สำนักหอสมุดกลาง พระจอมเกล้าฉ

ฟรอนต์เอ็นสำหรับโปรแกรมเน็ตเวิร์คซิมูเลชั่น

Network Simulation Front End

ณัฐพล สุธนะดิลก

นภดล เจียรวิจิตร์

วพ.
ณ342ฟ
2550

เลขหมู่.....82039
เลขทะเบียน.....
วัน,เดือน,ปี...-4 ก.ค. 2551

.b.....11943506
.i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2550

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2550

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ฟรอนต์เอ็นสำหรับโปรแกรมเน็ตเวิร์คซิมูเลชั่น

Network Simulation Front End

ผู้จัดทำ

1. นายณัฐพล สุธนะดิลก รหัสนักศึกษา 47010234

2. นายนภดล เจียรวิจิตร์ รหัสนักศึกษา 47010357

_____ อาจารย์ที่ปรึกษา

(อาจารย์ ธนัญชัย ตรีภาค)

_____ อาจารย์ที่ปรึกษา

(ผศ. ธนา หงษ์สุวรรณ)

_____ อาจารย์ที่ปรึกษา

(อาจารย์ อัครเดช วัชระภูพงษ์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟรอนต์เอ็นสำหรับโปรแกรมเน็ตเวิร์คซิมูเลชั่น

นาย ณัฐพล สุธนะดิลก 47010234
นาย นภดล เจียรวิจิตร์ 47010357
อาจารย์ธนัญชัย ตรีภาค อาจารย์ที่ปรึกษา
อาจารย์อัครเดช วัชระภูพงษ์ อาจารย์ที่ปรึกษาร่วม
ผศ.ธนา หงส์สุวรรณ อาจารย์ที่ปรึกษาร่วม
ปีการศึกษา 2550

บทคัดย่อ

ในปัจจุบันเครือข่ายของคอมพิวเตอร์มีอยู่ทุกหนแห่งตั้งแต่ขนาดเล็กเช่นเครือข่ายในบ้านไปจนขนาดใหญ่อย่าง เครือข่ายคอมพิวเตอร์ในบริษัทหรือองค์กรขนาดใหญ่ เครือข่ายคอมพิวเตอร์ในปัจจุบันจึงถือได้ว่ามีความสำคัญมาก การทำความเข้าใจเกี่ยวกับเครือข่ายในปัจจุบันนั้นมีโปรแกรมการจำลองเครือข่ายเกิดขึ้นมากมาย ส่วนใหญ่จะเน้นไปทางจำลองเพื่อทำความเข้าใจในทฤษฎีหรือการเชื่อมต่อต่างๆ มากกว่าที่จะพิจารณาด้านการสัญจรของข้อมูล โปรแกรมเน็ตเวิร์คซิมูเลชั่นเป็นหนึ่งในโปรแกรมที่จำลองเครือข่ายในลักษณะการจำลองเพื่อดูการสัญจรของข้อมูลที่วิ่งเข้าออก การเข้าคิวของข้อมูลหรือการสูญหายของข้อมูล โดยโปรแกรมเน็ตเวิร์คซิมูเลชั่นถือได้ว่ามีความสามารถค่อนข้างสูงสำหรับจำลองโพรโตคอลต่าง ๆ ได้หลากหลาย รวมไปถึงเครือข่ายไร้สาย โปรแกรมนี้มีจุดอ่อนตรงการใช้งานต้องกระทำผ่านโปรแกรมแปลคำสั่งภาษาทีซีแอล ซึ่งไม่สะดวก ผู้ใช้จำเป็นต้องทราบรูปแบบของคำสั่งและการติดตั้งโปรแกรมมีความยุ่งยาก ดังนั้นโครงงานนี้จึงมีเป้าหมายในการพัฒนาส่วนฟรอนต์เอ็นของโปรแกรมเน็ตเวิร์คซิมูเลชั่นเพื่อสะดวกต่อการใช้งานสำหรับผู้ที่สนใจใช้งานหรืออาจารย์ที่ต้องการโปรแกรมเพื่อส่งเสริมการเรียนรู้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Network Simulation Front End

Mr. Nuttapol	Sutanadilok	47010234
Mr. Napadon	Chianvijit	47010357
Mr. Thananchai	Treepak	Advisor
Mr. Akkradach	Watcharapupong	Co-Advisor
Asst.Prof. Thana	Hongsuwan	Co-Advisor

Academic Year 2007

ABSTRACT

Nowadays, computer networks exist worldwide from small size as household to large size network as computerize networks in large companies or organizations. The Computer network currently plays a very important role. Understanding in networks mostly focus on simulate network, which is based on theories or connectivity rather than data transfer. The network simulation is one of the programs which aim on data inflow and outflow, queuing or loss. The network simulation is of high capability for several simulate protocols and wireless networks. However, this program has a weak point in terms of the fact that it must be carried out through interpreters. Therefore, it is not convenient for users because they have to know specific command and the program installation is also difficult. This project thus aims at develop program front-end and is convenient for those who utilize the network simulation program.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานฟรอนต์เอ็นสำหรับโปรแกรมเน็ตเวิร์คซิมูเลชั่น มิอาจสำเร็จสมบูรณ์และลุล่วงไปได้ด้วยดี ถ้าขาดการให้โอกาส การดูแล และการให้คำแนะนำในการพัฒนาและจัดทำโครงการ การสั่งสอนที่เป็นอย่างดีเสมอมา จากท่านอาจารย์ที่ปรึกษา อาจารย์ธนัญชัย ตรีภาค ผู้ช่วยศาสตราจารย์ธนา หงษ์สุวรรณ อาจารย์อัครเดช วัชระภูพงษ์ และดร.อรัญญา วลัยรัชต์ซึ่งขอขอบพระคุณอาจารย์ทั้งสี่ท่านเป็นอย่างสูงและขอขอบพระคุณคณาอาจารย์ทุกท่านในภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ซึ่งผู้จัดทำขอขอบพระคุณอย่างสูงมา ณ ที่นี้

ขอขอบคุณห้องวิจัยและพัฒนาการรักษาความปลอดภัยข้อมูล (ISAG) ที่เป็นแหล่งประสิทธิ์ประสาทวิชาให้ความรู้ความเข้าใจ และสนับสนุนสถานที่และอุปกรณ์เครือข่ายสำหรับการใช้ในการพัฒนาโครงงาน สุดท้ายขอกราบขอบพระคุณ บิดา มารดา ครอบครัว และน้องเนมที่เป็นกำลังใจ และให้การสนับสนุนในด้านต่างๆ ทำให้ผู้จัดทำสามารถทำวิทยานิพนธ์ฉบับนี้ลุล่วงด้วยดี ทางผู้จัดทำโครงงานฟรอนต์เอ็นสำหรับโปรแกรมเน็ตเวิร์คซิมูเลชั่น จึงขอขอบพระคุณมา ณ ที่นี้

นายณัฐพล สุธนะดิลก

นายนภดล เจียรวิจิตร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VII

สารบัญรูป VIII

บทที่ 1 บทนำ

1.1 ความสำคัญและที่มาของโครงงาน 1

1.2 วัตถุประสงค์ของโครงงาน 1

1.3 ประโยชน์ที่คาดว่าจะได้รับ 2

1.4 ขอบเขตของโครงงาน 2

1.5 ขั้นตอนการดำเนินงาน 2

1.6 เนื้อหาของรายงาน 3

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

2.1 โปรแกรมเน็ตเวิร์คซิมูเลชั่น 4

2.1.1 โครงสร้างคำสั่งของโปรแกรมเน็ตเวิร์คซิมูเลชั่น 8

2.1.2 ส่วนประกอบของโปรแกรมเน็ตเวิร์คซิมูเลชั่น 12

2.1.3 โหนดและเร้าติ้ง 12

2.1.4 ลิงค์ 14

2.1.5 เอเจนต์โพรโตคอล 15

2.1.5.1 เอเจนต์ยูดีพี 15

2.1.5.2 เอเจนต์ทีซีพี 16

2.1.6 แอปพลิเคชั่น 17

2.1.7 ตารางเหตุการณ์ 18

2.2 เทคโนโลยีจาวาเว็บสตาร์ท 20

2.2.1 โครงสร้างพื้นฐานของกระบวนการดีพลอยเมนต์ 21

2.2.2 รายละเอียดของจาวาเว็บสตาร์ท 26

2.2.3 การรักษาความปลอดภัย 27

2.2.4 การเรียกใช้งานจาวาเว็บสตาร์ท 27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

บทที่ 3 การออกแบบและพัฒนา

3.1 บทนำ 29

3.2 โครงสร้างของโครงงาน 29

3.2.1 เครื่องมือที่ใช้ในการพัฒนา 29

3.3 รายละเอียดโปรแกรมที่พัฒนา 30

3.3.1 รายละเอียดส่วนนำเข้า 30

3.3.2 รายละเอียดส่วนแสดงผล 30

3.3.3 รายละเอียดฟังก์ชัน 30

3.3.4 โครงสร้างของซอฟต์แวร์ 31

3.3.4.1 ส่วนจำลองเครือข่าย 31

3.3.4.2 ส่วนการแสดงผล 32

3.3.4.3 ส่วนเซิร์ฟเวอร์ที่ให้บริการ 33

3.3.5 ขอบเขตและข้อจำกัดของโปรแกรมที่พัฒนา 33

3.4 การออกแบบและพัฒนา 33

3.4.1 ส่วนจำลองเครือข่าย 33

3.4.2 ส่วนการแสดงผล 39

3.4.3 ส่วนเซิร์ฟเวอร์ที่ให้บริการ 41

3.4.4 กระบวนการเรียกการทำงานของโปรแกรมฟรอนต์เอ็น 43

บทที่ 4 การทดลองและผลการทดลอง

4.1 บทนำ 46

4.2 การทดสอบในส่วนการเรียกใช้งานแอปพลิเคชั่น 46

4.3 การทดสอบในส่วนการสร้างเครือข่ายจำลอง 49

4.4 การทดสอบในส่วนการกำหนดตารางเหตุการณ์ 50

4.5 การทดลองในส่วนการสร้างไฟล์ทีซีแอลและการเชื่อมต่อกับเซิร์ฟเวอร์ 51

4.6 การทดสอบในส่วนการแสดงผลภาพเคลื่อนไหว 53

4.7 การทดลองการใช้งานฟังก์ชั่นของโปรแกรมฟรอนต์เอ็น 53

4.7.1 การทดลองการทำงานของโพรโตคอลทีซีพี 53

4.7.2 การทดลองการทำงานของโพรโตคอลยูดีพี 55

4.7.3 การทดลองการกำหนดค่าดอสของลิงค์และโพรโตคอลเลือกเส้นทาง 56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

4.7.4 การทดลองการทำงานของโพรโตคอลเลือกเส้นทาง 58

บทที่ 5 บทวิจารณ์และสรุป

5.1 สรุปผลการพัฒนา 60

5.2 ปัญหาและอุปสรรค 61

5.3 แนวทางในการพัฒนาต่อ 61

บรรณานุกรม 62

ภาคผนวก

ภาคผนวก ก. คู่มือการติดตั้งซอฟต์แวร์ 63

ภาคผนวก ข. ภาษาทีซีแอลและโอทีซีแอล 71

ภาคผนวก ค. ความหมายของไฟล์เอ็นเอเอ็ม 75

ภาคผนวก ง. คู่มือการใช้งานซอฟต์แวร์ 83

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ หน้า

ง.1 แสดงความสัมพันธ์ของการเชื่อมต่อ 87

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ หน้า

2.1 แสดงความสัมพันธ์ระหว่างอ็อบเจกต์ของภาษาโอทีซีแอลและภาษาซีพลัสพลัส 5

2.2 แสดงกระบวนการทำงานของโปรแกรมเน็ตเวิร์คซิมูเลชั่นโดยภาพรวม 5

2.3 มุมมองแบบสถาปัตยกรรมของโปรแกรมเน็ตเวิร์คซิมูเลชั่น 6

2.4 แสดงกระบวนการประมวลผลของโปรแกรมเน็ตเวิร์คซิมูเลชั่น 7

2.5 แสดงตัวอย่างไฟล์ทีอาร์ 7

2.6 แสดงตัวอย่างไฟล์เอ็นเอเอ็ม 8

2.7 ลิงค์และโหนด 9

2.8 การสร้างทีซีพีเอเจนต์ 10

2.9 การสร้างเอฟทีพี 11

2.10 แสดงลำดับของคลาสในโปรแกรมเน็ตเวิร์คซิมูเลชั่น 12

2.11 โครงสร้างภายในของโหนดแบบยูนิคาสต์ 13

2.12 โครงสร้างภายในของโหนดแบบมัลติคาสต์ 13

2.13 โครงสร้างภายในของลิงค์ 14

2.14 โครงสร้างการทำงานภายในของลิงค์ 15

2.15 เอเจนต์บางส่วนในโปรแกรมเน็ตเวิร์คซิมูเลชั่น 15

2.16 ตัวอย่างของแอปพลิเคชั่น 17

2.17 แสดงการทำงานของตารางเหตุการณ์ 18

2.18 ตัวอย่างของตารางเวลาเหตุการณ์ 19

2.19 ข้อมูลภายในเฮดเดอร์กลุ่มข้อมูลของโปรแกรมเน็ตเวิร์คซิมูเลชั่น 19

2.20 บทบาทและหน้าที่ของตัวช่วยแอปพลิเคชั่น 21

2.21 โครงสร้างพื้นฐานของการดีพลอยเมนต์ 21

2.22 โครงสร้างโดยสมบูรณ์ของการดีพลอยเมนต์ 22

2.23 วงจรชีวิตของจาวาเว็บสตาร์ท 28

3.1 โครงสร้างของโปรแกรม 32

3.2 แสดงขั้นตอนการสร้างโพรโตคอล 34

3.3 แสดงขั้นตอนการเลื่อนตำแหน่งของโพรโตคอล 35

3.4 แสดงขั้นตอนการสร้างลิงค์ 36

3.5 แสดงขั้นตอนการตั้งค่า 37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.6 แสดงขั้นตอนการแปลงเครือข่าย	38
3.7 ตัวอย่างไฟล์เอ็นเอเอ็ม	39
3.8 แสดงกระบวนการทำงานของส่วนการแสดงผล	40
3.9 ส่วนการสร้างเครือข่ายภายในไฟล์เอ็นเอเอ็ม	40
3.10 ส่วนการทำงานของกลุ่มข้อมูลภายในเครือข่ายจำลอง	41
3.11 แสดงการรับและส่งไฟล์ระหว่างไคลเอนต์และเซิร์ฟเวอร์	42
3.12 แสดงขั้นตอนการทำงานของฝั่งเซิร์ฟเวอร์	42
3.13 กระบวนการเรียกการทำงานฟรอนต์เอ็นโดยใช้งานจาวาเว็บสตาร์ท	43
3.14 ไฟล์เจเอ็นแอลพี	44
3.15 กระบวนการโดยรวมของการเรียกใช้งานฟรอนต์เอ็นผ่านจาวาเว็บสตาร์ท	45
4.1 การเรียกใช้งานเว็บไซต์ของโปรแกรมฟรอนต์เอ็น	46
4.2 การดาวน์โหลดเจเอ็นแอลพี	47
4.3 การเรียกใช้งานจาวาเว็บสตาร์ท	47
4.4 การแจ้งเตือนก่อนแอปพลิเคชั่นจะทำงาน	48
4.5 หน้าต่างของโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็น	48
4.6 การสร้างโหนดและการเชื่อมต่อโหนดด้วยลิงค์	49
4.7 การสร้างโพรโตคอลและการเชื่อมต่อเข้าด้วยกัน	49
4.8 การสร้างแอปพลิเคชั่นในเครือข่ายจำลอง	50
4.9 การกำหนดตารางเหตุการณ์ของเครือข่ายจำลอง	50
4.10 การแปลงเครือข่ายที่จำลองเป็นภาษาทีซีแอล	51
4.11 แสดงการทำงานของเซิร์ฟเวอร์ซึ่งคอยให้บริการ	52
4.12 ผลลัพธ์ที่ได้จากการประมวลผลไฟล์ภาษาทีซีแอล	52
4.13 การแสดงผลภาพเคลื่อนไหวด้วยโปรแกรมเน็ตเวิร์คอนิเมเตอร์	53
4.14 การทดลองการใช้งานโพรโตคอลทีซีพี	54
4.15 การกำหนดการทำงานของเครือข่ายโพรโตคอลทีซีพี	54
4.16 แสดงผลการทำงานของโพรโตคอลทีซีพีในส่วนสร้างการเชื่อมต่อ	55
4.17 แสดงผลการทำงานของโพรโตคอลทีซีพีในส่วนการรับส่งข้อมูล	55
4.18 การทดลองการใช้งานโพรโตคอลยูดีพี	55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.19 การกำหนดการทำงานของเครือข่ายโพรโตคอลยูดีพี 56

4.20 แสดงผลการทำงานของโพรโตคอลยูดีพี 56

4.21 เครือข่ายสำหรับทดลองการกำหนดค่าลิงค์คอส 57

4.22 การกำหนดการทำงานของเครือข่ายทดลองค่าลิงค์คอส 57

4.23 การเลือกเดินทางของกลุ่มข้อมูลในการกำหนดค่าลิงค์คอส 57

4.24 การเปลี่ยนแปลงเส้นทางการรับส่งข้อมูลเมื่อมีการเปลี่ยนแปลงค่าลิงค์คอส 58

4.25 โพรโตคอลเลือกเส้นทางในโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็น 58

4.26 แสดงการแลกเปลี่ยนเส้นทางของโพรโตคอลเลือกเส้นทาง 59

4.27 แสดงการรับส่งกลุ่มข้อมูลกรณีมีเส้นทางมากกว่าหนึ่งเส้นทางไปยังปลายทาง 59

ก.1 ไฟล์บีบอัดของโปรแกรมเน็ตเวิร์คซิมูเลชั่น 63

ก.2 แสดงการแจ้งเตือนทางเทอร์มินอล 64

ก.3 การแจ้งเตือนการกำหนดพารามิเตอร์ของโปรแกรมเน็ตเวิร์คซิมูเลชั่น 64

ก.4 แสดงการทดสอบของโปรแกรมเน็ตเวิร์คซิมูเลชั่น 65

ก.5 การกำหนดพารามิเตอร์ของโปรแกรมเน็ตเวิร์คซิมูเลชั่น 65

ก.6 แสดงการเรียกใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่นจากเทอร์มินอล 65

ก.7 การปรับแต่งการติดตั้งโปรแกรมคอมไพล์และใช้งานภาษาจาวา 66

ก.8 หน้าต่างยอมรับการใช้งานซอฟต์แวร์ 66

ก.9 การทดสอบการติดตั้งภาษาจาวา 67

ก.10 การทำเครื่องหมายโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็น 68

ก.11 ผลลัพธ์ที่ได้จากการทำเครื่องหมายไฟล์โปรแกรม 69

ก.12 แสดงสิทธิในการเข้าถึงไฟล์โปรแกรมเน็ตเวิร์คซิมูเลชั่น 70

ข.1 ตัวอย่างการสร้างคลาสในภาษาโอทีซีแอล 74

ง.1 แสดงการเรียกเว็บเพจของเซิร์ฟเวอร์ 83

ง.2 แสดงการเรียกใช้งานโปรแกรมฟรอนต์เอ็น 84

ง.3 ข้อความขออนุญาตใช้งานทรัพยากร 84

ง.4 โปรแกรมฟรอนต์เอ็น 85

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มาของโครงงาน

ในปัจจุบันเครือข่ายของคอมพิวเตอร์มีอยู่ที่หนแห่งตั้งแต่ขนาดเล็กเช่นเครือข่ายในบ้านไปจนขนาดใหญ่ เช่น เครือข่ายคอมพิวเตอร์ในบริษัทหรือองค์กรขนาดใหญ่ การทำความเข้าใจเกี่ยวกับเครือข่ายในปัจจุบันนั้นมีโปรแกรมการจำลองเครือข่ายเกิดขึ้นมากมาย ส่วนใหญ่จะเน้นไปทางจำลองเพื่อทำความเข้าใจในการเชื่อมต่อต่าง ๆ มากกว่าแบบที่พิจารณาด้านการจราจรของข้อมูล ในปัจจุบันโปรแกรมจำพวกการจำลองการวิ่งของกลุ่มข้อมูลยังมีไม่มาก ทำให้ผู้ที่ต้องการศึกษาหรือวิจัยทฤษฎีต่าง ๆ ทางด้านเครือข่ายหรือโปรโตคอลต่าง ๆ ทำได้ไม่สะดวกนัก โปรแกรมเน็ตเวิร์คซิมูเลชั่นเป็นหนึ่งในโปรแกรมที่จำลองเครือข่ายในลักษณะเพื่อการจำลองเครือข่ายเพื่อดูการจราจรของข้อมูลที่วิ่งเข้าออกหรือการเข้าคิวของข้อมูล โดยโปรแกรมเน็ตเวิร์คซิมูเลชั่นถือว่ามีความสามารถสูงสำหรับการจำลองโหนดในเครือข่าย โพรโตคอลต่าง ๆ อย่างโพรโตคอลทีซีพี ยูดีพี และแอปพลิเคชั่นหลาย ๆ ชนิดที่ใช้งานในปัจจุบัน รวมไปถึงเครือข่ายไร้สาย แต่โปรแกรมนี้มีจุดอ่อนตรงที่การใช้งาน โปรแกรมต้องกระทำผ่านโปรแกรมแปลคำสั่งภาษาโอทีซีแอล ซึ่งไม่สะดวกในการใช้งาน ผู้ใช้จำเป็นต้องทราบรูปแบบของคำสั่งเฉพาะของภาษาโอทีซีแอล และการติดตั้งโปรแกรมมีขั้นตอนที่ยุ่งยากซับซ้อน โครงงานนี้จึงมีเป้าหมายในการพัฒนาส่วนของฟรอนต์เอ็นซึ่งเป็นหน้าต่างผู้ใช้งานของโปรแกรมเพื่อสะดวกต่อการใช้งาน แต่ผู้ใช้งานจำเป็นต้องมีความรู้ความเข้าใจในตัวโปรแกรมเน็ตเวิร์คซิมูเลชั่นมาก่อนเนื่องจากมีการอ้างอิงตัวแปรโครงสร้างและชื่อต่าง ๆ มาจากโปรแกรมเน็ตเวิร์คซิมูเลชั่น รวมทั้งจำเป็นต้องมีความรู้เกี่ยวกับโพรโตคอลพื้นฐานในเครือข่ายคอมพิวเตอร์อีกด้วย นอกจากใช้งานได้ง่ายแล้วการใช้งานไม่จำเป็นต้องติดตั้งโปรแกรมเน็ตเวิร์คซิมูเลชั่นลงบนเครื่องที่ใช้งาน เพียงแต่จัดตั้งเซิร์ฟเวอร์เพื่อใช้งานทำให้ผู้ใช้สามารถใช้งานได้อย่างง่ายและรวดเร็ว อีกทั้งเป็นประโยชน์ต่อการศึกษาค้นคว้าโดยสามารถใช้ในการทดลองหรือเป็นสื่อการเรียนการสอนทางด้านเครือข่ายคอมพิวเตอร์ ซึ่งเป็นการประหยัดงบประมาณในการใช้งานอุปกรณ์จริง

1.2 วัตถุประสงค์ของโครงงาน

1. พัฒนาส่วนฟรอนต์เอ็นเพื่อให้ใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่นง่ายขึ้น
2. เพื่อลดขั้นตอนติดตั้งโปรแกรมเน็ตเวิร์คซิมูเลชั่นโดยมีการติดตั้งที่ง่ายและสะดวก
3. เพื่อศึกษาและทดลองทฤษฎีต่างๆ ทางด้านเครือข่ายคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้รับความรู้ความเข้าใจเกี่ยวกับการเขียนโปรแกรมเชิงวัตถุ
2. ได้รับความรู้ความเข้าใจเกี่ยวกับโพรโตคอลต่าง ๆ ของเครือข่ายคอมพิวเตอร์
3. ได้รับความรู้ความเข้าใจเกี่ยวกับการติดต่อสื่อสารระหว่างไคลเอนต์กับเซิร์ฟเวอร์
4. เพื่อให้มีผู้สนใจศึกษาหรือวิจัยโพรโตคอลต่าง ๆ ของเครือข่ายคอมพิวเตอร์มากขึ้น
5. เพื่อลดเวลาในการศึกษาวิจัยของผู้ที่ใช้งานหรือเริ่มต้นใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่น

1.4 ขอบเขตของโครงงาน

1. โปรแกรมสามารถทำงานโดยเป็นเว็บแอปพลิเคชั่น
2. โปรแกรมสามารถใช้งานได้โดยไม่ขึ้นกับระบบปฏิบัติการ
3. โปรแกรมสามารถจำลองเฉพาะเครือข่ายแบบต่อสายเท่านั้น
4. โปรแกรมไม่สามารถตรวจสอบความผิดพลาดทางลอจิกในการจำลองเครือข่ายที่สร้างขึ้นจากผู้ใช้อย่างการตั้งค่าไม่ถูกต้องหรือ การสร้างการเชื่อมต่อไม่ครบ
5. ผู้ใช้งานควรมีความรู้ความเข้าใจในตัวโปรแกรมเน็ตเวิร์คซิมูเลชั่นก่อนจะใช้งานในส่วนของฟรอนต์เอ็น เนื่องจากชื่อตัวแปรหรือโครงสร้างต่าง ๆ มีการอ้างอิงโดยตรงมาจากโปรแกรมเน็ตเวิร์คซิมูเลชั่นและควรมีความรู้เกี่ยวกับทีซีพีไอพีพื้นฐาน
6. เนื่องจากฟังก์ชันต่าง ๆ ของโปรแกรมเน็ตเวิร์คซิมูเลชั่นมีจำนวนมาก จึงรองรับเพียงบางส่วนดังนี้ Node, Queue, UDP, NULL, TCP Tahoe, TCP Vegas, TCP Reno, TCP New Reno, TCP Sack1, TCP Fack, TCP Full, TCP Sink, TCP Sink Del Ack, TCP Sink Sack1,TCP Sink Sack1 Del Ack ,Web Server, Web Client, CBR, Telnet, FTP, Exponential, Pareto

1.5 ขั้นตอนการดำเนินงาน

1. กำหนดขอบเขตของโครงงาน
2. วิเคราะห์หาทฤษฎีที่เกี่ยวข้องกับโครงงานทั้งหมด
3. วางแผนการทำโครงงาน
4. ศึกษาโครงสร้างและรูปแบบของภาษาทีซีแอลและโอทีซีแอลขั้นพื้นฐาน
5. ศึกษาโครงสร้างและวิธีการใช้งานฟังก์ชันต่างๆ ของโปรแกรมเน็ตเวิร์คซิมูเลชั่น
6. ศึกษาการใช้งานจาวาเพื่อใช้ในการสร้างหน้าต่างส่วนผู้ใช้งาน
7. ออกแบบการสร้างหน้าต่างส่วนผู้ใช้งานของตัวโปรแกรม
8. พัฒนาส่วนฟรอนต์เอ็นของโปรแกรมเน็ตเวิร์คซิมูเลชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. พัฒนาส่วนการติดต่อระหว่างฟรอนต์เอ็นและเครื่องเซิร์ฟเวอร์
10. ตรวจสอบและแก้ไขข้อผิดพลาด

1.6 เนื้อหาของรายงาน

1. บทที่หนึ่ง กล่าวถึงบทนำ ความเป็นมาของโครงงาน วัตถุประสงค์ ประโยชน์ที่คาดว่าจะได้รับ ขอบเขตของโครงงาน และขั้นตอนการดำเนินงาน
2. บทที่สอง อธิบายถึงทฤษฎีที่เกี่ยวข้องกับโครงงานทั้งหมด
3. บทที่สาม อธิบายถึงวิธีการออกแบบและพัฒนาโปรแกรม
4. บทที่สี่ อธิบายผลการทดลองและทดสอบการทำงานของโปรแกรม
5. บทที่ห้าเป็นบทสรุปผลการพัฒนาโครงงานทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 โปรแกรมเน็ตเวิร์คซิมูเลชั่น (Network Simulation version 2 หรือ NS2)

เครือข่ายคอมพิวเตอร์มีการพัฒนาตลอดตั้งแต่อดีตจนถึงปัจจุบัน ซึ่งในการพัฒนาจะต้องมีการวางโครงสร้างและระบบผ่านอุปกรณ์ต่าง ๆ ที่ใช้ในการเชื่อมต่อเครือข่ายหลากหลายชนิด แต่เนื่องจากการวางระบบเครือข่ายจะต้องใช้งบประมาณและระยะเวลามาก จึงไม่คุ้มที่จะทดสอบการทำงานของเครือข่ายโดยการทดสอบผ่านอุปกรณ์จริง ทำให้เกิดแนวคิดในการจำลองการทำงานของระบบเครือข่าย โดยพัฒนาโปรแกรมจำลองการทำงาน (Simulator) เพื่อศึกษาและทดสอบการทำงานของเครือข่าย

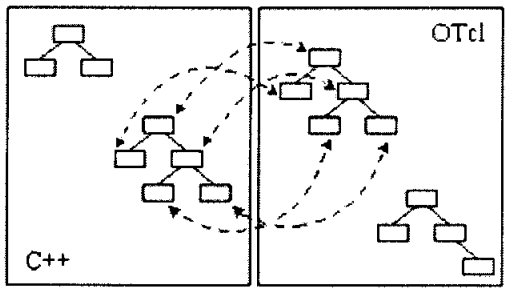
โปรแกรมเน็ตเวิร์คซิมูเลชั่น คือ โปรแกรมซึ่งพัฒนาเพื่อจำลองการทำงานของเครือข่ายในลักษณะเชิงวัตถุ (Object Oriented Simulator) พัฒนาโดยมหาวิทยาลัยเบิร์กเลย์ (U.C. Berkeley) เพื่อให้สามารถจำลองการทำงานได้หลายชนิดยกตัวอย่าง

- เครือข่ายไอพี (IP network)
- โพรโตคอล (Protocol) ในชั้นทรานสปอร์ตอย่าง ทีซีพี (TCP) และยูดีพี (UDP)
- ทราฟฟิคซอร์ด (Traffic Source) อย่าง เอฟทีพี (FTP) , เทลเน็ต (Telnet) , ซีบีอาร์ (CBR)
- การจัดการคิว (Queue) อย่าง ดรอปเทล (Drop Tail) , อาร์อีดี (RED)

โปรแกรมเน็ตเวิร์คซิมูเลชั่นเป็นส่วนหนึ่งของโปรเจค VINT (Virtual Inter Network Testbed) ซึ่งพัฒนาเครื่องมือสำหรับการจำลองการทำงานของเน็ตเวิร์ค โปรแกรมเน็ตเวิร์คซิมูเลชั่นถูกพัฒนาด้วยภาษาซีพลัสพลัส (C++) และใช้งานโปรแกรมแปลคำสั่งเป็นภาษาโอทีซีแอล (OTcl interpreter) ซึ่งเป็นภาษาทีซีแอลที่มีความสามารถในการโปรแกรมเชิงวัตถุ พัฒนาโดยสถาบันเทคโนโลยีแห่งแมสซาชูเซ็ท การใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่นผู้ใช้ควรมีความรู้ในภาษาโอทีซีแอลและความรู้ในด้านเน็ตเวิร์คควบคู่กัน

โปรแกรมเน็ตเวิร์คซิมูเลชั่นสนับสนุนลำดับชั้นคลาส (class hierarchy) ในภาษาซีพลัสพลัสและลำดับชั้นคลาสที่สัมพันธ์กันในโปรแกรมแปลคำสั่งโอทีซีแอล ลำดับชั้นของทั้งสองภาษามีความสัมพันธ์ที่ใกล้เคียงต่อกัน โดยจะมีความสัมพันธ์กันแบบ 1 ต่อ 1 ระหว่างลำดับชั้นคลาสของภาษาซีพลัสพลัสและลำดับชั้นคลาสของภาษาโอทีซีแอลดังรูปที่ 2.1 เมื่อผู้ใช้สร้างอ็อบเจกต์ของการจำลองผ่านโปรแกรมแปลคำสั่ง อ็อบเจกต์ของภาษาโอทีซีแอลจะถูกสร้างทันทีภายในโปรแกรมแปลคำสั่ง และจะสร้างอ็อบเจกต์ที่ตรงกันในลำดับชั้นคลาสในภาษาซีพลัสพลัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

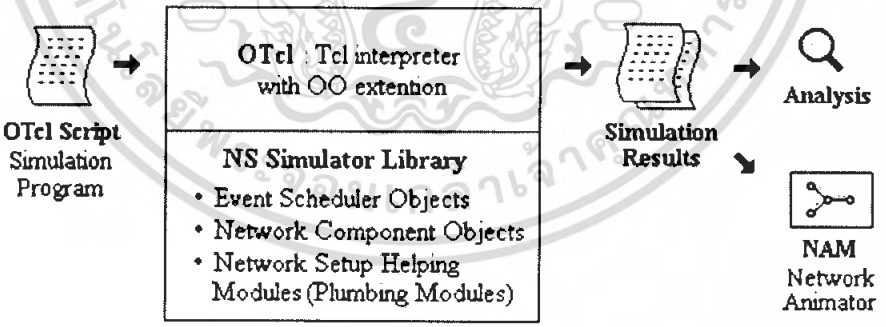


รูปที่ 2.1 แสดงความสัมพันธ์ระหว่างอ็อบเจกต์ของภาษาโอทีซีแอลและภาษาซีพลัสพลัส

เหตุผลที่โปรแกรมเน็ตเวิร์คซิมูเลชั่นพัฒนาโดยใช้งานสองภาษา ทั้งภาษาซีพลัสพลัสและภาษาโอทีซีแอลร่วมกัน เนื่องจากโปรแกรมจำลองต้องการการทำงาน 2 อย่างที่แตกต่างกัน

1. รายละเอียดในส่วนของโพรโตคอลที่จำลอง ซึ่งต้องการภาษาที่มีประสิทธิภาพเพื่อสามารถจัดการกับข้อมูลที่เป็นไบต์ หรือแพ็คเกตเฮดเดอร์ และสร้างขั้นตอนวิธี (algorithm) ที่สามารถประมวลผลข้อมูลขนาดใหญ่ได้ สำหรับงานเหล่านี้เวลาที่ใช้ในการทำงานเป็นสิ่งสำคัญส่วนในเรื่องของเทิร์นอราวด์ไทม์ (turn-around time) หรือเวลาที่ใช้ในการเริ่มต้นการจำลอง หาข้อผิดพลาด แก้ข้อผิดพลาด คอมไพล์และเริ่มทำงานใหม่จะมีความสำคัญน้อยกว่า

2. ในเครือข่ายจะมีตัวแปรเกี่ยวข้องหลายอย่าง ทั้งการตั้งค่า การเปลี่ยนแปลงค่าต่างๆ หรือ การดูข้อมูลจำนวนมาก ซึ่งต้องการความรวดเร็ว ในกรณีนี้ไอเทอเรชั่นไทม์ (iteration time) หรือเวลาเปลี่ยนรูปแบบที่จำลองและเริ่มต้นโปรแกรมใหม่ จะสำคัญมากกว่าเมื่อเทียบกับการตั้งค่าตั้งแต่เริ่มต้นการทำงานของโปรแกรม



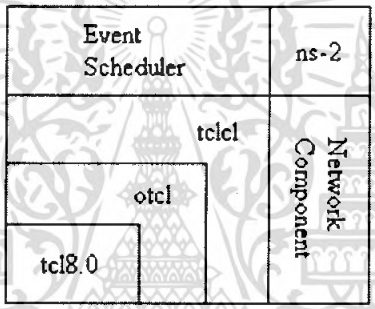
รูปที่ 2.2 แสดงกระบวนการทำงานของโปรแกรมเน็ตเวิร์คซิมูเลชั่นโดยภาพรวม

