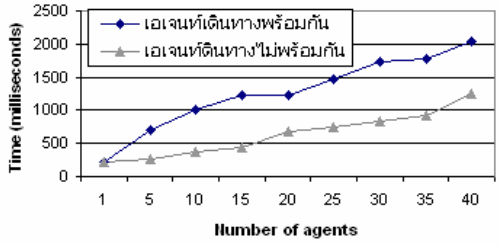
2. ข้อมูลการเสนอราคาต้องถูกเข้ารหัสด้วยกุญแจสาธารณะของตลาดกลาง หากเอเจนท์ของตลาดกลางถอดรหัสไม่ได้จะไม่รับการเสนอราคาประมูล

3. เมื่อเอเจนท์ของตลาดกลางถอดรหัสข้อมูลการเสนอราคาแล้วต้องสามารถตรวจสอบรหัสผู้ประมูลว่าได้ลงทะเบียนเป็นผู้ประมูลของตลาดกลางแล้ว หากยังไม่ได้ลงทะเบียนไว้จะไม่รับการเสนอราคาประมูล

จากการทดสอบการประมูลพบว่าเอเจนท์ทั้งหมดทำงานได้ถูกต้องตามตามเงื่อนไขที่กำหนดไว้

4.3 การทดสอบประสิทธิภาพของเอเจนท์แบบเคลื่อนที่

การวัดประสิทธิภาพของเอเจนท์แบบเคลื่อนที่พิจารณาจากเวลาในการเดินทางไปประมูลของเอเจนท์แบบเคลื่อนที่โดยเปรียบเทียบเวลาที่ใช้ในการเดินทางของเอเจนท์แบบเคลื่อนที่ไปยังตลาดอิเล็กทรอนิกส์ระหว่างการส่งเอเจนท์พร้อมกันกับไม่พร้อมกัน ผลการทดสอบแสดงดังภาพที่ 5



ภาพที่ 5: ผลการทดสอบการเปรียบเทียบเวลาในการเดินทางของเอเจนท์

จากผลการทดสอบสรุปได้ว่า ถ้ามีจำนวนเอเจนท์แบบเคลื่อนที่เข้ามาประมูลมากขึ้นจะใช้เวลาในการเดินทางไปประมูลเพิ่มขึ้น และหากมีจำนวนเอเจนท์มากเกินไปจะทำให้เซิร์ฟเวอร์ไม่สามารถให้บริการทรัพยากรแก่ทุกเอเจนท์ได้ ดังนั้นหากนำระบบเอเจนท์แบบเคลื่อนที่ไปใช้ในระบบงานจริงต้องพิจารณาคุณสมบัติเครื่องเซิร์ฟเวอร์ให้มีประสิทธิภาพสูงพอที่จะรองรับจำนวนของเอเจนท์แบบเคลื่อนที่ทั้งหมดที่จะเข้ามาประมูลได้

5. บทสรุปและงานที่จะทำในอนาคต

ในการประยุกต์ใช้เอเจนท์แบบเคลื่อนที่สำหรับการประมูลอิเล็กทรอนิกส์ของตลาดกลางยางพาราแห่งประเทศไทยผู้วิจัยได้ดำเนินการออกแบบการทำงานของเอเจนท์ที่ใช้ในระบบและพัฒนาโปรแกรมประยุกต์ต้นแบบตามโครงสร้างที่ออกแบบไว้เพื่อทดสอบการทำงานของระบบเอเจนท์ซึ่งจากการทดสอบพบว่าเอเจนท์สามารถทำงานได้ตามเงื่อนไขในการทดสอบ จากการวิเคราะห์ระบบงานของตลาดกลางยางพารา อำเภอหาดใหญ่ จังหวัดสงขลา และการพัฒนาระบบต้นแบบในห้องปฏิบัติการพบว่าประเด็นความปลอดภัย [11] เป็นเรื่องที่สำคัญยิ่งในการนำระบบต้นแบบไปใช้งานจริงซึ่งในอนาคตจำเป็นต้องพัฒนาส่วนนี้ให้รัดกุมยิ่งขึ้น

6. เอกสารอ้างอิง

[1] D. B. Lange and M. Oshima, “Programming and Deploying Java(TM) Mobile Agents with Aglets”, pp. 3-5. Addison Wesley Longman, Inc., Massachusetts, 2003.

[2] Object Management Group(OMG), “Mobile Agent Facility V 1.0,” March 2007; http://www.omg.org/docs/formal/00-01-02.pdf.

[3] M. Wooldridge and N. R. Jennings, “Intelligent Agents: Theory and Practice”, *The Knowledge Engineering Review*, vol. 10, no. 2, pp. 115-152, 1995.

[4] N. M. Karnik and A. R.Tripathi, "Design Issues in Mobile-Agent Programming Systems," *IEEE Concurrency*, vol. 6, no. 3, pp. 52-61, 1998.

[5] Foundation for Intelligent Physical Agents (FIPA), “FIPA Agent Management Specification,” March 2007; http://www.fipa.org/specs/fipa00023/SC00023K.html.

[6] A. C. B. Garcia, A. Lopes, and C. Bentes, “Electronic Auction with autonomous intelligent agents: Finding Opportunities by being there”, *Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial*. No.13, pp. 45-52, 2001.

[7] Telecom Italia. “Java Agent Development Framework (JADE),” March 2007; http://jade.tilab.com

[8] Tau Ceti Co-operative Ltd. “Legion of the Bouncy Castle Java cryptography APIs,” March 2007; http://www.bouncycastle.org/java.html.

[9] Apache Software Foundation. “Logging Services Log4j,” March 2007; http://logging.apache.org/log4j/docs/index.html.

[10] C. Kaufman, R. Perlman, and M.Speiner, “Network Security PRIVATE Communication in a PUBLIC World”, pp. 134-145. Prentice-Hall Inc., New Jersey, 1995.

[11] W. Jansen and T. Karygiannis, “NIST Special Publication 800-19 – Mobile Agent Security,” March 2007; http://csrc.nist.gov/publications/nistpubs/800-19/sp800-19.pdf.