บทที่ 3

การออกแบบระบบและการพัฒนา

กริดมักประกอบด้วยเครื่องคอมพิวเตอร์สมรรถนะสูงจำนวนหนึ่งภายในองค์กรจริงหรือจากหลายองค์กรจริงมาเชื่อมต่อกันในองค์กรเสมือน ถูกใช้ทำงานหนึ่งๆ เพื่อบรรลุวัตถุประสงค์ร่วมกันของกลุ่มเจ้าของเครื่องคอมพิวเตอร์ โครงสร้างทางกายภาพของเครือข่ายมักไม่เปลี่ยนแปลง เครื่องคอมพิวเตอร์เหล่านี้มักเป็นเครื่องคอมพิวเตอร์เซิร์ฟเวอร์ที่เป็นศูนย์กลางให้บริการแก่เครื่องลูกข่ายในองค์กร ซึ่งเครื่องเหล่านี้มีจำนวนไม่มาก

ในขณะที่เครือข่ายเพียร์ทูเพียร์มักประกอบด้วยเครื่องคอมพิวเตอร์ส่วนบุคคลจำนวนมากที่กระจายตัวอยู่บนเครือข่ายอินเตอร์เน็ต แต่ละเพียร์บนเครือข่ายสามารถเป็นทั้งผู้ให้และผู้ใช้บริการได้ เครือข่ายเพียร์ทูเพียร์มีโครงสร้างซ้อนทับที่มีความพลวัตสูง กล่าวคือแต่ละเพียร์สามารถเข้าและออกจากเครือข่ายได้เสมอ บริการบนเครือข่ายเพียร์ทูเพียร์ส่วนใหญ่เป็นบริการที่เกิดจากการร่วมมือกันระหว่างกลุ่มเพียร์ที่มีวัตถุประสงค์ร่วมกัน การให้และใช้บริการที่เกิดขึ้นบนเครือข่ายนี้จึงกระทำผ่านเพียร์กรุ๊ปที่เป็นจุดร่วมให้และใช้บริการ

การเชื่อมต่อกริดและเครือข่ายเพียร์ทูเพียร์ในงานวิจัยนี้ มุ่งเน้นที่การรวบรวมทรัพยากรบนเครือข่ายเพียร์ทูเพียร์มาเพิ่มเติมให้แก่กริด โดยมุ่งไปที่ทรัพยากรเนื้อที่เก็บข้อมูลที่กระจายตัวอยู่บนเครือข่ายเพียร์ทูเพียร์ การเชื่อมต่อเพื่อรวบรวมเนื้อที่เก็บข้อมูลนี้เป็นแบบไม่สมมาตร กล่าวคือระบบที่สร้างจะรวบรวมเนื้อที่เก็บข้อมูล รวมถึงข้อมูลจากเครือข่ายเพียร์ทูเพียร์มาเพิ่มให้แก่กริด แต่ไม่รวบรวมเนื้อที่เก็บข้อมูลรวมถึงข้อมูลจากกริดมาเพิ่มเติมให้แก่เครือข่ายเพียร์ทูเพียร์

การรวบรวมเนื้อที่เก็บข้อมูลจะกระทำผ่านระบบร่วมใช้แฟ้มข้อมูลที่ใช้งานกันบนเครือข่ายเพียร์ทูเพียร์ ซึ่งระบบร่วมใช้แฟ้มข้อมูลเป็นโปรแกรมประยุกต์ที่ใช้แพร่หลายบนเครือข่ายเพียร์ทูเพียร์อยู่เดิมแล้ว

3.1 แนวคิดการออกแบบระบบ

ในการเชื่อมต่อกริดและเครือข่ายเพียร์ทูเพียร์ จำเป็นอย่างยิ่งที่ต้องคำนึงถึงความแตกต่างของทั้งสองระบบ กริดข้อมูลนั้นให้ความสำคัญกับการจัดการข้อมูล อันรวมถึงการร่วมใช้กลุ่มข้อมูลขนาดใหญ่ที่มักจัดเก็บอยู่ในเครื่องคอมพิวเตอร์เซิร์ฟเวอร์ ในขณะที่ระบบร่วมใช้แฟ้มข้อมูลบนเพียร์ทูเพียร์จัดการกับข้อมูลขนาดเล็กที่มีอยู่จำนวนมากๆ ที่มักจัดเก็บอยู่บนเครื่อง

คอมพิวเตอร์ส่วนบุคคล จำนวนเพียร์ในเครือข่ายเพียร์ทูเพียร์อาจมีมากกว่ากริดโฮสต์ในกริดมาก และการปรากฏตัวของเพียร์ก็ขึ้นอยู่กับความต้องการของผู้ใช้บนเพียร์ ซึ่งส่งผลให้ทรัพยากรบนเพียร์ทูเพียร์มีความพลวัตสูง

ดังนั้นการออกแบบระบบเชื่อมต่อจึงมีหลักแนวคิดที่สำคัญสามประการ คือ

1. แต่ละเพียร์บนเครือข่ายเพียร์ทูเพียร์จะสามารถเข้าถึงได้โดยตรงจากเพียร์อื่นๆ แต่การเข้าถึงเนื้อที่ข้อมูลบนแต่ละเพียร์โดยตรงจากกริดจะไม่สะดวกและไม่เกิดประโยชน์เท่าที่ควร อันเนื่องมาจากจำนวนเพียร์ที่มีอยู่มาก เมื่อพิจารณาปัญหาการเข้าถึงกลุ่มทรัพยากรเช่นนี้บนกริดแล้ว จะเห็นว่าแนวทางแก้ปัญหาคือการใช้รูปแบบการประมวลผลแบบคลัสเตอร์ ซึ่งมีโหนดมาสเตอร์ที่สามารถติดต่อกับภายนอกได้ ในขณะที่โหนดมาสเตอร์เองสามารถติดต่อกับโหนดลูกที่เป็นโหนดประมวลผล เราเลือกใช้กลไกการสร้างเพียร์กรุ๊ปที่ทำหน้าที่คล้ายการจัดกลุ่มทรัพยากรแบบคลัสเตอร์ โดยเพียร์กรุ๊ปคือกลไกการรวมรวบเพียร์ (ซึ่งรวมถึงทรัพยากรเพียร์) ที่มีความสนใจบริการหนึ่งร่วมกันมารวมกลุ่มกัน โดยเราจะติดต่อผ่านซุปเปอร์เพียร์ที่เก็บรวบรวมข้อมูลเพียร์ต่างๆ ในเพียร์กรุ๊ปไว้ด้วย
2. การเชื่อมต่อเป็นแบบไม่สมมาตร (Asymmetric) นั่นคือ เพียร์กรุ๊ปจะทำตัวเป็นเสมือนทรัพยากรกริด ให้บริการแก่ทุกกริดโฮสต์บนกริด ในขณะที่กริดโฮสต์ไม่ได้ทำตัวเป็นเพียร์บนเครือข่ายเพียร์ทูเพียร์
3. การเชื่อมต่อต้องโปร่งใส (Transparent) ต่อผู้ใช้กริดและโปรแกรมประยุกต์บนกริด กล่าวคือทำให้ผู้ใช้กริดและโปรแกรมประยุกต์บนกริดสามารถจัดเก็บ (Put) และเรียกคืน (Get) บนเนื้อที่จัดเก็บเสมือนนี้ได้เสมือนกระทำกับกริดโฮสต์อื่นๆ การเข้าถึงเนื้อที่จัดเก็บเสมือนไม่ควรเป็นวิธีการใหม่หรือใช้คำสั่งใหม่ที่ทำให้ผู้ใช้กริดต้องเรียนรู้เพิ่มหรือแก้ไขวิธีการเขียนโปรแกรมประยุกต์บนกริด