โปรแกรมเน็ตเวิร์คซิมูเลชั่นจึงเลือกใช้งานภาษาสองภาษาคือ ภาษาซีพลัสพลัสและภาษาโอทีซีแอล โดยภาษาซีพลัสพลัสมีคุณสมบัติ คือ ทำงานได้รวดเร็วแต่แก้ไขและเปลี่ยนแปลงการทำงานได้ช้า ทำให้ภาษานี้เหมาะกับการดูแลรายละเอียดของโพรโตคอลหรือทำงานในแบ็กเอ็นด์ (backend) ส่วนภาษาโอทีซีแอลทำงานช้าแต่สามารถเปลี่ยนแปลงการทำงานได้เร็วและเป็นแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เชิงโต้ตอบ (interactive) ทำให้เหมาะจะใช้สำหรับการตั้งค่าหรือเปลี่ยนแปลงค่าต่างในการจำลอง โดยโปรแกรมเน็ตเวิร์คซิมูเลชั่นมีกระบวนการเชื่อมโยงให้อ็อบเจกต์และตัวแปรปรากฏบนทั้งสองภาษาตรงกัน

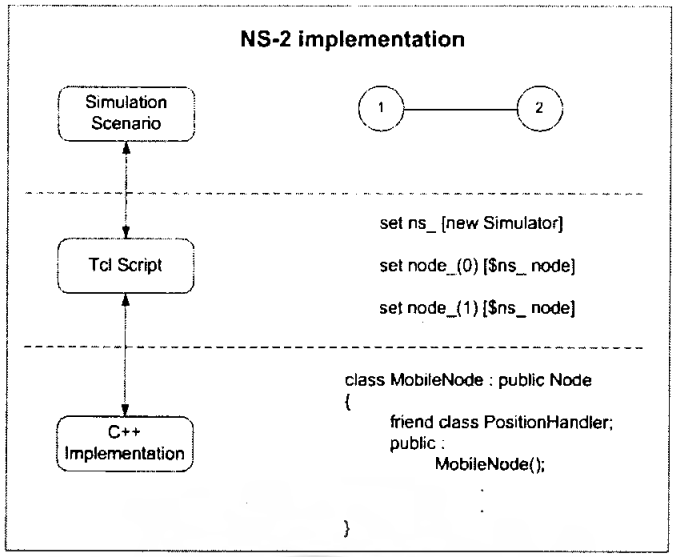
สำหรับการกำหนดค่าต่าง ๆ และการติดตั้งจะใช้งานภาษาโอทีซีแอล ส่วนภาษาซีพลัสพลัสจะใช้ในงานที่เกี่ยวข้องกับการไหลของข้อมูล และใช้เมื่อต้องการเปลี่ยนแปลงซอร์สโค๊ด (Source Code) เดิมที่มีอยู่เพื่อให้รองรับการทำงานที่ต้องการ ตัวอย่างเช่น ลิงค์ ซึ่งเป็นอ็อบเจกต์ของภาษาโอทีซีแอลจะเกี่ยวข้องกับเวลาที่สูญเสียในการประกอบข้อมูลกลับ , การเข้าคิว และการสูญหายของข้อมูล แต่ถ้าต้องการทำสิ่งที่นอกเหนือจากนั้น อย่างการเข้าคิวแบบพิเศษ หรือการจำลองการสูญเสียของข้อมูล จำเป็นต้องมีการสร้างอ็อบเจกต์ของภาษาซีพลัสพลัสหรือถ้าต้องการสร้างโพรโตคอลชนิดใหม่เพิ่มเติม จะต้องสร้างอ็อบเจกต์ของภาษาซีพลัสพลัสเพื่อประกาศสร้างโพรโตคอลจำลองชนิดใหม่



รูปที่ 2.3 มุมมองแบบสถาปัตยกรรมของโปรแกรมเน็ตเวิร์คซิมูเลชั่น

ผู้ใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่นทั่วไปจะใช้งานภาษาโอทีซีแอลในการสร้างเครือข่าย โดยเขียนเป็นชุดคำสั่งในแต่ละการจำลอง จากนั้นจะนำชุดคำสั่งภาษาโอทีซีแอลที่ได้ประมวลผลกับโปรแกรมเน็ตเวิร์คซิมูเลชั่น โดยโปรแกรมเน็ตเวิร์คซิมูเลชั่นจะมีการเชื่อมโยงภาษาโอทีซีแอลเข้ากับภาษาซีพลัสพลัสเพื่อการประมวลผลดังรูปที่2.4และจะได้ไฟล์ผลลัพธ์ของเครือข่ายตามชุดคำสั่งที่ได้จำลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 แสดงกระบวนการประมวลผลของโปรแกรมเน็ตเวิร์คซิมูเลชั่น

ผลลัพธ์จากการประมวลผลของโปรแกรมเน็ตเวิร์คซิมูเลชั่นมี 2 รูปแบบคือ

1. ไฟล์ทีอาร์หรือไฟล์เทรซ (tr หรือ trace file) ภายในไฟล์จะเก็บข้อมูลการทำงานของเครือข่ายที่ได้จำลอง โดยจะแสดงการไหลของกลุ่มข้อมูลที่มีการรับส่งภายในเครือข่าย และสามารถนำเอาไฟล์ทีอาร์ไปใช้ในการวิเคราะห์ประสิทธิภาพของเครือข่ายที่ได้จำลอง

```
+ 0.1 0 3 cbr 40 ------- 0 0.0 4.1 0 0
- 0.1 0 3 cbr 40 ------- 0 0.0 4.1 0 0
r 0.11016 0 3 cbr 40 ------- 0 0.0 4.1 0 0
+ 0.11016 3 4 cbr 40 ------- 0 0.0 4.1 0 0
- 0.11016 3 4 cbr 40 ------- 0 0.0 4.1 0 0
r 0.12032 3 4 cbr 40 ------- 0 0.0 4.1 0 0
+ 0.12032 4 3 ack 40 ------- 1 4.1 0.0 0 1
- 0.12032 4 3 ack 40 ------- 1 4.1 0.0 0 1
r 0.13048 4 3 ack 40 ------- 1 4.1 0.0 0 1
```

รูปที่ 2.5 แสดงตัวอย่างไฟล์ทีอาร์

2. ไฟล์เอ็นเอเอ็มหรือไฟล์เน็ตเวิร์คอนิเมเตอร์ (NAM file) เป็นผลลัพธ์อีกชนิดหนึ่งของโปรแกรมเน็ตเวิร์คซิมูเลชั่น ซึ่งลักษณะจะคล้ายกับไฟล์ทีอาร์ แต่จะมีข้อมูลมากกว่าเนื่องจากสามารถนำไปแสดงการทำงานในลักษณะภาพเคลื่อนไหวได้ โดยปกติไฟล์เอ็นเอเอ็มจะแสดงภาพเคลื่อนไหวโดยใช้งานร่วมกับโปรแกรมเน็ตเวิร์คอนิเมเตอร์(NetworkAnimator)ซึ่งภาพเคลื่อนไหวที่ได้จะแสดงถึงการเดินทางของกลุ่มข้อมูลภายในเครือข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
+ -t 0.5 -s 0 -d 3 -p cbr -e 40 -c 0 -i 0 -a 0 -x {0.0 4.1 0 ------- null}
- -t 0.5 -s 0 -d 3 -p cbr -e 40 -c 0 -i 0 -a 0 -x {0.0 4.1 0 ------- null}
h -t 0.5 -s 0 -d 3 -p cbr -e 40 -c 0 -i 0 -a 0 -x {0.0 4.1 -1 ------- null}
r -t 0.51016 -s 0 -d 3 -p cbr -e 40 -c 0 -i 0 -a 0 -x {0.0 4.1 0 ------- null}
+ -t 0.51016 -s 3 -d 4 -p cbr -e 40 -c 0 -i 0 -a 0 -x {0.0 4.1 0 ------- null}
- -t 0.51016 -s 3 -d 4 -p cbr -e 40 -c 0 -i 0 -a 0 -x {0.0 4.1 0 ------- null}
h -t 0.51016 -s 3 -d 4 -p cbr -e 40 -c 0 -i 0 -a 0 -x {0.0 4.1 -1 ------- null}
r -t 0.52032 -s 3 -d 4 -p cbr -e 40 -c 0 -i 0 -a 0 -x {0.0 4.1 0 ------- null}
```

รูปที่ 2.6 แสดงตัวอย่างไฟล์เอ็นเอเอ็ม

การใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่น ผู้ใช้งานควรจะมีความรู้ทางด้านเครือข่ายคอมพิวเตอร์ อีกทั้งควรมีความรู้ของภาษาโอทีซีแอล เพื่อที่จะช่วยในการสร้างเครือข่ายจำลอง และสามารถใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่นได้อย่างมีประสิทธิภาพ

2.1.1 โครงสร้างคำสั่งของโปรแกรมเน็ตเวิร์คซิมูเลชั่น (NS Script Structure)

หลักการสร้างไฟล์ภาษาทีซีแอลเพื่อจำลองการทำงานของเครือข่าย

1. กำหนดเริ่มต้นการจำลองและกำหนดจุดสิ้นสุดการทำงาน โปรแกรมเน็ตเวิร์คซิมูเลชั่นจะเริ่มต้นการจำลองด้วยคำสั่ง

```
set ns [new Simulator]
```

คำสั่งนี้เป็นการสร้างอ็อบเจกต์ของคลาส Simulator ซึ่งในที่นี้จะใช้ชื่อตัวแปรเป็น ns และถ้าต้องการไฟล์ผลลัพธ์ออกมาเพื่อแสดงผล ต้องใช้คำสั่ง

```
#Open the NAM trace file
set namfile [open out.nam w]
$ns namtrace-all $namfile
```

คำสั่ง open คือ การเปิดไฟล์ออกมา จากนั้นใช้คำสั่ง namtrace-all เพื่อเขียนผลลัพธ์ออกทางไฟล์ที่ชื่อ out.nam โดยไฟล์เอ็นเอเอ็ม (nam)จะเก็บผลลัพธ์ของการจำลองการทำงานเพื่อไปใช้ในการแสดงผลในรูปแบบของกราฟฟิก

w คือ พารามิเตอร์สำหรับเปิดไฟล์เพื่อเขียนข้อมูลลงในไฟล์ที่ชื่อว่า out.nam และจุดสิ้นสุดของโปรแกรมคือการเรียกใช้งานกระบวนคำสั่งที่ชื่อ finish โดยสร้างดังนี้

```
#Define a 'finish' procedure
proc finish {} {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        global  ns  namfile
        $ns  flush-trace
        close  $namfile
        exec  nam  out.nam &
        exit  0
    }
```

กระบวนคำสั่ง finish จะเรียกใช้เพื่อจบการทำงานของการจำลอง โดยในกระบวนคำสั่งนี้ได้ทำการปิดไฟล์ที่ชี้โดยตัวแปรที่ชื่อ namfile คำสั่ง exec nam out.nam & เป็นการเรียกการแสดงผลของโปรแกรมเน็ตเวิร์คอนิเมเตอร์ ส่วนคำสั่ง exit 0 เป็นการออกจากโปรแกรม การเรียกกระบวนคำสั่ง finish สามารถเรียกการทำงานดังนี้

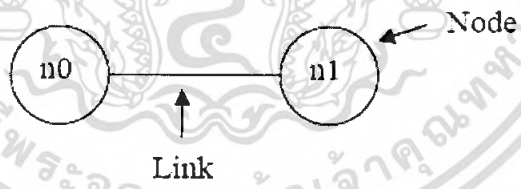
```
$ns  at  125.0  "finish"
```

จะเป็นการเรียกกระบวนคำสั่ง finish ที่เวลา 125 วินาที

คำสั่งที่จะเรียกการจำลองให้ทำงาน จะใช้คำสั่ง run ในคลาส Simulator ดังนี้

```
$ns  run
```

2. ประกาศสร้างลิงค์(link)และโหนด (node) ภายในเครือข่ายที่ต้องการจำลอง



รูปที่ 2.7 ลิงค์และโหนด

โหนดเปรียบเสมือนกับชั้นกายภาพ (physical layer) ของแบบอ้างอิงโอเอสไอ (OSI Model) ซึ่งการสร้างโหนดสามารถสร้างได้โดยประกาศดังนี้

```
set  n0  [$ns  node]
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งข้างต้นเป็นการประกาศโหนดโดยมีตัวแปรชื่อ n0 และหลังจากสร้างโหนดแล้ว ต้องสร้างลิงค์เชื่อมโยงระหว่างโหนดซึ่งลิงค์เปรียบเสมือนชั้นเชื่อมโยงข้อมูล (Data link layer) โดยการสร้างลิงค์ใช้คำสั่ง

```
$ns duplex-link $n0 $n2 10Mb 10ms DropTail
```

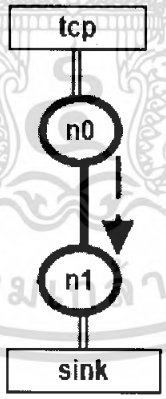
เป็นการเชื่อมโหนด 2 โหนดคือ n0 และ n2 เข้าด้วยกันโดยใช้ลิงค์ที่มีความเร็ว 10 เมกกะบิท (Mb) มีค่าดีเลย์ (delay) 10 ms และการเข้าคิวแบบดรอปเทล ซึ่งเป็นกระบวนการคิวชนิดหนึ่งที่มีในโปรแกรมเน็ตเวิร์คซิมูเลชั่น ขนาดของคิวสามารถกำหนดได้โดยใช้คำสั่ง

```
$ns queue-limit $n0 $n2 20
```

เป็นการกำหนดขนาดคิวของลิงค์ให้มีขนาด 20 (โดยปกติคิวจะมีขนาด 50)

3. การสร้างเอเจนต์ (agent) และแอปพลิเคชั่น (application)

หลังจากสร้างโครงข่ายของเน็ตเวิร์คแล้ว จะเป็นการกำหนดเอเจนต์และแอปพลิเคชั่นที่ใช้ในการจำลองเน็ตเวิร์ค เอเจนต์เปรียบเสมือนกับชั้นเคลื่อนย้ายข้อมูล (Transport layer) และแอปพลิเคชั่นเปรียบเสมือนชั้นแอปพลิเคชั่น (Application layer) จะยกตัวอย่างการสร้างเอฟทีพี (FTP) บนทีซีพีเอเจนต์ (TCP agent)



Create TCP

รูปที่ 2.8 การสร้างทีซีพีเอเจนต์

การสร้างทีซีพีเอเจนต์และเพิ่มเข้าไปยังโหนดที่ต้องการโดยใช้คำสั่ง

```
set tcp [new Agent/TCP]
$ns attach-agent $n0 $tcp
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งข้างต้นเป็นการสร้างเอเจนต์ให้กับโหนดต้นทาง (source node)
การสร้างเอเจนต์ให้กับโหนดปลายทาง สามารถสร้างได้โดยใช้คำสั่งดังนี้

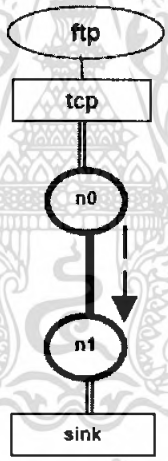
```
set sink [new Agent/TCPSink]
$ns attach-agent $n2 $sink
```

เมื่อสร้างโหนดและกำหนดเอเจนต์ที่โหนดแล้ว การเชื่อมต่อเอเจนต์เข้าด้วยกันสามารถใช้คำสั่งดังนี้

```
$ns connect $tcp $sink
```

จากนั้นกำหนดแอปพลิเคชั่นในโหนดที่มีเอเจนต์เพื่อสร้างทราฟฟิกซอร์ด ในที่นี้จะใช้งานเอฟทีพีบนทีซีพีเอเจนต์ ปกติแล้วทีซีพีจะไม่สร้างข้อมูลที่ส่งออกไป โดยสามารถใช้คำสั่ง

```
set ftp [new Application/FTP]
$ftp attach-agent $tcp
```



Create FTP

รูปที่ 2.9 การสร้างเอฟทีพี

4. การกำหนดเหตุการณ์ (Scheduling Event)

การกำหนดเหตุการณ์ต่าง ๆ ในโปรแกรมเน็ตเวิร์คซิมูเลชั่นสามารถใช้คำสั่ง

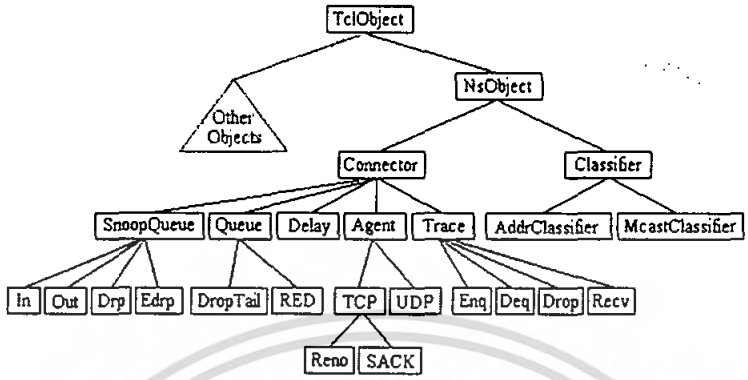
```
$ns at <time> <event>
```

โดยเหตุการณ์ต่าง ๆ ที่กำหนดไว้จะเริ่มการทำงานโดยคำสั่ง $ns run

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 ส่วนประกอบของโปรแกรมเน็ตเวิร์คซิมูเลชั่น (Network Components)

ส่วนประกอบของโปรแกรมเน็ตเวิร์คซิมูเลชั่น ส่วนใหญ่แล้วเป็นส่วนประกอบทางด้านเครือข่าย โดยสามารถแสดงได้ดังนี้



รูปที่ 2.10 แสดงลำดับของคลาสในโปรแกรมเน็ตเวิร์คซิมูเลชั่น

ส่วนบนสุดของลำดับต้นไม้ คือ คลาส TclObject ซึ่งเป็นซูเปอร์คลาส (superclass) หรือคลาสแม่ของทุก ๆ โอทีซีแอลอ็อบเจกต์ไลบรารี่ (OTcl library objects) ส่วนคลาส NsObject เป็นซูเปอร์คลาสของทุกคลาสที่เป็นส่วนประกอบของเครือข่ายพื้นฐานที่ดูแลจัดการกับกลุ่มข้อมูล ลิงค์และโหนด

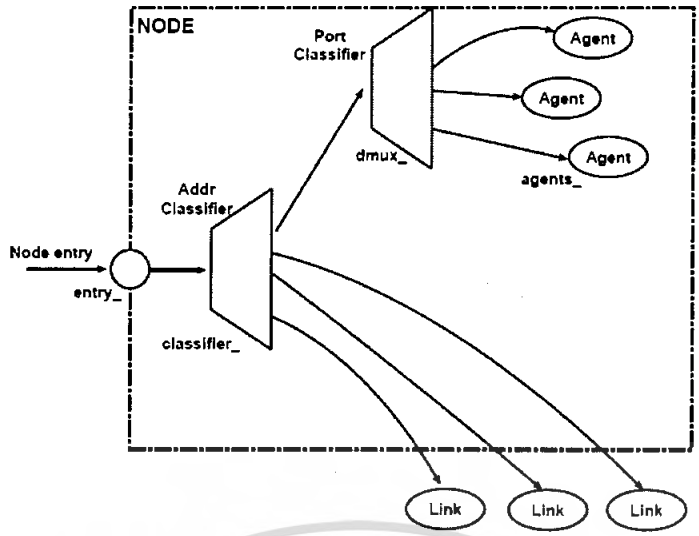
ส่วนของเครือข่ายขั้นพื้นฐานแบ่งออกเป็นสองคลาสย่อยคือ คลาส Connector และ คลาส Classifier ซึ่งขึ้นกับจำนวนของเส้นทางที่ข้อมูลวิ่ง ถ้ามีข้อมูลออกเพียงเส้นทางเดียวจะอยู่ภายใต้คลาส Connector แต่ถ้ามีข้อมูลออกได้หลายเส้นทาง ก็จะอยู่ภายใต้คลาส Classifier

2.1.3 โหนดและเร้าติ้ง (node and routing)

โปรแกรมเน็ตเวิร์คซิมูเลชั่น โหนด หมายถึง อุปกรณ์ในเครือข่าย โดยอาจจะเป็นได้ทั้งคอมพิวเตอร์ อุปกรณ์เลือกเส้นทาง หรืออุปกรณ์อื่น ๆ ซึ่งโปรแกรมเน็ตเวิร์คซิมูเลชั่นมีโหนดสองชนิด คือ โหนดชนิดยูนิคาสต์ และโหนดชนิดมัลติคาสต์

โหนดชนิดยูนิคาสต์ (unicast) จะมีแอดเดรสคลาสซิไฟเออร์ (address classifier) ชนิดยูนิคาสต์เร้าติ้ง (unicast routing) และพอร์ตคลาสซิไฟเออร์ (port classifier) ซึ่งโดยปกติแล้วเมื่อสร้างโหนดขึ้นมาจะเป็นชนิดยูนิคาสต์ โครงสร้างภายในของโหนดชนิดยูนิคาสต์จะเป็นดังรูปที่ 2.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

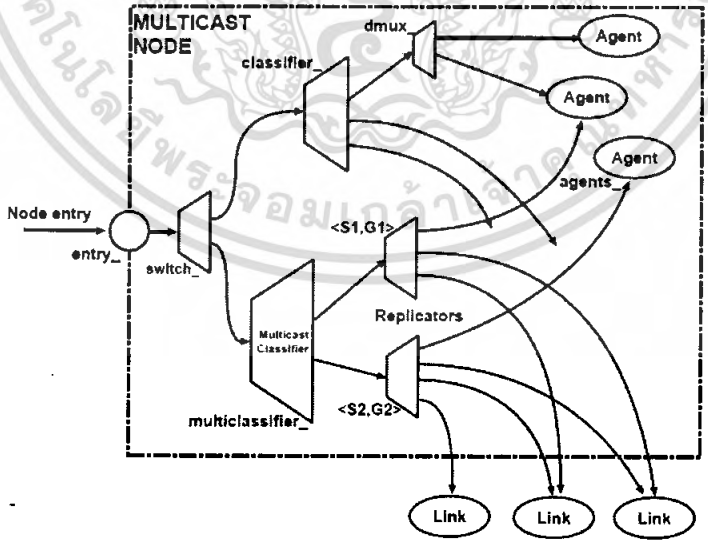


รูปที่ 2.11 โครงสร้างภายในของโหนดชนิดยูนิคาสต์

การสร้างโหนดชนิดยูนิคาสต์ในโปรแกรมเน็ตเวิร์คซิมูเลชั่น จะใช้คำสั่ง

```
set ns [new Simulator]
$ns node
```

ส่วนโหนดชนิดมัลติคาสต์ (multicast node) จะมีคลาสซิไฟเออร์ชนิดมัลติคาสต์เร้าติ้ง (multicast routing) ซึ่งจะมีโครงสร้างภายในดังรูปที่ 2.12



รูปที่ 2.12 โครงสร้างภายในของโหนดชนิดมัลติคาสต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

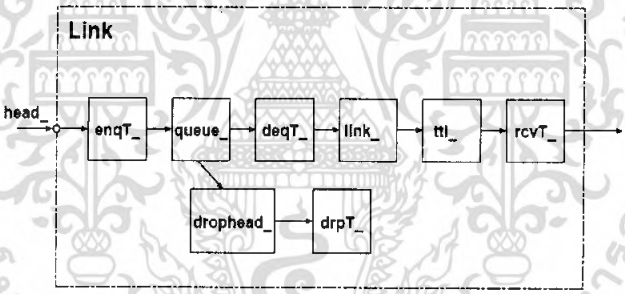
การใช้งานโหนดชนิดมัลติคาสต์จะต้องประกาศโดยตรงเมื่อสร้างไฟล์ภาษาโอทีซีแอลในส่วนประกาศสร้างโหนด โดยการสร้างโหนดแบบมัลติคาสต์จะมีรูปแบบดังนี้

```
set ns [new Simulator -multicast on]
```

เมื่อสร้างโหนดแล้วจะสามารถกำหนดข้อมูลต่าง ๆ ได้อย่าง โพรโตคอลเลือกเส้นทาง (routing protocol) การเพิ่มเอเจนต์อย่างทีซีพี ยูดีพี หรือการกำหนดแอดเดรสให้กับโหนด

2.1.4 ลิงค์ (Link)

ลิงค์ (link) เป็นส่วนประกอบในการเชื่อมโยงเครือข่ายคอมพิวเตอร์ โดยเป็นตัวกลางในการนำพากลุ่มข้อมูลจากที่หนึ่งไปยังอีกที่หนึ่ง ชนิดของลิงค์มีทั้งลิงค์สื่อสารทางเดียว (simplex link) และลิงค์สื่อสารสองทาง (duplex link) ซึ่งโปรแกรมเน็ตเวิร์คซิมูเลชั่นสามารถจำลองการทำงานของลิงค์ได้ทั้งแบบลิงค์สื่อสารทางเดียวหรือลิงค์สื่อสารสองทาง โดยเมื่อสร้างลิงค์สื่อสารสองทาง โปรแกรมเน็ตเวิร์คซิมูเลชั่นจะจัดการแปลงให้เป็นลิงค์สื่อสารทางเดียว 2 เส้นที่มีทิศทางตรงข้ามกัน โครงสร้างภายในของลิงค์สื่อสารทางเดียวมีลักษณะดังรูปที่ 2.13



รูปที่ 2.13 โครงสร้างภายในของลิงค์

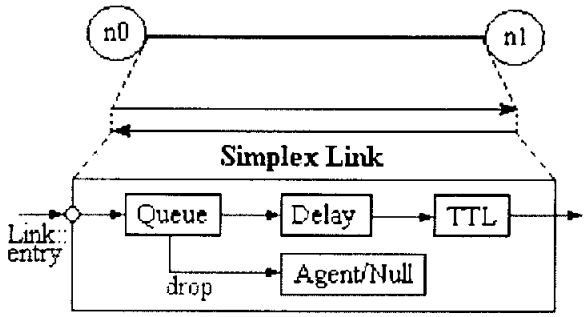
คำสั่งการสร้างลิงค์ของโปรแกรมเน็ตเวิร์คซิมูเลชั่น มีลักษณะดังนี้

```
set ns [new Simulator]
$ns simplex-link <source_node> <destination_node> <BW> <delay> <queue_type>
$ns duplex-link <source_node> <destination_node> <BW> <delay> <queue_type>
```

การเข้าคิวของกลุ่มข้อมูลจะจัดการในอ็อบเจกต์ของลิงค์สื่อสารทางเดียว ซึ่งถ้ากลุ่มข้อมูลถูกนำออกจากคิว (dequeue) ออกมา จะถูกส่งไปจัดการในออบเจ็กต์ดีเลย์ (delay object) ส่วนกลุ่มข้อมูลดรอป (drop packet) จะส่งไปยังเอเจนต์ว่าง (Null Agent) และทิ้งไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อ็อบเจกต์ทีทีแอล (TTL object) จะใช้คำนวนในแต่ละกลุ่มข้อมูลเมื่อผ่านแต่ละโหนดไป ซึ่งการทำงานภายในของลิงค์จะแสดงดังรูปที่ 2.14



รูปที่ 2.14 โครงสร้างการทำงานภายในของลิงค์

2.1.5 เอเจนต์โพรโตคอล (Protocol Agent)

โพรโตคอลในชั้นทรานสปอร์ตจะทำหน้าที่รองรับการทำงานของแอปพลิเคชั่น และส่งต่อไปยังแอปพลิเคชั่นปลายทางที่ถูกต้อง โดยในโปรแกรมเน็ตเวิร์คซิมูเลชั่นมีเอเจนต์ซึ่งรวมเอาโพรโตคอลหลาย ๆ ชนิดเข้าไว้ด้วยกัน (ตั้งแต่ชั้นทรานสปอร์ตขึ้นไป) อย่าง ทีซีพี ยูดีพี เป็นต้น ซึ่งจะมีดังรูปที่ 2.15

TCP/FullTcp	a more full-functioned TCP with 2-way traffic
TCP/Vegas	a "Vegas" TCP sender
TCP/Vegas/RBP	a Vegas TCP with "rate based pacing"
TCP/Vegas/RBP	a Reno TCP with "rate based pacing"
TCP/Asym	an experimental Tahoe TCP for asymmetric links
TCP/Reno/Asym	an experimental Reno TCP for asymmetric links
TCP/Newreno/Asym	an experimental Newreno TCP for asymmetric links
TCPSink	a Reno or Tahoe TCP receiver (not used for FullTcp)
TCPSink/DelAck	a TCP delayed-ACK receiver
TCPSink/Asym	an experimental TCP sink for asymmetric links
TCPSink/Sack1	a SACK TCP receiver
TCPSink/Sack1/DelAck	a delayed-ACK SACK TCP receiver
UDP	a basic UDP agent

รูปที่ 2.15 เอเจนต์บางส่วนในโปรแกรมเน็ตเวิร์คซิมูเลชั่น

2.1.5.1 เอเจนต์ยูดีพี (UDP Agent)

เอเจนต์ยูดีพีเป็นเอเจนต์ตัวหนึ่งในชั้นทรานสปอร์ต ซึ่งทำหน้าเอาข้อมูลที่รับต่อจากชั้นแอปพลิเคชั่นและแบ่งข้อมูลเป็นส่วนย่อย (ถ้าจำเป็น) เพื่อที่จะส่งออกไปยังปลายทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานเอเจนต์ยูดีพีต้องสร้างโหนดขึ้นมาก่อน จึงจะสามารถเพิ่มเอเจนต์เข้าไปในโหนดที่ต้องการได้ คำสั่งที่ใช้ในการสร้างมีดังนี้

```
set n0 [$ns node]
set udp [new Agent/UDP]
$ns attach-agent $n0 $udp
```

คำสั่งข้างต้นเป็นการเพิ่มยูดีพีเอเจนต์เข้าไปในโหนด n0

2.1.5.2 เอเจนต์ทีซีพี (TCP Agent)

ทีซีพีเอเจนต์เป็นเอเจนต์ชนิดหนึ่งในโปรแกรมเน็ตเวิร์คซิมูเลชั่น ซึ่งทีซีพีเอเจนต์สามารถแบ่งออกเป็นสองชนิดคือ

1. วันเวย์ทีซีพี (one-way TCP) ซึ่งแบ่งย่อยได้อีก 2 ชนิดคือ

เอเจนต์ทีซีพีผู้ส่ง (TCP sender agent) ซึ่งทำหน้าที่ในการรับข้อมูลจากชั้นแอปพลิเคชั่นและส่งต่อไปยังปลายทาง จะประกอบไปด้วย

Agent/TCP

Agent/TCP/Reno

Agent/TCP/Newreno

Agent/TCP/Sack1

Agent/TCP/Vegas

Agent/TCP/Fack

อีกชนิดหนึ่งคือ เอเจนต์ทีซีพีผู้รับ (TCP receiver (Sink)) ซึ่งจะทำหน้าที่รับข้อมูลเพื่อส่งต่อไปยังแอปพลิเคชั่น จะประกอบไปด้วย

Agent/TCPSink

Agent/TCPSink/DelAck

Agent/TCPSink/Sack1

Agent/TCPSink/Sack1/DelAck

2. ทูเวย์ทีซีพี (two-way TCP) ซึ่งสามารถทำหน้าที่ได้ทั้งรับ และส่งข้อมูล ในปัจจุบันโปรแกรมเน็ตเวิร์คซิมูเลชั่นมีอยู่เพียงชนิดเดียวคือ

Agent/TCP/FullTcp

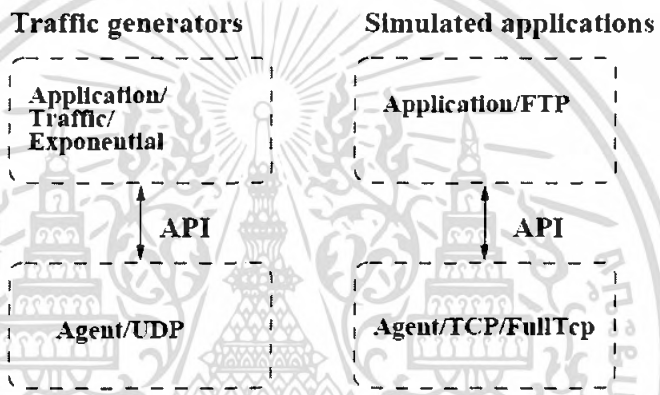
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานเอเจนต์ทีซีพีต้องสร้างโหนดก่อน จากนั้นค่อยเพิ่มเอเจนต์เข้าไปในโหนด ซึ่งตัวอย่างคำสั่งจะเป็นดังนี้

```
set tcp [new Agent/TCP]
$ns attach-agent $n0 $tcp
```

2.1.6 แอปพลิเคชั่น (Application)

แอปพลิเคชั่นจะอยู่ส่วนบนของทรานสปอร์ตเอเจนต์ (transport agent) ของโปรแกรมเน็ตเวิร์คซิมูเลชั่น ซึ่งจะแบ่งคลาสของแอปพลิเคชั่นออกเป็นสองกลุ่ม คือ ตัวสร้างทราฟฟิก (traffic generator) และตัวจำลองแอปพลิเคชั่น (simulated application) ซึ่งแสดงให้เห็นดังรูปที่ 2.16



รูปที่ 2.16 ตัวอย่างของแอปพลิเคชั่น

โปรแกรมเน็ตเวิร์คซิมูเลชั่นมีคลาสที่ทำหน้าที่ในชั้นแอปพลิเคชั่นอยู่หลายคลาส ซึ่งคลาส Application เป็นหนึ่งในหลาย ๆ คลาสแอปพลิเคชั่นที่มีอยู่ในโปรแกรม การใช้งานของคลาสแอปพลิเคชั่นจะคล้าย ๆ กัน โดยจะต้องสร้างโหนดและเอเจนต์ขึ้นมารองรับแอปพลิเคชั่นก่อน คำสั่งในการใช้งานมีดังนี้

```
set ftp [new Application/FTP]
$ftp attach-agent $tcp
```

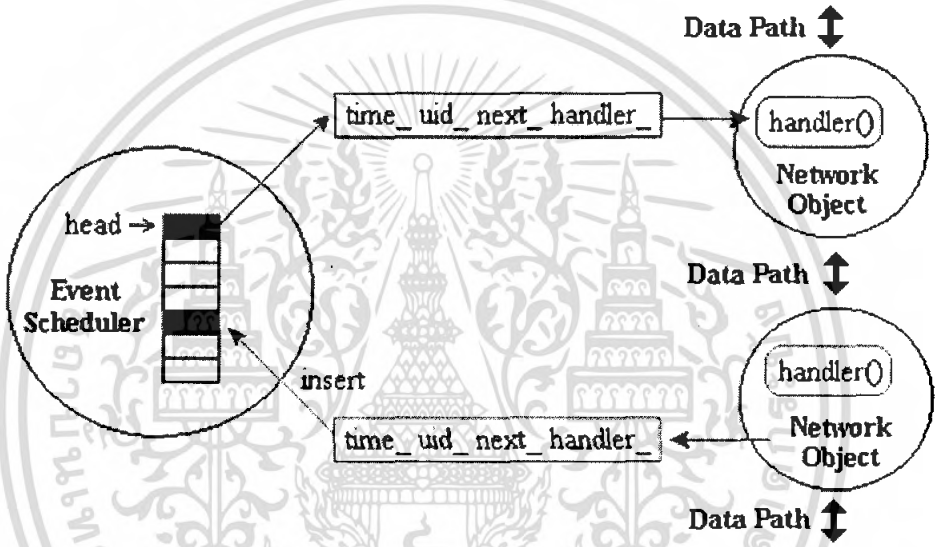
หรือจะใช้ในรูปแบบ

```
set ftp [$tcp attach-app FTP]
```

82039
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.7 ตารางเหตุการณ์ (Event Scheduler)

ตารางเหตุการณ์ในโปรแกรมเน็ตเวิร์คซิมูเลชั่นจะเป็นตัวกำหนดการทำงานของเครือข่ายทั้งหมด ซึ่งถ้าไม่มีการกำหนดตารางเหตุการณ์ จะไม่สามารถควบคุมการทำงานของการจำลองได้ โดยสิ่งที่ใช้งานตารางเหตุการณ์มากที่สุด คือ ส่วนประกอบของเครือข่ายซึ่งใช้ในการจำลองการส่งของกลุ่มข้อมูล หรือจัดการเวลา ดังแสดงในรูปที่ 2.17 แต่ละอ็อบเจกต์ของเครือข่ายมีการใช้งานตารางเหตุการณ์ เส้นทางของกลุ่มข้อมูลระหว่างแต่ละออบเจ็กต์ของเครือข่าย จะใช้งานแยกกับเส้นทางของเหตุการณ์ต่าง ๆ โดยกลุ่มข้อมูลจะถูกส่งจากอ็อบเจกต์ของเครือข่ายหนึ่งไปยังอีกอ็อบเจกต์หนึ่งโดยใช้งานเมทธอด send(Packet* p) ซึ่งเป็นเมทธอดของผู้ส่ง และเมทธอด recv(Packet*, Handler* h = 0) ซึ่งเป็นเมทธอดของผู้รับ



รูปที่ 2.17 แสดงการทำงานของตารางเหตุการณ์

โปรแกรมเน็ตเวิร์คซิมูเลชั่นมีตารางเหตุการณ์ที่แตกต่างกันอยู่ 2 ชนิด คือ

1. ตารางเวลาแบบเรียลไทม์ (real-time scheduler) เป็นการจำลองการทำงานของเน็ตเวิร์คสามารถที่จะติดต่อกับเน็ตเวิร์คของจริงได้ ซึ่งในปัจจุบันอยู่ระหว่างการพัฒนา

2. ตารางเวลาแบบนอนเรียลไทม์ (non-real-time scheduler) ในโปรแกรมเน็ตเวิร์คซิมูเลชั่น ตารางเวลาแบบนี้จะมีการทำงานให้เลือกอยู่ 3 รูปแบบ คือ ลิสต์ (List) , ฮีป (Heap) และคาเลนดาร์ (Calendar) ซึ่งโดยปกติจะใช้งานแบบคาเลนดาร์ การใช้งานตารางเวลาจะใช้เมื่อต้องการกำหนดการทำงานของเหตุการณ์ต่าง ๆ ที่เกิดขึ้นอย่าง กำหนดให้เอฟทีพีเริ่มทำงาน จบการทำงานของการจำลอง ในการสั่งให้ทำงานนั้น จะใช้คำสั่ง at time “string” โดยจะเรียกว่าแอตอีเวนท์ (AtEvent) ณ เวลาที่กำหนดไว้ ซึ่งคลาส AtEvent นี้เป็นคลาสลูกของคลาส Event ตัวอย่างการกำหนดตารางเวลาจะแสดงได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

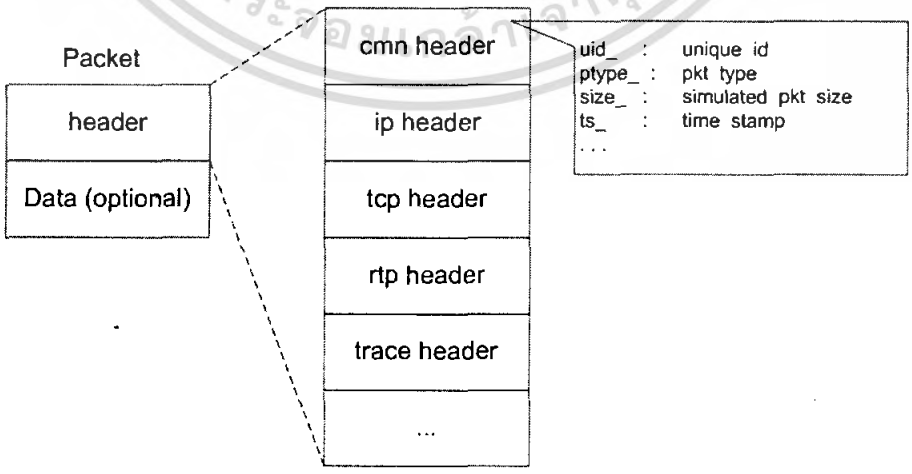
```
...
set ns [new Simulator]
$ns use-scheduler Heap
$ns at 300.5 "complete_sim"
...

proc complete_sim {} {
...
}
```

รูปที่ 2.18 ตัวอย่างของตารางเวลาเหตุการณ์

การกำหนดตารางเวลาเหตุการณ์จะต้องสร้างอ็อบเจกต์ของคลาส Simulator เพื่อใช้ในการกำหนด โดยในตัวอย่างจะเรียกกระบวนคำสั่งที่ชื่อ complete_sim ที่เวลา 300.5 วินาที ในส่วนที่เป็นสายอักขระนั้น อาจจะใส่คำสั่งอื่น ๆ เพื่อกำหนดการทำงานของการจำลองเครือข่ายก็ได้อย่างเช่น