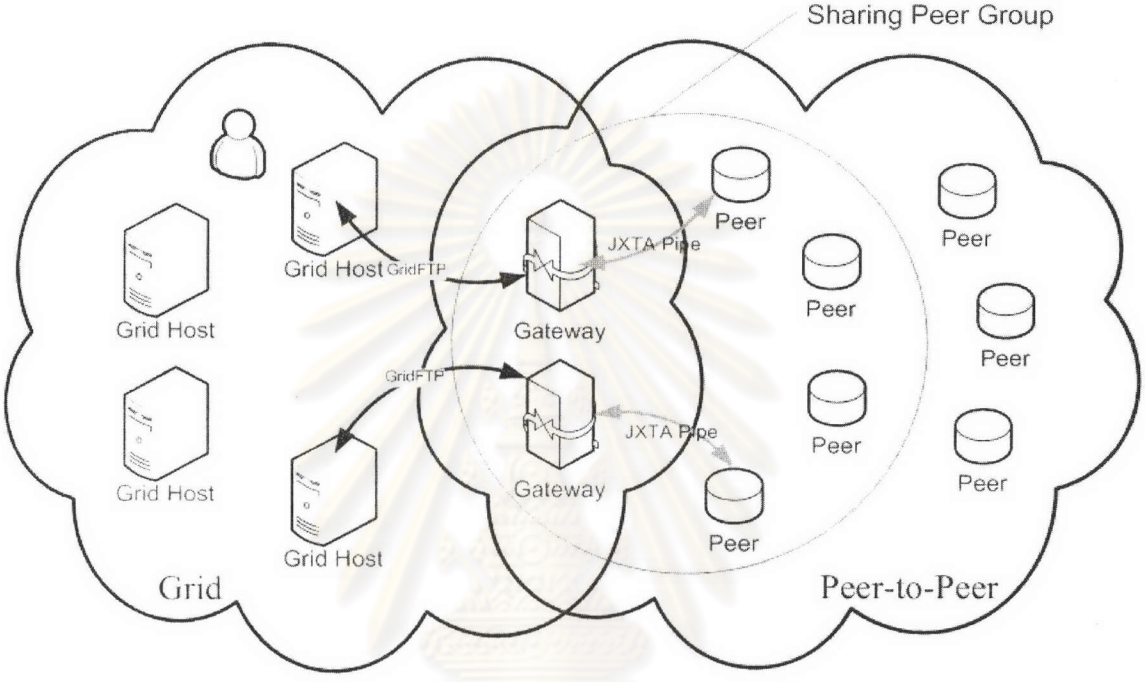
3.2 สถาปัตยกรรมของระบบ

ระบบที่เชื่อมต่อแล้วประกอบด้วยสามส่วนองค์ประกอบย่อยคือ กริดโฮสต์ เพียร์ เกตเวย์

กริดโฮสต์คือเครื่องที่ผู้ใช้กริดหรือโปรแกรมประยุกต์บนกริดทำงานอยู่ ซึ่งกริดโฮสต์กระจายตัวอยู่บนกริด ในขณะที่เพียร์ซึ่งกระจายตัวอยู่บนเครือข่ายเพียร์ทูเพียร์ก็สามารถเข้าร่วมเพียร์กรุ๊ปและรันโปรแกรมประยุกต์ร่วมใช้แฟ้มข้อมูล ที่ใช้เป็นกลไกในการเข้าใช้เนื้อที่จัดเก็บข้อมูลและเข้าถึงแฟ้มข้อมูล ช่วยทำให้เกิดเนื้อที่จัดเก็บข้อมูลเสมือนเดียวได้

เกตเวย์เป็นสมาชิกของทั้งกริดและเครือข่ายเพียร์ทูเพียร์ในเวลาเดียวกัน มีความสามารถของทั้งสองระบบ สำหรับการอิมพลีเมนต์แล้ว เกตเวย์มีความสามารถจากการรันทั้งโกลบัสทูลคิทและจักซ์ตา เกตเวย์วางตัวอยู่บนทรัพยากรกริดเพื่อให้มั่นใจว่าจะให้บริการได้ถาวร โดยทำหน้าที่เป็นจุดบริการ (Access point) แก่กริดโฮสต์สู่เนื้อที่จัดเก็บข้อมูลเสมือน

แบบจำลองเชิงนามธรรมของการเชื่อมต่อเป็นดังรูปที่ 3.1



รูปที่ 3.1 แบบจำลองเชิงนามธรรมของการเชื่อมต่อ

3.3 กลไกเนื้อที่เก็บข้อมูลเสมือน

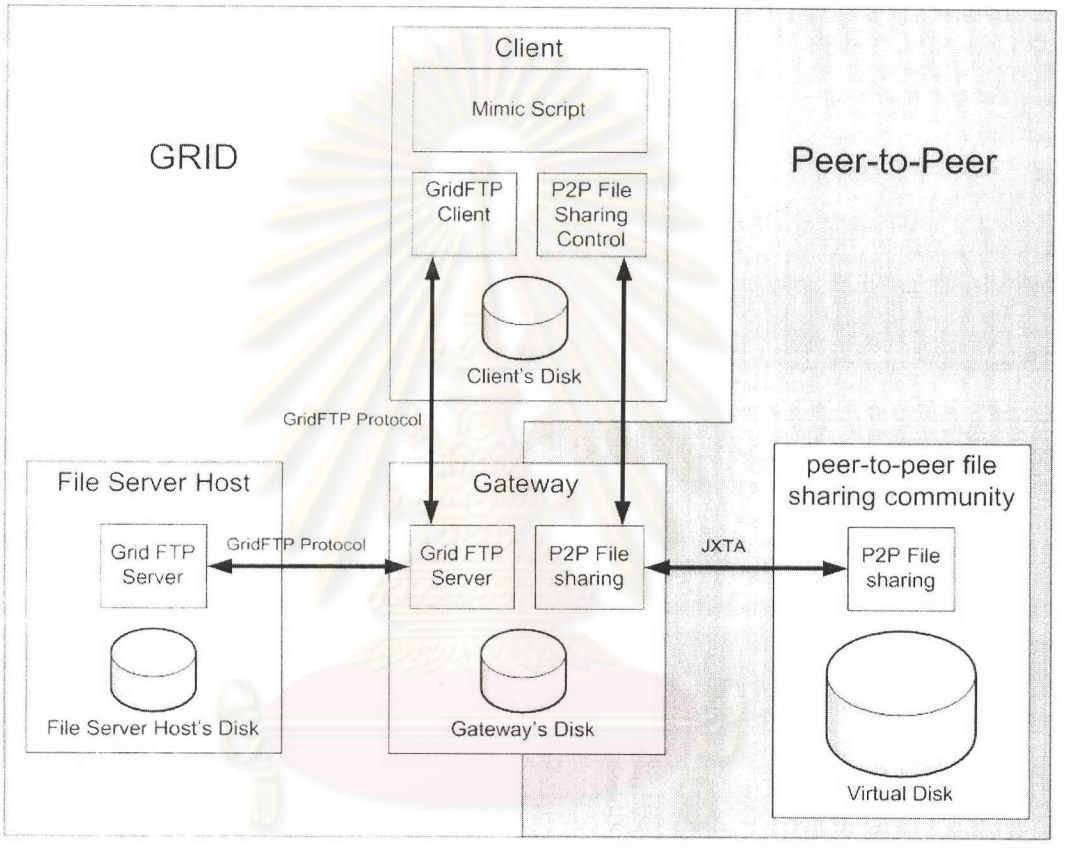
ระบบจะประกอบด้วยสามส่วนประกอบหลักคือ ไคลเอนต์สำหรับเรียกใช้บริการถ่ายโอนข้อมูลที่สามารถใช้โพรโทคอลเข้าถึงแฟ้มข้อมูล p2p:// ที่อยู่บนเนื้อที่เก็บข้อมูลเสมือน ส่วนเชื่อมต่อที่เป็นเสมือนผู้เชื่อมเนื้อที่เก็บข้อมูลเสมือนให้กับกริด และระบบร่วมใช้แฟ้มข้อมูลแบบเพียร์ทูเพียร์ที่มีโปรแกรมประยุกต์ร่วมใช้แฟ้มข้อมูลเป็นเครื่องมือร่วมใช้แฟ้มข้อมูล ดังรูปที่ 2

ส่วนไคลเอนต์ช่วยซ่อนกระบวนการเข้าถึงเนื้อที่เก็บข้อมูลเสมือนต่อผู้ใช้และโปรแกรมประยุกต์บนกริด โดยระบุการเข้าถึงด้วยโพรโทคอลเข้าถึงข้อมูล p2p:// ที่เพิ่มแก่กริดเอฟทีพีไคลเอนต์ซึ่งงานวิจัยนี้เลือกใช้กริดเอฟทีพีไคลเอนต์ชื่อ globus-url-copy ที่นิยมและแพร่หลาย ใช้โพรโทคอลกริดเอฟทีพีมาตรฐาน และมีอยู่ในโกลบัสทูลคิทแล้ว

ส่วนเชื่อมต่อทำหน้าที่ปฏิบัติงานการถ่ายโอนแฟ้มข้อมูลระหว่างทรัพยากรของระบบด้วยโพรโทคอลการถ่ายโอนที่เหมาะสม กล่าวคือการถ่ายโอนระหว่างส่วนเชื่อมต่อกับเครือข่ายเพียร์ทู

เพียร์ก็จะใช้กลไกการถ่ายโอนไปป์ของจักซ์ตา ส่วนการถ่ายโอนระหว่างส่วนเชื่อมต่อกับกริดโฮสต์ก็จะใช้กลไกการถ่ายโอนตามโพรโทคอลกริดเอฟทีพี

ระบบร่วมใช้แฟ้มข้อมูลแบบเพียร์ทูเพียร์ทำงานบนเฟรมเวิร์คจักซ์ตาโดยใช้โปรแกรมประยุกต์เจนูแชร์ (Jnushare) เจนูแชร์ใช้จีไอเอสพีเป็นกลไกดัชนีแบบกระจายบนเครือข่ายเพียร์ทูเพียร์ กลไกดัชนีแบบกระจายบนเครือข่ายเพียร์ทูเพียร์นี้ใช้เป็นเครื่องมือในการกระจายความรับผิดชอบดัชนีของข้อมูลและข้อมูล