```
$ns at 1.0 "$ftp start"
```

ในโปรแกรมเน็ตเวิร์คซิมูเลชั่นกลุ่มข้อมูลจะประกอบด้วยเฮดเดอร์และข้อมูลต่าง ๆ ซึ่งแพ็คเกตเฮดเดอร์ (packet header) เหล่านี้จะถูกกำหนดค่าเริ่มต้นเมื่ออ็อบเจกต์ของคลาส Simulator ถูกสร้างขึ้น ภายในเฮดเดอร์ของกลุ่มข้อมูลจะประกอบด้วยไอพีเฮดเดอร์ (IP header) ทีซีพีเฮดเดอร์ (TCP header) อาร์ทีพีเฮดเดอร์ (RTP header) และองค์ประกอบอื่น ๆ จะถูกประกาศไว้ ซึ่งหมายความว่าไม่ว่าเฮดเดอร์จะถูกใช้งานหรือไม่ เฮดเดอร์ของกลุ่มข้อมูลก็จะถูกสร้างขึ้น



รูปที่ 2.19 ข้อมูลภายในเฮดเดอร์กลุ่มข้อมูลของโปรแกรมเน็ตเวิร์คซิมูเลชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยทั่วไปแล้วกลุ่มข้อมูลในโปรแกรมเน็ตเวิร์คซิมูเลชั่นจะมีเพียงเฮดเดอร์เท่านั้น แต่กลุ่มข้อมูลสามารถที่จะนำเอาข้อมูลจริงบรรจุได้ โดยกำหนดพื้นที่สำหรับบรรจุข้อมูลไว้ ซึ่งในตัวโปรแกรมเน็ตเวิร์คซิมูเลชั่นไม่จำเป็นต้องกำหนดให้แต่ละกลุ่มข้อมูลบรรจุข้อมูล เพราะเป็นเพียงการจำลองการทำงาน แต่ก็สามารถจะกำหนดให้กลุ่มข้อมูลบรรจุข้อมูลได้เช่นกัน โดยทำการเพิ่มหรือเปลี่ยนแปลงเอเจนต์ที่มีเพื่อจำลองการทำงาน

2.2 เทคโนโลยีจาวาเว็บสตาร์ท (Java Web Start Technology)

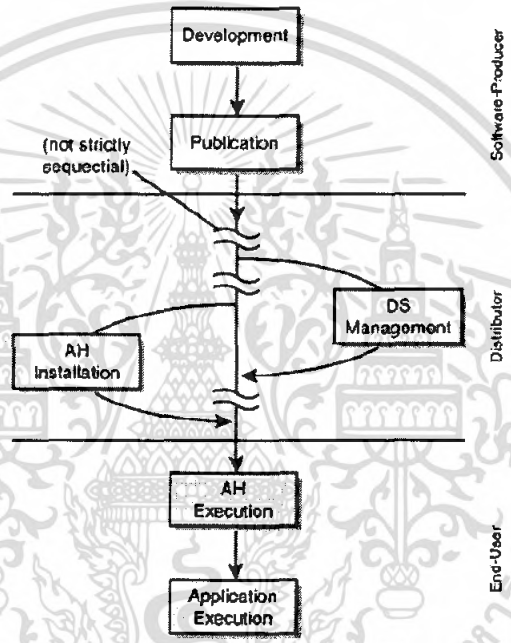
จาวาเว็บสตาร์ท (Java Web Start หรือ JAWS) คือ เทคโนโลยีดีพลอยเมนต์ (deployment) ที่มีอยู่ในภาษาจาวา โดยเป็นตัวช่วยให้สามารถเข้าใช้งานแอปพลิเคชั่นได้ผ่านทางเว็บเบราว์เซอร์ (Web browser) หรือผ่านทางเครือข่าย เมื่อผู้ใช้งานเรียกการทำงานผ่านลิงค์ซึ่งเป็นตัวสั่งการทำงานไฟล์เจเอ็นแอลพี (JNLP หรือ Java Network Launch Protocol) จะเป็นการเรียกให้เว็บเบราว์เซอร์ เริ่มการทำงานจาวาเว็บสตาร์ท โดยจะดาวน์โหลดและเริ่มการทำงานแอปพลิเคชั่นอัตโนมัติ ซึ่งก่อนที่จะอธิบายเรื่องจาวาเว็บสตาร์ท จะกล่าวถึงหัวข้อเรื่องซอฟต์แวร์ดีพลอยเมนต์ (Software Deployment) ซึ่งเป็นกระบวนการในการพัฒนาโปรแกรม ตั้งแต่การพัฒนา ทดสอบ ติดตั้ง และใช้งานแอปพลิเคชั่นผ่านทางเน็ตเวิร์คโดยจะมีตัวช่วยแอปพลิเคชั่น (Application Helper หรือ AH) หรือเรียกอีกอย่างหนึ่งว่าแอปพลิเคชั่นคอนเทนเนอร์ (Application Container หรือ AC) ซึ่งเป็นโปรแกรมที่จะช่วยในการดาวน์โหลด ติดตั้ง และเรียกใช้งานแอปพลิเคชั่น ที่ถูกนำไปเก็บไว้บนเซิร์ฟเวอร์ (Server) เพื่อให้ไคลเอนต์ (client) สามารถใช้บริการ และยังมีหน้าที่คอยดูแล จัดการโปรแกรม หรือการใช้งานทรัพยากรภายในเครื่อง โดยที่ชุดคำสั่งของแอปพลิเคชั่นไม่ได้อยู่บนเครื่องไคลเอนต์ แต่จะอยู่บนเซิร์ฟเวอร์ผู้ให้บริการ โดย AH เป็นตัวที่จะช่วยในการเชื่อมต่อระหว่างดีพลอยเมนต์เซิร์ฟเวอร์ (Deployment Server หรือ DS) กับสภาวะแวดล้อมของไคลเอนต์ (Client Environment)เข้าด้วยกัน ซึ่งในส่วนของเซิร์ฟเวอร์จะมีการสร้างนโยบายดีพลอยเมนต์ (Deployment Policies) ที่จะคอยช่วยในการตัดสินว่าควรจะติดตั้ง แอปพลิเคชั่นตัวไหนให้เหมาะสมอย่างเช่น ถ้าเพลตฟอร์ม (platform) คือโทรศัพท์มือถือ ก็จะติดตั้งแอปพลิเคชั่นที่เหมาะสมกับโทรศัพท์มือถือ แต่ถ้าเป็นโน๊ตบุ๊ค (notebook) หรือพีซีก็จะติดตั้งแอปพลิเคชั่นอีกตัวหนึ่ง ซึ่งสามารถที่จะแสดงการทำงานได้ดังรูปที่ 2.20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.20 บทบาทและหน้าที่ของตัวช่วยแอปพลิเคชั่น

2.2.1 โครงสร้างพื้นฐานของกระบวนการดีพลอยเมนต์

กระบวนการดีพลอยเมนต์แบ่งออกเป็น 6 ขั้นตอนใหญ่ ๆ ดังนี้



รูปที่ 2.21 โครงสร้างพื้นฐานของการดีพลอยเมนต์

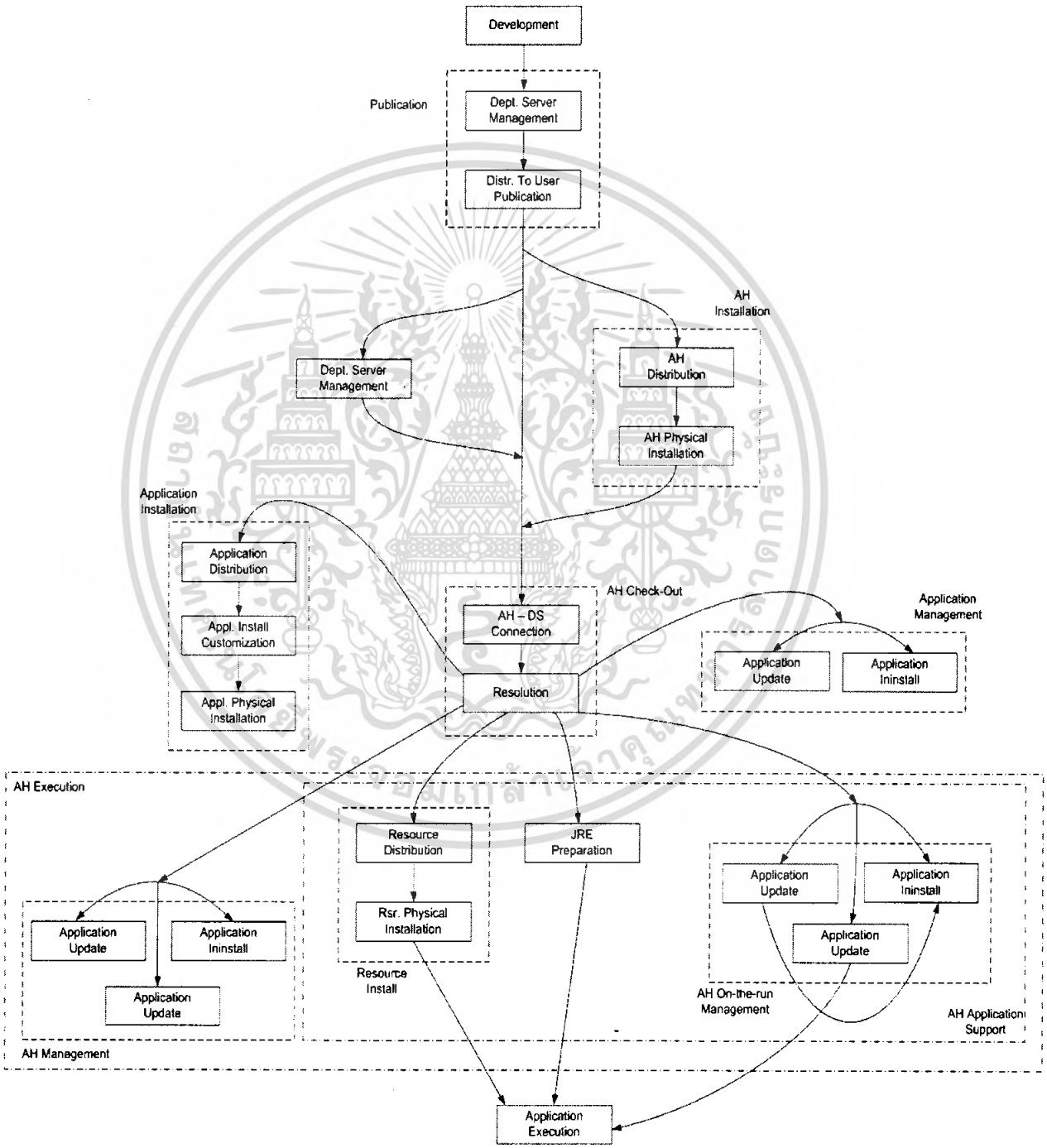
จากรูปที่ 2.21 กระบวนการดีพลอยเมนต์จะมีลำดับการทำงานดังนี้

1. ในช่วงซอฟต์แวร์โปรดิวเซอร์ (software producer) จะเริ่มจากการวิเคราะห์ ออกแบบและพัฒนาซอฟต์แวร์ ซึ่งอยู่ในขั้นตอนการพัฒนา (development) จากนั้นจะนำไปสู่การออกไปเผยแพร่ในขั้นตอนการประกาศ (publication)
2. ในช่วงของการแจกจ่าย (distributor) จะเป็นช่วงที่คอยดูแลจัดการการเรียกใช้งานแอปพลิเคชั่นของไคลเอนต์ ซึ่งขึ้นกับนโยบายดีพลอยเมนต์ด้วยว่าในแต่ละแอปพลิเคชั่นจะทำงานในแพลตฟอร์มใด โดยตัวแอปพลิเคชั่นจะถูกนำเก็บไว้ในที่ดีพลอยเมนต์เซิร์ฟเวอร์ (DS) เพื่อให้ผู้ใช้งานสามารถเรียกใช้งานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ในช่วงของผู้ใช้งานปลายทาง (end-user) ในไคลเอนต์จะต้องติดตั้งตัวช่วยแอปพลิเคชั่นเพื่อช่วยในการจัดการ การเรียกใช้งานแอปพลิเคชั่นโดยเมื่อผู้ใช้งานเรียกใช้แอปพลิเคชั่นซึ่งถูกนำไปเก็บไว้บนดีพลอยเมนต์เซิร์ฟเวอร์ ตัวช่วยแอปพลิเคชั่นจะเป็นตัวที่จะติดต่อไปยังดีพลอยเมนต์เซิร์ฟเวอร์ และดาวน์โหลดมาเก็บไว้ในเครื่องไคลเอนต์ เมื่อดาวน์โหลดเสร็จเรียบร้อยก็จะติดตั้งและเรียกแอปพลิเคชั่นขึ้นมาทำงาน

กระบวนการย่อยทั้งหมดสามารถแบ่งได้ดังรูปที่ 2.22



รูปที่ 2.22 โครงสร้างโดยสมบูรณ์ของการดีพลอยเมนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กระบวนการย่อยภายในจะประกอบด้วย

1. ขั้นตอนการพัฒนา
2. ขั้นตอนการประกาศ
3. การติดตั้งโดยตัวช่วยแอปพลิเคชั่น
4. การตรวจสอบสถานะของตัวช่วยแอปพลิเคชั่น
5. การติดตั้งแอปพลิเคชั่น
6. การเตรียม JRE โดยที่มีตัวช่วยแอปพลิเคชั่นรองรับ
7. การเรียกการทำงานของแอปพลิเคชั่น
8. การติดตั้งทรัพยากรโดยที่มีตัวช่วยแอปพลิเคชั่นรองรับ
9. การจัดการในขณะที่แอปพลิเคชั่นกำลังทำงาน
10. การจัดการแอปพลิเคชั่น
11. การจัดการตัวช่วยแอปพลิเคชั่น

ในแต่ละส่วนสามารถอธิบายได้ดังนี้

1. ขั้นตอนการพัฒนา (Development)

ในเฟสของการพัฒนาแอปพลิเคชั่นเป็นที่ส่วนสำคัญ โดยผลลัพธ์ที่ได้จะเป็นทรัพยากรซอฟต์แวร์ที่สามารถนำไปติดตั้งในดีพลอยเมนต์เซิร์ฟเวอร์เพื่อให้ผู้ใช้สามารถเรียกใช้งานได้

2. ขั้นตอนการประกาศ (Publication)

ในส่วนนี้จะประกอบด้วย 2 ส่วน คือ

- การประกาศแจกจ่ายจากผู้ผลิต (Producer to Distributor Publication) ในขั้นตอนนี้เป็นกระบวนการภายในของดีพลอยเมนต์เซิร์ฟเวอร์ โดยจะเป็นการนำเอาทรัพยากรต่าง ๆ ที่ถูกสร้างหรือพัฒนาขึ้นมา ติดตั้งไว้ภายในในดีพลอยเมนต์เซิร์ฟเวอร์
- การประกาศแจกจ่ายไปยังผู้ใช้งาน (Distributor to End-User Publication) ในขั้นตอนนี้เป็นการเรียกใช้งานทรัพยากรต่าง ๆ ซึ่งติดตั้งอยู่บนดีพลอยเมนต์เซิร์ฟเวอร์โดยตัวช่วยแอปพลิเคชั่นที่ติดตั้งอยู่บนเครื่องของไคลเอนต์จะเป็นตัวเรียกใช้งาน

3. การติดตั้งโดยตัวช่วยแอปพลิเคชั่น (Application Helper Installation)

ในเฟสนี้ตัวช่วยแอปพลิเคชั่นจะทำการติดตั้งตัวเองลงบนเครื่องของไคลเอนต์ที่ยังไม่มีการติดตั้ง โดยจะประกอบด้วย 2 ส่วนดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การกระจายตัวช่วยแอปพลิเคชั่น (AH Distribution) คือ การเผยแพร่ตัวช่วยแอปพลิเคชั่นเหมือนกับทรัพยากรอื่น ๆ ในกรณีที่ไคลเอนต์ไม่มีตัวช่วยแอปพลิเคชั่นติดตั้งอยู่บนเครื่องก็จะทำการดาวน์โหลดตัวช่วยแอปพลิเคชั่นไปติดตั้ง
- การติดตั้งตัวช่วยแอปพลิเคชั่น (AH Physical Installation) หลังจากที่มีการดาวน์โหลดตัวช่วยแอปพลิเคชั่น แล้วก็จะติดตั้งลงบนเครื่องของไคลเอนต์

4. การตรวจสอบสถานะของตัวช่วยแอปพลิเคชั่น (AH Check-Out)

ในเฟสนี้จะเป็นการตรวจสอบสถานะ และเลือกทำตามสถานะที่เกิดขึ้น โดยจะมีอยู่ 2 ส่วน ดังนี้

- การติดต่อระหว่างตัวช่วยแอปพลิเคชั่นและดีพลอยเมนต์เซิร์ฟเวอร์ (AH-DS Connection) เป็นการตรวจสอบสถานะการติดต่อระหว่างตัวช่วยแอปพลิเคชั่นและดีพลอยเมนต์เซิร์ฟเวอร์โดยถ้าไม่สามารถติดต่อกันได้ ตัวช่วยแอปพลิเคชั่นจะทำงานในสถานะออฟไลน์และจะเลือกทำงานตามนโยบายที่ได้วางเอาไว้
- ผลลัพธ์ที่ได้ (Resolution) คือ ผลลัพธ์ที่ได้จากการตรวจสอบการเชื่อมต่อตัว ช่วยและดีพลอยเมนต์เซิร์ฟเวอร์โดยจะตัดสินว่าควรที่จะทำอย่างไรต่อไป

5. การติดตั้งแอปพลิเคชั่น (Application Installation)

ในเฟสนี้เป็นการติดตั้งแอปพลิเคชั่นลงบนไคลเอนต์ เมื่อมีการเรียกใช้งานเป็นครั้งแรก ซึ่งการติดตั้งจะถูกจัดการโดยตัวช่วยแอปพลิเคชั่นให้อัตโนมัติ ซึ่งสามารถแบ่งออกเป็น 3 ส่วน ดังนี้

- การกระจายแอปพลิเคชั่น (Application Distribution) ในขั้นตอนนี้เป็นการดาวน์โหลดแอปพลิเคชั่นซึ่งติดตั้งอยู่บนดีพลอยเมนต์เซิร์ฟเวอร์ไปยังไคลเอนต์
- การเตรียมตัวการติดตั้งแอปพลิเคชั่น (Application Installation Customization) เป็นการเตรียมตัวของไคลเอนต์ ก่อนที่จะทำการติดตั้งแอปพลิเคชั่นลงบนไคลเอนต์
- การติดตั้งแอปพลิเคชั่น (Application Physical Installation) เป็นการติดตั้งแอปพลิเคชั่นที่ได้ดาวน์โหลดจากดีพลอยเมนต์เซิร์ฟเวอร์ ลงบนเครื่องของไคลเอนต์

6. การเตรียม JRE โดยที่มีตัวช่วยแอปพลิเคชั่นรองรับ (JRE Preparation During AH Application Support)

ตัวช่วยแอปพลิเคชั่นจะเรียกใช้สภาวะแวดล้อมของจาวา (Java Runtime Environment หรือ JRE) เวอร์ชั่นที่ตรงกัน โดยจะดูจากนโยบายดีพลอยเมนต์ ผลที่ได้จากขั้นตอนนี้คือการกำหนดค่าให้แก่ JRE เพื่อจะใช้ในขั้นตอนการทำงานต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. การเรียกการทำงานของแอปพลิเคชั่น (Application Execution)

ในเฟสนี้แอปพลิเคชั่นพร้อมจะทำงานโดย JRE จะช่วยในการเตรียมสภาวะแวดล้อม

8. การติดตั้งทรัพยากรโดยที่มีตัวช่วยแอปพลิเคชั่นรองรับ (Resource Installation During AH Application Support)

ในขณะที่แอปพลิเคชั่นกำลังทำงานอยู่นั้น ตัวช่วยแอปพลิเคชั่นจะดูแลดังนี้

- การกระจายของทรัพยากร (Resource Distribution) การทำงานในขั้นนี้จะคล้ายกับเฟสการกระจายอื่น ๆ ซึ่งจะติดต่อกลับไปยังดีพลอยเมนต์เซิร์ฟเวอร์โดยจะร้องขอทรัพยากรที่ต้องการใช้ จากนั้นก็จะดาวน์โหลดมาติดตั้งบนเครื่องของไคลเอนต์
- การติดตั้งทรัพยากร (Resource Physical Installation) เมื่อทรัพยากรได้ถูกดาวน์โหลดจากดีพลอยเมนต์เซิร์ฟเวอร์ มาแล้วก็จะติดตั้งลงบนเครื่องของไคลเอนต์ซึ่งจะจัดการอัตโนมัติโดยตัวช่วยแอปพลิเคชั่น

9. การจัดการในขณะที่แอปพลิเคชั่นกำลังทำงาน (On-The-Run Application Management During AH Application Suuport)

ในเฟสนี้ตัวช่วยแอปพลิเคชั่นจะทำหน้าที่ช่วยเหลือในขณะที่แอปพลิเคชั่นกำลังทำงานดังนี้

- การค้นหาข้อผิดพลาด (Debugging) เมื่อเกิดข้อผิดพลาดขึ้น ตัวช่วยแอปพลิเคชั่นจะทำหน้าที่ช่วยแก้ไขร่วมกับ ดีพลอยเมนต์เซิร์ฟเวอร์ซึ่ง JRE ที่ใช้งานจะต้องรองรับการทำงานด้วย
- การอัพเดตในขณะทำงาน (On-The-Run Updating) ในขณะที่แอปพลิเคชั่นกำลังทำงาน อาจจะมีบางส่วนเกิดการเปลี่ยนแปลงทำให้ต้องมีการอัพเดตข้อมูล
- การจัดการอื่น ๆ ในขณะทำงาน (Other-On-The-Run Management) ในขณะทำงานอาจจะมีบางอย่างเกิดขึ้นได้อย่างเช่นการอัพเดตของข้อมูล เพื่อเป็นประโยชน์ต่อการทำงาน

10. การจัดการแอปพลิเคชั่น (Application Management)

ในเฟสนี้จะประกอบด้วยการทำงานของตัวช่วยแอปพลิเคชั่น ในขณะที่ทำการติดตั้งแอปพลิเคชั่น ซึ่งแอปพลิเคชั่นไม่ได้ทำงานอยู่ในขณะนั้น โดยประกอบด้วยดังนี้

- การอัพเดตแอปพลิเคชั่น (Application Update) แอปพลิเคชั่นจะอัพเดตเมื่อตัวช่วยแอปพลิเคชั่นบนเครื่องของไคลเอนต์ ติดต่อไปยังเซิร์ฟเวอร์ซึ่งจะทำให้เกิดการติดตั้งหรือการย้ายออกของทรัพยากร โดยทรัพยากรที่จำเป็นเท่านั้นที่จะถูกจัดการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การยกเลิกการติดตั้งแอปพลิเคชั่น (Application Uninstall) ในส่วนนี้จะทำการยกเลิกการติดตั้งของแอปพลิเคชั่นออกจากเครื่องของไคลเอนต์

11. การจัดการตัวช่วยแอปพลิเคชั่น (AH Management)

การจัดการตัวช่วยแอปพลิเคชั่นสามารถที่จะเปลี่ยนแปลงได้ โดยอาจจะเป็นตัวช่วยแอปพลิเคชั่นเองหรือโปรแกรมอื่น ๆ ซึ่งในขั้นนี้จะประกอบด้วย 3 ขั้นตอนดังนี้

- การอัพเดตตัวช่วยแอปพลิเคชั่น (AH Update) เมื่อถึงเวลาตัวช่วยแอปพลิเคชั่นจะต้องมีการอัพเดตตัวเองเพื่อเป็นเวอร์ชั่นใหม่ ๆ
- การกำหนดค่าของแอปพลิเคชั่นและตัวช่วยแอปพลิเคชั่น (AH and Application Configuration) เป็นวิธีที่จะจัดการตัวช่วยแอปพลิเคชั่นเหมือนกับการจัดการแอปพลิเคชั่นโดยตัวช่วยแอปพลิเคชั่นและแอปพลิเคชั่นสามารถทำงานได้ ทั้งจากดีพลอยเมนต์เซิร์ฟเวอร์หรือบนเครื่องของไคลเอนต์ก็ได้
- การจัดการอื่น ๆ ของตัวช่วยแอปพลิเคชั่น (Other AH Management) คือ การที่สามารถใช้งานตัวช่วยแอปพลิเคชั่นให้ทำงานอื่น ๆ ได้ภายในเครื่องของไคลเอนต์

2.2.2 รายละเอียดของจาวาเว็บสตาร์ท

มีดังต่อไปนี้

1. จาวาเว็บสตาร์ทสามารถที่จะเรียกแอปพลิเคชั่นโปรแกรมที่พัฒนาด้วยภาษาจาวาได้ โดยสามารถเรียกแอปพลิเคชั่นที่อยู่บนเว็บเซิร์ฟเวอร์ได้ และสามารถที่จะเรียกผ่านแพลตฟอร์มใดก็ได้อย่าง วินโดวส์ 98/NT/2000/ME/XP ลินุกซ์หรือระบบปฏิบัติการอื่น ๆ
2. จาวาเว็บสตาร์ทสามารถรองรับ Java Standard Edition (JSE) ที่มีอยู่หลายเวอร์ชั่นได้ ซึ่งก็ขึ้นกับว่า ตัวแอปพลิเคชั่นมีต้องการเวอร์ชั่นอะไร โดยจาวาเว็บสตาร์ทสามารถที่จะดาวน์โหลดและติดตั้งได้อัตโนมัติ เมื่อเวอร์ชั่นที่ต้องการไม่ได้ติดตั้งอยู่บนเครื่องของไคลเอนต์
3. จาวาเว็บสตาร์ทอนุญาตให้แอปพลิเคชั่นสามารถทำงานได้โดยเป็นอิสระจากชนิดของเว็บเบราว์เซอร์ และสามารถที่จะทำงานได้ในรูปแบบออฟไลน์ ในกรณีไม่สามารถเรียกผ่านเบราว์เซอร์ หรือสามารถเรียกผ่านช็อตคัทที่อยู่บนเดสทอปได้เสมือนกับเรียกโปรแกรมที่อยู่บนเครื่องของไคลเอนต์
4. จาวาเว็บสตาร์ทได้สืบทอดและรองรับในเรื่องของความปลอดภัย (security) ของจาวาแพลตฟอร์ม ซึ่งแอปพลิเคชั่นที่ถูกเรียกใช้งานจะทำงานอยู่ในระบบที่ถูกป้องกันไว้ภายในแซนด์บ็อกซ์ (sandbox) และไม่สามารถที่จะเข้าใช้งานทรัพยากรในเครื่องหรือภายในเน็ตเวิร์คได้ ซึ่งตรงจุดนี้ทำให้ผู้ใช้งานรู้สึกปลอดภัยที่จะใช้งานแอปพลิเคชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. แอปพลิเคชั่นที่ทำงานด้วยจาวาเว็บสตาร์ทจะมีการแคช (cache) แอปพลิเคชั่นเก็บไว้ภายในเครื่องไคลเอนต์ที่เรียกใช้งาน โดยโปรแกรมที่ดาวน์โหลดไปแล้วจะเปรียบเสมือนกับว่าติดตั้งอยู่บนเครื่องของไคลเอนต์

เทคโนโลยีที่ใช้งานร่วมและเป็นส่วนหนึ่งของจาวาเว็บสตาร์ทก็คือ โพรโตคอลในการเรียกใช้งานจาวาผ่านเน็ตเวิร์ค (Java Network Launching Protocol & API หรือ JNLP) ซึ่งเทคโนโลยีนี้ถูกพัฒนาขึ้นโดยจาวาคอมมิวนิตี้โปรเซส (Java Community Process หรือ JCP) โดยเจเอ็นแอลพีเทคโนโลยีจะใช้ในการประกาศรูปแบบต่าง ๆ ในการที่จะเรียกใช้งานแอปพลิเคชั่น

2.2.3 การรักษาความปลอดภัย (Security)

จาวาเว็บสตาร์ทถูกพัฒนาขึ้นบนรากฐานของ Java SE ซึ่งจะมีความปลอดภัยในการทำงานโดยแอปพลิเคชั่นที่ทำงาน นั้นจะทำงานอยู่ภายในสภาวะแวดล้อมที่จำกัดหรือแซนด์บ็อกซ์ โดยจะป้องกันไม่ให้ตัวแอปพลิเคชั่นสามารถที่จะเข้าใช้งานไฟล์หรือเน็ตเวิร์คได้ ทำให้มีความปลอดภัยในการใช้งานมากขึ้น แต่แอปพลิเคชั่นสามารถจะร้องขอการเข้าใช้งานทรัพยากรต่าง ๆ ในเครื่องไคลเอนต์ได้โดยจะแสดงข้อความเตือนออกมาก่อนที่แอปพลิเคชั่นจะทำงานในครั้งแรก โดยที่ข้อความเตือนจะแสดงข้อมูลเกี่ยวกับผู้พัฒนาตัวแอปพลิเคชั่นนั้น ๆ ออกมา โดยถ้าเลือกเชื่อใจ แอปพลิเคชั่นจะถูกเรียกขึ้นมาทำงานทันที

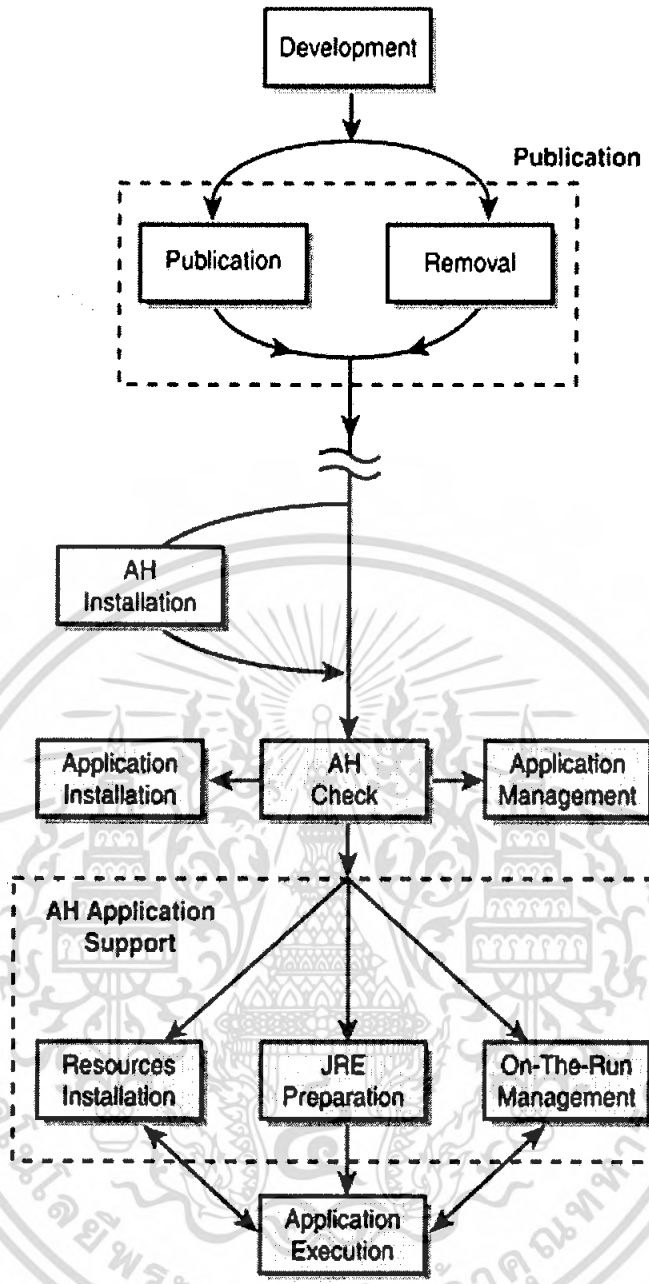
2.2.4 การเรียกใช้งานจาวาเว็บสตาร์ท

จาวาเว็บสตาร์ทสามารถที่จะเรียกแอปพลิเคชั่นที่พัฒนาด้วยภาษาจาวาได้โดยตรงจากเว็บ ซึ่งสามารถที่จะเรียกได้ 3 วิธีคือ

1. เรียกจากเว็บเบราว์เซอร์ได้โดยการกดที่ลิงค์ของไฟล์เจเอ็นแอลพี
2. เรียกได้จากไอคอนบนเดสทอปหรือสตาร์ทเมนู
3. เรียกได้จากจาวาแคชวิวเวอร์ (Java Cache Viewer)

เมื่อมีการเรียกใช้งานจาวาเว็บสตาร์ท จะมีติดต่อไปยังเว็บเซิร์ฟเวอร์ในแต่ละครั้งที่มีการเรียกใช้งานแอปพลิเคชั่น เพื่อตรวจสอบการอัพเดตเวอร์ชั่นใหม่ ๆ ของแอปพลิเคชั่น โดยวงจรการทำงานของจาวาเว็บสตาร์ทจะแสดงดังรูปที่ 2.23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.23 วงจรชีวิตของจาวาเว็บสตาร์ท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและพัฒนา

3.1 บทนำ

โครงงานฟรอนต์เอ็นสำหรับโปรแกรมเน็ตเวิร์คซิมูเลชั่นพัฒนาขึ้นเพื่อให้การใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่นสามารถใช้งานได้ง่ายขึ้น และลดขั้นตอนในการติดตั้งโปรแกรมที่ซับซ้อนลง ทำให้สามารถศึกษาหรือวิจัยการทำงานของโพรโตคอลต่างๆ ได้โดยไม่ต้องเสียเวลาในการศึกษาคำสั่งภาษาทีซีแอลก่อน อีกทั้งยังใช้งานและเข้าใจได้ง่ายเนื่องจากเป็นกราฟฟิกยูสเซอร์อินเทอร์เฟส (GUI) และมีการใช้เทคโนโลยีจาวาเว็บสตาร์ทเข้ามาช่วยในการเรียกใช้งานโปรแกรมฟรอนต์เอ็นทำให้สามารถเรียกใช้งานผ่านเว็บเบราว์เซอร์ได้โดยไม่ต้องติดตั้งโปรแกรม

3.2 โครงสร้างของโครงงาน

3.2.1 เครื่องมือที่ใช้ในการพัฒนา

- จาวาเทคโนโลยี (Java Technology)

เนื่องจากภาษาจาวาเป็นภาษาที่มีความนิยมสูง สามารถนำไลบารีที่มีอยู่แล้วมาพัฒนาต่อได้ทำให้ลดเวลาในการพัฒนาโปรแกรม อีกทั้งเป็นภาษาที่เปิดกว้างทางซอสโค๊ดและสามารถใช้งานได้ในทุกระบบปฏิบัติการ นอกจากนั้นยังมีเทคโนยีจาวาเว็บสตาร์ท (Java Web Start) ทำให้สามารถเรียกโปรแกรมใช้งานผ่านเครือข่ายทำได้สะดวก โดยไม่ต้องสร้างตัวโปรแกรมเป็นจาวาแอปเพล็ต (Java Applet) ที่มีข้อจำกัดในการเขียนและใช้งานมากกว่าโปรแกรมจาวาธรรมดา ในส่วนของโปรแกรมที่ช่วยในการพัฒนาในโครงงานนี้ใช้โปรแกรมเน็ตบีนส์ (NetBeans) เนื่องจากเป็นโปรแกรมที่ไม่เสียค่าใช้จ่าย และสามารถติดตั้งใช้งานได้ง่าย

- ระบบปฏิบัติการ (Operating System)

โครงการนี้มีการใช้ระบบปฏิบัติการยูบันตู (Ubuntu) เวอร์ชั่น 7.04 โดยจะใช้เป็นส่วนของเซิร์ฟเวอร์ เนื่องจากสามารถติดตั้งโปรแกรมเน็ตเวิร์คซิมูเลชั่นได้สะดวกและสมบูรณ์ แต่อาจไม่จำเป็นต้องใช้ระบบปฏิบัติการยูบันตูก็ได้เพราะสามารถใช้ระบบปฏิบัติการลินุกซ์ตัวใดก็ได้ และใช้ระบบปฏิบัติวินโดว์ในการพัฒนาส่วนของฟรอนต์เอ็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เว็บเบราว์เซอร์ (Web Browser)

เป็นตัวกลางสำหรับสื่อสารกันระหว่างฝั่งไคลเอนต์กับเซิร์ฟเวอร์เพื่อใช้ในการเรียกใช้งานโปรแกรมฟรอนต์เอ็นสำหรับเน็ตเวิร์คซิมูเลชั่น เว็บเบราว์เซอร์ที่สามารถใช้งานได้ยกตัวอย่าง อินเตอร์เน็ตเอ็กซ์โพลเรอร์ (Internet Explorer) ไฟล์ฟ็อกซ์ (Firefox)

- โปรแกรมเน็ตเวิร์คซิมูเลชั่นเวอร์ชั่น 2 (Network Simulator 2)

เป็นโปรแกรมหลักที่ทำหน้าที่ในการรับคำสั่งเป็นไฟล์ภาษาทีซีแอล และนำไปประมวลผลได้ออกมาเป็นไฟล์เอ็นเอเอ็ม เพื่อนำไปแสดงภาพเคลื่อนไหวต่อไป ซึ่งตัวโปรแกรมจะถูกติดตั้งบนเครื่องเว็บเซิร์ฟเวอร์

- เอชทีทีพีเซิร์ฟเวอร์ (HTTP Server)

เพื่อใช้เป็นเว็บเซิร์ฟเวอร์ในการบริการเว็บเพจสำหรับเรียกการทำงานโปรแกรมส่วนของฟรอนต์เอ็น โดยโครงการนี้ใช้งานโปรแกรมอาปาเช่ (Apache)

3.3 รายละเอียดโปรแกรมที่พัฒนา (Software Specification)

3.3.1 รายละเอียดส่วนนำเข้า

- ผู้ใช้งานเรียกใช้งานส่วนฟรอนต์เอ็นผ่านทางเว็บเบราว์เซอร์
- ผู้ใช้งานสามารถสร้างเครือข่ายที่ต้องการจำลองขึ้นมา โดยใช้งานในรูปแบบของกราฟฟิก และกำหนดตารางเวลาของการทำงานต่างๆ ของเครือข่ายได้
- ผู้ใช้งานสามารถเปิดไฟล์เครือข่ายที่เคยสร้างขึ้นไว้ก่อนหน้าขึ้นมาทำงานต่อได้

3.3.2 รายละเอียดส่วนแสดงผล

- โปรแกรมจะทำงานขึ้นมาโดยผ่านทางเว็บเบราว์เซอร์จากหน้าเว็บเพจ
- โปรแกรมจะแสดงเครือข่ายที่จำลองขึ้นมาในรูปแบบของกราฟฟิกโดยสามารถกำหนดค่าตัวแปรต่างๆ ได้โดยผ่านหน้าต่างส่วนผู้ใช้งาน
- โปรแกรมสามารถแปลงเครือข่ายที่จำลองขึ้นเป็นไฟล์คำสั่งภาษาทีซีแอลได้
- โปรแกรมสามารถบันทึกเครือข่ายที่สร้างขึ้นลงในเครื่องผู้ใช้ได้
- โปรแกรมแสดงผลลัพธ์ของการจำลองเครือข่ายที่สร้างขึ้นในรูปแบบของภาพเคลื่อนไหว

3.3.3 รายละเอียดฟังก์ชัน

- โปรแกรมสามารถแปลงการจำลองเครือข่ายที่สร้างขึ้นในรูปกราฟฟิกเป็นคำสั่งไฟล์ทีซีแอลเพื่อส่งไปประมวลผลที่เครื่องเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ในส่วนของเซิร์ฟเวอร์ของโปรแกรมเน็ตเวิร์คซิมูเลชั่นสามารถรับอินพุทเป็นไฟล์ทีซีแอลและประมวลผลส่งกลับมาในรูปของไฟล์เอ็นเอเอ็มได้
- ฟรอนต์เอ็นส่วนการแสดงผลสามารถรับอินพุทเป็นไฟล์เอ็นเอเอ็มจากเครื่องเซิร์ฟเวอร์แล้วนำไปแสดงผลเป็นภาพเคลื่อนไหวได้
- โปรแกรมสามารถบันทึกและโหลดเครือข่ายที่สร้างขึ้นจากผู้ใช้ลงไฟล์ได้
- โปรแกรมทำงานโดยไม่ขึ้นกับระบบปฏิบัติการ
- โปรแกรมสามารถกำหนดค่าเริ่มต้นให้กับบางตัวแปรได้อย่างชื่อของโหนด
- โปรแกรมสามารถเชื่อมต่อโหนดต่าง ๆ ตามกฎเกณฑ์ของโปรแกรมเน็ตเวิร์คซิมูเลชั่น
- โปรแกรมสามารถจำลองและแสดงการรับส่งข้อมูลระหว่างเครื่องภายในเครือข่ายได้ตามตารางเวลาที่ผู้ใช้กำหนดไว้
- ส่วนของเซิร์ฟเวอร์สามารถรองรับการติดต่อขอใช้งานโปรแกรมจากเครื่องผู้ใช้งานได้หลายๆ เครื่องพร้อมกัน

3.3.4 โครงสร้างของซอฟต์แวร์ (Design)

ในการพัฒนาส่วนฟรอนต์เอ็น โปรแกรมจะแบ่งเป็นองค์ประกอบได้ 3 ส่วน ดังต่อไปนี้

3.3.4.1 ส่วนจำลองเครือข่าย

ในส่วนนี้เป็นส่วนที่ผู้ใช้งานมีการติดต่อด้วยโดยตรงซึ่งเป็นหน้าต่างผู้ใช้งานเพื่อความสะดวกต่อการใช้งาน ผู้ใช้งานสามารถจำลองเครือข่ายที่ต้องการจำลองในรูปของกราฟฟิก และสามารถตั้งค่าโพรโตคอลหรือลิงค์ต่าง ๆ ได้ อีกทั้งสามารถที่จะบันทึกหรือโหลดเครือข่ายที่จำลองขึ้นจากไฟล์ nsf (Network Simulation Front End) ได้ โดยหลังจากผู้ใช้งานจำลองเครือข่ายและตั้งค่าตัวแปรต่างๆ เสร็จแล้ว ส่วนนี้จะแปลงเครือข่ายนั้นเป็นไฟล์คำสั่งภาษาทีซีแอลเพื่อ สั่งงานให้โปรแกรมเน็ตเวิร์คซิมูเลชั่นทำงาน โดยฟรอนต์เอ็นจะส่งไฟล์ผ่านเครือข่ายไปให้กับเครื่องเซิร์ฟเวอร์ที่มีโปรแกรมเน็ตเวิร์คซิมูเลชั่นทำงานนำไปประมวลต่อไป ในส่วนนี้จะแบ่งการทำงานเป็นสองส่วนหลักๆ คือ

1. การสร้างเครือข่ายและการการกำหนดการทำงาน

ผู้ใช้งานสามารถสร้างเครือข่ายในรูปแบบต่าง ๆ แล้ววางได้อย่างอิสระ และสามารถกำหนดค่าการทำงานต่าง ๆ ของเครือข่ายได้

2. แปลงเครือข่ายเป็นคำสั่งภาษาทีซีแอล

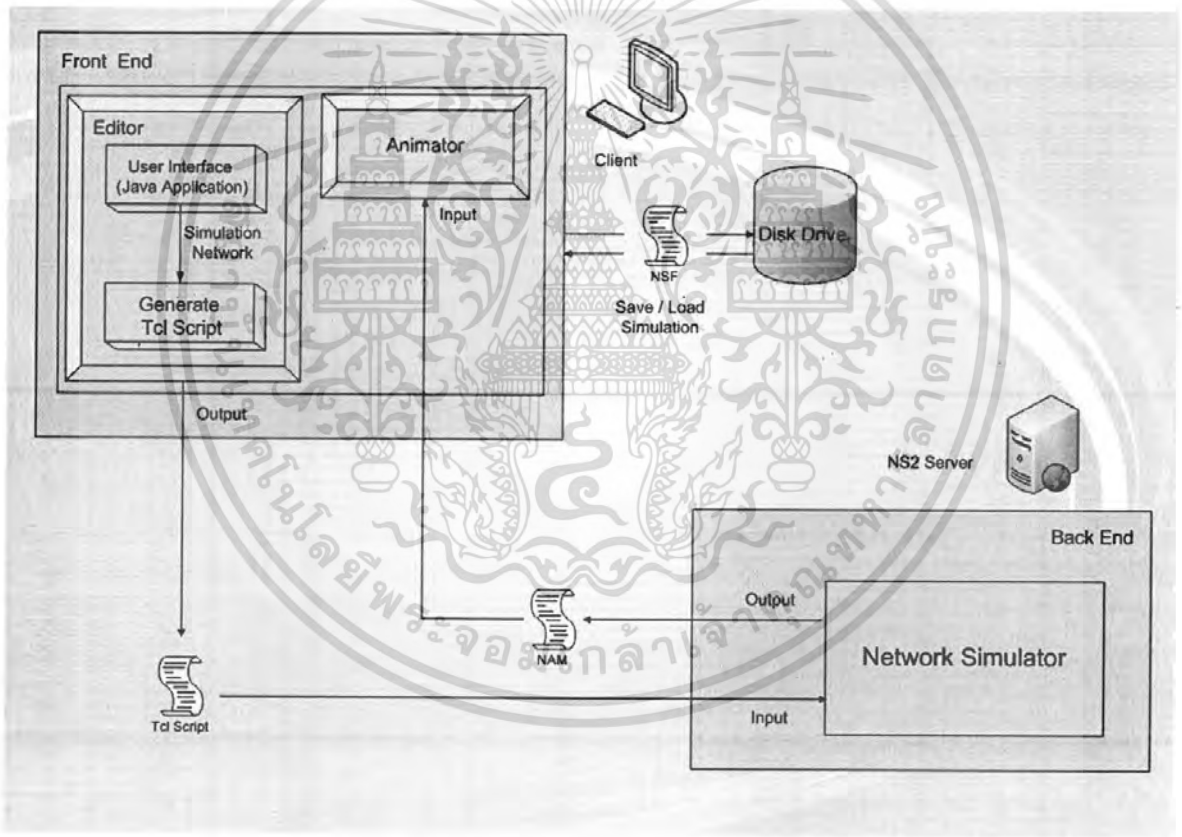
หลังจากผู้ใช้สร้างเครือข่ายจำลองเสร็จและทำการกำหนดตารางเหตุการณ์ต่าง ๆ โปรแกรมจะนำการตั้งค่าของโหนด การเชื่อมต่อ และเหตุการณ์ที่เกิดขึ้นแปลงเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งภาษาทีซีแอลเพื่อใช้สั่งในโปรแกรมเน็ตเวิร์คซิมูเลชั่นประมวลที่ทำงานอยู่บนเครื่องเซิร์ฟเวอร์

3.3.4.2 ส่วนการแสดงผล (Animator)

ในส่วนนี้จะทำหน้าที่แสดงผลลัพธ์ในลักษณะภาพเคลื่อนไหวที่ได้จากโปรแกรมเน็ตเวิร์คซิมูเลชั่นที่เครื่องเซิร์ฟเวอร์ส่งมาในรูปของไฟล์เอ็นเอเอ็ม ซึ่งลักษณะไฟล์เอ็นเอเอ็มจะอธิบายถึงข้อมูลการทำงานของกลุ่มข้อมูลในเครือข่ายที่สร้างขึ้นในเวลาต่าง ๆ โดยส่วนแสดงผลจะทำการอ่านไฟล์เอ็นเอเอ็มและนำเอาเหตุการณ์ต่าง ๆ มาแสดงผลเป็นภาพเคลื่อนไหว โดยภาพที่แสดงออกมาคือ ลักษณะเครือข่าย และการทำงานของกลุ่มข้อมูลภายในเครือข่าย นอกจากนั้นยังมีส่วนอธิบายข้อมูลของแต่ละกลุ่มข้อมูลภายในโปรแกรม



รูปที่ 3.1 แสดงโครงสร้างของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.4.3 ส่วนเซิร์ฟเวอร์ที่ให้บริการ

ในส่วนของแบ็กเอ็นนั้นจะมีองค์ประกอบสามส่วนคือ โปรแกรมเน็ตเวิร์คซิมูเลชั่น โปรแกรมให้บริการเว็บเซิร์ฟเวอร์ และโปรแกรมจาวาสำหรับเชื่อมต่อกับฟรอนต์เอ็น โดยโปรแกรมเน็ตเวิร์คซิมูเลชั่นจะเป็นโปรแกรมหลักในการทำงาน ทำหน้าที่ประมวลผลไฟล์ภาษาทีซีแอลซึ่งส่งมาจากไคลเอนต์ โดยผลลัพธ์ที่ได้จะอยู่ในรูปของไฟล์เอ็นเอเอ็ม โปรแกรมให้บริการเว็บเซิร์ฟเวอร์จะทำหน้าที่ให้บริการเว็บเพจสำหรับผู้ใช้งานเรียกใช้งานโปรแกรมฟรอนต์เอ็นจากเว็บบราวเซอร์ โดยในส่วนนี้มีการนำเทคโนโลยีจาวาเว็บสตาร์ทเพื่อใช้ในการเรียกใช้งานส่วนของฟรอนต์เอ็น และส่วนโปรแกรมจาวาสำหรับติดต่อกับฟรอนต์เอ็นจะทำหน้าที่รอคอยไฟล์ทีซีแอล นำไฟล์ทีซีแอลไปประมวลผลโดยเรียกใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่น และส่งไฟล์เอ็นเอเอ็มกลับไปยังส่วนฟรอนต์เอ็น โดยโครงสร้างทั้งหมดแสดงดังรูปที่ 3.1

3.3.5 ขอบเขตและข้อจำกัดของโปรแกรมที่พัฒนา

- โปรแกรมมีการอ้างอิงตัวแปรต่างๆ จากโปรแกรมเน็ตเวิร์คซิมูเลชั่น ดังนั้นผู้ใช้งานควรมีความรู้ความเข้าใจในตัวโปรแกรมเน็ตเวิร์คซิมูเลชั่น
- โปรแกรมไม่สามารถตรวจสอบความผิดพลาดที่เกิดขึ้นจากผู้ใช้งานอย่าง การตั้งค่าตัวแปรไม่ถูกต้อง หรือเชื่อมต่อลิงค์ไม่ครบ
- โปรแกรมรองรับการจำลองเครือข่ายเฉพาะแบบต่อสาย
- เครื่องผู้ใช้งานที่ขอใช้งานโปรแกรมต้องมีการติดตั้งสภาพแวดล้อมของโปรแกรมภาษาจาวา (Java Runtime Environment หรือ JRE) ลงอยู่ในเครื่องเพื่อให้สามารถเรียกใช้งานโปรแกรมภาษาจาวาได้

3.4 การออกแบบและพัฒนา

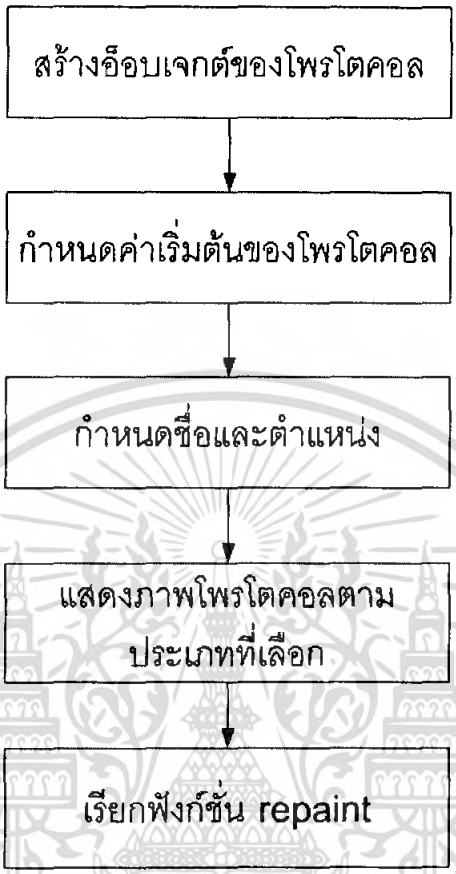
โครงงานฟรอนต์เอ็นสำหรับโปรแกรมเน็ตเวิร์คซิมูเลชั่นถูกออกแบบตามวัตถุประสงค์ของโครงงานสองข้อคือเพื่อให้ใช้โปรแกรมเน็ตเวิร์คซิมูเลชั่นได้ง่าย โดยไม่ต้องมีขั้นตอนในการคิดคั้งจากวัตถุประสงค์ข้อที่สองโปรแกรมฟรอนต์เอ็นจึงต้องทำงานเป็นเว็บแอปพลิเคชั่น ทำให้ต้องออกแบบเป็นสองส่วนคือส่วนของฟรอนต์เอ็นและส่วนแบ็กเอ็นหรือส่วนเซิร์ฟเวอร์ที่ให้บริการ และจากวัตถุประสงค์ข้อแรกโปรแกรมฟรอนต์เอ็นจึงถูกออกแบบเป็นกราฟฟิกยูสเซอร์อินเทอร์เฟส โดยส่วนของฟรอนต์เอ็นจะแบ่งย่อยออกเป็นสองส่วนหลักดังนี้

3.4.1 ส่วนจำลองเครือข่าย

ในส่วนการจำลองเครือข่ายทำหน้าที่สร้างเครือข่ายและแปลงเครือข่ายเป็นไฟล์คำสั่งภาษาทีซีแอล ส่วนการจำลองเครือข่ายถูกออกแบบเป็นกราฟฟิกยูสเซอร์อินเทอร์เฟส เพื่อรองรับการแสดงโพรโตคอลและลิงค์ต่างๆ เป็นภาพและการตั้งค่าต่างๆ ผ่านหน้าต่างผู้ใช้งาน โดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

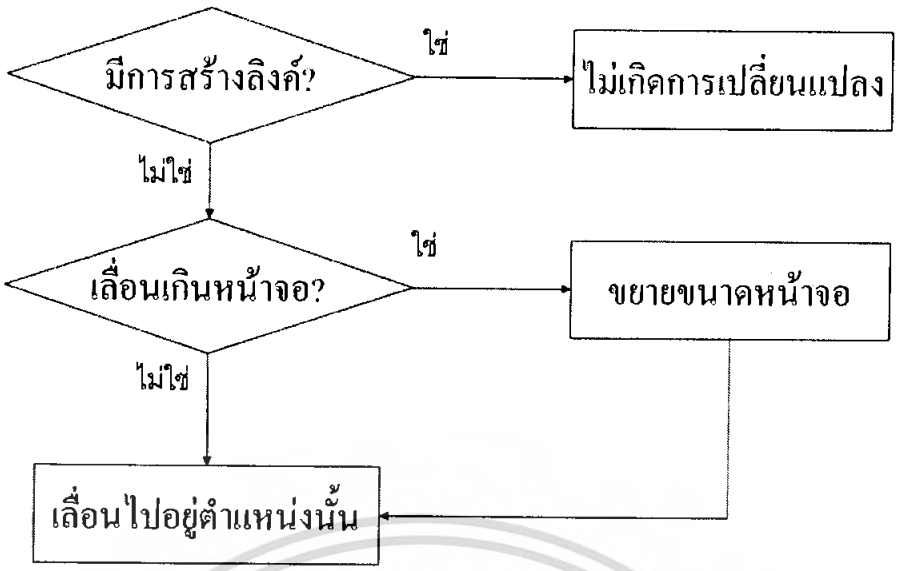
กระบวนการที่เกิดขึ้นจากผู้ใช้ต่างๆ คือ การสร้างโพรโตคอล การตั้งค่า การลากเพื่อเปลี่ยนตำแหน่ง การคลิกซ้อนสองครั้ง (Double Click) และการสร้างลิงค์



รูปที่ 3.2 แสดงขั้นตอนการสร้างโพรโตคอล

ในขั้นตอนการสร้างโพรโตคอลจากรูปที่ 3.2 เริ่มต้นเมื่อผู้ใช้เลือกโพรโตคอลที่ต้องการสร้าง จะทำการสร้างอ็อบเจกต์ของโพรโตคอลนั้นขึ้นมาเก็บไว้ในเว็กเตอร์ตามประเภทของโพรโตคอล จากนั้นทำการกำหนดค่าเริ่มต้นของโพรโตคอล รวมถึงชื่อและตำแหน่งซึ่งจะถูกกำหนดโดยอัตโนมัติ โดยชื่อจะกำหนดจากชื่อของประเภทโพรโตคอลต่อตัวเลขที่เพิ่มขึ้นตามจำนวนที่สร้าง และตำแหน่งจะถูกกำหนดโดยคำนวณไม่ให้โพรโตคอลที่สร้างขึ้นมาใหม่ทับตำแหน่งกับตำแหน่งเดิมเพื่อให้ง่ายต่อการใช้งาน ขั้นตอนต่อมาคือการเรียกฟังก์ชันการแสดงภาพตามประเภทโพรโตคอลนั้นๆ ขั้นตอนสุดท้ายคือทำการเรียกฟังก์ชัน repaint() เพื่อวาดภาพโพรโตคอลที่สร้างขึ้น

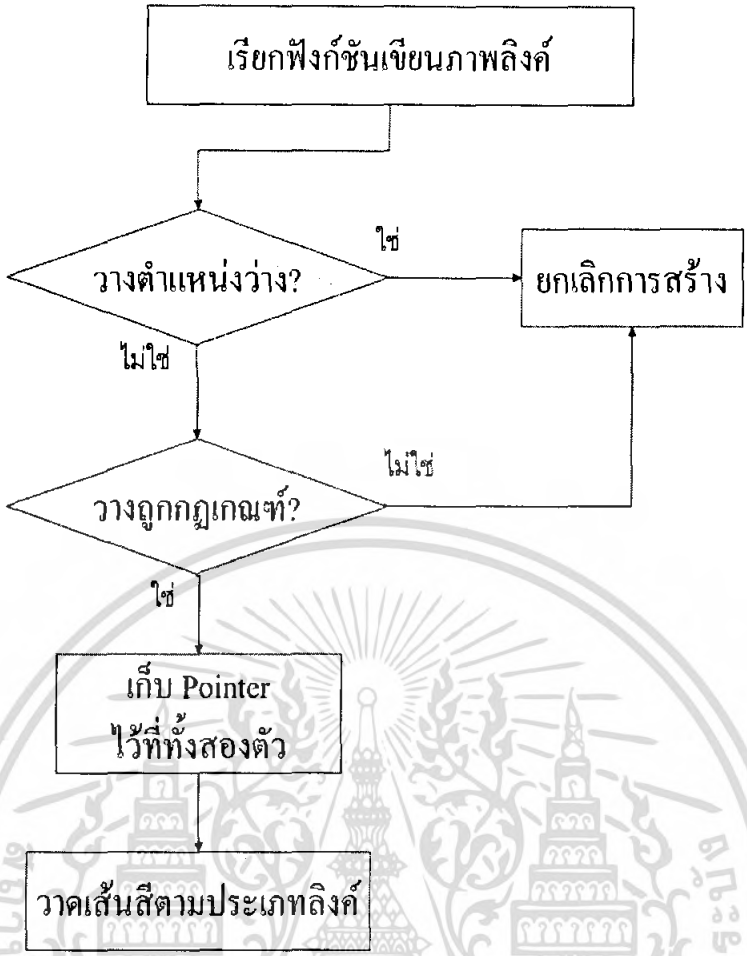
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 แสดงขั้นตอนการเลื่อนตำแหน่งของโพรโตคอล

จากรูปที่ 3.3 แสดงการเปลี่ยนแปลงตำแหน่งของโพรโตคอล เมื่อผู้ใช้งานกดเมาส์ปุ่มซ้ายค้างไว้จะเป็นการเลื่อนตำแหน่งของภาพที่แสดงโพรโตคอล โดยในขั้นตอนการเลื่อนตำแหน่งนั้นเริ่มแรกต้องทำการตรวจสอบการสร้างลิงค์ โดยถ้าโพรโตคอลนั้นกำลังสร้างลิงค์อยู่ขณะนั้นจะไม่สามารถทำการเลื่อนตำแหน่งได้เนื่องจากการแสดงลิงค์จะผิดพลาด หลังจากนั้นจะทำการตรวจสอบว่าตำแหน่งที่เลื่อนนั้นมีการตกขอบจอหรือไม่ โดยถ้ามีการเลื่อนตกขอบทางด้านซ้าย ซ้ายบนและด้านบนจะไม่ยอมให้มีการเลื่อน แต่ถ้าการเลื่อนเกินขนาดหน้าจอไปทางขวา ขวาล่างและด้านล่างจะทำการขยายหน้าจอ โดยมีปุ่มให้เลื่อนขึ้นลงหรือซ้ายขวา

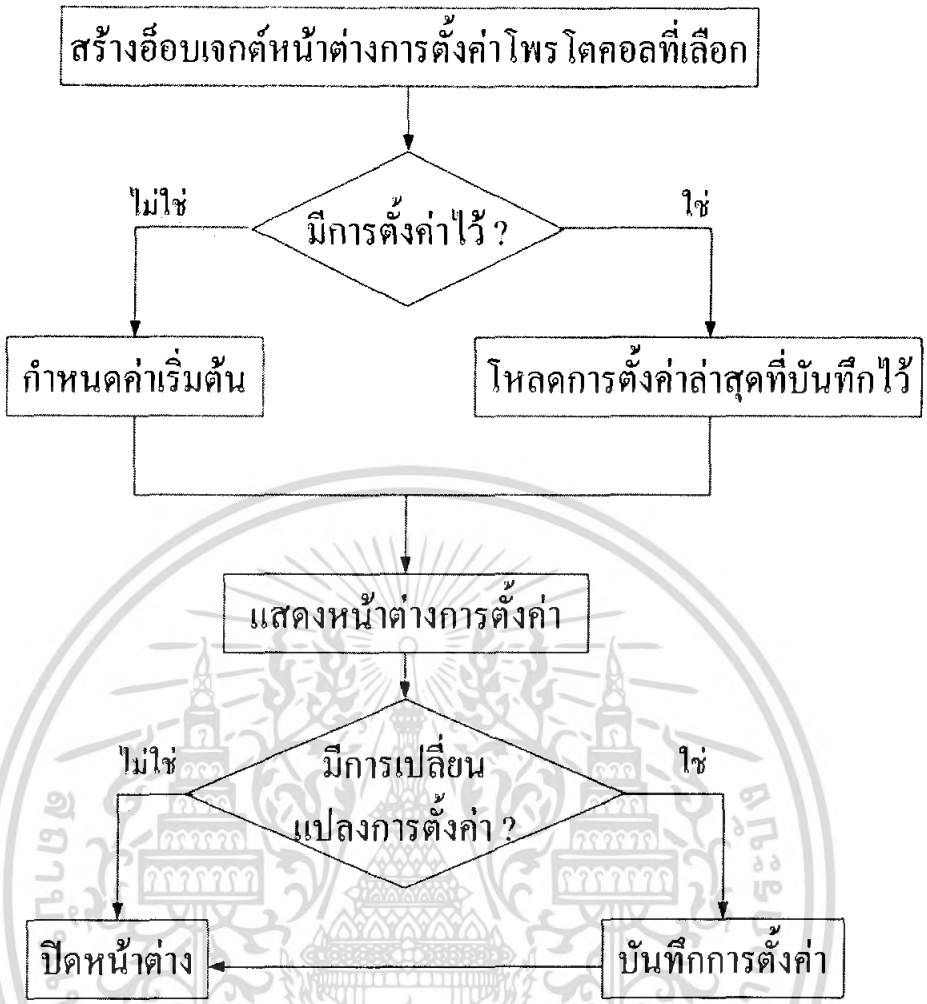
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 แสดงขั้นตอนการสร้างลิงค์

เมื่อผู้ใช้กดเมาส์ปุ่มขวาที่ภาพจะเป็นการสร้างการเชื่อมต่อหรือลิงค์ จากรูปที่ 3.4 จะแสดงการสร้างลิงค์ ขั้นตอนแรกจะทำการเรียกฟังก์ชันวาดเส้นตรงโดยมีจุดเริ่มต้นเป็นตำแหน่งของโพรโตคอลนั้น และมีจุดปลายเป็นตำแหน่งที่เคอร์เซอร์อยู่ขณะนั้น เมื่อผู้ใช้กดเมาส์ปุ่มขวาหรือซ้ายอีกครั้งหนึ่ง โปรแกรมจะทำการตรวจสอบว่าตำแหน่งที่กดนั้นมีโพรโตคอลอยู่หรือเป็นตำแหน่งว่าง โดยถ้าเป็นตำแหน่งว่างจะทำการยกเลิกการสร้างลิงค์นั้น แต่ถ้าเป็นตำแหน่งที่มีโพรโตคอลวางอยู่ โปรแกรมจะทำการตรวจสอบว่าการเชื่อมต่อนั้นถูกต้องตามกฎเกณฑ์ของโปรแกรมเน็ตเวิร์คซิมูเลชั่นหรือไม่ ถ้าถูกต้องโปรแกรมจะทำการวาดเส้นตรงแสดงการเชื่อมต่อระหว่างกัน โดยเส้นตรงจะมีสีแตกต่างกันตามประเภทของการเชื่อมต่อ และเก็บพ้อยต์เตอร์ของกันและกันเพื่อใช้ในการอ้างอิง

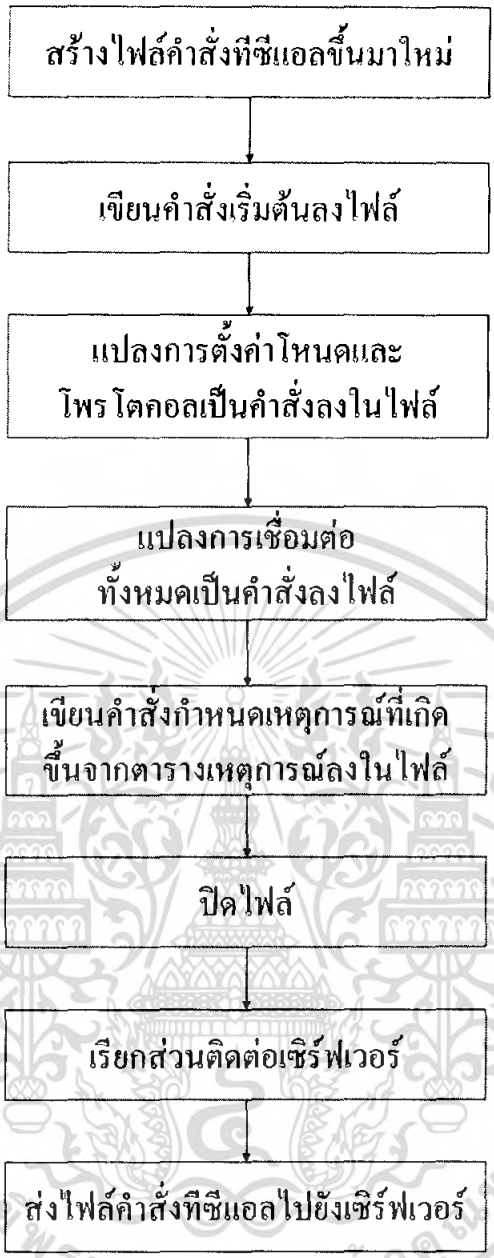
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 แสดงขั้นตอนการตั้งค่า

เมื่อผู้ใช้งานคลิกเมาส์ซ้อนกันสองครั้ง (Double Click) ตรงตำแหน่งที่มีภาพแสดงโพรโตคอลอยู่จะเปิดการเรียกหน้าต่างการตั้งค่าขึ้นมา กระบวนการทำงานของการตั้งค่าแสดงดังรูปที่ 3.5 โดยการทำงานขั้นตอนแรกคือสร้างอ็อบเจกต์ของหน้าต่างการตั้งค่า หลังจากนั้นเรียกค่าต่างๆ ที่เก็บไว้ในอ็อบเจกต์ของโพรโตคอลที่ต้องการนำมาแสดงในหน้าต่างการตั้งค่า ซึ่งถ้าไม่มีค่าที่บันทึกจะทำการกำหนดค่าเริ่มต้นให้ หลังจากแสดงหน้าต่างการตั้งค่าแล้วถ้าผู้ใช้มีการบันทึกเปลี่ยนแปลงโปรแกรมจะเก็บค่าต่างๆ ไว้ในอ็อบเจกต์ของโพรโตคอลที่เลือกตั้งค่า แต่ถ้าไม่ต้องการเปลี่ยนแปลงค่า โปรแกรมจะปิดหน้าต่างการตั้งค่าโดยไม่มีการบันทึก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 แสดงขั้นตอนการแปลงเครือข่าย

หลังจากผู้ใช้งานทำการจำลองเครือข่ายและตั้งค่าต่างๆ รวมถึงตารางเหตุการณ์เสร็จแล้ว ขั้นตอนต่อไปคือการแปลงเครือข่ายเป็นคำสั่งไฟล์ภาษาทีซีแอล เริ่มแรกจะสร้างไฟล์นามสกุลทีซีแอลขึ้นมาโดยยังไม่มีข้อมูลใดๆ ต่อมาทำการเขียนส่วนหัวและการประกาศทั่วไป จากนั้นเริ่มอ่านข้อมูลของโหนด โพรโตคอลต่างๆ และการเชื่อมต่อทั้งหมด จากเวกเตอร์และทำการแปลงออกมาเป็นคำสั่งไฟล์ภาษาทีซีแอล ขั้นตอนต่อไปหลังจากประกาศโพรโตคอลและลิงค์เสร็จคือการกำหนดเวลาในการเกิดเหตุการณ์ต่างๆ จากตารางเหตุการณ์ เมื่อเสร็จแล้วทำการปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไฟล์และเรียกส่วนติดต่อไปยังเครื่องเซิร์ฟเวอร์เพื่อทำการส่งไฟล์คำสั่งภาษาทีซีแอลไปให้และรอรับไฟล์เอ็นเอเอ็ม

3.4.2 ส่วนแสดงผล (Animator)

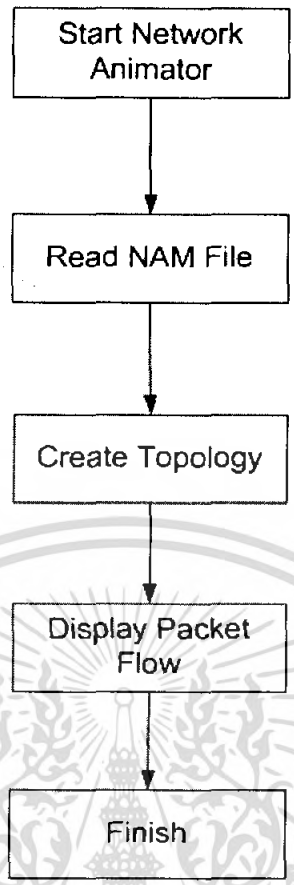
ในส่วนนี้จะทำหน้าที่แสดงผลลัพธ์ที่ได้จากโปรแกรมเน็ตเวิร์คซิมูเลชั่น โดยเครื่องเซิร์ฟเวอร์ส่งมาในรูปของไฟล์เอ็นเอเอ็ม (ดังรูปที่ 3.7) และนำไปแสดงผลการทำงานในลักษณะภาพเคลื่อนไหว ส่วนนี้จึงออกแบบเป็นหน้าต่างผู้ใช้งานเนื่องจากผู้ใช้งานสามารถควบคุมลักษณะการแสดงผลได้ ซึ่งจะแบ่งส่วนแสดงผลเป็นสองส่วนคือ ส่วนแสดงภาพเคลื่อนไหว และส่วนแสดงรายละเอียดการทำงานของกลุ่มข้อมูล

```
V -t * -v 1.0a5 -a 0
W -t * -x 318 -y 256
A -t * -n 1 -p 0 -o 0xffffffff -c 31 -a 1
A -t * -h 1 -m 2147483647 -s 0
c -t * -i 0 -n black
c -t * -i 1 -n black
n -t * -a 0 -s 0 -S UP -v circle -c black -i black -x 141 -y 179 -Z 0.0
n -t * -a 1 -s 1 -S UP -v circle -c black -i black -x 318 -y 256 -Z 0.0
l -t * -s 1 -d 0 -S UP -r 2000000 -D 0.01 -c black
q -t * -s 0 -d 1 -a 0.5
q -t * -s 1 -d 0 -a 0.5
n -t 0 -s 0 -S DLABEL -l "Node0" -L ""
n -t 0 -s 1 -S DLABEL -l "Node1" -L ""
+ -t 1 -s 0 -d 1 -p cbr -e 210 -c 1 -i 0 -a 1 -x {0.0 1.0 0 ------- null}
- -t 1 -s 0 -d 1 -p cbr -e 210 -c 1 -i 0 -a 1 -x {0.0 1.0 0 ------- null}
h -t 1 -s 0 -d 1 -p cbr -e 210 -c 1 -i 0 -a 1 -x {0.0 1.0 -1 ------- null}
+ -t 1.00375 -s 0 -d 1 -p cbr -e 210 -c 1 -i 1 -a 1 -x {0.0 1.0 1 ------- null}
- -t 1.00375 -s 0 -d 1 -p cbr -e 210 -c 1 -i 1 -a 1 -x {0.0 1.0 1 ------- null}
h -t 1.00375 -s 0 -d 1 -p cbr -e 210 -c 1 -i 1 -a 1 -x {0.0 1.0 -1 ------- null}
+ -t 1.0075 -s 0 -d 1 -p cbr -e 210 -c 1 -i 2 -a 1 -x {0.0 1.0 2 ------- null}
- -t 1.0075 -s 0 -d 1 -p cbr -e 210 -c 1 -i 2 -a 1 -x {0.0 1.0 2 ------- null}
h -t 1.0075 -s 0 -d 1 -p cbr -e 210 -c 1 -i 2 -a 1 -x {0.0 1.0 -1 ------- null}
r -t 1.01084 -s 0 -d 1 -p cbr -e 210 -c 1 -i 0 -a 1 -x {0.0 1.0 0 ------- null}
+ -t 1.01125 -s 0 -d 1 -p cbr -e 210 -c 1 -i 3 -a 1 -x {0.0 1.0 3 ------- null}
- -t 1.01125 -s 0 -d 1 -p cbr -e 210 -c 1 -i 3 -a 1 -x {0.0 1.0 3 ------- null}
h -t 1.01125 -s 0 -d 1 -p cbr -e 210 -c 1 -i 3 -a 1 -x {0.0 1.0 -1 ------- null}
r -t 1.01459 -s 0 -d 1 -p cbr -e 210 -c 1 -i 1 -a 1 -x {0.0 1.0 1 ------- null}
+ -t 1.015 -s 0 -d 1 -p cbr -e 210 -c 1 -i 4 -a 1 -x {0.0 1.0 4 ------- null}
```

รูปที่ 3.7 ตัวอย่างไฟล์เอ็นเอเอ็ม

โดยผลลัพธ์ที่แสดงออกมาคือ ลักษณะเครือข่ายที่ได้กำหนด การทำงานของกลุ่มข้อมูลภายในเครือข่าย และยังมีส่วนอธิบายของแต่ละกลุ่มข้อมูลซึ่งทำงานภายในโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 แสดงกระบวนการทำงานของส่วนการแสดงผล

จากรูปที่ 3.8 จะเห็นกระบวนการทำงานของส่วนการแสดงผล โดยเมื่อส่วนฟรอนด์เอ็นได้รับไฟล์เอ็นเอเอ็มแล้ว จะเริ่มแสดงผลการทำงานโดยอ่านแต่ละเหตุการณ์ภายในไฟล์เอ็นเอเอ็ม และนำมาสร้างเป็นเครือข่าย โดยส่วนการสร้างเครือข่ายจะอยู่ในส่วนต้นของไฟล์เอ็นเอเอ็ม ดังรูปที่ 3.9

```
V -t * -v 1.0a5 -a 0
W -t * -x 318 -y 256
A -t * -n 1 -p 0 -o 0xffffffff -c 31 -a 1
A -t * -h 1 -m 2147483647 -s 0
c -t * -i 0 -n black
c -t * -i 1 -n black
n -t * -a 0 -s 0 -S UP -v circle -c black -i black -x 141 -y 179 -Z 0.0
n -t * -a 1 -s 1 -S UP -v circle -c black -i black -x 318 -y 256 -Z 0.0
l -t * -s 1 -d 0 -S UP -r 2000000 -D 0.01 -c black
q -t * -s 0 -d 1 -a 0.5
q -t * -s 1 -d 0 -a 0.5
n -t 0 -s 0 -S DLABEL -l "Node0" -L ""
n -t 0 -s 1 -S DLABEL -l "Node1" -L ""
```

รูปที่ 3.9 ส่วนการสร้างเครือข่ายภายในไฟล์เอ็นเอเอ็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อสร้างเครือข่ายเสร็จเรียบร้อย ก็จะอ่านแต่ละเหตุการณ์ภายในเครือข่ายที่ได้สร้างขึ้น และนำมาแสดงการทำงานของแต่ละกลุ่มข้อมูล โดยแต่ละเหตุการณ์ภายในไฟล์เอ็นเอเอ็มแสดงได้ดังรูปที่ 3.10

```
+ -t 1 -s 0 -d 1 -p cbr -e 210 -c 1 -i 0 -a 1 -x {0.0 1.0 0 ------- null}
- -t 1 -s 0 -d 1 -p cbr -e 210 -c 1 -i 0 -a 1 -x {0.0 1.0 0 ------- null}
h -t 1 -s 0 -d 1 -p cbr -e 210 -c 1 -i 0 -a 1 -x {0.0 1.0 -1 ------- null}
+ -t 1.00375 -s 0 -d 1 -p cbr -e 210 -c 1 -i 1 -a 1 -x {0.0 1.0 1 ------- null}
- -t 1.00375 -s 0 -d 1 -p cbr -e 210 -c 1 -i 1 -a 1 -x {0.0 1.0 1 ------- null}
h -t 1.00375 -s 0 -d 1 -p cbr -e 210 -c 1 -i 1 -a 1 -x {0.0 1.0 -1 ------- null}
+ -t 1.0075 -s 0 -d 1 -p cbr -e 210 -c 1 -i 2 -a 1 -x {0.0 1.0 2 ------- null}
- -t 1.0075 -s 0 -d 1 -p cbr -e 210 -c 1 -i 2 -a 1 -x {0.0 1.0 2 ------- null}
h -t 1.0075 -s 0 -d 1 -p cbr -e 210 -c 1 -i 2 -a 1 -x {0.0 1.0 -1 ------- null}
r -t 1.01084 -s 0 -d 1 -p cbr -e 210 -c 1 -i 0 -a 1 -x {0.0 1.0 0 ------- null}
+ -t 1.01125 -s 0 -d 1 -p cbr -e 210 -c 1 -i 3 -a 1 -x {0.0 1.0 3 ------- null}
- -t 1.01125 -s 0 -d 1 -p cbr -e 210 -c 1 -i 3 -a 1 -x {0.0 1.0 3 ------- null}
h -t 1.01125 -s 0 -d 1 -p cbr -e 210 -c 1 -i 3 -a 1 -x {0.0 1.0 -1 ------- null}
r -t 1.01459 -s 0 -d 1 -p cbr -e 210 -c 1 -i 1 -a 1 -x {0.0 1.0 1 ------- null}
+ -t 1.015 -s 0 -d 1 -p cbr -e 210 -c 1 -i 4 -a 1 -x {0.0 1.0 4 ------- null}
```

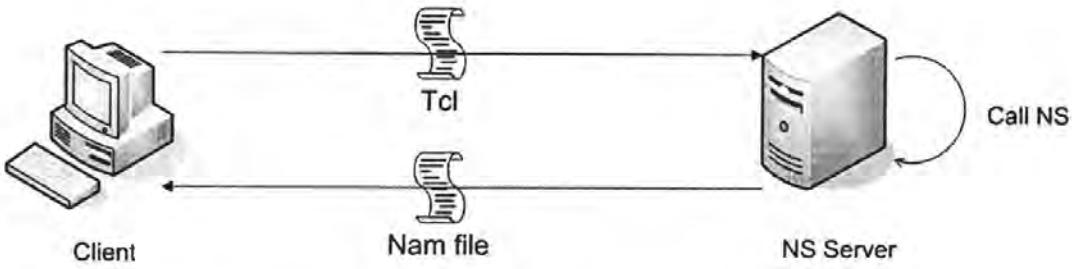
รูปที่ 3.10 ส่วนการทำงานของกลุ่มข้อมูลภายในเครือข่ายจำลอง

3.4.3 ส่วนเซิร์ฟเวอร์ที่ให้บริการ (NS2 Server)

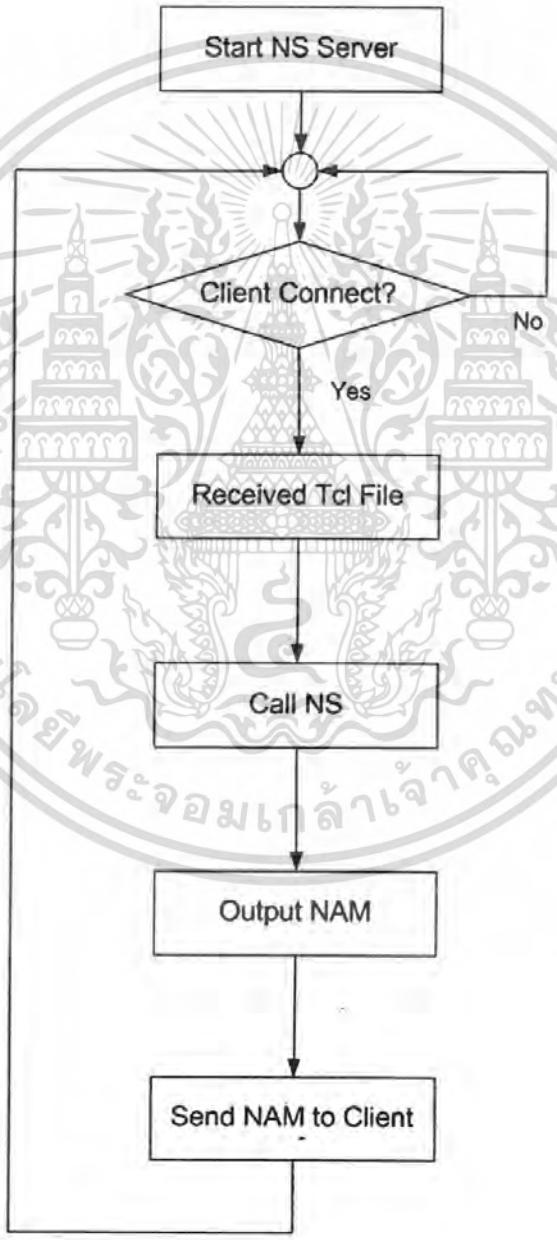
ในส่วนของเซิร์ฟเวอร์ของโปรแกรมเน็ตเวิร์คซิมูชั่นได้มีการออกแบบโดยมีการใช้งานระบบปฏิบัติการลินุกซ์ ซึ่งในโครงงานนี้ใช้ระบบปฏิบัติการยูบันตูเนื่องจากสามารถติดตั้งได้สะดวก โดยทำการติดตั้งโปรแกรมเน็ตเวิร์คซิมูเลชั่น (วิธีการติดตั้งจะอยู่ในส่วนภาคผนวก) มีการติดตั้งโปรแกรมให้บริการเว็บเซิร์ฟเวอร์อาปาเช่ เพื่อสามารถให้บริการเว็บเพจสำหรับการเรียกการทำงานของจาวาเว็บสตาร์ท และใช้ในการเก็บโปรแกรมฟรอนต์เอ็นสำหรับโปรแกรมเน็ตเวิร์คซิมูเลชั่น

ในส่วนฟรอนต์เอ็นจะมีส่วนที่ใช้ในการติดต่อระหว่างเครื่องเซิร์ฟเวอร์ โดยเริ่มแรกเครื่องเซิร์ฟเวอร์จะมีการเปิดช่องทางการเชื่อมต่อเพื่อรอรับการติดต่อจากฝั่งไคลเอนต์ เมื่อส่วนฟรอนต์เอ็นติดต่อไปยังเซิร์ฟเวอร์ ไคลเอนต์จะส่งไฟล์ภาษาทีซีแอลไปยังเซิร์ฟเวอร์ โดยโปรแกรมเน็ตเวิร์คซิมูเลชั่นซึ่งติดตั้งอยู่บนเครื่องเซิร์ฟเวอร์จะประมวลผลไฟล์ภาษาทีซีแอล ซึ่งผลลัพธ์ที่ได้ออกมาคือไฟล์เอ็นเอเอ็ม หลังจากนั้นฝั่งเซิร์ฟเวอร์จะทำการส่งไฟล์เอ็นเอเอ็มกลับไปยังเครื่องของผู้ใช้ซึ่งโปรแกรมฟรอนต์เอ็น จะรับเอาไฟล์เอ็นเอเอ็มและนำไปแสดงผลการทำงาน ซึ่งกระบวนการของฝั่งเซิร์ฟเวอร์จะแสดงดังรูปที่ 3.11 และรูปที่ 3.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 แสดงการรับและส่งไฟล์ระหว่างไคลเอนต์และเซิร์ฟเวอร์

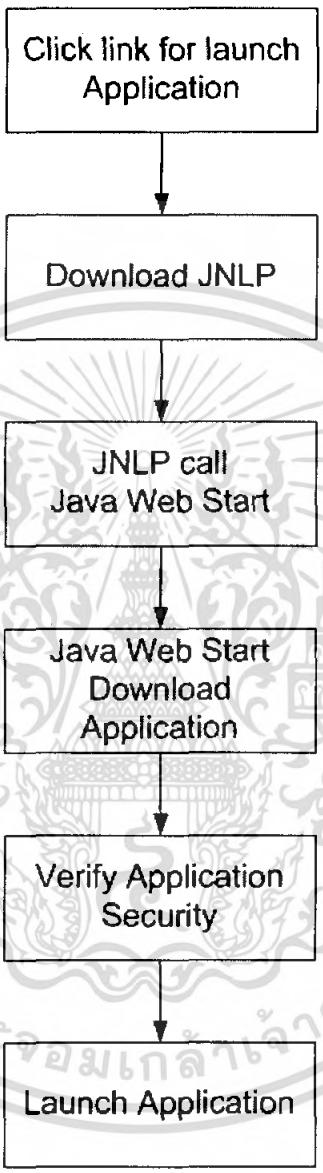


รูปที่ 3.12 แสดงขั้นตอนการทำงานของฝั่งเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.4 กระบวนการเรียกการทำงานของโปรแกรมฟรอนต์เอ็น

การเรียกใช้งานโปรแกรมฟรอนต์เอ็นจะเรียกผ่านทางหน้าเว็บเพจของเซิร์ฟเวอร์โปรแกรมเน็ตเวิร์คซิมูเลชั่น โดยมีการใช้งานจาวาเว็บสตาร์ทช่วยในการเรียกใช้ ซึ่งกระบวนการทำงานสามารถแสดงได้ดังรูปที่ 3.13



รูปที่ 3.13 กระบวนการเรียกการทำงานฟรอนต์เอ็นโดยใช้งานจาวาเว็บสตาร์ท

กระบวนการเรียกใช้งานของฟรอนต์เอ็นจะเริ่มจากผู้ใช้เปิดหน้าเว็บเพจของโปรแกรม จากนั้นเมื่อมีการกดลิงค์ของไฟล์เจเอ็นแอลพี ไฟล์เจเอ็นแอลพีจะเรียกการทำงานของจาวาเว็บสตาร์ท โดยจาวาเว็บสตาร์ทจะดาวน์โหลดจาวาแอปพลิเคชั่นจากเซิร์ฟเวอร์ ซึ่งเมื่อดาวน์โหลดจาวาแอปพลิเคชั่นจะมีการตรวจสอบการรักษาความปลอดภัยของจาวาแอปพลิเคชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่จาวาแอปพลิเคชั่นมีการเข้าใช้งานทรัพยากรของเครื่องไคลเอนต์ จาวาแอปพลิเคชั่นจะต้องมีการทำเครื่องหมาย (Sign) เพื่อให้สามารถเชื่อใจแอปพลิเคชั่นและเข้าใช้งานทรัพยากรของไคลเอนต์ได้ อีกส่วนที่ต้องกำหนดสำหรับการเข้าใช้งานทรัพยากรคือภายในไฟล์เจเอ็นแอลพีซึ่งต้องกำหนดให้เข้าใช้ทรัพยากรบนเครื่องของไคลเอนต์ได้ หลังจากมีการตรวจสอบความปลอดภัยเรียบร้อย จาวาเว็บสตาร์ทจะเริ่มการทำงานของจาวาแอปพลิเคชั่น

การกำหนดการทำงานของจาวาเว็บสตาร์ทจะกำหนดภายในไฟล์เจเอ็นแอลพี ซึ่งลักษณะของไฟล์เจเอ็นแอลพีจะเป็นไฟล์เอ็กแอมแอล (xml) ชนิดหนึ่ง ภายในจะมีการกำหนดการทำงานด่าง ๆ ซึ่งลักษณะของไฟล์เจเอ็นแอลพีมีดังรูปที่ 3.14

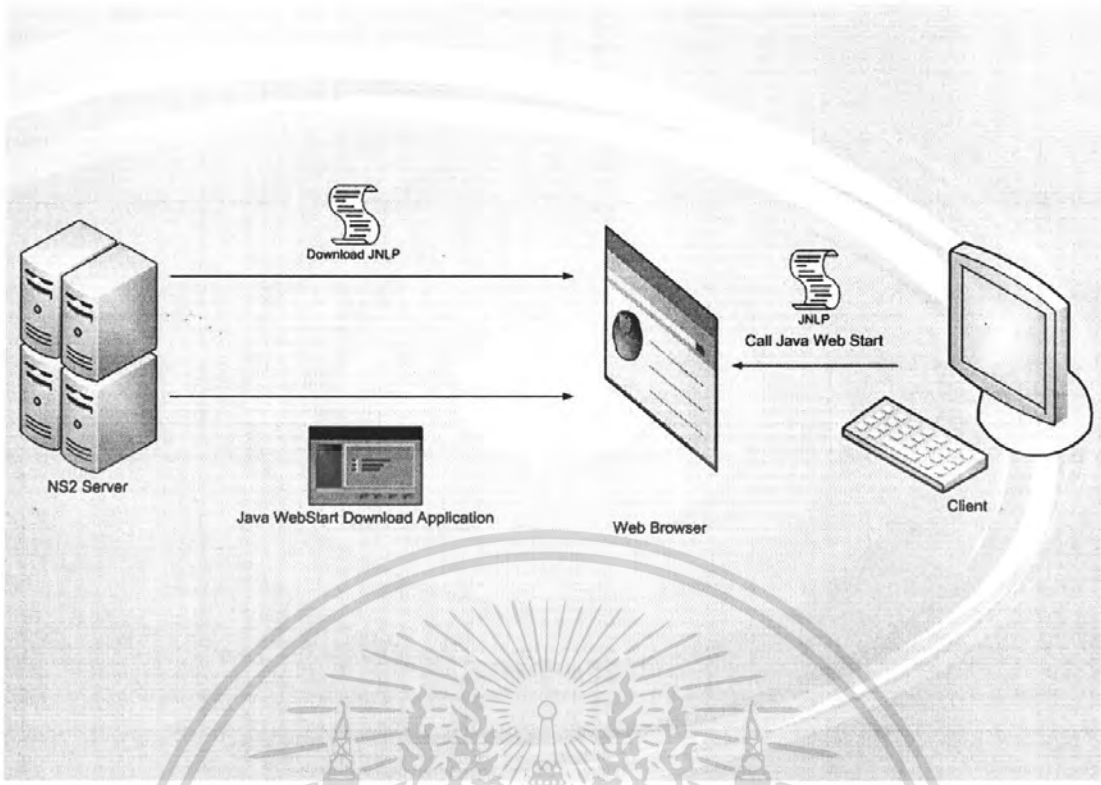
```
<?xml version="1.0" encoding="utf-8"?>
<jnlp spec="0.2 1.0" codebase="http://161.246.5.21/">
<information>
        <title>NSFE</title>
        <vendor>Network Simulation Front End</vendor>
        <homepage href=""/>
        <description>NS</description>
        <description kind="short">NS</description>
        <offline-allowed/>
</information>
<security>
        <all-permissions/>
</security>
<update check = "always" policy = "always"/>
<resources>
        <j2se version="1.3+" href="http://java.sun.com/products/autodl/j2se"/>
        <j2se version="1.3+"/>
        <jar href="NSFE.jar" main="true" download="eager"/>
</resources>
<application-desc main-class="Editor.MainClass"/>
</jnlp>
```

รูปที่ 3.14 ไฟล์เจเอ็นแอลพี

ในส่วนของการกำหนดการเข้าใช้งานทรัพยากรของเครื่องไคลเอนต์จะอยู่ในส่วนการรักษาความปลอดภัย (<security>) ซึ่งจากรูปที่ 3.14 จะเป็นการกำหนดให้สามารถเข้าใช้งานทรัพยากรในเครื่องของไคลเอนต์ได้ (<all-permissions/>)

การทำงานโดยรวมของการเรียกการใช้งานจาวาแอปพลิเคชั่นผ่านจาวาเว็บสตาร์ทสามารถแสดงได้ดังรูปที่ 3.15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 กระบวนการโดยรวมของการเรียกใช้งานฟรอนต์เอ็นผ่านจาวาเว็บสตาร์ท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

4.1 บทนำ

ในบทนี้จะเป็นการทดสอบการใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็น โดยจะแบ่งขั้นตอนการทดสอบทั้งในส่วนของโปรแกรมฟรอนต์เอ็น ส่วนการทำงานของเซิร์ฟเวอร์ การทำงานของเน็ตเวิร์คออนิเมเตอร์ จากนั้นจะเป็นการทดสอบการใช้งานฟังก์ชั่นต่าง ๆ ของโปรแกรมฟรอนต์เอ็นเน็ตเวิร์คซิมูเลชั่น

ในการทดลอบการทำงานจะกำหนดองค์ประกอบที่ใช้ในการทดสอบดังนี้

1. เครื่องเซิร์ฟเวอร์ของโปรแกรมเน็ตเวิร์คซิมูเลชั่น

เซิร์ฟเวอร์โปรแกรมเน็ตเวิร์คซิมูเลชั่นจะเลือกใช้งานระบบปฏิบัติการยูบันตูเวอร์ชั่น 7.04 โดยจะติดตั้งโปรแกรมเน็ตเวิร์คซิมูเลชั่นเวอร์ชั่น 2.31 และมีการติดตั้งโปรแกรมอาปาเช่เพื่อเปิดให้บริการเว็บเซิร์ฟเวอร์เพื่อเรียกใช้งานโปรแกรมเน็ตเวิร์คซิมูเลเตอร์ฟรอนต์เอ็น

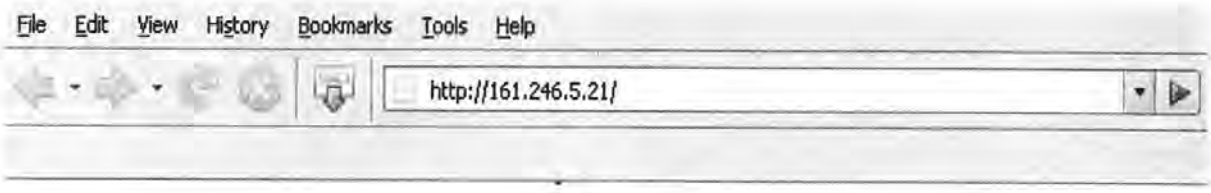
2. เครื่องไคลเอนต์

ไคลเอนต์ที่เรียกใช้งาน จะใช้ระบบปฏิบัติการวินโดว์เอ็กซ์พี และใช้งานเว็บเบราว์เซอร์ไฟล์ฟ็อกซ์ โดยเครื่องของไคลเอนต์ได้ติดตั้งโปรแกรมสภาพแวดล้อมของภาษาจาวา

4.2 การทดลองในส่วนการเรียกใช้งานแอปพลิเคชั่น

ในส่วนนี้จะเป็นการเรียกใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็นผ่านทางเว็บไซต์ โดยการเรียกใช้งานโปรแกรมจะเรียกโดยการกดลิงค์บนเว็บเพจซึ่งเป็นลิงค์ที่จะไปเรียกการทำงานของจาวาเว็บสตาร์ท

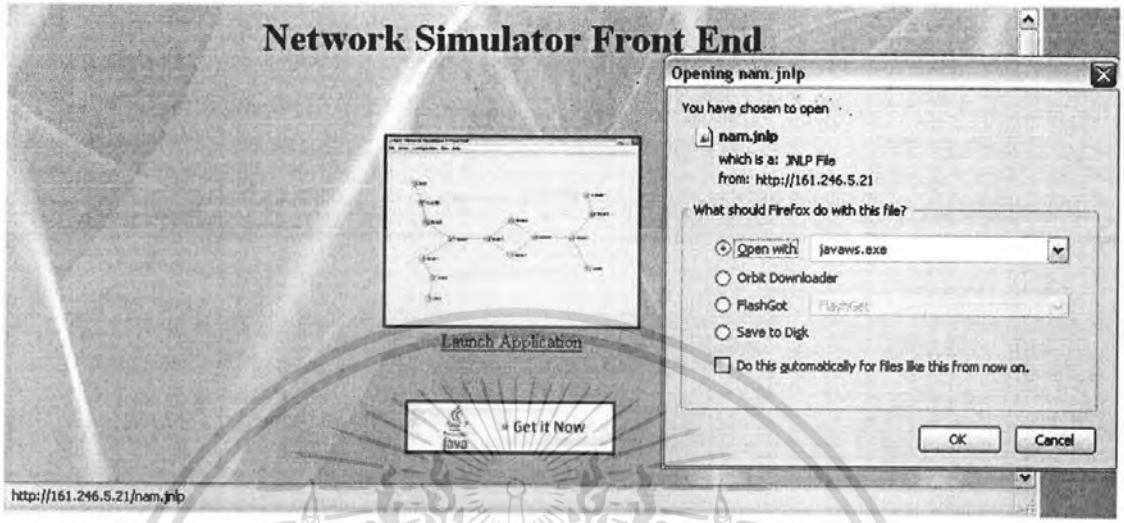
1. เครื่องทางฝั่งไคลเอนต์เข้าใช้งานเว็บไซต์ของโปรแกรมเน็ตเวิร์คซิมูเลเตอร์ฟรอนต์เอ็น ซึ่งในที่นี้จะเรียกไปที่ http://161.246.5.21/ ดังรูปที่ 4.1



รูปที่ 4.1 การเรียกใช้งานเว็บไซต์ของโปรแกรมฟรอนต์เอ็น

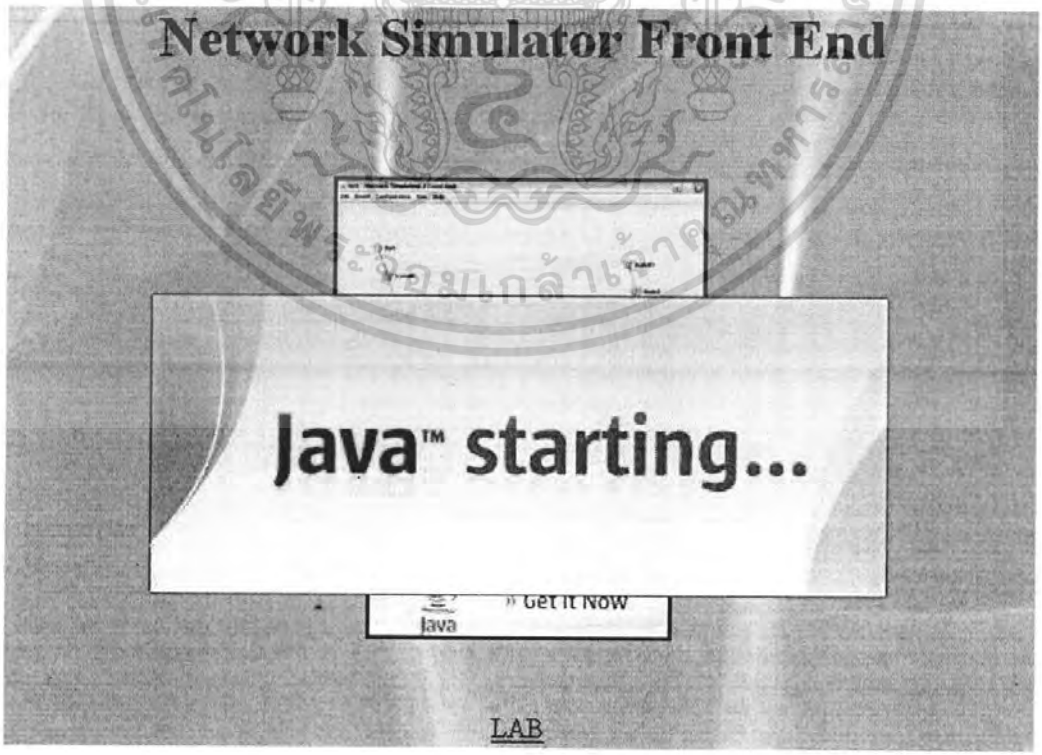
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เว็บไซต์ของโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็นสามารถเรียกการใช้งานโปรแกรมฟรอนต์เอ็นได้โดยการกดลิงค์บนเว็บเพจ ซึ่งเมื่อกดแล้วจะเป็นการดาวน์โหลดเจเอ็นแอลพีเพื่อใช้ในการเรียกการทำงานของจาวาเว็บสตาร์ทดังรูปที่ 4.2



รูปที่ 4.2 การดาวน์โหลดเจเอ็นแอลพี

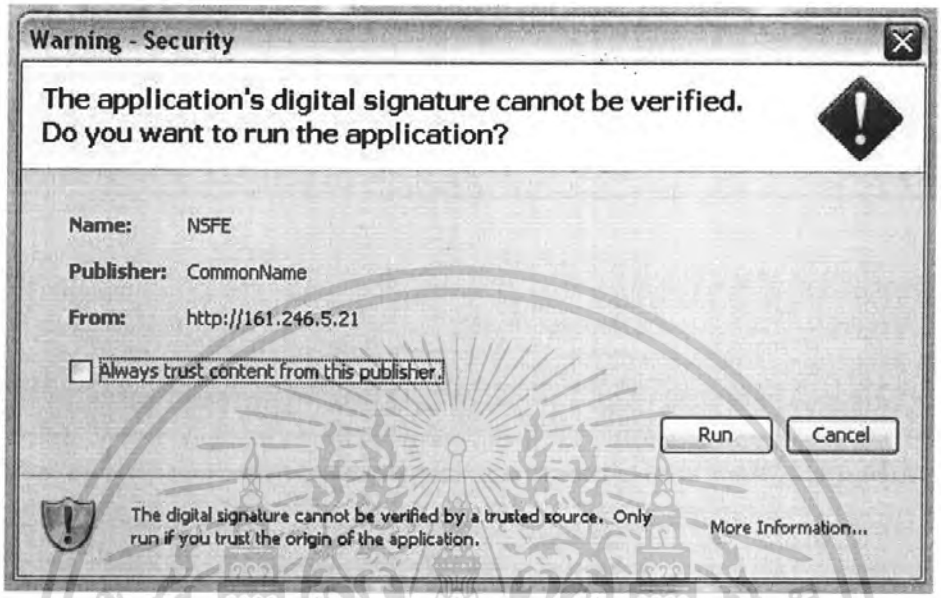
3. หลังจากดาวน์โหลดไฟล์เจเอ็นแอลพีแล้ว โปรแกรมจาวาเว็บสตาร์ทจะเริ่มทำงานดังรูปที่ 4.3



รูปที่ 4.3 การเรียกใช้งานจาวาเว็บสตาร์ท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เมื่อโปรแกรมจาวาเว็บสตาร์ทเริ่มทำงาน จะทำการดาวน์โหลดและตรวจสอบแอปพลิเคชั่นที่ได้เรียกมาโดยจะมีการตรวจสอบความปลอดภัยของแอปพลิเคชั่น และสอบถามความต้องการใช้งานแอปพลิเคชั่นดังรูปที่ 4.4 ซึ่งในกรณีที่มีการเข้าใช้งานทรัพยากรของเครื่องไคลเอนต์จะต้องมีตรวจสอบความปลอดภัยก่อนจะใช้งานแอปพลิเคชั่น



รูปที่ 4.4 การแจ้งเตือนก่อนแอปพลิเคชั่นจะทำงาน

5. เมื่อเรียกการทำงานของโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็น โปรแกรมจะถูกเรียกขึ้นมาทำงานดังรูปที่ 4.5 โดยโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็นจะถูกดาวน์โหลดและเก็บไว้ภายในแคชของจาวาเว็บสตาร์ท

รูปที่ 4.5 หน้าต่างของโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

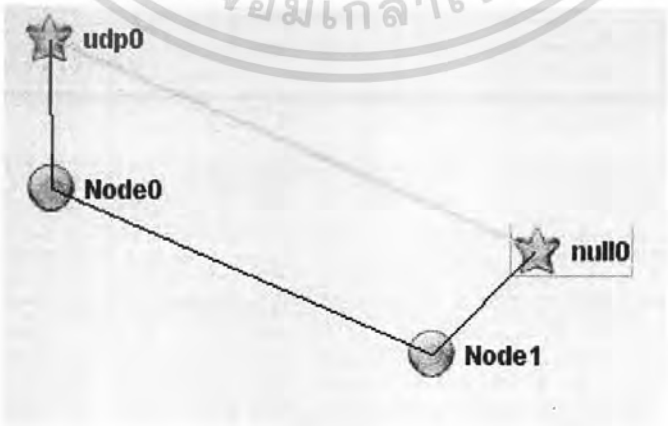
4.3 การทดลองในส่วนการสร้างเครือข่ายจำลอง

ในส่วนนี้จะเป็นการทดลองการใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็น โดยจะทำการสร้างเครือข่ายจำลองง่าย ๆ ซึ่งประกอบไปด้วยโหนด จำนวน 2 โหนด ลิงค์ซึ่งเชื่อมต่อระหว่างโหนด โพรโตคอลเอเจนต์ในชั้นทรานสปอร์ตซึ่งใช้งานโพรโตคอลยูดีพี และแอปพลิเคชั่นในการจำลองเครือข่ายจะใช้งานซีบีอาร์

1. เริ่มต้นสร้างโหนดในเครือข่ายจำลอง โดยสร้างโหนดจำนวนสองโหนด และเชื่อมต่อโหนดทั้งสองด้วยลิงค์ ดังรูปที่ 4.6

รูปที่ 4.6 การสร้างโหนดและการเชื่อมต่อโหนดด้วยลิงค์

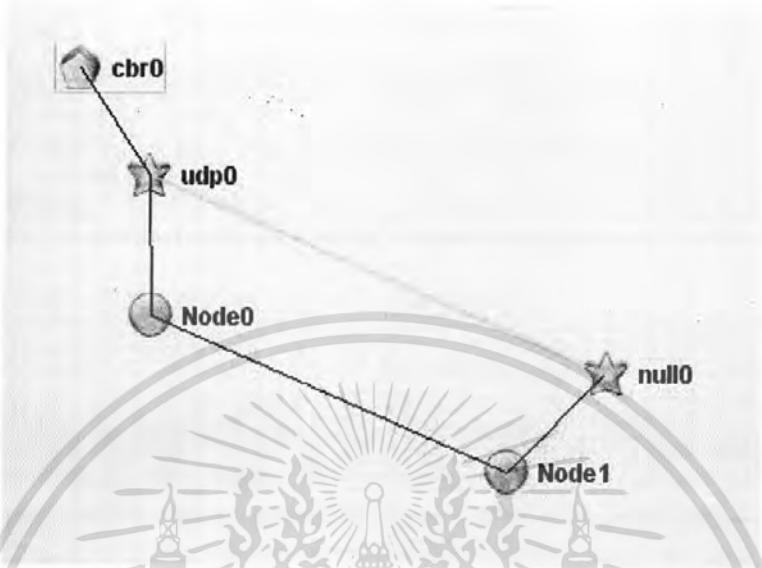
2. สร้างโพรโตคอลยูดีพี และ Null เพื่อใช้เป็นโพรโตคอลในชั้นทรานสปอร์ต จากนั้นกำหนดให้แต่ละโหนดทำงานโพรโตคอลที่สร้างขึ้นมาโดยเชื่อมโพรโตคอลเข้ากับโหนด และเชื่อมต่อโพรโตคอลทั้งสองเข้าด้วยกันดังรูปที่ 4.7



รูปที่ 4.7 การสร้างโพรโตคอลและการเชื่อมต่อเข้าด้วยกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

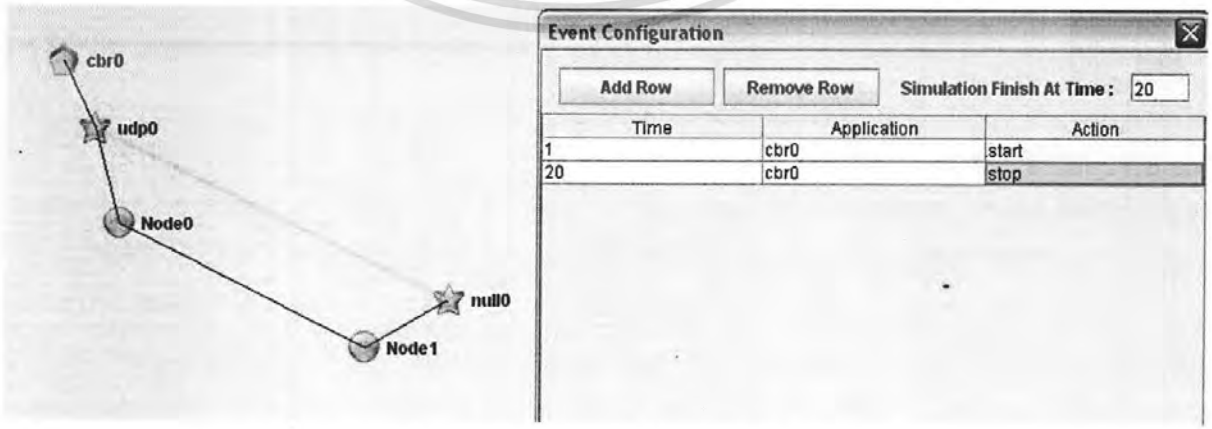
3. สร้างแอปพลิเคชั่นซีบีอาร์ซึ่งกำหนดให้ทำงานในการเครือข่ายที่ได้จำลอง โดยใช้งานโพรโตคอลยูดีพี ดังรูปที่ 4.8 ซึ่งเป็นการใช้งานซีบีอาร์ส่งข้อมูลจากโหนด Node0 ไปยังโหนด Node1



รูปที่ 4.8 การสร้างแอปพลิเคชั่นในเครือข่ายจำลอง

4.4 การทดลองในส่วนการกำหนดตารางเหตุการณ์

การทดสองส่วนนี้เป็นการทดลองกำหนดตารางเวลาของการจำลองเครือข่าย ซึ่งจะในเครือข่ายจะมีการสร้างโหนดจำนวน 2 โหนด มีการใช้งานโพรโตคอลชั้นทรานสปอร์ตชนิดยูดีพี และใช้งานแอปพลิเคชั่นซีบีอาร์ โดยตารางเวลาจะกำหนดได้ดังรูปที่ 4.9 ซึ่งในการทดลองจะกำหนดเวลาการจำลองทั้งหมดไว้ที่ 20 วินาที โดยแอปพลิเคชั่นซีบีอาร์จะเริ่มทำงานที่เวลา 1 วินาทีและจะสิ้นสุดการทำงานที่เวลา 20 วินาที



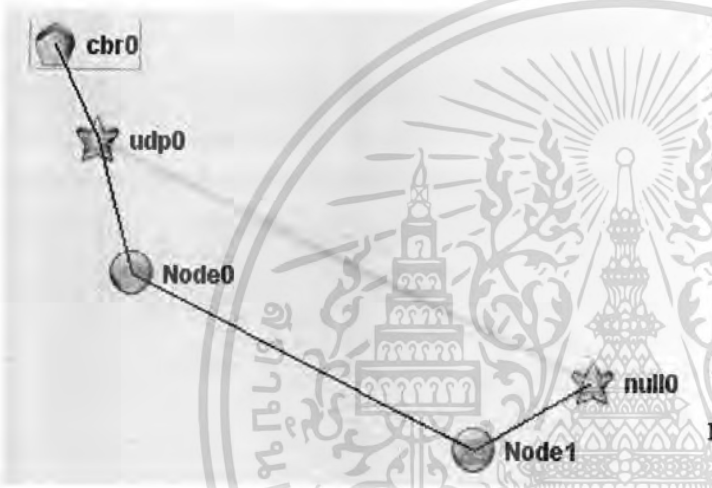
รูปที่ 4.9 การกำหนดตารางเหตุการณ์ของเครือข่ายจำลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 การทดลองในส่วนการสร้างไฟล์ทีซีแอลและการเชื่อมต่อกับเซิร์ฟเวอร์

ในส่วนนี้เป็นการทดลองการเชื่อมต่อเข้ากับเซิร์ฟเวอร์ซึ่งติดตั้งโปรแกรมเน็ตเวิร์คซิมูเลชั่นและคอยให้บริการไคลเอนต์

1. โปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็นจะแปลงเครือข่ายที่สร้างจากโปรแกรมฟรอนต์เอ็นให้เป็นไฟล์ภาษาทีซีแอลดังรูปที่ 4.10 จากนั้นจะส่งผลลัพธ์จากการประมวลผลที่ได้ซึ่งเป็นไฟล์เอ็นเอเอ็ม กลับคืนยังไคลเอนต์เพื่อแสดงผลต่อไป