รูปที่ 3.2 แสดงส่วนส่วนประกอบหลักของการเชื่อมต่อ

อธิบายการทำงานของแต่ละส่วนของกลไกจัดเก็บข้อมูลเสมือนดังนี้

3.3.1 ส่วนไคลเอนต์

ส่วนไคลเอนต์ทำหน้าที่สร้างความโปร่งใสในการเรียกใช้การถ่ายโอนข้อมูลที่สามารถเข้าถึงเนื้อที่เก็บข้อมูลเสมือนให้แก่ผู้ใช้และโปรแกรมประยุกต์บนกริด โดยใช้กลไกการแทรกขั้นตอนด้วยบทคำสั่งเลียนแบบ (Mimic script) แทนการเรียกใช้คำสั่งถ่ายโอนข้อมูลจริง ด้วยวิธีการนี้จะทำให้ผู้ใช้และโปรแกรมประยุกต์บนกริดสามารถเรียกใช้การถ่ายโอนข้อมูลในรูปแบบคำสั่งเดิม โดยสามารถเข้าถึงเนื้อที่เก็บข้อมูลเสมือนได้ด้วยการระบุโพรโทคอลเข้าถึงข้อมูลที่กำหนดใหม่ คือ p2p:// โดยการถ่ายโอนข้อมูลระหว่างทรัพยากรกริดและส่วนเชื่อมต่อจะเป็นไป

ตามโพรโทคอลกริดเอฟทีพี บทคำสั่งเลียนแบบและคำสั่งจริงที่เป็นไคลเอนต์ของกริดเอฟทีพีมีชื่อ globus-url-copy ซึ่งเป็นไคลเอนต์ของกริดเอฟทีพีที่เป็นที่นิยมและมีมากับโกลบัสทูลคิทอยู่แล้ว

รูปแบบคำสั่งเดิมของกริดเอฟทีพี

```
globus-url-copy Source_URL Destination_URL
```

โดย Source_URL ระบุถึง URL ของแฟ้มข้อมูลที่จะถูกคัดลอกและ Destination_URL ระบุถึง URL ของแฟ้มข้อมูลเป้าหมายที่จะสร้างเนื่องจากการคัดลอก

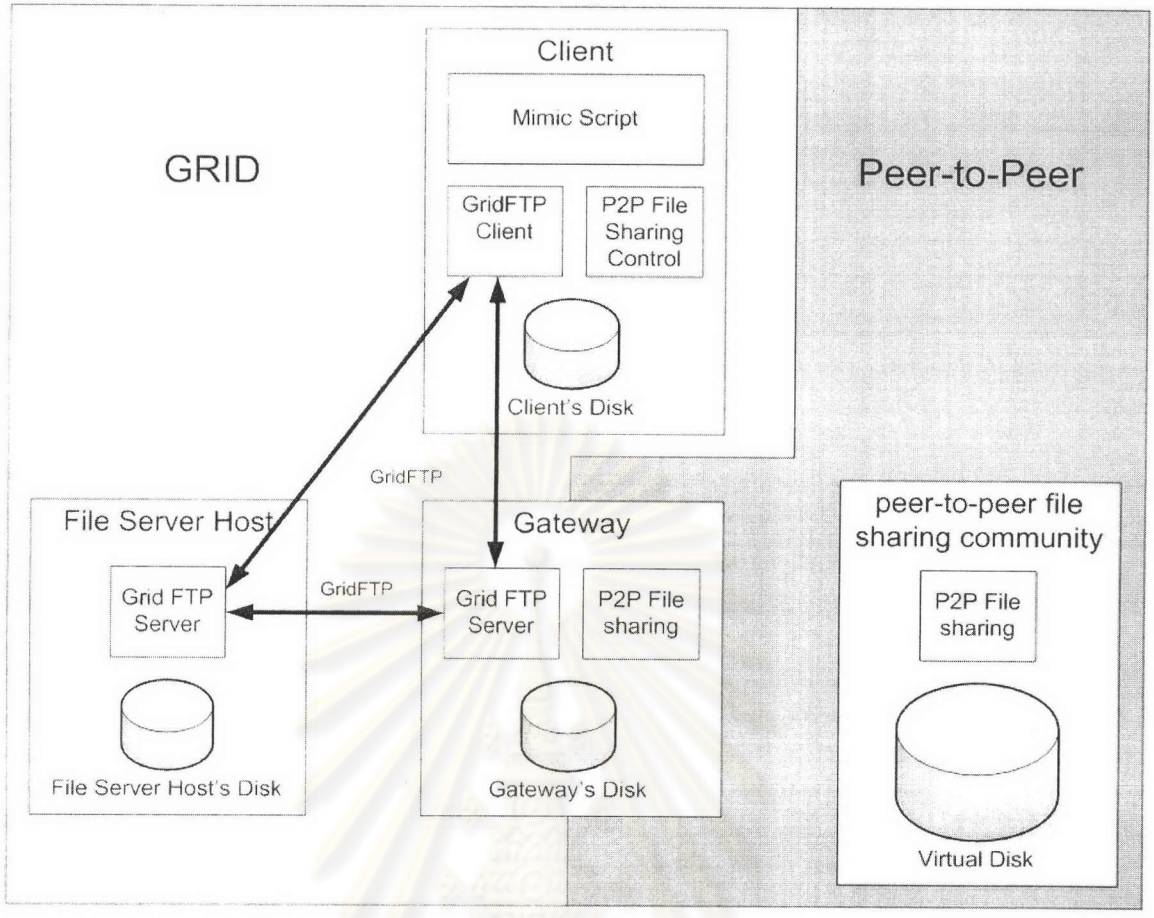
โดยโพรโทคอลเข้าถึงข้อมูลประกอบด้วย http:// https:// ftp:// gsiftp:// file:// สังเกตว่าโพรโทคอลเข้าถึงข้อมูล file:// จะใช้ในการระบุ URL ของแฟ้มข้อมูลที่โลคอล

เมื่อมีบทคำสั่งเลียนแบบ การเรียกใช้จะยังคงรูปแบบคำสั่งเดิม แต่มีโพรโทคอลเข้าถึงข้อมูลเพิ่มมาคือ p2p:// ซึ่งเป็นโพรโทคอลเข้าถึงข้อมูลที่กำหนดโฮสต์เป้าหมายซึ่งมีส่วนเชื่อมต่อให้บริการอยู่ ซึ่งทราบได้จากผู้ดูแลระบบกริดพร้อมพอร์ต โดยการอ้างอิงแฟ้มข้อมูลเป็นแบบสมบูรณ์ ด้วยกลไกสร้างเนื้อที่เก็บข้อมูลเสมือน จะทำให้ทุกส่วนเชื่อมต่อที่ร่วมใช้แฟ้มข้อมูลบนเพียร์กรุ๊ปเดียวกัน เห็นระบบจัดเก็บแฟ้มข้อมูลเหมือนกัน ตัวอย่างการอ้างอิงด้วยการใช้โพรโทคอลเข้าถึงข้อมูล p2p เช่น p2p://sky.cp.eng.chula.ac.th:9811/foo.dat

3.3.1.1 กรณีที่ต้นทางและปลายทางในคำสั่งไม่มีการอ้าง p2p://

ในกรณีที่ Source_URL และ Destination_URL ไม่มีการใช้โพรโทคอลเข้าถึงข้อมูล p2p:// บทคำสั่งเลียนแบบของส่วนไคลเอนต์จะส่งต่อคำสั่งโดยตรงไปยังคำสั่งจริง globus-url-copy ซึ่งจะใช้โพรโทคอลกริดเอฟทีพีในการคัดลอกข้อมูลตามเดิม การส่งถ่ายข้อมูลเกิดขึ้นได้ระหว่างทรัพยากรกริดดังแสดงรูปที่ 3.3

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.3 การเชื่อมต่อที่เกิดขึ้นได้เมื่อคำสั่งไม่มีการอ้าง p2p://