```
set ns [new Simulator]

$ns color 0 black
$ns color 1 black

set f [open out.tr w]
$ns trace-all $f
set nf [open 161.246.22.55.nam w]
$ns namtrace-all $nf
proc finish {} {
    global ns nf f
    $ns flush-trace
    puts "Simulation completed."
    close $f
    exit 0
}

Node set multiPath_ 1

set Node0 [$ns node]
$Node0 set X_ 141
$Node0 set Y_ 179
$Node0 set Z_ 0.0
$ns at 0.0 "$Node0 label Node0"

set Node1 [$ns node]
$Node1 set X_ 318
$Node1 set Y_ 256
$Node1 set Z_ 0.0
$ns at 0.0 "$Node1 label Node1"
```

รูปที่ 4.10 การแปลงเครือข่ายที่จำลองเป็นภาษาทีซีแอล

2. ส่วนฟรอนต์เอ็นจะติดต่อไปยังเซิร์ฟเวอร์เพื่อส่งไฟล์ภาษาทีซีแอลที่สร้างได้จากเครือข่ายประมวลผลกับโปรแกรมเน็ตเวิร์คซิมูเลชั่นซึ่งติดตั้งบนเซิร์ฟเวอร์ โดยเซิร์ฟเวอร์จะรอการติดต่อจากไคลเอนต์ตลอดเวลา ซึ่งในฝั่งของเซิร์ฟเวอร์จะแสดงการทำงานดังรูปที่ 4.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Server Waiting for Client Connect
New Connection Accepted
File Name :161.246.22.55.tcl
Save Complete
Starting
Call NS Finish
Send Complete
Connection close by Client
```

รูปที่ 4.11 แสดงการทำงานของเซิร์ฟเวอร์ซึ่งคอยให้บริการ

3. ผลลัพธ์ที่ได้จากการประมวลผลของโปรแกรมเน็ตเวิร์คซิมูเลชั่น คือไฟล์เอ็นเอเอ็มจะถูกส่งไปยังโปรแกรมฟรอนต์เอ็น ซึ่งลักษณะของไฟล์เอ็นเอเอ็มมีลักษณะดังรูปที่ 4.12

```
V -t * -v 1.0a5 -a 0
W -t * -x 318 -y 256
A -t * -n 1 -p 0 -o 0xffffffff -c 31 -a 1
A -t * -h 1 -m 2147483647 -s 0
c -t * -i 0 -n black
c -t * -i 1 -n black
n -t * -a 0 -s 0 -S UP -v circle -c black -i black -x 141 -y 179 -Z 0.0
n -t * -a 1 -s 1 -S UP -v circle -c black -i black -x 318 -y 256 -Z 0.0
l -t * -s 1 -d 0 -S UP -r 2000000 -D 0.01 -c black
q -t * -s 0 -d 1 -a 0.5
q -t * -s 1 -d 0 -a 0.5
n -t 0 -s 0 -S DLABEL -l "Node0" -L ""
n -t 0 -s 1 -S DLABEL -l "Node1" -L ""
+ -t 1 -s 0 -d 1 -p cbr -e 210 -c 1 -i 0 -a 1 -x (0.0 1.0 0 ------- null)
- -t 1 -s 0 -d 1 -p cbr -e 210 -c 1 -i 0 -a 1 -x (0.0 1.0 0 ------- null)
h -t 1 -s 0 -d 1 -p cbr -e 210 -c 1 -i 0 -a 1 -x (0.0 1.0 -1 ------- null)
+ -t 1.00375 -s 0 -d 1 -p cbr -e 210 -c 1 -i 1 -a 1 -x (0.0 1.0 1 ------- null)
- -t 1.00375 -s 0 -d 1 -p cbr -e 210 -c 1 -i 1 -a 1 -x (0.0 1.0 1 ------- null)
h -t 1.00375 -s 0 -d 1 -p cbr -e 210 -c 1 -i 1 -a 1 -x (0.0 1.0 -1 ------- null)
+ -t 1.0075 -s 0 -d 1 -p cbr -e 210 -c 1 -i 2 -a 1 -x (0.0 1.0 2 ------- null)
- -t 1.0075 -s 0 -d 1 -p cbr -e 210 -c 1 -i 2 -a 1 -x (0.0 1.0 2 ------- null)
h -t 1.0075 -s 0 -d 1 -p cbr -e 210 -c 1 -i 2 -a 1 -x (0.0 1.0 -1 ------- null)
r -t 1.01084 -s 0 -d 1 -p cbr -e 210 -c 1 -i 0 -a 1 -x (0.0 1.0 0 ------- null)
+ -t 1.01125 -s 0 -d 1 -p cbr -e 210 -c 1 -i 3 -a 1 -x (0.0 1.0 3 ------- null)
- -t 1.01125 -s 0 -d 1 -p cbr -e 210 -c 1 -i 3 -a 1 -x (0.0 1.0 3 ------- null)
h -t 1.01125 -s 0 -d 1 -p cbr -e 210 -c 1 -i 3 -a 1 -x (0.0 1.0 -1 ------- null)
r -t 1.01459 -s 0 -d 1 -p cbr -e 210 -c 1 -i 1 -a 1 -x (0.0 1.0 1 ------- null)
+ -t 1.015 -s 0 -d 1 -p cbr -e 210 -c 1 -i 4 -a 1 -x (0.0 1.0 4 ------- null)
```

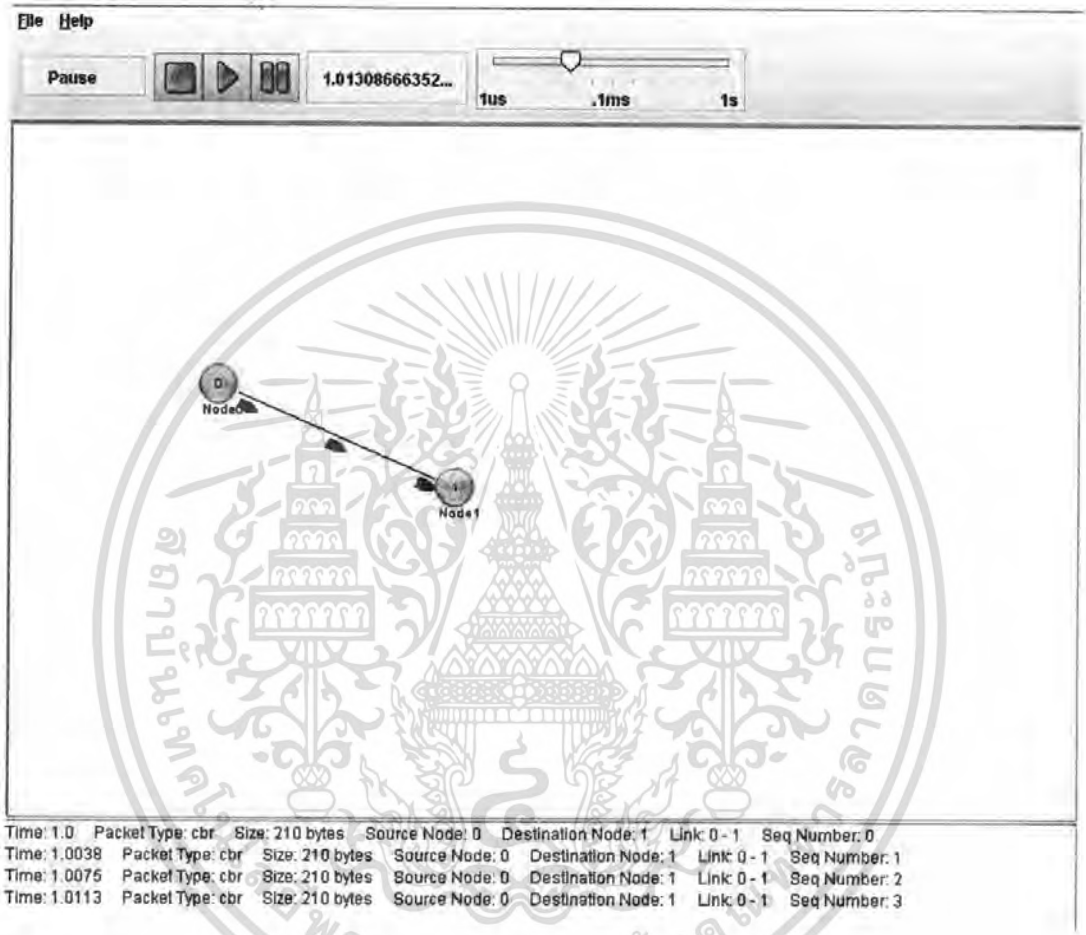
รูปที่ 4.12 ผลลัพธ์ที่ได้จากการประมวลผลไฟล์ภาษาทีซีแอล

ไฟล์เอ็นเอเอ็มที่ได้จะนำไปใช้ในการแสดงผลการทำงานของเครือข่ายที่ได้จำลองขึ้นมาด้วยโปรแกรมเน็ตเวิร์คอนิเมเตอร์ (Network Animator)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 การทดลองในส่วนการแสดงผลภาพเคลื่อนไหว

ผลลัพธ์ที่ได้จากโปรแกรมเน็ตเวิร์คซิมูเลชั่น คือไฟล์เอ็นเอเอ็มซึ่งเก็บข้อมูลการทำงานของกลุ่มข้อมูล ซึ่งสามารถแสดงผลเป็นภาพเคลื่อนไหวได้ จะใช้งานโปรแกรมเน็ตเวิร์คอนิเมเตอร์ในการแสดงผล โดยจะแสดงการทำงานของกลุ่มข้อมูลและแสดงข้อมูลของกลุ่มข้อมูลแต่ละอัน ดังรูปที่ 4.13



รูปที่ 4.13 การแสดงผลภาพเคลื่อนไหวด้วยโปรแกรมเน็ตเวิร์คอนิเมเตอร์

4.7 การทดลองการใช้งานฟังก์ชั่นของโปรแกรมฟรอนต์เอ็น

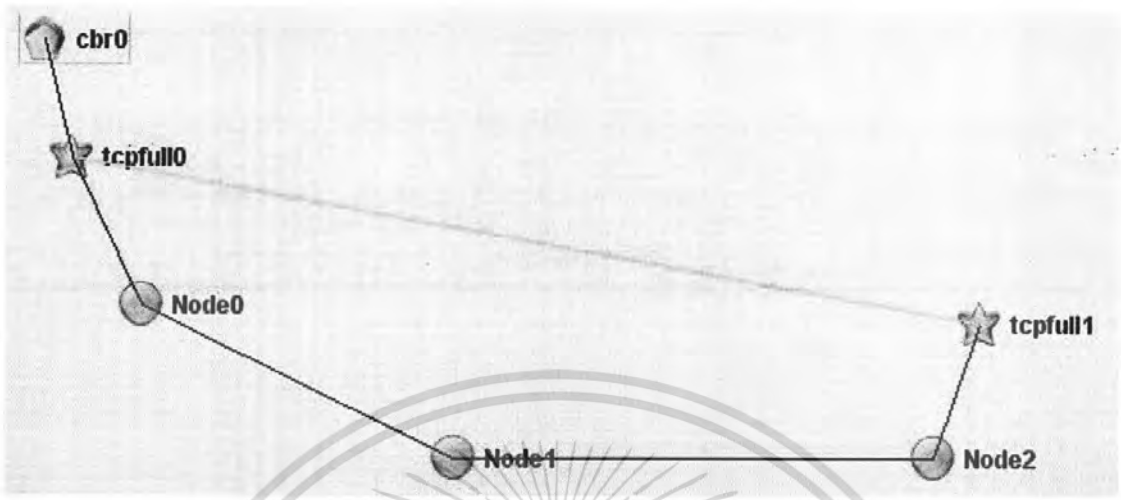
โปรแกรมฟรอนต์เอ็นสามารถจำลองการทำงานได้หลายอย่าง โดยใช้งานฟังก์ชั่นของโปรแกรมเน็ตเวิร์คซิมูเลชั่น เพื่อใช้การกำหนดการจำลองได้

4.7.1. การทดลองการทำงานของโพรโตคอลทีซีพี

โปรแกรมเน็ตเวิร์คซิมูเลชั่นมีโพรโตคอลทีซีพีอยู่หลายชนิด อย่างทีซีพีตัวรับ ทีซีพีตัวส่ง ซึ่งแต่ละชนิดจะมีการทำงานที่แตกต่างกันตามแต่ละชนิด

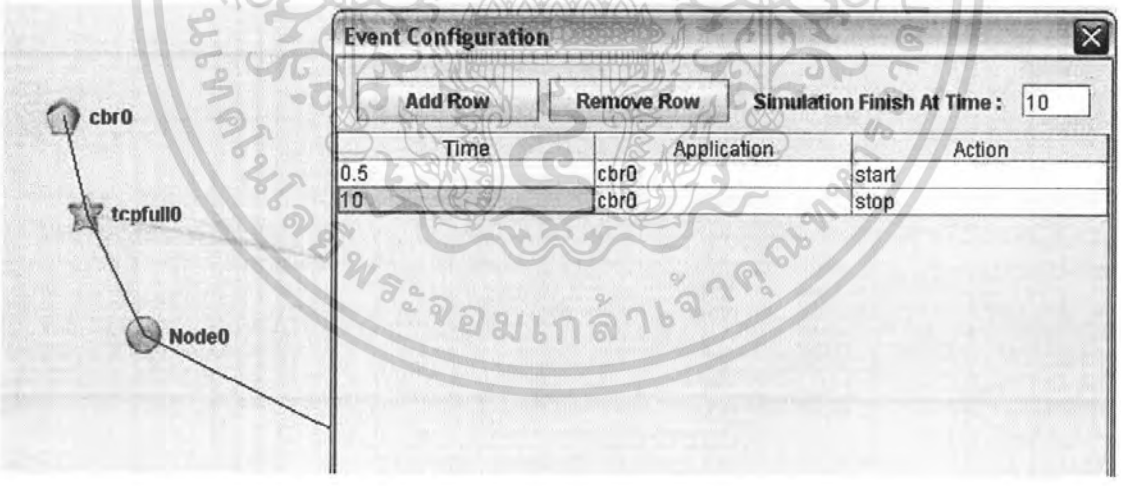
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. กำหนดการทำงานโดยสร้างเครือข่ายดังรูปที่ 4.14 ซึ่งเป็นการสร้างโหนด 4 โหนด และมีการกำหนดโพรโตคอลทีซีพีทำงานภายในเครือข่ายที่จำลอง



รูปที่ 4.14 การทดลองการใช้งานโพรโตคอลทีซีพี

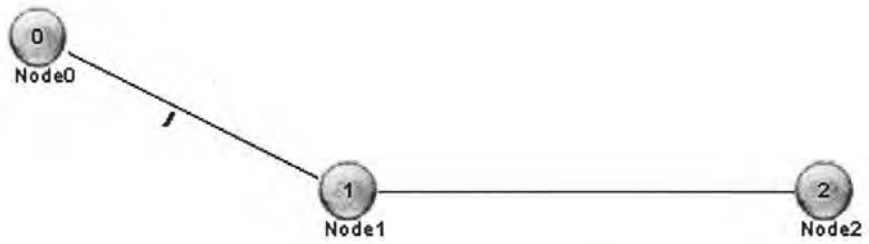
2. กำหนดตารางเวลาการทำงานของเครือข่ายดังรูปที่ 4.15 โดยจะกำหนดเวลาในการจำลองการทำงานเป็นระยะเวลา 10 วินาที และให้แอปพลิเคชั่นซีบีอาร์เริ่มทำงานที่เวลา 0.5 วินาที



รูปที่ 4.15 การกำหนดการทำงานของเครือข่ายโพรโตคอลทีซีพี

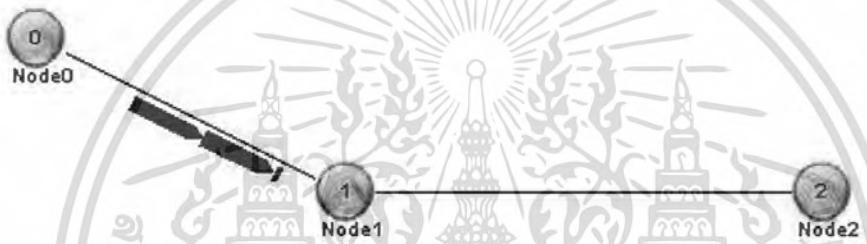
3. เมื่อกำหนดการทำงานเรียบร้อย จึงเริ่มการจำลองการทำงานของโพรโตคอลทีซีพี ซึ่งผลที่ได้จะแสดงดังรูปที่ 4.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 แสดงผลการทำงานของโพรโตคอลทีซีพีในส่วนสร้างการเชื่อมต่อ

ในการเริ่มต้นการเชื่อมต่อของโพรโตคอลทีซีพีจะเริ่มจากการสร้างการเชื่อมต่อด้วยวิธี 3 เวย์แฮนด์เชก เมื่อสร้างการเชื่อมต่อเสร็จแล้วก็เริ่มทำการส่งข้อมูล ซึ่งแสดงได้ดังรูปที่ 4.17

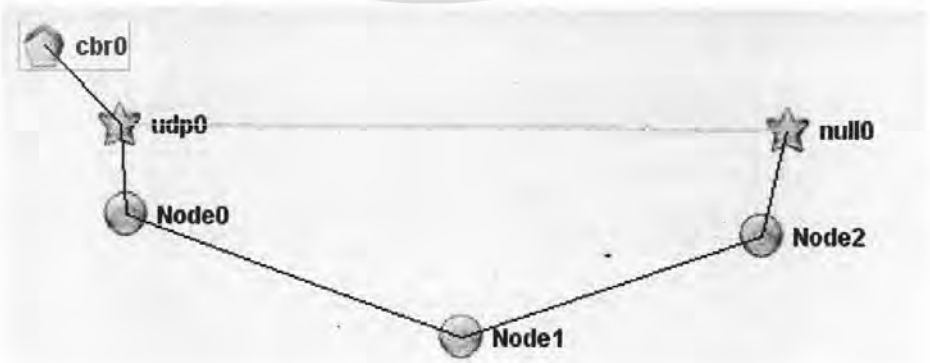


รูปที่ 4.17 แสดงผลการทำงานของโพรโตคอลทีซีพีในส่วนการรรับส่งข้อมูล

4.7.2 การทดลองการทำงานของโพรโตคอลยูดีพี

โปรแกรมเน็ตเวิร์คซิมูเลชั่นสามารถจำลองการทำงานของโพรโตคอลยูดีพี โดยจะเชื่อมต่อเข้ากับ Null ซึ่งทดลองได้ดังนี้

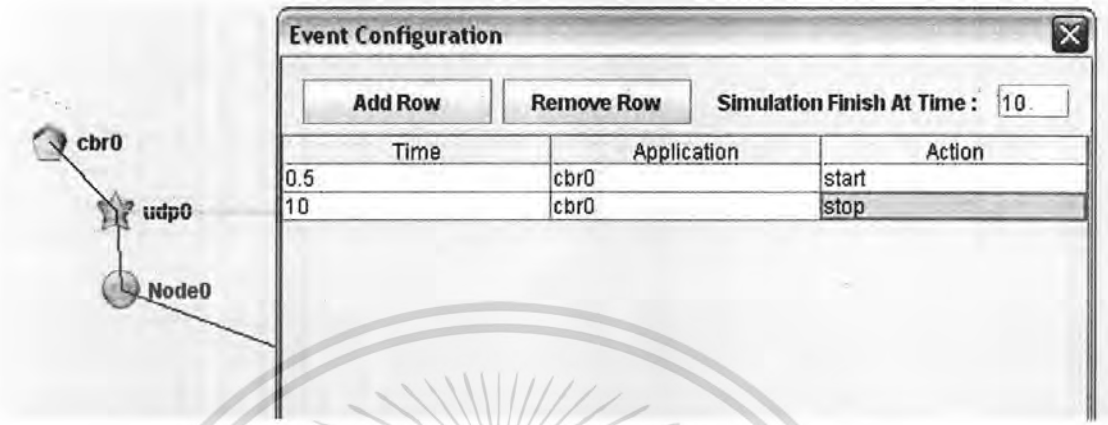
1. สร้างเครือข่ายโดยกำหนดให้มีโหนดจำนวน 3 โหนด ทำงานภายในเครือข่าย และมีการใช้งานโพรโตคอลยูดีพี และ Null ดังรูปที่ 4.18



รูปที่ 4.18 การทดลองการใช้งานโพรโตคอลยูดีพี

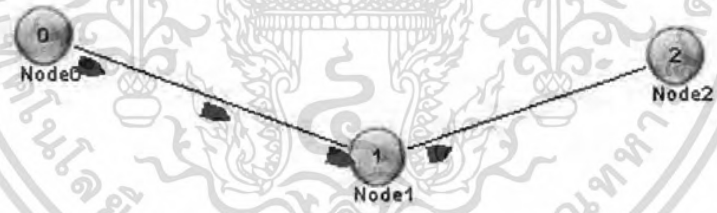
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. กำหนดตารางเวลาของเครือข่ายทดลองการทำงานโพรโตคอลยูดีพีดังรูปที่ 4.19 ซึ่งเป็นการกำหนดระยะเวลาการจำลองเป็นเวลา 10 วินาที โดยให้โพรโตคอลยูดีพีเริ่มการทำงานที่เวลา 0.5 วินาที และหยุดการทำงานที่เวลา 10 วินาที



รูปที่ 4.19 การกำหนดการทำงานของเครือข่ายโพรโตคอลยูดีพี

3. เริ่มต้นการจำลองการทำงานของเครือข่าย โดยโพรโตคอลยูดีพีจะไม่มีการสร้างการเชื่อมต่อก่อนที่จะส่งข้อมูลซึ่งแสดงดังรูปที่ 4.20



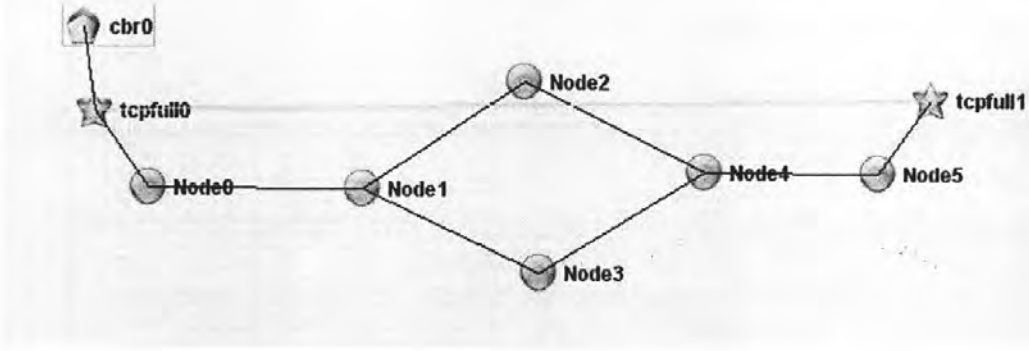
รูปที่ 4.20 แสดงผลการทำงานของโพรโตคอลยูดีพี

4.7.3 การทดลองการกำหนดค่าคอสของลิงค์และโพรโตคอลเลือกเส้นทาง

โปรแกรมเน็ตเวิร์คซิมูเลชั่นสามารถกำหนดค่าคอสของลิงค์ได้ โดยค่าคอสของลิงค์จะเป็นค่าที่ใช้ในการตัดสินใจเลือกเส้นทางสำหรับการรับส่งกลุ่มข้อมูล

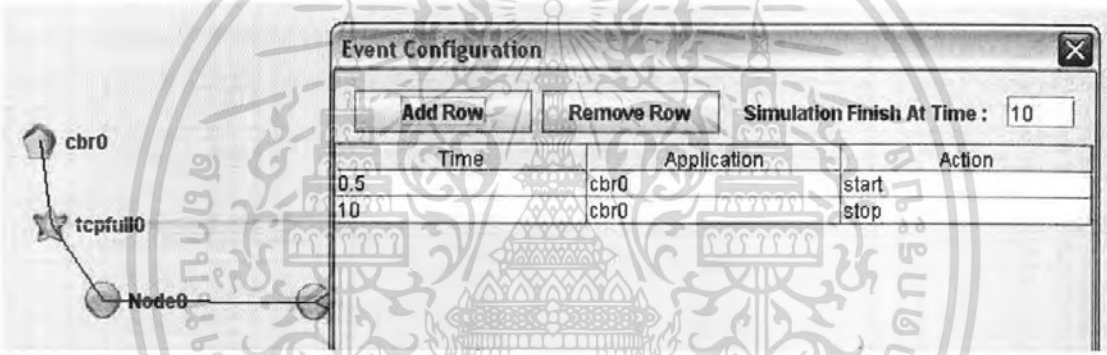
1. สร้างเครือข่ายเพื่อใช้ในการทดลองการกำหนดค่าคอสของลิงค์โดยกำหนดให้มีโหนดในการทดลองจำนวน 6 โหนด โดยให้ปลายเส้นทางของเครือข่ายใช้งานโพรโตคอลทีซีพีดังรูปที่ 4.21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



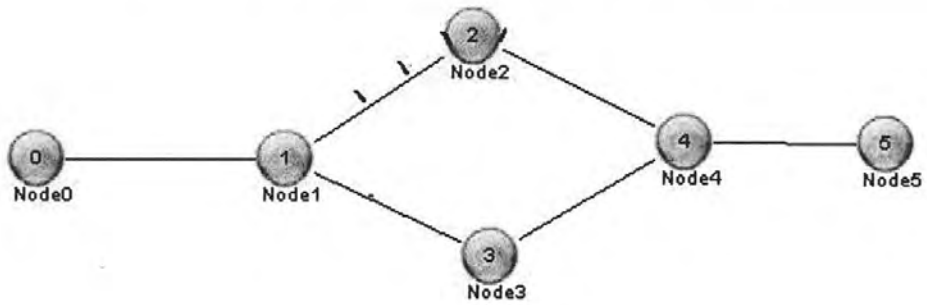
รูปที่ 4.21 เครือข่ายสำหรับทดลองการกำหนดค่าลิงค์คอส