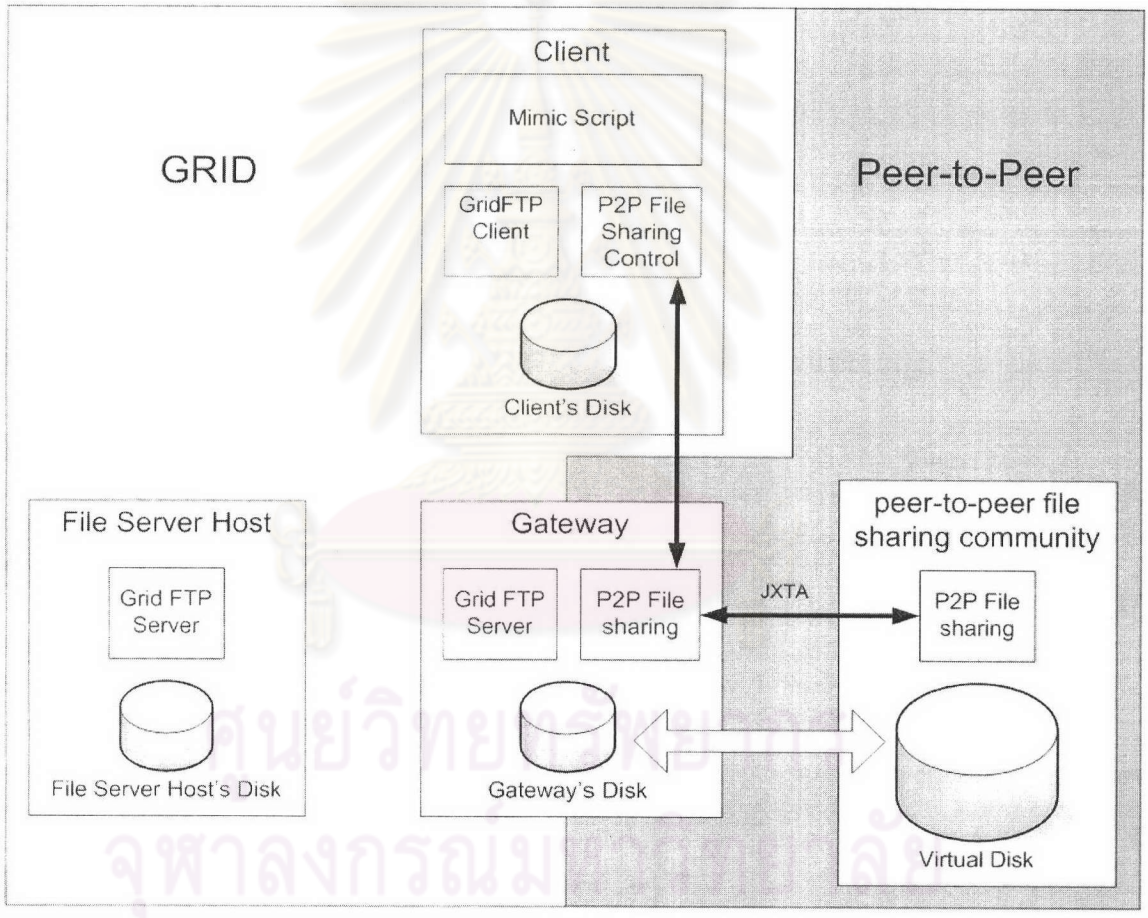
จากรูปจะเห็นว่า ส่วนเชื่อมต่อที่อยู่บนโฮสต์ในกริด ก็เป็นดังเช่นทรัพยากรกริดทั่วไป บทคำสั่งเลียนแบบจะเรียกใช้กริดเอฟทีพีไคลเอนต์ คือ globus-url-copy ในการสื่อสารกับกริดเอฟทีพีเซิร์ฟเวอร์บนโฮสต์ในกริดรวมถึงโฮสต์ที่มีส่วนเชื่อมต่อด้วย จากรูปการถ่ายโอนข้อมูลระหว่างโฮสต์โดยมีกริดเอฟทีพีไคลเอนต์เป็นบุคคลที่สามในการควบคุมสามารถทำได้

3.3.1.2 กรณีต้นทางและปลายทางในคำสั่งอ้าง p2p://

ในกรณีที่ Source_URL และ Destination_URL ใช้โพรโทคอลเข้าถึงข้อมูล p2p:// ทั้งคู่นั้น บทคำสั่งเลียนแบบจะใช้ส่วนควบคุมการร่วมใช้แฟ้มข้อมูลแบบเพียร์ทูเพียร์ที่ไคลเอนต์ในการส่งคำร้องไปยังส่วนการร่วมใช้แฟ้มข้อมูลแบบเพียร์ทูเพียร์ที่ส่วนเชื่อมต่อของโฮสต์ที่ได้จาก Source_URL เพื่อให้ส่วนเชื่อมต่อเรียกคืนแฟ้มข้อมูลมาเก็บไว้ยังพาธที่ได้ใน Desination_URL บนส่วนเชื่อมต่อ สังเกตว่าโฮสต์ใน Destination_URL จะถูกละเลยความสนใจ เนื่องจากแม้จะถูกอ้างส่วนเชื่อมต่อที่โฮสต์ที่แตกต่างจาก Source_URL แต่ก็จะถูกจัดเก็บลงบนเนื้อที่จัดเก็บข้อมูลเสมือนหนึ่งเดียวกัน กรณีนี้จึงเป็นเสมือนการคัดลอกแฟ้มข้อมูลบนเนื้อที่จัดเก็บข้อมูลเสมือนก้อนเดียวกัน ซึ่งหากพาธที่ตามหลังโฮสต์และชื่อแฟ้มข้อมูลของ Source_URL และ Destination_URL

ตรงกัน บทคำสั่งเลียนแบบจะตอบกลับความล้มเหลวเพื่อแจ้งเตือนกลับไป รวมถึงกรณีที่แฟ้มข้อมูลเป้าหมายใน Destination_URL มีอยู่เดิมแล้วด้วย

การคัดลอกแฟ้มข้อมูลจากเครือข่ายเพียร์ทูเพียร์มายังส่วนเชื่อมต่อใช้กลไกระบบจัดการเนื้อหา (Content Management System: CMS) ของจักซ์ตา ซึ่งใช้ไปป์เป็นกลไกส่งถ่ายข้อมูล หลังจากคัดลอกแฟ้มข้อมูลมาที่ส่วนเชื่อมต่อที่พาธที่กำหนดแล้ว การจัดเก็บแฟ้มข้อมูลใหม่จะปล่อยให้เป็นไปตามกระบวนการความรับผิดชอบแฟ้มข้อมูลใหม่นี้ สังเกตว่าระบบจัดเก็บแฟ้มข้อมูลบนเนื้อที่จัดเก็บข้อมูลเสมือนนี้ยังไม่รองรับสารบัญการทำซ้ำ (Replication catalog) ว่าแฟ้มข้อมูลทั้งสองคือแฟ้มข้อมูลเดียวกัน การส่งถ่ายข้อมูลเกิดขึ้นระหว่างส่วนเชื่อมต่อและเครือข่ายเพียร์ทูเพียร์ดังรูปที่ 3.4



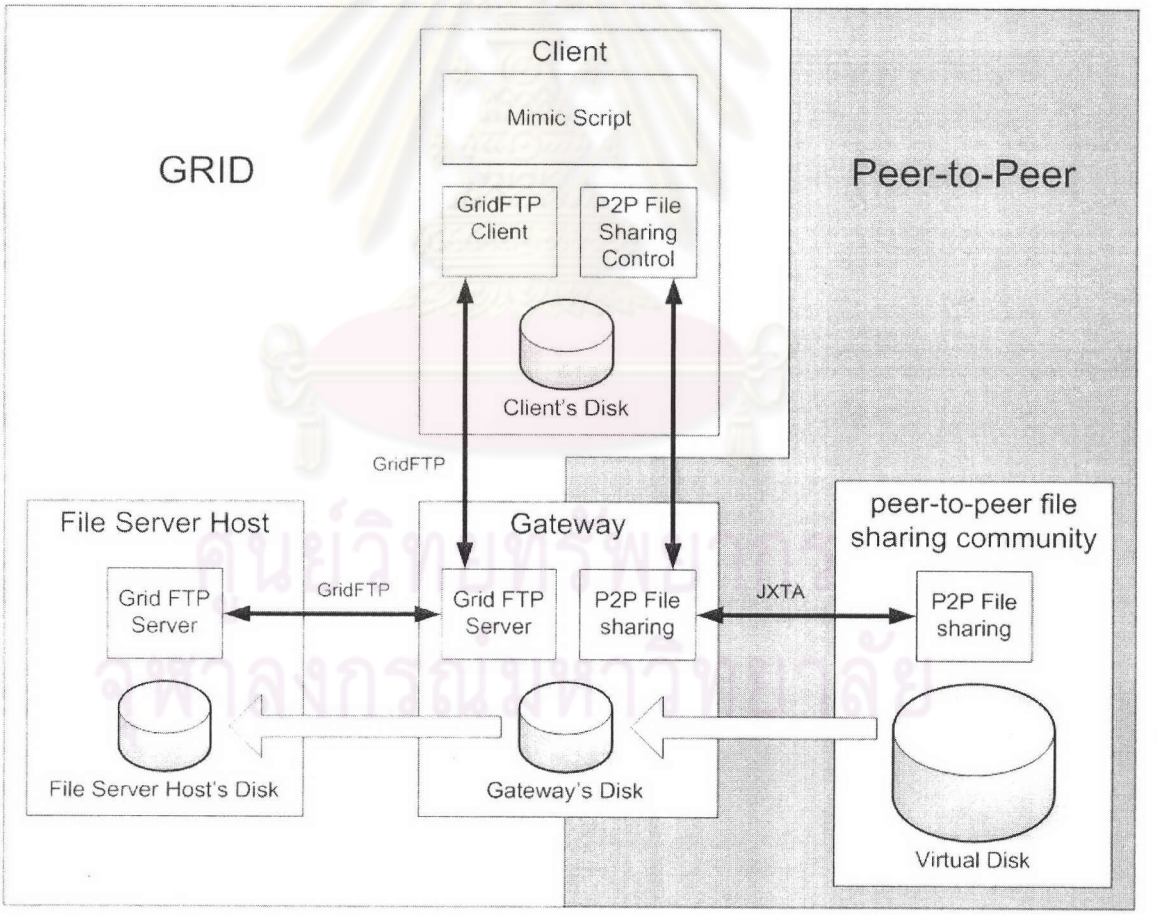
รูปที่ 3.4 การเชื่อมต่อที่เกิดขึ้นเมื่อคำสั่งอ้าง p2p:// ทั้งต้นทางและปลายทาง