2. กำหนดตารางเวลาของเครือข่ายในการทดลองดังรูปที่ 4.22



รูปที่ 4.22 การกำหนดการทำงานของเครือข่ายทดลองค่าลิงค์คอส

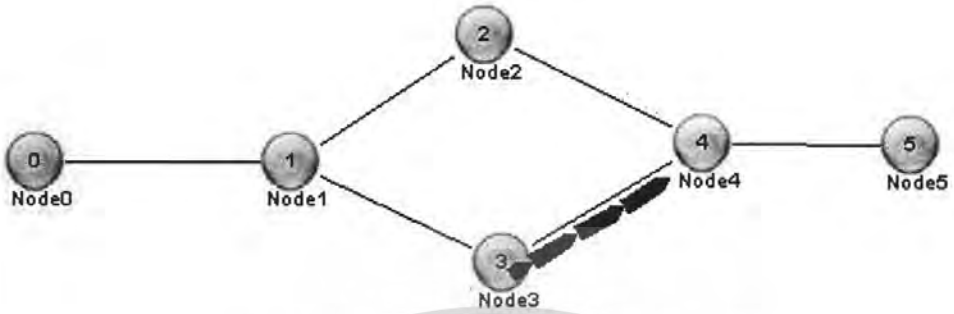
3. โดยปกติแล้วค่าคอสของลิงค์ในโปรแกรมเน็ตเวิร์คซิมูเลชั่นมีค่าเป็น 1 ซึ่งในการทดลองไม่ได้มีการกำหนดโพรโตคอลเลือกเส้นทาง ทำให้การเดินทางของกลุ่มข้อมูลจะเลือกเส้นทางเพียงเส้นทางเดียวในการรับส่งข้อมูล ดังรูปที่ 4.23



รูปที่ 4.23 การเลือกเดินทางของกลุ่มข้อมูลในการกำหนดค่าลิงค์คอส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เมื่อมีการเปลี่ยนแปลงค่าลิงค์คอสของลิงค์ระหว่างโหนด 2 และ โหนด 4 โดยเปลี่ยนแปลงค่าลิงค์คอสจาก 1 ให้มีค่าเป็น 2 จะเกิดการเปลี่ยนแปลงเส้นทางในการรับส่งข้อมูลของเครือข่าย ดังรูปที่ 4.24



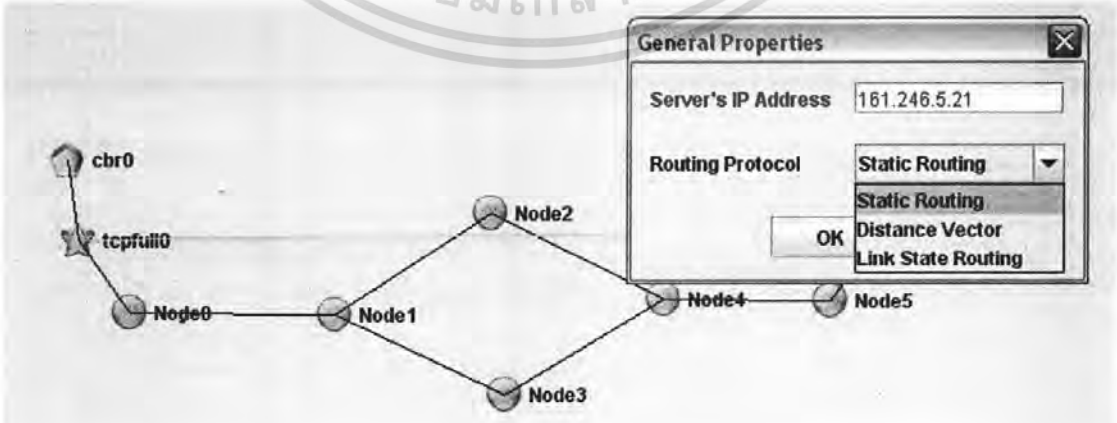
รูปที่ 4.24 การเปลี่ยนแปลงเส้นทางการรับส่งข้อมูลเมื่อมีการเปลี่ยนแปลงค่าลิงค์คอส

4.7.4 การทดลองการทำงานของโพรโตคอลเลือกเส้นทาง

โปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็นสามารถจำลองการทำงานของโพรโตคอลเลือกเส้นทางได้ทั้งชนิด Static , Distance Vector และ Link State ซึ่งแต่ละชนิดจะมีการทำงานที่แตกต่างกัน

1. โพรโตคอลเลือกเส้นทางชนิด static จะเลือกใช้งานเส้นทางเส้นเดิมตลอดในกรณีที่เส้นทางแต่ละเส้นมีค่าคอสเท่ากัน ซึ่งโดยปกติโปรแกรมเน็ตเวิร์คซิมูเลชั่นจะใช้งานชนิด static

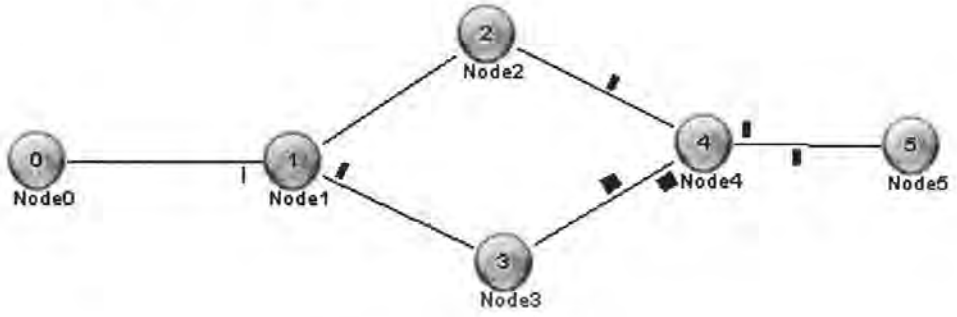
2. โพรโตคอลเลือกเส้นทางชนิด Distance Vector และ โพรโตคอลเลือกเส้นทางชนิด Link State เป็นโพรโตคอลเลือกเส้นทางชนิดไดนามิกที่มีในโปรแกรมเน็ตเวิร์คซิมูเลเตอร์ ซึ่งสามารถที่จะเลือกเส้นทางในการรับส่งกลุ่มข้อมูลได้อัตโนมัติ ซึ่งการกำหนดสามารถทำได้ดังรูปที่ 4.25



รูปที่ 4.25 โพรโตคอลเลือกเส้นทางในโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็น

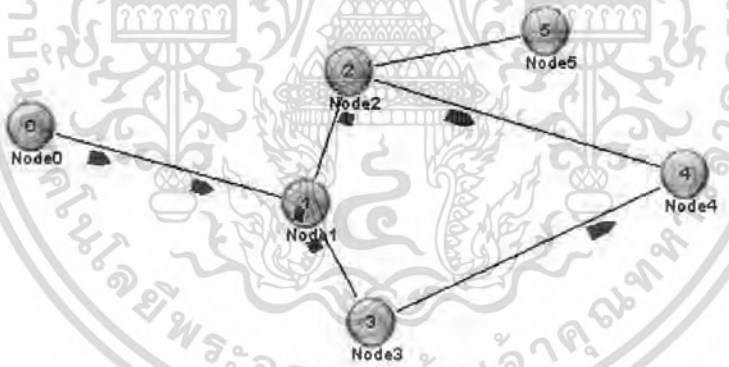
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การทำงานของโพรโตคอลเลือกเส้นทาง จะมีการรับส่งกลุ่มข้อมูลเพื่อเรียนรู้เส้นทางสำหรับรับส่งข้อมูล ซึ่งแสดงได้รูปที่ 4.26



รูปที่ 4.26 แสดงการแลกเปลี่ยนเส้นทางของโพรโตคอลเลือกเส้นทาง

4. ในกรณีที่มีเส้นทางให้เลือกมากกว่าหนึ่งเส้นทางซึ่งมีค่าคอส (Cost) เท่ากัน ที่สามารถไปยังโหนดปลายทางได้ โพรโตคอลเลือกเส้นทางชนิดไดนามิก อย่าง Distance Vector และ Link State จะเลือกใช้งานเส้นทางทุกเส้นโดยจะสลับกันใช้งาน ดังรูปที่ 4.27



รูปที่ 4.27 แสดงการรับส่งกลุ่มข้อมูลกรณีมีเส้นทางมากกว่าหนึ่งเส้นทางไปยังปลายทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และสรุป

5.1 สรุปผลการพัฒนา

โครงงานนี้เป็นโครงงานใหม่ที่ได้จัดทำขึ้นมา เพื่อให้ผู้ที่ใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่นอยู่ก่อนแล้วหรือผู้ที่เริ่มต้นใช้งานสามารถใช้งานได้ง่ายจากเดิมที่ต้องพิมพ์คำสั่งผ่านโปรแกรมแปลคำสั่งภาษาโอทีซีแอลกระทำผ่านหน้าต่างผู้ใช้งาน การติดตั้งและเรียกใช้งานทำได้ง่ายโดยมีเซิร์ฟเวอร์โปรแกรมเน็ตเวิร์คซิมูเลชั่นเป็นผู้ให้บริการ เนื่องจากเดิมการติดตั้งใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่นนั้นมีการติดตั้งที่ค่อนข้างยุ่งยาก ต้องใช้ไลบราลี่หลายตัวซึ่งไม่ได้ระบุไว้ และตัวโปรแกรมเน็ตเวิร์คซิมูเลชั่นออกแบบมาเพื่อใช้งานในระบบปฏิบัติการลินุกซ์ ดังนั้นผู้ใช้งานที่ต้องการใช้งานในระบบปฏิบัติการวินโดว์ต้องใช้โปรแกรมจำลองสภาพแวดล้อมระบบปฏิบัติการลินุกซ์ขึ้นมาทำให้เปลืองทรัพยากรระบบและเกิดความยุ่งยากไม่สะดวกต่อการใช้งาน โดยตัวโปรแกรมเป็นเว็บแอปพลิเคชั่นซึ่งสามารถเรียกใช้งานผ่านเว็บเบราว์เซอร์โดยใช้เทคโนโลยีจาวาเว็บสตาร์ท ทำให้โปรแกรมไม่ต้องเขียนเป็นจาวาแอ็บเพล็ดจึงมีข้อจำกัดน้อยกว่า อีกทั้งผู้ใช้ยังสามารถใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่นได้โดยไม่ขึ้นกับระบบปฏิบัติการ เนื่องจากใช้งานภาษาจาวาและยังเป็นภาษาที่มีความนิยมสูงทำให้มีไลบราลี่หลากหลายช่วยลดเวลาในการพัฒนา

ฟรอนต์เอ็นจะประกอบไปด้วยสองส่วนหลักๆ คือ ส่วนการจำลองเครือข่าย และส่วนการแสดงผลการทำงาน ส่วนการจำลองเครือข่ายจะแปลงเครือข่ายกราฟฟิกที่ผู้ใช้งานสร้างขึ้นเป็นไฟล์คำสั่งภาษาทีซีแอลจากนั้นจะส่งไฟล์ภาษาทีซีแอลไปยังเครื่องเซิร์ฟเวอร์ที่ติดตั้งโปรแกรมเน็ตเวิร์คซิมูเลชั่น โปรแกรมเน็ตเวิร์คซิมูเลชั่นจะประมวลผลไฟล์ภาษาทีซีแอลและผลลัพธ์ซึ่งเป็นไฟล์เอ็นเอเอ็มจะถูกส่งกลับไปยังฝั่งของไคลเอนต์ จากนั้นฟรอนต์เอ็นในส่วนของการแสดงผลในรูปแบบภาพเคลื่อนไหวจะอ่านข้อมูลจากไฟล์เอ็นเอเอ็มแล้วแสดงผลลัพธ์ออกมาทางหน้าจอในลักษณะของภาพเคลื่อนไหว

การพัฒนาในส่วนของการจำลองเครือข่าย และการแสดงผลการทำงาน ได้พัฒนาโดยการใช้งานภาษาจาวาซึ่งเป็นภาษาที่ไม่ขึ้นกับระบบปฏิบัติการ ทำให้สามารถเรียกใช้งานได้บนระบบปฏิบัติการได้ทุกชนิด ในส่วนของเซิร์ฟเวอร์ซึ่งให้บริการมีการใช้งานเว็บเซิร์ฟเวอร์อาปาเช่ และการเขียนโปรแกรมในส่วนเน็ตเวิร์คโปรแกรมมิ่ง ซึ่งคอยรับการติดต่อจากไคลเอนต์ส่วนของฟรอนต์เอ็น นอกจากนี้มีการใช้งานจาวาเทคโนโลยีอย่างจาวาเว็บสตาร์ทในการเรียกใช้งานโปรแกรมฟรอนต์เอ็นผ่านเน็ตเวิร์ค

โปรแกรมฟรอนต์เอ็นสำหรับเน็ตเวิร์คซิมูเลชั่นนี้แม้ว่าจะใช้ฟังก์ชันของโปรแกรมเน็ตเวิร์คซิมูเลชั่นได้เพียงส่วนหนึ่งแต่ก็สามารถใช้ศึกษาหรือวิจัยโพรโตคอลหรือทดลองทฤษฎีต่างๆ ของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครือข่ายคอมพิวเตอร์ได้เป็นอย่างดีอีกทั้งมีการติดตั้งและใช้งานได้ง่าย ในแง่ของลดเวลาในศึกษาหรือวิจัยหรืออาจารย์ผู้สอนวิชาที่เกี่ยวกับกับเครือข่ายก็สามารถนำโปรแกรมนี้ไปใช้เพื่อส่งเสริมการเรียนรู้ได้ ซึ่งผู้พัฒนาหวังว่าจะทำให้ผู้สนใจที่ใช้โปรแกรมเน็ตเวิร์คซิมูเลชั่นสามารถนำไปใช้ให้เกิดประโยชน์สูงสุดกับการศึกษาหรือวิจัยได้

5.2 ปัญหาอุปสรรคและแนวทางแก้ไข

1. คู่มือของโปรแกรมเน็ตเวิร์คซิมูเลชั่นมีการอธิบายไม่ละเอียดเพียงพอ การเข้าใจหลักการและโครงสร้างต่างๆ ทำได้ค่อนข้างยาก ซึ่งแนวทางแก้ไข คือ การเรียนรู้คำสั่งจากแหล่งอื่นนอกเหนือจากคู่มือของโปรแกรมเน็ตเวิร์คซิมูเลชั่น
2. คู่มือและข้อมูลของโปรแกรมเน็ตเวิร์คซิมูเลชั่นมีจำนวนน้อย ปัญหานี้เองทำให้โปรแกรมที่พัฒนาขึ้นมาจำเป็นต้องนำมาเพียงบาง แนวทางแก้ไขคือ ศึกษาหาข้อมูลจากการค้นหาในเว็บไซต์
3. การติดตั้งโปรแกรมเน็ตเวิร์คซิมูเลชั่นซึ่งการติดตั้งทำได้ค่อนข้างยาก เนื่องจากต้องติดตั้งบนระบบปฏิบัติการลินุกซ์ และต้องติดตั้งไลบารี่เพิ่มเติมซึ่งต้องใช้เวลาในการศึกษาการติดตั้ง แนวทางการแก้ไขคือ ค้นหาวิธีการติดตั้งโปรแกรมจากอินเตอร์เน็ต
4. ปัญหาด้านการพัฒนา เนื่องจากภาษาจาวาไม่สามารถรองรับการกระทำบน จียูไอที่มีลักษณะอื่นนอกจากสี่เหลี่ยม ทำให้การสร้างการกระทำบนเส้นตรง แสดงการเชื่อมต่อทำได้ยาก และการแสดงผลข้อมูลที่วิ่งก็ไม่สามารถอธิบายข้อมูลลึกไปยังแต่ละตัวได้ ซึ่งแนวทางการแก้ไขคือ ค้นหาวิธีจากอินเตอร์เน็ต และหนังสือ

5.3 แนวทางในการพัฒนาต่อ

1. พัฒนาให้โปรแกรมฟรอนต์เอ็นรวมกับโปรแกรมเน็ตเวิร์คซิมูเลชั่นเป็นโปรแกรมเดียว
2. พัฒนาให้โปรแกรมฟรอนต์เอ็นส่วนของการแสดงผลให้มีความสามารถมากขึ้น เช่นการออกแบบโพรโตคอลผ่านฟรอนต์เอ็น และสามารถเก็บเป็นข้อมูลของผู้ใช้ได้
3. พัฒนาให้โปรแกรมส่วนจำลองเครือข่ายให้ตรวจสอบข้อผิดพลาดที่เกิดจากผู้ใช้ได้มากขึ้น
4. พัฒนาให้โปรแกรมฟรอนต์เอ็นรองรับฟังก์ชันของโปรแกรมเน็ตเวิร์คซิมูเลชั่นมากขึ้น พัฒนาในส่วนของการวิเคราะห์ค่าในเน็ตเวิร์คจำลอง รวมถึงการจำลองเครือข่ายไร้สาย
5. ปรับปรุงการทำงานของโปรแกรมให้รวดเร็วยิ่งขึ้น และใช้ทรัพยากรของระบบน้อยลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

Mauro Marinilli. 2004. **Java Deployment with JNLP and Webstart**. Indiana : Sams Publishing.

Welch , Brent B. 2003. **Practical programming in Tcl & Tk**. New Jersey :Prentice-Hall.

John K. Ousterhout. 1994. **Tcl and the Tk toolkit**. Boston : Addison-Wesley Professional

Harvey M. Deitel , Paul J. Deitel. 2004. **Java How to Program.**New Jersey :Prentice-Hall.

วีระศักดิ์ ซึงถาวร. 2543. **JAVA PROGRAMMING Volume I.** กรุงเทพฯ: ซีเอ็ดยูเคชั่น

วีระศักดิ์ ซึงถาวร. 2545. **JAVA PROGRAMMING Volume II.** กรุงเทพฯ: ซีเอ็ดยูเคชั่น

วีระศักดิ์ ซึงถาวร. 2548. **JAVA PROGRAMMING Volume III.** กรุงเทพฯ: ซีเอ็ดยูเคชั่น

VINT Group. 2000. **The Network Simulator.** [Online].
Avialable : http:// www.isi.edu/nsnam/ns

Jae Chaung , Mark Claypool. 2000. **Ns by Example.** [Online]
Avialable : http://nile.wpi.edu/NS/

Sun Develop Network. 2007. **Java Technology.** [Online]
Avialable : http://java.sun.com

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.
คู่มือการติดตั้งซอฟต์แวร์

การติดตั้งโปรแกรมเน็ตเวิร์คซิมูชั่น

การติดตั้งโปรแกรมเน็ตเวิร์คซิมูเลชั่น จะติดตั้งบนระบบปฏิบัติการลินุกซ์ซึ่งในที่นี้ได้เลือกใช้งานยูบันตู (Ubuntu) เวอร์ชั่น 7.04 โดยหลังจากติดตั้งเสร็จเรียบร้อยให้ทำการอัพเดตระบบปฏิบัติการให้เรียบร้อย จากนั้นจะติดตั้งโปรแกรมเน็ตเวิร์คซิมูเลชั่นเวอร์ชั่น 2.31 ซึ่งดาวน์โหลดได้จาก www.isi.edu/nsnam/ns/

รูปที่ ก.1 ไฟล์บีบอัดของโปรแกรมเน็ตเวิร์คซิมูเลชั่น

คลายไฟล์บีบอัดของโปรแกรมเน็ตเวิร์คซิมูเลชั่นออกด้วยคำสั่ง

```
#tar -xzvf ns-allinone-2.31.tar.gz
```

จากนั้นเปลี่ยนชื่อโฟลเดอร์จาก ns-allinone-2.31 เป็น NS และติดตั้งไลบรารี่เพิ่มเติมซึ่งดาวน์โหลดโดยใช้คำสั่ง

```
# apt-get install -f build-essential libxt-dev libxt6 libsm-dev libsm6  libice-dev libice6 libxmu-dev
```

เมื่อติดตั้งไลบรารี่เพิ่มเติมเสร็จเรียบร้อยแล้ว ให้ไปยังโฟลเดอร์ NS ที่ได้คลายออกมา จากนั้นสามารถติดตั้งโปรแกรมเน็ตเวิร์คซิมูเลชั่นโดยใช้คำสั่ง

```
# ./install
```

เมื่อติดตั้งเสร็จเรียบร้อยแล้ว จะได้รับแจ้งทางเทอร์มินอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Nam has been installed successfully.

Ns-allinone package has been installed successfully.
Here are the installation places:
tcl8.4.14:      /home/serverns/Desktop/NS/{bin,include,lib}
tk8.4.14:               /home/serverns/Desktop/NS/{bin,include,lib}
otcl:           /home/serverns/Desktop/NS/otcl-1.13
tclcl:          /home/serverns/Desktop/NS/tclcl-1.19
ns:             /home/serverns/Desktop/NS/ns-2.31/ns
nam:    /home/serverns/Desktop/NS/nam-1.13/nam
xgraph: /home/serverns/Desktop/NS/xgraph-12.1
gt-itm:   /home/serverns/Desktop/NS/itm, edriver, sgb2alt, sgb2ns, sgb2comns, sg
b2hierns
```

รูปที่ ก.2 แสดงการแจ้งเตือนทางเทอร์มินอล

จากนั้นจะมีการแจ้งเตือนให้กำหนดพาธของโปรแกรมเน็ตเวิร์คซิมูเลชั่น

```
Please put /home/serverns/Desktop/NS/bin:/home/serverns/Desktop/NS/tcl8.4.14/uni
x:/home/serverns/Desktop/NS/tk8.4.14/unix
into your PATH environment; so that you'll be able to run itm/tclsh/wish/xgraph.

IMPORTANT NOTICES:

(1) You MUST put /home/serverns/Desktop/NS/otcl-1.13, /home/serverns/Desktop/NS/
lib,
    into your LD_LIBRARY_PATH environment variable.
    If it complains about X libraries, add path to your X libraries
    into LD_LIBRARY_PATH.
    If you are using csh, you can set it like:
            setenv LD_LIBRARY_PATH <paths>
    If you are using sh, you can set it like:
            export LD_LIBRARY_PATH=<paths>

(2) You MUST put /home/serverns/Desktop/NS/tcl8.4.14/library into your TCL_LIBRA
RY environmental
    variable. Otherwise ns/nam will complain during startup.
```

รูปที่ ก.3 การแจ้งเตือนการกำหนดพาธของโปรแกรมเน็ตเวิร์คซิมูเลชั่น

หลังจากติดตั้งโปรแกรมเน็ตเวิร์คซิมูเลชั่นเสร็จเรียบร้อยแล้ว สามารถตรวจสอบการติดตั้งโดยใช้คำสั่ง

```
# ./validate
```

ภายในโฟลเดอร์ ns-2.31 โดยโปรแกรมจะทดสอบการทำงานทั้งทั้งหมด เมื่อทดสอบการทำงานเสร็จเรียบร้อยและไม่มีข้อผิดพลาด โปรแกรมจะแสดงข้อความดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
These messages are NOT errors and can be ignored:
    warning: using backward compatibility mode
    This test is not implemented in backward compatibility mode

validate overall report: all tests passed
```

รูปที่ ก.4 แสดงการทดสอบของโปรแกรมเน็ตเวิร์คซิมูเลชั่น

การกำหนดพาธของโปรแกรมเน็ตเวิร์คซิมูเลชั่นสามารถกำหนดได้โดยแก้ไขไฟล์ environment ซึ่งอยู่ภายใต้ /etc/ ซึ่งในที่นี้โปรแกรมได้ถูกติดตั้งไว้ที่ /home/serverns/Desktop/NS/ ซึ่งในไฟล์ environment จะมีการเพิ่มเติมดังนี้

```
PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/home/serverns/Desktop/NS/bin"
LANG="en_US.UTF-8"
```

รูปที่ ก.5 การกำหนดพาธของโปรแกรมเน็ตเวิร์คซิมูเลชั่น

ทดลองรีสตาร์ทระบบปฏิบัติการ และเรียกโปรแกรมเน็ตเวิร์คซิมูเลชั่นจากเทอร์มินอลโดยใช้คำสั่ง

```
# ns
```

โดยภายในโปรแกรมเทอร์มินอล ถ้าเครื่องหมาย "%" ปรากฏ แสดงว่าการติดตั้งโปรแกรมเน็ตเวิร์คซิมูเลชั่นเสร็จสิ้นแล้ว ดังรูป

```
serverns@serverns-desktop:~$ ns
%
```

รูปที่ ก.6 แสดงการเรียกใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่นจากเทอร์มินอล

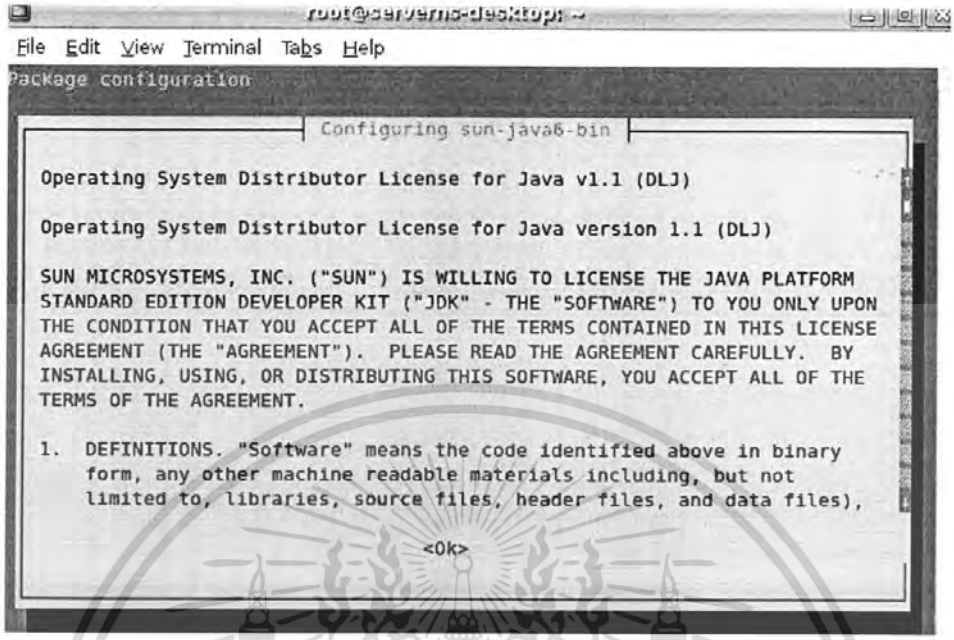
การติดตั้งโปรแกรมคอมไพล์และใช้งานภาษาจาวา

โปรแกรมคอมไพล์และใช้งานภาษาจาวาจะต้องติดตั้งเพื่อให้สามารถเรียกใช้งานโปรแกรมที่พัฒนาด้วยภาษาจาวาได้ โดยตัวที่ติดตั้งคือ Java Standard Edition เวอร์ชั่น 6.0 ซึ่งในระบบปฏิบัติการยูบันตู สามารถติดตั้งโดยใช้คำสั่ง

```
# apt-get install sun-java6-jdk
```

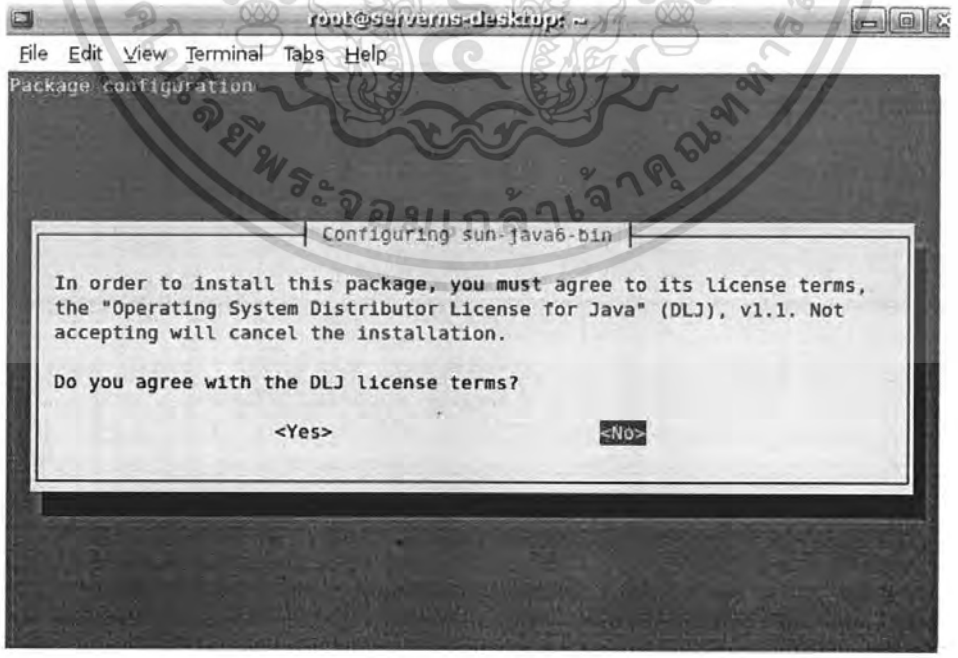
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยหน้าต่างของเทอร์มินอลจะสอบถามอีกครั้งว่าต้องการติดตั้งหรือไม่ ให้ตอบ y จากนั้น เมื่อระบบปฏิบัติการดาวน์โหลดโปรแกรมติดตั้งจาวาเสร็จเรียบร้อยจะแสดลหน้าต่างดังนี้



รูปที่ ก.7 การปรับแต่งการติดตั้งโปรแกรมคอมไพล์และใช้งานภาษาจาวา

เลื่อนเคอร์เซอร์ไปตำแหน่ง OK จากนั้นปุ่มกด Enter จะขึ้นหน้าต่างต่อไปซึ่งเป็นหน้ายอมรับการใช้งานซอฟต์แวร์



รูปที่ ก.8 หน้าต่างยอมรับการใช้งานซอฟต์แวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลือกตอบ Yes เพื่อยอมรับการใช้งานซอฟต์แวร์ จากนั้นจะเป็นการติดตั้งภาษาจาวาซึ่งเมื่อติดตั้งเสร็จเรียบร้อยแล้ว สามารถตรวจสอบการทำงานได้โดยการพิมพ์คำสั่ง

```
# java –version
```

ผลที่ได้จะแสดงเวอร์ชั่นของภาษาจาวาที่ติดตั้งออกมาดังนี้

รูปที่ ก.9 การทดสอบการติดตั้งภาษาจาวา

การติดตั้งเว็บเซิร์ฟเวอร์

เว็บเซิร์ฟเวอร์ซึ่งติดตั้งเพื่อใช้ในการเรียกโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็น โดยจะเรียกใช้งานผ่านหน้าเว็บไซต์ ซึ่งในที่นี้เลือกใช้งานเว็บเซิร์ฟเวอร์อาปาเช่2 (apache2) พร้อมติดตั้งไลบรารี่สำหรับรองรับการใช้งานพีเอชพีเวอร์ชั่น 5 (PHP 5) โดยการติดตั้งใช้คำสั่งภายในเทอร์มินอลดังนี้

```
# apt-get install apache2 libapache2-mod-php5
```

เมื่อติดตั้งเสร็จเรียบร้อยแล้ว ให้ตรวจสอบการติดตั้งโดยทดลองเปิดเว็บเบราว์เซอร์ไปที่ http://localhost/

หลังจากทดสอบการติดตั้งแล้ว ให้เปลี่ยนแปลงค่าการทำงานของอาปาเช่โดยไปที่ /etc/apache2/sites-available จากนั้นเปลี่ยนแปลงค่าของไฟล์ default โดยเปลี่ยนแปลงในส่วนของการกำหนดค่า DocumentRoot ให้มีค่าเป็น /var/www/apache2-default

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การนำโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็นติดตั้งในเซิร์ฟเวอร์

หลังจากที่ได้คอมไพล์ซอสโค้ดทั้งหมดของโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็นซึ่งถ้าใช้งานโปรแกรมเน็ตบีนส์ (NetBeans) ในการพัฒนา โปรแกรมเน็ตบีนส์จะช่วยสร้างไฟล์ Java Archive (jar) ให้อัตโนมัติ ซึ่งเราสามารถนำไปใช้งานได้ทันที แต่ถ้าต้องการให้สามารถใช้งานได้ผ่านทางเน็ตเวิร์ค จะต้องทำเครื่องหมาย (sign)ไฟล์ เพื่อให้สามารถเข้าใช้งานทรัพยากรบนเครื่องไคลเอนต์ที่เรียกใช้งานโปรแกรม

การทำเครื่องหมายไฟล์จาร์สามารถทำได้โดยสิ่งแรกต้องสร้างคีย์ขึ้นมา โดยใช้โปรแกรมของจาวา คือ โปรแกรมคีย์ทูล (keytool) ซึ่งติดตั้งมาพร้อมกันในขณะติดตั้ง JDK โดยการสร้างคีย์จะใช้คำสั่ง

```
keytool -genkey -alias MyAlias -keystore MyStore -keypass MyKeyPass -dname
"cn=CommonName" -storepass MyStorePass
```

ผลที่ได้จากโปรแกรมคีย์ทูลจะสร้างไฟล์คีย์ขึ้นมา ซึ่งมีชื่อไฟล์เป็น MyStore จากนั้นจะใช้คำสั่ง jarsigner เพื่อทำเครื่องหมายไฟล์จาร์โดยมีรูปแบบคำสั่งดังนี้

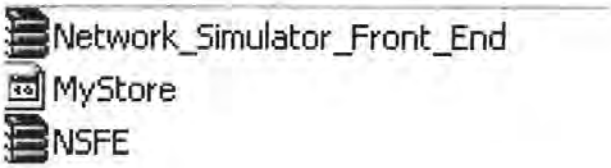
```
jarsigner -keystore MyStore -storepass MyStorePass -keypass MyKeyPass -signedjar NSFE.jar
Network_Simulator_Front_End.jar MyAlias
```

ทั้งโปรแกรม keytool และ jarsigner จะมีพารามิเตอร์บางตัวที่ต้องเหมือนกัน ซึ่งถ้าไม่เหมือนกันจะทำให้การทำเครื่องหมายไฟล์ไม่สำเร็จ โดยผลที่ได้จากการทำงานจะมีไฟล์ใหม่ซึ่งชื่อว่า NSFE.jar เกิดขึ้น

```
C:¥Documents and Settings¥FokyBelmont¥Desktop¥Test>keytool -genkey -alias MyAlia
s -keystore MyStore -keypass MyKeyPass -dname "cn=CommonName" -storepass MyStore
Pass

C:¥Documents and Settings¥FokyBelmont¥Desktop¥Test>jarsigner -keystore MyStore -
storepass MyStorePass -keypass MyKeyPass -signedjar NSFE.jar Network_Simulator_F
ront_End.jar MyAlias

Warning:
The signer certificate will expire within six months.

C:¥Documents and Settings¥FokyBelmont¥Desktop¥Test>
```

รูปที่ ก.10 การทำเครื่องหมายโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.11 ผลลัพธ์ที่ได้จากการทำเครื่องหมายไฟล์โปรแกรม

หลังจากทำเครื่องหมายไฟล์เรียบร้อยแล้ว ให้สร้างไฟล์ JNLP (Java Network Launch Protocol) ซึ่งเป็นไฟล์ที่จะใช้ในการเรียกใช้งานแอพพลิเคชั่นผ่านทางเน็ตเวิร์คซึ่งเป็นเทคโนโลยีของจาวา โครงสร้างของไฟล์ JNLP มีลักษณะดังนี้

```
<?xml version="1.0" encoding="utf-8"?>
<jnlp spec="0.2 1.0" codebase="http://192.168.23.133/">
<information>
        <title>NSFE</title>
        <vendor>Foky</vendor>
    <homepage href=""/>
    <description>NS</description>
    <description kind="short">NS</description>
     <offline-allowed/>
</information>
  <security>
    <all-permissions/>
 </security>
<resources>
   <j2se version="1.3+" href="http://java.sun.com/products/autodl/j2se"/>
   <j2se version="1.3+"/>
   <jar href="NSFE.jar" main="true" download="eager"/>
</resources>
<application-desc main-class="Editor.MainClass"/>
</jnlp>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อได้ทั้งแอพพลิเคชั่นซึ่งทำเครื่องหมายไฟล์และไฟล์ JNLP แล้ว ให้นำทั้งสองไฟล์ใส่ไว้ในโฟลเดอร์ /var/www/ apache2-default

ทำการเปลี่ยนค่ากำหนดสิทธิของไฟล์ NSFE.jar และ nsfe.jnlp โดยกำหนดให้มีค่าเป็น 755 เพื่อให้ผู้ใช้งานทั่วไปสามารถเรียกใช้งานแอพพลิเคชั่นได้