3.3.1.3 กรณีต้นทางในคำสั่งอ้าง p2p:// และปลายทางในคำสั่งไม่อ้าง p2p://

ในกรณี Source_ URL ใช้โพรโทคอลเข้าถึงข้อมูล p2p:// และ Destination_URL ไม่ใช้โพรโทคอลเข้าถึงข้อมูล p2p:// บทคำสั่งเลียนแบบจะใช้ส่วนควบคุมการร่วมใช้แฟ้มข้อมูลแบบเพียร์ทูเพียร์ที่ไคลเอนต์ในการส่งคำร้องไปยังส่วนการร่วมใช้แฟ้มข้อมูลแบบเพียร์ทูเพียร์ที่ส่วน

เชื่อมต่อของโฮสต์ที่ได้จาก Source_URL เพื่อให้ส่วนเชื่อมต่อเรียกคืนแฟ้มข้อมูลมาเก็บไว้ยังพาธที่ได้ในเนื้อที่พักชั่วคราวบนส่วนเชื่อมต่อ จากนั้นบทคำสั่งเลียนแบบจะคัดลอกข้อมูลที่เนื้อที่พักชั่วคราวบนส่วนเชื่อมต่อมายังส่วนไคลเอนต์ด้วยการเรียกใช้กริดเอฟทีพีไคลเอนต์ globus-url-copy โดยมี Source_URL เป็นโฮสต์เกตเวย์ที่ติดต่อตามด้วยพาธเนื้อที่พักชั่วคราวและชื่อแฟ้มข้อมูล ส่วน Destination_URL ให้คงเดิมของการเรียกใช้เดิม

สังเกตว่ากรณีที่ Destination_URL ของการเรียกใช้บทคำสั่งเลียนแบบมีโพรโทคอลเข้าถึงแฟ้มข้อมูลเป็น file:// กริดเอฟทีพีไคลเอนต์ของไคลเอนต์จะรับข้อมูลจากส่วนการเชื่อมต่อ แต่หาก Destination_URL ของการเรียกใช้บทคำสั่งเลียนแบบมีโพรโทคอลเข้าถึงแฟ้มข้อมูลเป็น http:// หรือ https:// หรือ ftp:// หรือ gsiftp:// แล้วกริดเอฟทีพีไคลเอนต์จะทำการสั่งกริดเอฟทีพีเซิร์ฟเวอร์บนส่วนเชื่อมต่อถ่ายโอนข้อมูลจากเนื้อที่พักข้อมูลชั่วคราวของตนไปยังกริดเอฟทีพีเซิร์ฟเวอร์ปลายทางที่ระบุไว้ใน Destination_URL ที่พาธที่กำหนดไว้ ซึ่งกระบวนการนี้เป็นการถ่ายโอนข้อมูลโดยบุคคลที่สาม ดังรูปที่ 3.5



รูปที่ 3.5 การถ่ายโอนข้อมูลโดยบุคคลที่สามเมื่อต้นทางในคำสั่งอ้าง p2p:// และปลายทางในคำสั่งไม่อ้าง p2p:// และ file://

3.3.1.4 **กรณีต้นทางในคำสั่งไม่อ้าง p2p:// และปลายทางในคำสั่งอ้าง p2p://**

ในกรณี Source_ URL ไม่ใช้โพรโทคอลเข้าถึงข้อมูล p2p:// และ Destination_URL ใช้โพรโทคอลเข้าถึงข้อมูล p2p:// จะคล้ายกันกับ กรณี Source_ URL ใช้โพรโทคอลเข้าถึงข้อมูล p2p:// และ Destination_URL ไม่ใช้โพรโทคอลเข้าถึงข้อมูล p2p:// คือการใช้ส่วนเชื่อมต่อเป็นตัวกลางพักข้อมูลชั่วคราว โดยสลับขั้นการถ่ายโอนด้วยการถ่ายโอนแฟ้มข้อมูลจากโฮสต์ต้นทางที่ได้จาก Source_URL มาที่ส่วนเชื่อมต่อก่อน ส่วนการถ่ายโอนจากส่วนเชื่อมต่อไปยังเครือข่ายเพียร์ทูเพียร์ สังเกตว่าจะไม่ต้องสั่งด้วยส่วนไคลเอนต์ เนื่องจากจะต้องผ่านกระบวนการกระจายความรับผิดชอบก่อน จากนั้นเพียร์รับผิดชอบบนเครือข่ายเพียร์ทูเพียร์จะดึงข้อมูลไปอัตโนมัติ

3.3.2 **ส่วนเชื่อมต่อ**

เพื่อให้การถ่ายโอนแฟ้มข้อมูลระหว่างระบบกริดและระบบเพียร์ทูเพียร์สอดคล้องกับกลไกการทำงานของแต่ละระบบ โดยไม่ส่งผลกระทบต่อระบบใดระบบหนึ่ง หรือต้องมีการเปลี่ยนแปลงระบบใดระบบหนึ่ง ส่วนเชื่อมต่อจึงจำเป็นต้องมีความสามารถด้านการถ่ายโอนแฟ้มข้อมูลทั้งสองทำหน้าที่รับภาระการแปลงกลไกการเข้าถึงแฟ้มข้อมูลบนระบบหนึ่งไปยังอีกระบบหนึ่งด้วยกลไกการเข้าถึงแฟ้มข้อมูลที่สอดคล้องกับแต่ละระบบ โดยในงานวิจัยนี้มุ่งเน้นการเข้าถึงแฟ้มข้อมูลที่กระจายตัวอยู่บนระบบเพียร์ทูเพียร์จากระบบกริดแบบทางเดียว

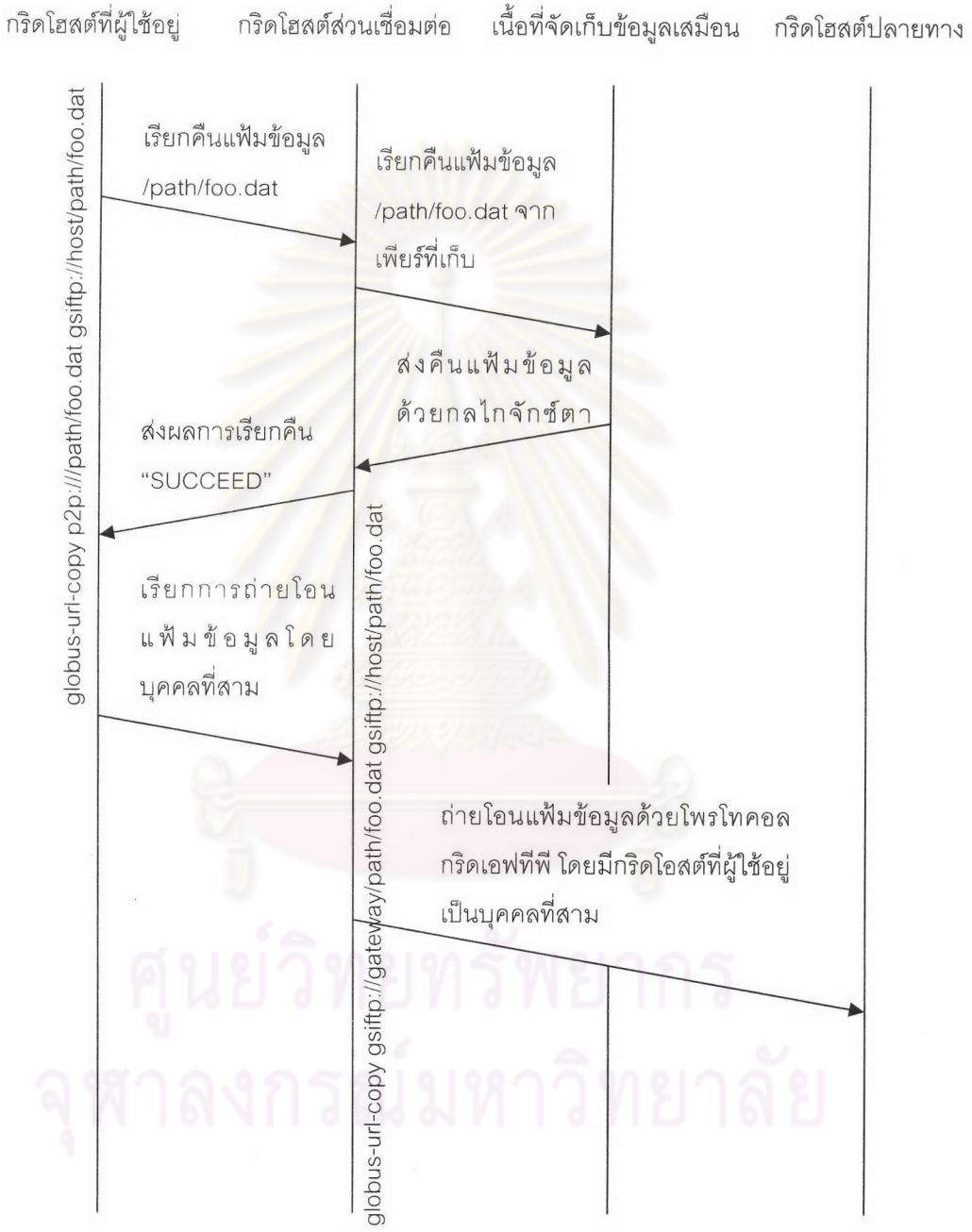
ส่วนเชื่อมต่อของระบบที่ออกแบบประกอบด้วยสองส่วนหลักคือ