```
drwxr-xr-x 3 root root   4096 2008-01-31 14:43 .
drwxr-xr-x 4 root root   4096 2007-07-23 06:01 ..
-rw-r--r-- 1 root root    366 2008-01-31 14:52 index.html
-rwxr-xr-x 1 root root   3377 2008-01-12 17:25 Java.gif
drwxr-xr-x 2 root root   4096 2008-01-31 14:46 Lab
-rwxr-xr-x 1 foky foky    984 2008-01-31 14:24 nsfe.jnlp
-rwxr-xr-x 1 root root 762111 2008-01-13 15:55 NSFE.jar
-rwxr-xr-x 1 root root   5801 2008-01-12 17:57 NSFE S.jpg
-rwxr-xr-x 1 root root  20468 2008-01-13 01:14 wall.jpg
```

รูปที่ ก.12 แสดงสิทธิในการเข้าถึงไฟล์โปรแกรมเน็ตเวิร์คซิมูเลชั่น

กำหนดลิงค์ในการเรียกใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็นภายในไฟล์ภาษา HTML ดังนี้

```
<a href=nsfe.jnlp>Launch Application</a><br>
```

เมื่อมีการกดลิงค์ซึ่งกำหนดไปยังไฟล์ JNLP จะเป็นการเรียกใช้งานจาวาเว็บสตาร์ท โดยจะเรียกใช้งานโปรแกรมเน็ตเวิร์คซิมูเลชั่นฟรอนต์เอ็นโดยอัตโนมัติ

ในการเริ่มต้นการทำงานของเซิร์ฟเวอร์แต่ละครั้งจะต้องมีการเรียกโปรแกรมซึ่งทำหน้าที่ในการรับการส่งไฟล์ภาษาโอทีซีแอลซึ่งรับมาจากไคลเอนต์เพื่อนำไปประมวลผลกับโปรแกรมเน็ตเวิร์คซิมูเลชั่นซึ่งติดตั้งบนเซิร์ฟเวอร์ โดยการเรียกใช้งานสามารถเรียกได้โดยใช้คำสั่ง

```
# java ServerNS
```

โดยฝั่งเซิร์ฟเวอร์จะคอยรอการติดต่อจากแอพพลิเคชั่นซึ่งเรียกใช้งานโดยฝั่งไคล์เอนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

ภาษาทีซีแอลและโอทีซีแอล

ภาษาทีซีแอล (Tcl หรือ Tool Command Language) ได้ถูกพัฒนาขึ้นโดย Dr. John Ousterhout แห่งมหาวิทยาลัยเบิร์กเลย์ ในปี ค.ศ. 1989 ภาษาทีซีแอลได้ถูกออกแบบให้มีขนาดเล็ก ง่ายต่อเปลี่ยนแปลงและสามารถพัฒนาเพิ่มเติมได้ โดยสามารถใช้งานได้หลายแพลตฟอร์มอย่างเช่น วินโดส์ (Windows) , ยูนิกซ์ (Unix) และแมคอินทอช (Macintosh) ผู้พัฒนาโปรแกรมสามารถใช้งานภาษาทีซีแอลและส่วนขยายของภาษาทีซีแอลในการช่วยพัฒนาโปรแกรมได้ โดยส่วนขยายของภาษาทีซีแอลที่ใช้งานในโปรแกรมเน็ตเวิร์คซิมูเลเตอร์จะใช้งานภาษาโอทีซีแอลซึ่งเป็นส่วนขยายเพื่อสามารถสร้างโปรแกรมเชิงวัตถุได้

พื้นฐานของภาษาโอทีซีแอล

ลักษณะคำสั่งพื้นฐานของทีซีแอลจะมีลักษณะดังนี้

command arg1 arg2 arg3 ...

ซึ่งคำสั่ง (command) เป็นชื่อของคำสั่งที่มีอยู่ในภาษาทีซีแอลหรืออาจจะเป็นกระบวนคำสั่ง (Procedure) ก็ได้ โดยการเว้นวรรคจะเป็นการแยกคำสั่งกับอาร์กิวเมนต์ และใช้การขึ้นบรรทัดใหม่หรือเซมิโคลอนในการจบคำสั่งในบรรทัดนั้น ๆ ซึ่งในภาษาทีซีแอลทุกอย่างจะถูกมองเป็นสายอักขระ (String)

1 . การกำหนดค่าให้กับตัวแปร จะใช้คำสั่ง

```
set <ชื่อตัวแปร> <ค่าที่กำหนด>
```

ตัวอย่าง กำหนดค่าตัวแปร b มีค่าเท่ากับ 5 จะใช้คำสั่ง

```
set b 5
```

2. การกำหนดต่าจากตัวแปรหนึ่งไปยังอีกตัวแปรหนึ่ง จะใช้คำสั่ง

```
set <ตัวแปรที่ถูกกำหนดค่า> <ตัวแปรที่กำหนดค่าให้>
```

ตัวอย่าง กำหนดค่าของตัวแปร x ให้มีค่าเท่ากับตัวแปร a จะใช้คำสั่ง

```
Set x $a
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การคำนวณทางคณิตศาสตร์ จะใช้คำสั่ง expr ตัวอย่างต้องการกำหนดค่าให้ x มีค่าเท่ากับผลรวมของ a และ b จะใช้คำสั่ง

```
set x [expr $a + $b]
```

4. เครื่องหมาย # หมายถึงการคอมเมนต์ (comment) ซึ่งถ้ามีเครื่องหมายนี้อยู่หน้าบรรทัดใด บรรทัดนั้นจะไม่ถูกนำมาประมวลผล

5. การสร้างไฟล์ จะใช้คำสั่ง open ร่วมกับ w ซึ่งเป็นการเปิดไฟล์และกำหนดให้เขียนลงไฟล์ที่กำหนดไว้ โดยจะใช้คำสั่ง

```
set file1 [open <ชื่อไฟล์> w]
```

เป็นการกำหนดให้ตัวแปร file1 ชี้ไปยังไฟล์ที่ได้สร้างขึ้น โดยเป็นการสร้างไฟล์เพื่อเขียน

6. การแสดงผลออกทางหน้าจอ จะใช้คำสั่ง

```
put "ข้อความที่ต้องการแสดง"
```

เป็นการแสดงผลออกทางหน้าจอ ตัวอย่าง

```
put "Hello World"
```

จะเป็นการแสดงข้อความ Hello World ออกทางหน้าจอ

7. การเรียกการทำงานของคำสั่ง จะเป็นการเรียกคำสั่งที่มีอยู่บนเครื่องให้ทำงานโดยจะใช้คำสั่ง

```
exec <คำสั่งที่ต้องการเรียก> &
```

8. โครงสร้างของคำสั่ง if

คำสั่ง if จะเป็นการเลือกทำตามเงื่อนไขที่ได้กำหนดไว้ ซึ่งในภาษาทีซีแอลมีโครงสร้างดังนี้

```
if { เงื่อนไข } {
        <คำสั่งที่ต้องการประมวลผล>
} else {
        <คำสั่งที่ต้องการประมวลผล>
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. โครงสร้างของคำสั่งทำซ้ำ

ในภาษาโอทีซีแอลจะมีคำสั่งทำซ้ำหลายชนิด ซึ่งมีดังนี้

```
for {กำหนดค่าเริ่มต้น} {เงื่อนไข} {ปรับปรุงค่านิพจน์} {
        <คำสั่งที่ต้องการประมวลผล>
}
```

```
while { เงื่อนไข } {
        < คำสั่งที่ต้องการประมวลผล>
}
```

```
foreach <ชื่อของตัวแปร> { ลิสต์ } {
        <คำสั่งที่ต้องการประมวลผล>
}
```

ตัวอย่างการใช้งาน

```
for {set i 0} {$i < 5} {incr i } {
        puts "$i"
}
```

จะเป็นการแสดงค่าของ i ทุกครั้งในการวนรอบโดย i มีค่าเริ่มต้นที่ 0 และค่อย ๆ เพิ่มค่า i ทีละ 1 ทุกครั้งของการวนรอบ

10. สายอักขระ (String) และลิสต์ (List)

ภาษาโอทีซีแอลจะมีการจัดเก็บค่าของข้อมูลในตัวแปร โดยค่าที่ได้จะเป็นสายอักขระ ถ้าต้องการเก็บค่าตัวเลยในตัวแปร ค่าตัวเลขที่ได้นั้นจะถูกแปลเป็นสายอักขระ

ลิสต์เป็นชนิดพิเศษของสายอักขระ โดยมีหลักการทำงาน คือ จะแบ่งข้อมูลคำออกเป็นส่วน ๆ โดยการตรวจสอบช่องว่าง (Space)

ตัวอย่างการสร้างลิสต์

```
set simple_list "Alice Bob Caroll Trudy"
```

จะเป็นการสร้างลิสต์ที่มีสมาชิก 4 คำ คือ Alice , Bob , Caroll , Trudy

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. การสร้างกระบวนคำสั่ง (procedure)

```
proc <ชื่อของกระบวนคำสั่ง> {พารามิเตอร์} {
        <คำสั่งที่ต้องการ>
        Return $ค่าที่ส่งคืน
}
```

12. การสร้างคลาส

- การประกาศสร้างคลาสจะใช้คำสงวน คือ Class <ชื่อของคลาส>
- เมื่อต้องการสร้างเมทธอดของคลาสให้ใช้คำว่า instrproc <ชื่อของเมทธอด>
- เมื่อต้องการสร้าง constructor ให้ใช้คำสั่ง init
- เมื่อต้องการสร้างตัวชี้ (pointer) ซึ่งชี้ตัวเองให้ใช้คำว่า self
- ถ้าต้องการประกาศตัวแปรที่เป็นอินสแตนซ์ (instance) ของภาษาโอทีซีแอล จะใช้คำสั่ง instvar
- เมื่อต้องการทำการสืบทอดคลาสจะใช้คำสั่ง superclass เพื่อบอกคลาสที่มีคำสั่ง superclass นำหน้านั้นเป็นคลาสบรรพบุรุษ

ตัวอย่างการสร้างคลาส

```
#add a member function call "greet"
Class mom
mom instproc greet { } {
        $self instvar age_
        puts "Sage_ year old mom say: How are you doing?"
}

#Create a child class of "mom" called "kid"
#and overide the member function "greet"
Class kid -superclass  mom
Kid instproc  greet { } {
        $self instvar age_
        puts "Sage_ year old kid say: What's up, dude?"
}
```

รูปที่ ข.1 ตัวอย่างการสร้างคลาสในภาษาโอทีซีแอล

13. การประกาศตัวแปรโกลบอล (Global)

ในภาษาโอทีซีแอลสามารถประกาศตัวแปรโกลบอลได้โดยการใช้คำสั่ง

```
global <ชื่อตัวแปรโกลบอล>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

ความหมายของไฟล์เอ็นเอเอ็ม

ในส่วนนี้จะเป็นรายละเอียดของไฟล์เอ็นเอเอ็มสำหรับในแต่ละเหตุการณ์ที่เกิดขึ้น

: คอมเมนต์ซึ่งจะไม่สนใจประมวลในบรรทัดนี้

T : เป็นเหตุการณ์ที่ใช้ในการประสานเวลา

-t <time> เวลา

n : โหนด

-t	<time>	เวลา
-s	<int>	หมายเลขของโหนด
-v	<shape>	รูปร่างของโหนด (วงกลม , หกเหลี่ยม)
-c	<color>	ค่าสีของโหนด
-z	<double>	ขนาดของโหนด
-a	<int>	แอดเดรสของโหนด
-x	<double>	ค่าตำแหน่ง x
-y	<double>	ค่าตำแหน่ง y
-i	<color>	ป้ายแสดงสี
-b	<string>	ป้าย
-l	<string>	ป้ายแสดงชื่อของโหนด
-o	<color>	ค่าสีก่อนหน้าที่จะเปลี่ยนแปลง
-S	<string>	สถานะของโหนด(ทำงาน , ไม่ทำงาน , ค่าสีที่กำหนด)
-L	<string>	ป้ายแสดงชื่อของโหนดก่อนหน้าที่มีการเปลียนแปลง
-p	<string>	ตำแหน่งของป้ายแสดงชื่อของโหนด
-P	<string>	ตำแหน่งของป้ายแสดงชื่อของโหนดก่อนหน้าการเปลี่ยนแปลง
-i	<color>	ค่าสีภายในของป้ายที่แสดง
-I	<color>	ค่าสีภายในของป้ายที่แสดงก่อนหน้าที่มีการเปลี่ยนแปลง
-e	<color>	ค่าสีของป้าย
-E	<color>	ค่าสีของป้ายก่อนหน้าที่จะมีการเปลี่ยนแปลง
-u	<string>	ค่าความเร็วในแกน x

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-U	<string>	ค่าความเร็วในแกน x
-V	<string>	ค่าความเร็วในแกน y
-T	<double>	เวลาที่โหนดจะหยุดทำงาน
-w	<flag>	การจำลองโหนดไร้สาย

l : ลิงค์

-t	<time>	เวลา
-s	<int>	หมายเลขโหนดต้นทาง
-d	<int>	หมายเลขโหนดปลายทาง
-r	<double>	อัตราการส่งข้อมูล
-D	<double>	ค่าดีเลย์
-h	<double>	ค่าความยาว
-O	<orientation>	ค่ากำหนดทิศทาง
-b	<string>	ป้ายแสดงชื่อของลิงค์
-c	<color>	ค่าสีของลิงค์
-o	<color>	ค่าสีของลิงค์ก่อนหน้าการเปลี่ยนแปล
-S	<string>	สถานะ (ทำงาน , ไม่ทำงาน)
-l	<string l>	ป้ายแสดงชื่อของลิงค์
-L	<string>	ป้ายแสดงชื่อของลิงค์ก่อนหน้าที่จะมีการเปลี่ยนแปลง
-e	<color>	ป้ายแสดงชื่อสีของลิงค์
-E	<color>	ป้ายแสดงชื่อสีของลิงค์ก่อนหน้าที่จะมีการเปลี่ยนแปลง

+ : การเข้าคิวของกลุ่มข้อมูล

-t	<time>	เวลา
-s	<int>	หมายเลขโหนดต้นทาง
-d	<int>	หมายเลขโหนดปลายทาง
-e	<int>	ส่วนขยาย
-a	<int>	ค่าสีของกลุ่มข้อมูล
-i	<int>	หมายเลขของกลุ่มข้อมูล
-l	<int>	ค่าพลังงานของกลุ่มข้อมูล
-c	<string>	การติดต่อของกลุ่มข้อมูล
-x	<comment>	คอมเมนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-p	<string>	ชนิดของกลุ่มข้อมูล
-k	<string>	ชนิดของกลุ่มข้อมูล

- : การออกจากคิวของกลุ่มข้อมูล

-t	<time>	เวลา
-s	<int>	หมายเลขโหนดต้นทาง
-d	<int>	หมายเลขโหนดปลายทาง
-e	<int>	ส่วนขยาย
-a	<int>	ค่าสีของกลุ่มข้อมูล
-i	<int>	หมายเลขของกลุ่มข้อมูล
-l	<int>	ค่าพลังงานของกลุ่มข้อมูล
-c	<string>	การติดต่อของกลุ่มข้อมูล
-x	<comment>	คอมเมนต์
-p	<string>	ชนิดของกลุ่มข้อมูล
-k	<string>	ชนิดของกลุ่มข้อมูล

h : การผ่านโหนดหรือฮอป

-t	<time>	เวลา
-s	<int>	หมายเลขโหนดต้นทาง
-d	<int>	หมายเลขโหนดปลายทาง
-e	<int>	ส่วนขยาย
-a	<int>	ค่าสีของกลุ่มข้อมูล
-i	<int>	หมายเลขของกลุ่มข้อมูล
-l	<int>	ค่าพลังงานของกลุ่มข้อมูล
-c	<string>	การติดต่อของกลุ่มข้อมูล
-x	<comment>	คอมเมนต์
-p	<string>	ชนิดของกลุ่มข้อมูล
-k	<string>	ชนิดของกลุ่มข้อมูล
-R	<double>	รัศมีการกระจายของการสื่อสารไร้สาย
-D	<double>	รัศมีการกระจายของการสื่อสารไร้สาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

r : การรับกลุ่มของข้อมูล

-t	<time>	เวลา
-s	<int>	หมายเลขโหนดต้นทาง
-d	<int>	หมายเลขโหนดปลายทาง
-e	<int>	ส่วนขยาย
-a	<int>	ค่าสีของกลุ่มข้อมูล
-i	<int>	หมายเลขของกลุ่มข้อมูล
-l	<int>	ค่าพลังงานของกลุ่มข้อมูล
-c	<string>	การติดต่อของกลุ่มข้อมูล
-x	<comment>	คอมเมนต์
-p	<string>	ชนิดของกลุ่มข้อมูล
-k	<string>	ชนิดของกลุ่มข้อมูล

d : การดรอปของกลุ่มข้อมูล

-t	<time>	เวลา
-s	<int>	หมายเลขโหนดต้นทาง
-d	<int>	หมายเลขโหนดปลายทาง
-e	<int>	ส่วนขยาย
-a	<int>	ค่าสีของกลุ่มข้อมูล
-i	<int>	หมายเลขของกลุ่มข้อมูล
-l	<int>	ค่าพลังงานของกลุ่มข้อมูล
-c	<string>	การติดต่อของกลุ่มข้อมูล
-x	<comment>	คอมเมนต์
-p	<string>	ชนิดของกลุ่มข้อมูล
-k	<string>	ชนิดของกลุ่มข้อมูล

E : การเข้าคิวของเซสชั่น

-t	<time>	เวลา
-s	<int>	หมายเลขโหนดต้นทาง
-d	<int>	หมายเลขโหนดปลายทาง
-e	<int>	ส่วนขยาย
-a	<int>	ค่าสีของกลุ่มข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-i	<int>	หมายเลขของกลุ่มข้อมูล
-l	<int>	ค่าพลังงานของกลุ่มข้อมูล
-c	<string>	การติดต่อของกลุ่มข้อมูล
-x	<comment>	คอมเมนต์
-p	<string>	ชนิดของกลุ่มข้อมูล
-k	<string>	ชนิดของกลุ่มข้อมูล

D : การออกคิวของเซสชั่น

-t	<time>	เวลา
-s	<int>	หมายเลขโหนดต้นทาง
-d	<int>	หมายเลขโหนดปลายทาง
-e	<int>	ส่วนขยาย
-a	<int>	ค่าสีของกลุ่มข้อมูล
-i	<int>	หมายเลขของกลุ่มข้อมูล
-l	<int>	ค่าพลังงานของกลุ่มข้อมูล
-c	<string>	การติดต่อของกลุ่มข้อมูล
-x	<comment>	คอมเมนต์
-p	<string>	ชนิดของกลุ่มข้อมูล
-k	<string>	ชนิดของกลุ่มข้อมูล

P : การครอปของเซสชั่น

-t	<time>	เวลา
-s	<int>	หมายเลขโหนดต้นทาง
-d	<int>	หมายเลขโหนดปลายทาง
-e	<int>	ส่วนขยาย
-a	<int>	ค่าสีของกลุ่มข้อมูล
-i	<int>	หมายเลขของกลุ่มข้อมูล
-l	<int>	ค่าพลังงานของกลุ่มข้อมูล
-c	<string>	การติดต่อของกลุ่มข้อมูล
-x	<comment>	คอมเมนต์
-p	<string>	ชนิดของกลุ่มข้อมูล
-k	<string>	ชนิดของกลุ่มข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

a : เอเจนต์

-t	<time>	เวลา
-s	<int>	หมายเลขโหนดต้นทาง
-d	<int>	หมายเลขโหนดปลายทาง
-x	<flag>	การย้ายออกของเอเจนต์
-n	<string>	ชื่อของเอเจนต์

f : ฟีเจอร์

-t	<time>	เวลา
-s	<int>	หมายเลขโหนดต้นทาง
-d	<int>	หมายเลขโหนดปลายทาง
-x	<flag>	การยกเลิกฟีเจอร์
-T	<char>	ชนิดของฟีเจอร์
-n	<string>	ชื่อของฟีเจอร์
-a	<string>	เอเจนต์
-v	<string>	ค่าของฟีเจอร์
-o	<string>	ค่าของฟีเจอร์ก่อนจะมีการเปลี่ยนแปลง

G : กลุ่มของโหนด

-t	<time>	เวลา
-n	<string>	ชื่อกลุ่มของโหนด
-i	<int>	หมายเลขโหนด
-a	<int>	หมายเลขกลุ่มของโหนด
-x	<flag>	การยกเลิกจากกลุ่มของโหนด

L : การเชื่อมต่อเครือข่ายท้องถิ่น

-t	<time>	เวลา
-s	<int>	หมายเลขโหนดต้นทาง
-d	<int>	หมายเลขโหนดปลายทาง
-o	<orientation>	ค่ากำหนดทิศทาง
-O	<orientation>	ค่ากำหนดทิศทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

m : เครื่องหมายของโหนด

-t	<time>	เวลา
-n	<string>	ชื่อของเครื่องหมาย
-s	<int>	หมายเลขโหนด
-c	<string>	ค่าสีของโหนด
-h	<string>	รูปร่าง (วงกลม , สี่เหลี่ยม, หกเหลี่ยม)
-X	<flag>	การยกเลิกเครื่องหมาย

R : การเลือกเส้นทาง

-t	<time>	เวลา
-s	<int>	หมายเลขโหนดต้นทาง
-d	<int>	หมายเลขโหนดปลายทาง
-g	<int>	กลุ่มมัลติคาสต์
-p	<packet source>	หมายเลขกลุ่มข้อมูลต้นทาง
-n	<flag>	ค่าแคชที่ดิดลบ
-x	<flag>	เส้นทางได้ถูกยกเลิก
-T	<double>	เวลายกเลิก
-m	<string>	โหมดการทำงาน

c : ตารางกำหนดค่าสี

-t	<time>	เวลา
-i	<int>	หมายเลข
-n	<string>	ค่าสี

q : การสร้างคิวของกลุ่มข้อมูล

-t	<time>	เวลา
-s	<int>	หมายเลขโหนดต้นทาง
-d	<int>	หมายเลขโหนดปลายทาง
-a	<orientation>	ค่ากำหนดทิศทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

X : รูปร่างของเครือข่ายท้องถิ่น

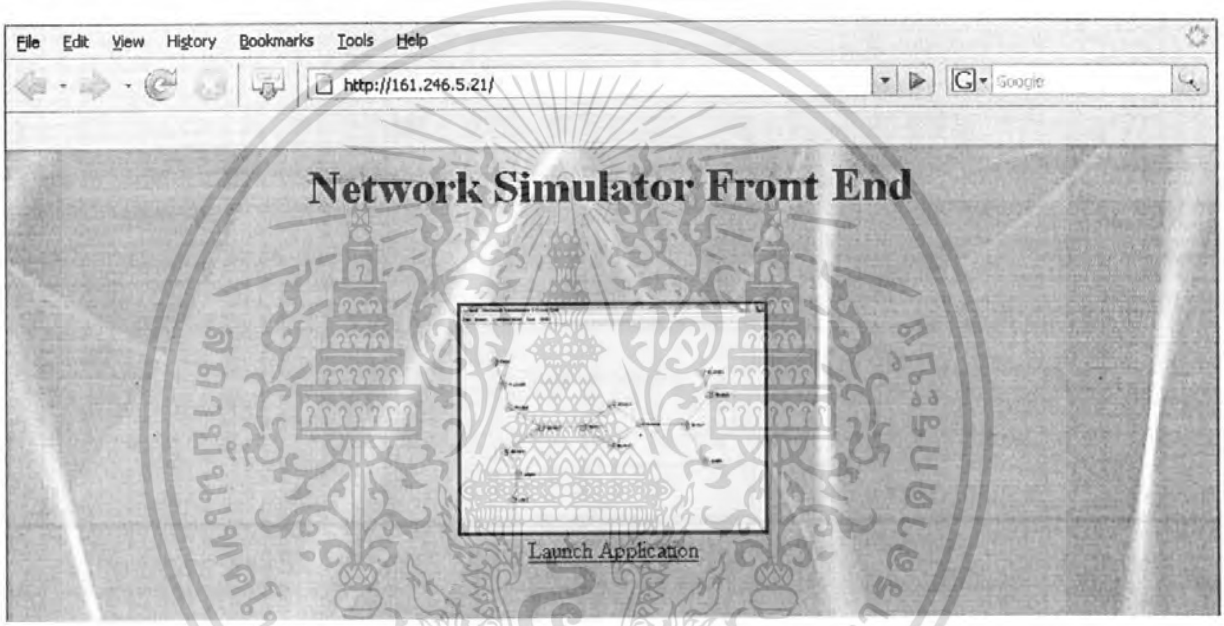
-t	<time>	เวลา
-n	<string>	ชื่อของเครือข่าย
-r	<double>	อัตรา
-D	<double>	ค่าดีเลย์
-o	<orientation>	ค่ากำหนดทิศทาง
-O	<orientation>	ค่ากำหนดทิศทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง.
คู่มือการใช้งานซอฟต์แวร์

วิธีการใช้งานซอฟต์แวร์

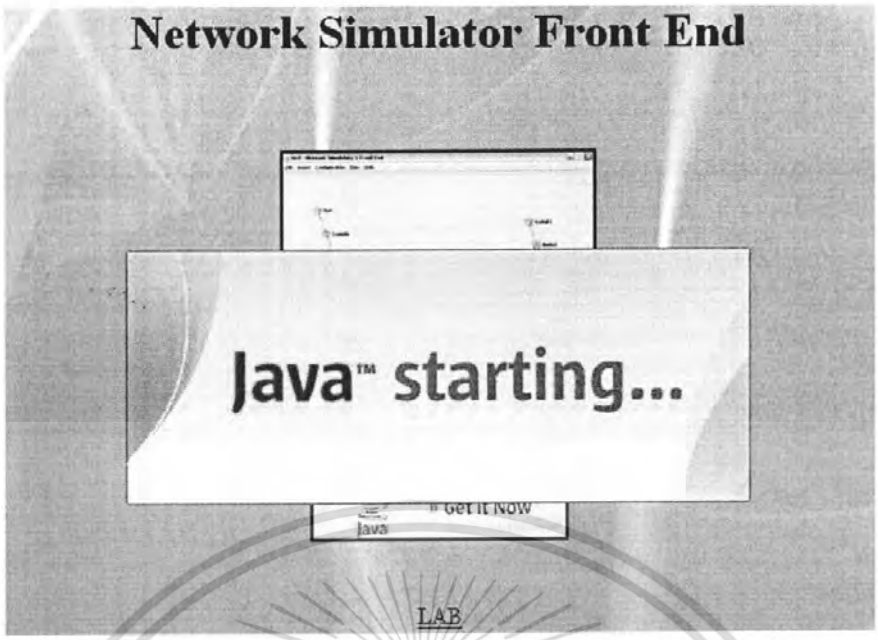
1. ผู้ใช้งานเรียกโปรแกรมเว็บเบราว์เซอร์อย่าง อินเตอร์เน็ตเอ็กซ์โพลเลอร์ หรือไฟล์ฟ็อกซ์ โดยระบุที่อยู่ของหน้าเว็บเพจที่เครื่องเซิร์ฟเวอร์โปรแกรมเน็ตเวิร์คซิมูเลชั่นให้บริการอยู่



รูปที่ ง.1 แสดงการเรียกเว็บเพจของเซิร์ฟเวอร์

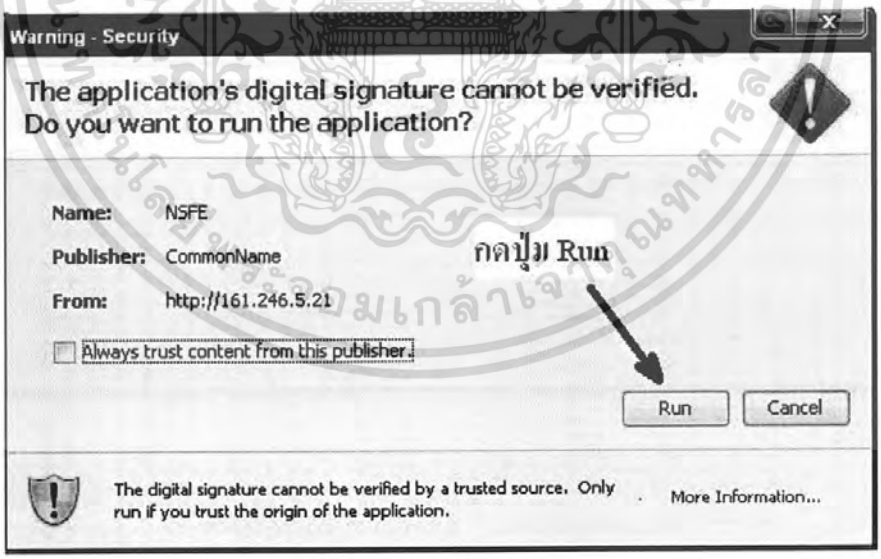
2. การเรียกใช้งานโปรแกรมฟรอนต์เอ็นสามารถเรียกโดยกดคลิงค์บนหน้าเว็บเพจซึ่งจะทำหน้าที่เรียกใช้งานจาวาเว็บสตาร์ทอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ง.2 แสดงการเรียกใช้งานโปรแกรมฟรอนต์เอ็น

3. จาวาเว็บสตาร์ทจะแจ้งเตือนในเรื่องความปลอดภัยในการเรียกใช้งาน ซึ่งโปรแกรมฟรอนต์เอ็นจะมีการเข้าใช้งานทรัพยากรของเครื่องบางส่วน



รูปที่ ง.3 ข้อความขออนุญาตใช้งานทรัพยากร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากนั้นโปรแกรมฟรอนต์เอ็นจะทำงาน

รูปที่ ง.4 โปรแกรมฟรอนต์เอ็น

หน้าต่างของโปรแกรมฟรอนต์เอ็น ประกอบด้วยดังนี้

- ปุ่ม File
  - New เป็นการเริ่มต้นการจำลองใหม่โดยโปรแกรมจะลบการตั้งค่าเดิมทั้งหมด
  - Open เพื่อใช้ในการเปิดไฟล์นามสกุล .nsf และคืนค่าการจำลองที่บันทึกไว้
  - Save เพื่อใช้ในการบันทึกไฟล์นามสกุล .nsf ค่าต่างๆ ในการจำลองจะถูกบันทึกลงไฟล์เดิมที่กำลังเปิดอยู่
  - Save Asเพื่อใช้ในการบันทึกไฟล์นามสกุล .nsf เป็นไฟล์ใหม่
- ปุ่ม Insert
  - Node เพื่อสร้างโหนดใหม่ขึ้นมา
  - Transport เพื่อใช้ในการสร้างเอเจนต์ประเภท Transport
    - UDP เพื่อใช้ในการสร้างเอเจนต์ประเภท UDP
    - NULL เพื่อใช้ในการสร้างเอเจนต์ประเภท NULL
    - TCP Tahoe เพื่อใช้ในการสร้างเอเจนต์ประเภท TCP Tahoe
    - TCP Vegas เพื่อใช้ในการสร้างเอเจนต์ประเภท TCP Vegas
    - TCP Reno เพื่อใช้ในการสร้างเอเจนต์ประเภท TCP Reno

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- TCP New Reno เพื่อใช้ในการสร้างเอเจนต์ประเภท TCP New Reno
- TCP Sack1 เพื่อใช้ในการสร้างเอเจนต์ประเภท TCP Sack1
- TCP Fack เพื่อใช้ในการสร้างเอเจนต์ประเภท TCP Fack
- TCP Full เพื่อใช้ในการสร้างเอเจนต์ประเภท TCP Full
- TCP Sink เพื่อใช้ในการสร้างเอเจนต์ประเภท TCP Sink
- TCP Sink Del Ack เพื่อใช้ในการสร้างเอเจนต์ประเภท TCP Sink Delay Ack
- TCP Sink Sack1 เพื่อใช้ในการสร้างเอเจนต์ประเภท TCP Sink Sack1
- TCP Sink Sack1 Del Ack เพื่อใช้ในการสร้างเอเจนต์ประเภท TCP Sink Sack1 Delay Ack

- Traffic เพื่อใช้ในการสร้างแอปพลิเคชั่นประเภททราฟฟิก
  - Web Client เพื่อใช้ในการสร้างแอปพลิเคชั่นประเภทเว็บไคล์เอนต์
  - Web Server เพื่อใช้ในการสร้างแอปพลิเคชั่นประเภทเว็บเซิร์ฟเวอร์
  - FTP เพื่อใช้ในการสร้างแอปพลิเคชั่นประเภทเอฟทีพี
  - Telnet เพื่อใช้ในการสร้างแอปพลิเคชั่นประเภทเทลเน็ต
  - CBR เพื่อใช้ในการสร้างแอปพลิเคชั่นประเภทซีบีอาร์
  - Pareto เพื่อใช้ในการสร้างแอปพลิเคชั่นประเภทพาร์เร โต
  - Exponential เพื่อใช้ในการสร้างแอปพลิเคชั่นประเภทเอ็กซ์โพเน็นเชียล

- ปุ่ม Configuration เพื่อใช้ในการตั้งค่าต่างๆ
  - Event Handler เพื่อเปิดหน้าต่างสำหรับตั้งค่าเหตุการณ์ต่างๆ ในเครือข่าย
  - General Properties เพื่อตั้งค่าหมายเลขไอพีของเครื่องเซิร์ฟเวอร์และตั้งค่าชนิดของโปรโตคอลเลือกเส้นทาง
- ปุ่ม Run เพื่อใช้ในการจำลองเครือข่ายหรือเปิดส่วนการแสดงผล
  - Generate เพื่อแสดงผลลัพธ์ของการจำลองเครือข่ายที่สร้างขึ้น
  - Network Animator เพื่อเรียกการทำงานของโปรแกรมเน็ตเวิร์คอนิเมเตอร์
- ปุ่ม Help
  - About เพื่อเปิดข้อมูลของผู้พัฒนาโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อในโปรแกรมฟรอนต์เอ็นมีรูปแบบเงื่อนไขซึ่งสามารถสรุปเป็นตารางดังนี้

ตารางที่ ง.1 แสดงความสัมพันธ์ของการเชื่อมต่อ

ชื่อ	เชื่อมต่อได้เพียงหนึ่ง	เชื่อมต่อได้มากกว่าหนึ่ง
Node	----	ทุกชนิดยกเว้น Application ประเภท Traffic
UDP	Null , Node	Application ประเภท Traffic ทุกชนิด ยกเว้น Telnet FTP , Web Client , Web Server
NULL	UDP , Node	----
TCP Tahoe , TCP Vegas TCP Reno , TCP New Reno TCP Sack1 , TCP Fack	TCP Sink ทุกชนิด , Node	Application ประเภท Traffic ทุกชนิดเว้น Telnet Web Client , Web Server
TCP Full	TCP Sink ทุกชนิด , Node	Application ประเภท Traffic ทุกชนิด
TCP Sink Del Ack TCP Sink Sack1 TCP Sink Sack1 Del Ack	TCP ทุกชนิด , Node	Application ประเภท Traffic ทุกชนิดเว้น Telnet Web Client , Web Server
Web Client	Web Server , Node	----
Web Server	Web Client , Node	----
FTP	TCP ทุกชนิด	----
Telnet	TCP Full , Node	----
CBR	TCP ทุกชนิด , UDP	----
Pareto	TCP ทุกชนิด , UDP	----
Exponentail	TCP ทุกชนิด , UDP	----

ในการเชื่อมต่อ แต่ละสีสามารถบอกความสัมพันธ์ของการเชื่อมต่อดังนี้

สีดำ แสดงการเชื่อมต่อระหว่างโหนดกับโหนด

สีแดง แสดงการเชื่อมต่อระหว่างเอเจนต์กับโหนด

สีเหลือง แสดงการเชื่อมต่อระหว่างเอเจนต์กับเอเจนต์

สีน้ำเงิน แสดงการเชื่อมต่อระหว่างเอเจนต์กับแอปพลิเคชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้