- ส่วนรองรับกลไกการถ่ายโอนแฟ้มข้อมูลแบบกริด
- ส่วนรองรับกลไกการถ่ายโอนแฟ้มข้อมูลแบบเพียร์ทูเพียร์

3.3.2.1 **ส่วนรองรับกลไกการถ่ายโอนแฟ้มข้อมูลแบบกริด**

ส่วนรองรับกลไกการถ่ายโอนแฟ้มข้อมูลแบบกริดสร้างขึ้นด้วยกริดเอฟทีพีเซิร์ฟเวอร์ กริดเอฟทีพีเซิร์ฟเวอร์นี้จะทำหน้าที่รองรับการควบคุมการถ่ายโอนแฟ้มข้อมูลระหว่างกริดโฮสต์อื่นๆ กับกริดโฮสต์ที่ส่วนเชื่อมต่อเองนี้ทำงานอยู่ตามโพรโทคอลกริดเอฟทีพีมาตรฐาน กริดเอฟทีพีเซิร์ฟเวอร์ที่ส่วนเชื่อมต่อจะทำหน้าที่รับแฟ้มข้อมูลจากกริดโฮสต์ต่างๆ เมื่อผู้ใช้กริดหรือโปรแกรมประยุกต์บนกริดต้องการจัดเก็บแฟ้มข้อมูลลงในเนื้อที่จัดเก็บแฟ้มข้อมูลเสมือนบนระบบเพียร์ทูเพียร์ และกริดเอฟทีพีเซิร์ฟเวอร์ที่ส่วนเชื่อมต่อจะทำหน้าที่รับคำสั่งการถ่ายโอนแฟ้มข้อมูลจากกริดโฮสต์ที่ส่วนเชื่อมต่อนี้ทำงานอยู่ไปยังกริดโฮสต์ใดๆ รวมถึงกริดโฮสต์ที่ผู้ใช้กริดหรือโปรแกรมประยุกต์บนกริดเป็นผู้ส่งคำสั่งการถ่ายโอนแฟ้มข้อมูลด้วย การถ่ายโอนแฟ้มข้อมูลจากกริดโฮสต์ที่ส่วนเชื่อมต่อนี้ทำงานอยู่ไปยังกริดโฮสต์ใดๆ ที่ไม่ใช่กริดโฮสต์ที่ผู้ใช้กริดหรือโปรแกรมประยุกต์บน

กริดทำงานอยู่ซึ่งมีส่วนไคลเอนต์เป็นผู้ส่งคำสั่งการถ่ายโอนแฟ้มข้อมูล จะทำงานในรูปแบบการถ่ายโอนโดยมีกริดโฮสต์ที่ผู้ใช้กริดหรือโปรแกรมประยุกต์บนกริดทำงานอยู่ซึ่งเป็นผู้ส่งคำสั่งการถ่ายโอนแฟ้มข้อมูลเป็นดังบุคคลที่สาม รูปที่ 3.6 แสดงขั้นตอนการถ่ายโอนแฟ้มข้อมูลของส่วนรองรับกลไกการถ่ายโอนแฟ้มข้อมูลแบบกริด



รูปที่ 3.6 ขั้นตอนของกลไกการถ่ายโอนแฟ้มข้อมูลแบบกริด

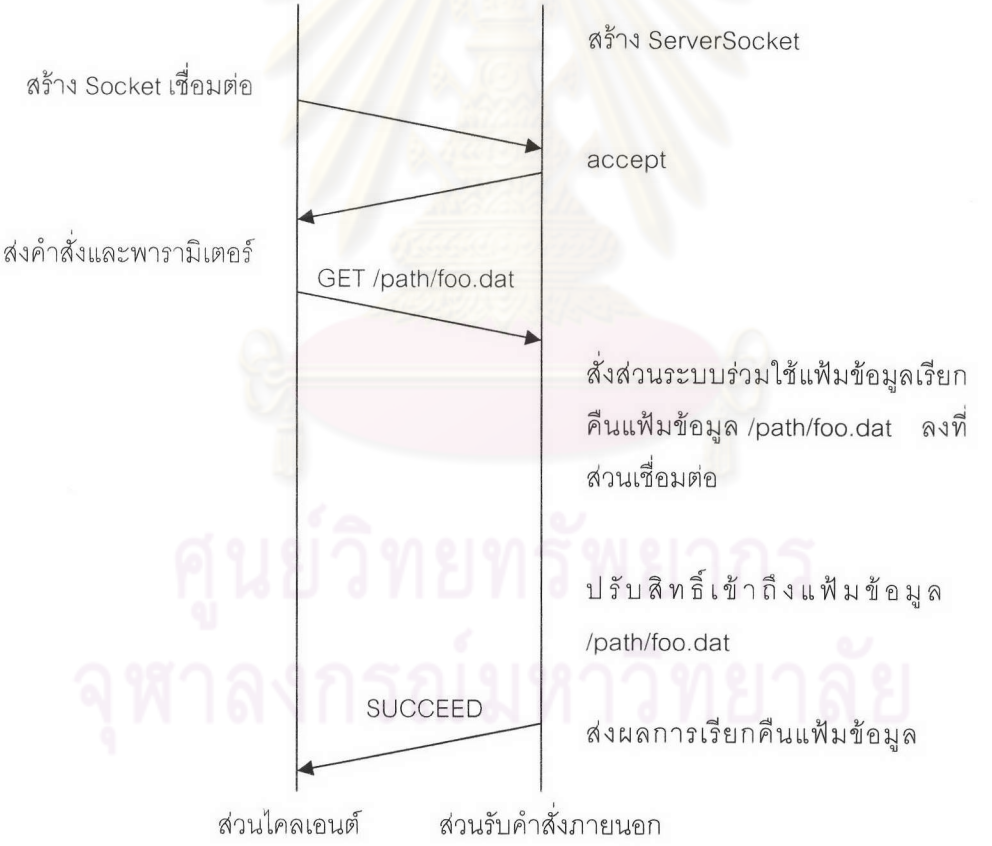
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

3.3.2.2 ส่วนรองรับกลไกการถ่ายโอนแฟ้มข้อมูลแบบเพียร์ทูเพียร์

ส่วนรองรับกลไกการถ่ายโอนแฟ้มข้อมูลแบบเพียร์ทูเพียร์ประกอบด้วยสองส่วนหลักคือ

- ส่วนรับคำสั่งภายนอกจากส่วนไคลเอนต์
- ส่วนควบคุมระบบร่วมใช้แฟ้มข้อมูล

ส่วนรับคำสั่งภายนอกจากส่วนไคลเอนต์ ทำหน้าที่รอรับการเชื่อมต่อและคำสั่งจัดการแฟ้มข้อมูลบนเนื้อที่จัดเก็บข้อมูลเสมือนบนระบบเพียร์ทูเพียร์จากส่วนไคลเอนต์ที่ผู้ใช้กริดหรือโปรแกรมประยุกต์บนกริดเรียกใช้ คำสั่งภายนอกจากส่วนไคลเอนต์เป็นคำสั่งปฏิบัติการพื้นฐาน ได้แก่ จัดเก็บ “PUT”, เรียกคืน “GET” และแสดงรายการ “LIST” โดยส่วนรับคำสั่งภายนอกจากส่วนไคลเอนต์นี้จะเรียกใช้การทำงานในส่วนควบคุมระบบร่วมใช้แฟ้มข้อมูลต่ออีกทีในโหมดตามสั่ง (manual) รูปที่ 3.7 แสดงขั้นตอนการโต้ตอบระหว่างส่วนไคลเอนต์และส่วนรับคำสั่งภายนอกจากส่วนไคลเอนต์ของส่วนเชื่อมต่อ



รูปที่ 3.7 ขั้นตอนของส่วนรับคำสั่งภายนอกจากส่วนไคลเสวนควบคุมระบบร่วมใช้แฟ้มข้อมูล

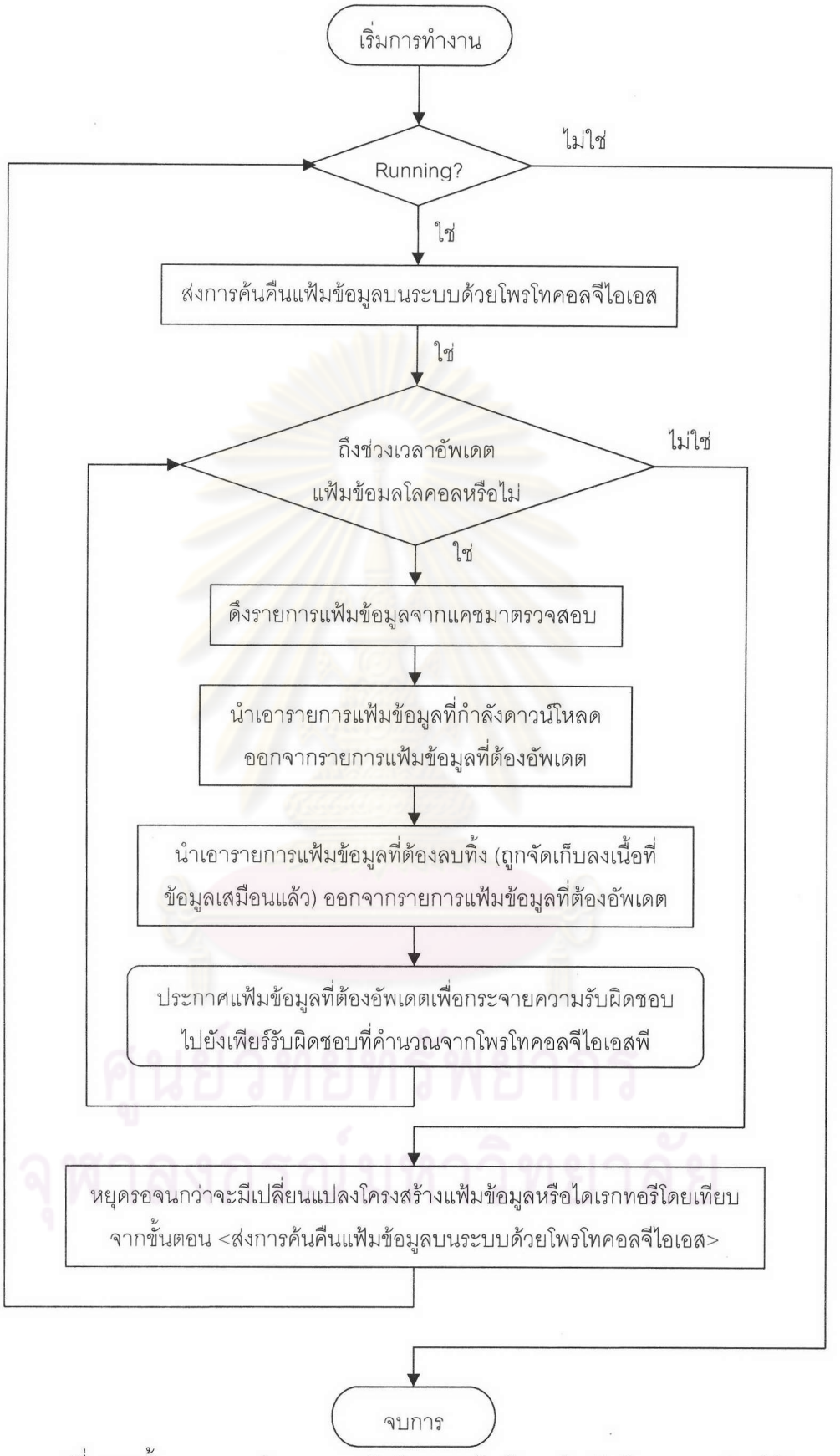
ส่วนควบคุมระบบร่วมใช้แฟ้มข้อมูลบนระบบเพียร์ทูเพียร์ประกอบด้วยสองส่วนหลักคือตัวจัดการการร่วมใช้ (Share Manager) และส่วนจีไอเอสพี โดยทั้งสองส่วนหลักนี้ทำงานบนแพลตฟอร์มจักซ์ตาสำหรับการสร้างระบบเครือข่ายเพียร์ทูเพียร์

ตัวจัดการการร่วมใช้ทำหน้าที่ดูแลการร่วมใช้แฟ้มข้อมูลภายในระบบเพียร์ทูเพียร์โดยทำงานบนโพรโทคอลจักซ์ตา การดูแลการร่วมใช้แฟ้มข้อมูลประกอบด้วยการทำงานย่อยคือ

- การอัพเดตแฟ้มข้อมูลแบบโลคอลบนส่วนเชื่อมต่อ พร้อมทั้งการสร้างและประกาศแอดเวอทิสเมนต์ของแฟ้มข้อมูลแบบโลคอล
- การสร้างแคชของคอนเทนต์แอดเวอทิสเมนต์ (Content advertisement) ที่ได้รับของแต่ละแฟ้มข้อมูลภายในระบบร่วมใช้แฟ้มข้อมูล โดยทำการแมปคอนเทนต์แอดเวอทิสเมนต์สำหรับแต่ละแฟ้มข้อมูลให้ถูกต้อง
- การอัพเดตแคชของรายการไดเรกทอรีและแฟ้มข้อมูลตามคำสั่งของส่วนไคลเอนต์หรือจากการกระตุ้นอันเนื่องจากการเกิดขึ้นใหม่ของไดเรกทอรีหรือแฟ้มข้อมูล
- เรียกใช้จีไอเอสพีสำหรับการกำหนดความรับผิดชอบแก่เพียร์รับผิดชอบ และประกาศความรับผิดชอบไปยังเพียร์รับผิดชอบ
- บันทึกการส่งแฟ้มข้อมูลไปยังเพียร์รับผิดชอบ และลบแฟ้มข้อมูลที่ถูกส่งต่อไปยังเพียร์รับผิดชอบแล้ว
- เรียกคืนแฟ้มข้อมูลจากเพียร์รับผิดชอบที่คำนวณจากโพรโทคอลจีไอเอสพีผ่านบริการซีเอ็มเอสของจักซ์ตา

หน้าที่ที่สำคัญอีกประการของตัวจัดการการร่วมใช้คือการปฏิบัติงานตามคำสั่งที่ได้รับจากส่วนไคลเอนต์และให้ผลตอบกลับของการปฏิบัติงานแก่ส่วนรองรับการถ่ายโอนแฟ้มข้อมูลแบบเพียร์ทูเพียร์เพื่อส่งผลไปยังส่วนไคลเอนต์

การดูแลแฟ้มข้อมูลแบบโลคอลและคอนเทนต์แอดเวอทิสเมนต์ของแฟ้มข้อมูลแบบโลคอล รวมถึงโครงสร้างไดเรกทอรีและแฟ้มข้อมูลของเนื้อที่จัดเก็บแฟ้มข้อมูลเสมือนที่สร้างขึ้นจากคอนเทนต์แอดเวอทิสเมนต์ของเพียร์อื่นๆ ในระบบจะถูกแยกไปเป็นตัวจัดการแฟ้มข้อมูล (File Manager)



รูปที่ 3.8 ขั้นตอนการทำงานของตัวจัดการแฟ้มข้อมูลในตัวจัดการการร่วมใช้

ตัวจัดการแฟ้มข้อมูลทำงานแบบเธรด เพื่อทำการดูแลโครงสร้างไดเรกทอรีและแฟ้มข้อมูลภายในเนื้อที่จัดเก็บแฟ้มข้อมูลเสมือนเป็นระยะๆ โดยใช้ข้อมูลในคอนเทนต์แอดเวอทิสเมนต์ที่ได้จากการเรียกค้นคืนรายการที่พาธต่างๆ ผ่านโพรโทคอลจีไอเอสพี รวมถึงการดูแลแฟ้มข้อมูลแบบโลคอลที่อาจมีการสร้างแอดเวอทิสเมนต์ใหม่เพื่อให้แฟ้มข้อมูลคงอยู่ต่อเมื่อแอดเวอทิสเมนต์เดิมหมดอายุ หรือลบทิ้งเมื่อจัดเก็บลงเนื้อที่เก็บข้อมูลเสมือนแล้วเมื่อเป็นแฟ้มข้อมูลที่ต้องจัดเก็บ ขั้นตอนการทำงานอย่างคร่าวแสดงไว้ดังรูปที่ 3.8

3.3.2.3 รายละเอียดกลไกการเรียกคืนแฟ้มข้อมูลจากเนื้อที่จัดเก็บข้อมูลเสมือนของส่วนเชื่อมต่อ

การเรียกคืนแฟ้มข้อมูลจากเนื้อที่จัดเก็บข้อมูลเสมือนจะกระทำโดยตัวจัดการการร่วมใช้ เมื่อตัวจัดการการร่วมใช้ได้รับข้อมูลพาธและชื่อแฟ้มข้อมูลที่อ้างอิงบนเนื้อที่จัดเก็บข้อมูลเสมือนแล้ว ตัวจัดการการร่วมใช้จะเทียบข้อมูลพาธและชื่อแฟ้มข้อมูลจากแคชของตนที่ถูกอัพเดตสม่ำเสมอ ซึ่งจะได้รายการของแอดเวอทิสเมนต์ประเภทคอนเทนต์แอดเวอทิสเมนต์ของแฟ้มข้อมูลดังกล่าว แฟ้มข้อมูลหนึ่งๆ อาจมีคอนเทนต์แอดเวอร์ทิสเมนต์มากกว่าหนึ่งซึ่งได้รับจากหลายเพียร์รับผิดชอบแฟ้มข้อมูลดังกล่าวนั้น ตัวอย่างคอนเทนต์แอดเวอรทิสเมนต์ของแฟ้มข้อมูลคือ

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE jxta:ContentAdvertisement>
<jxta:ContentAdvertisement>
      <name> /peer2/data2.txt </name>
      <cid> md5:ff9cf2d690d888cb337f6bf4526b6130 </cid>
      <length> 5 </length>
      <metadata>
            <module> sharemanager </module>
            <expire> 20051228-195218 </expire>
            <rate> 1 </rate>
            <scheme>
                  <name> jnushare </name>
                  <content-type> text/litexml </content-type>
            </scheme>
      </metadata>
      <address> jxta://uuid-59616261646162614A78746150325033DE0C11F43
E7A4B0C88E28B7BAADD733A03/CMS/jxta-NetGroup
      </address>
</jxta:ContentAdvertisement>
```

สำหรับแต่และแฟ้มข้อมูลจะมีคอนเทนต์แอดเวอทิสเมนต์ที่ต่างกันที่เกิดจากเพียร์รับผิดชอบซึ่งเก็บแฟ้มข้อมูลนั้นไว้บนเนื้อทีจัดเก็บข้อมูลของตน แอดเวอทิสเมนต์ของแฟ้มข้อมูล

เดียวกันนี้จะต้องมีชื่อแฟ้มข้อมูลในอิลิเมนต์ name และ อิลิเมนต์ cid ที่เหมือนกัน ซึ่งเป็นอิลิเมนต์ที่บรรจุชื่อแฟ้มข้อมูลและคอนเทนต์ไอดีที่ได้จากค่าแฮชของเนื้อข้อมูลภายในแฟ้มข้อมูล

สำหรับแต่ละคอนเทนต์แอดเวอทิสเมนต์จะมีวันและเวลาของการหมดอายุของแอดเวอทิสเมนต์ ซึ่งการดึงแฟ้มข้อมูลจะต้องตรวจสอบวันหมดอายุของแอดเวอทิสเมนต์ก่อนเสมอแม้ตัวจัดการการร่วมใช้จะทำหน้าที่ตรวจสอบและลบแอดเวอทิสเมนต์อยู่เป็นระยะๆ อยู่แล้ว

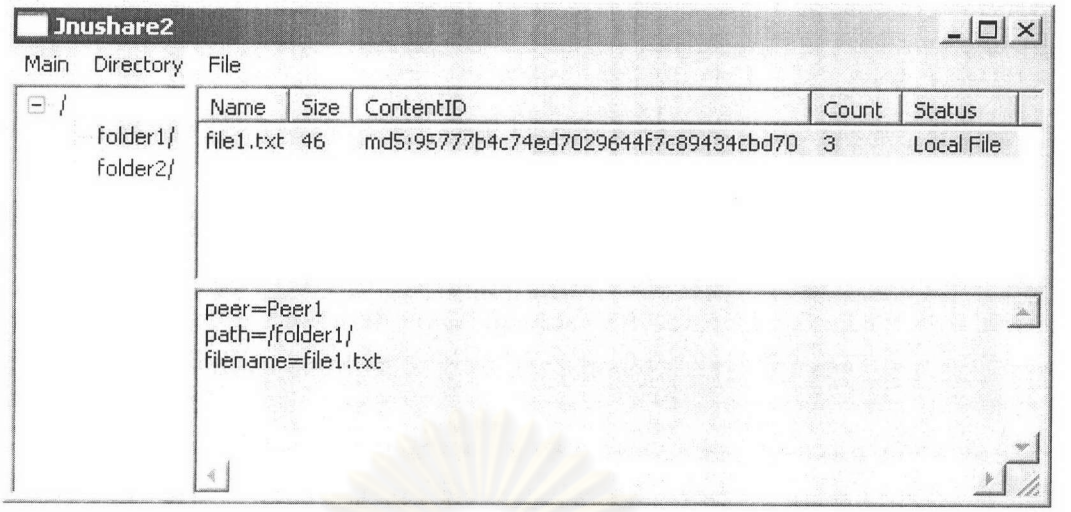
ตัวจัดการการร่วมใช้จะเรียกคืนแฟ้มข้อมูลด้วยบริการซีเอ็มเอสของจักซ์ตาด้วยตำแหน่งปลายทางที่กำหนดไว้ในอิลิเมนต์ address ที่อยู่ในคอนเทนต์แอดเวอทิสเมนต์

สังเกตว่า ในการอิมพลีเมนต์นี้มีอิลิเมนต์ rate ที่ตัวจัดการการร่วมใช้เป็นผู้กำหนดค่าให้แก่แอดเวอทิสเมนต์ที่ตนสร้างและประกาศให้แก่ผู้อื่น ปกติจะมีค่า 1 เมื่อแอดเวอทิสเมนต์ยังไม่หมดอายุ ตัวจัดการร่วมใช้สามารถกำหนดแอดเวอทิสเมนต์ที่ตนเก็บไว้ (อาจเป็นแอดเวอทิสเมนต์ที่ผู้อื่นสร้าง) ให้มีค่า -1 เมื่อแอดเวอทิสเมนต์นั้นหมดอายุ ผลรวมของค่าอิลิเมนต์ rate จากทุกแอดเวอทิสเมนต์โดยมีหนึ่งแอดเวอทิสเมนต์สำหรับหนึ่งแฟ้มข้อมูลบนแต่ละเพียร์รับผิดชอบเป็นตัวเลขบ่งชี้ความเข็มแข็งอย่างง่ายของแฟ้มข้อมูลนั้น

แฟ้มข้อมูลที่เรียกคืนจากเนื้อที่จัดเก็บข้อมูลเสมือนมาลงสู่ส่วนเชื่อมต่อ จะถูกเปลี่ยนสิทธิ์เข้าถึงแฟ้มข้อมูลเป็นสามารถเข้าถึงได้จากทุกผู้ใช้ที่อยู่ในกลุ่มผู้ใช้กริด โดยผู้บริหารระบบจะต้องกำหนดให้ผู้ใช้กริดเป็นสมาชิกของกลุ่มผู้ใช้กริดไว้ก่อนหน้าด้วย

3.3.3 ระบบร่วมใช้แฟ้มข้อมูลบนเครือข่ายเพียร์ทูเพียร์

เพียร์บนเครือข่ายเพียร์ทูเพียร์จะมีโปรแกรมประยุกต์เจนูแซร์ที่ถูกดัดแปลงโดยผู้วิจัย โปรแกรมประยุกต์เจนูแซร์คือโปรแกรมประยุกต์สำหรับการร่วมใช้แฟ้มข้อมูลบนเครือข่ายเพียร์ทูเพียร์ ระบบจัดเก็บแฟ้มข้อมูลแบบไดเรกทอรีที่ทุกเพียร์เห็นระบบแฟ้มข้อมูลเหมือนกัน เสมือนเป็นเนื้อจัดเก็บข้อมูลก้อนเดียวกันดังแสดงในรูปที่ 3.9 แฟ้มข้อมูลบนระบบแฟ้มข้อมูลจะถูกกระจายบนเครือข่ายเพียร์ทูเพียร์ตามตำแหน่งเพียร์ผู้เป็นเจ้าของแฟ้มข้อมูล แต่สามารถเห็นโครงสร้างแฟ้มข้อมูลแบบไดเรกทอรีได้เหมือนกัน



รูปที่ 3.9 โปรแกรมประยุกต์เจนูแชร์รุ่น 2

การกำหนดตำแหน่งจัดเก็บและค้นคืนแฟ้มข้อมูลบนระบบแบ่งร่วมแฟ้มข้อมูลจะใช้กลไกจีไอเอสพี เจนูแชร์ใช้จีไอเอสพีสร้างดัชนีคำหลักคู่กับข้อมูลแฟ้มข้อมูล ข้อมูลแฟ้มข้อมูลนี้คือคอนเทนต์แอดเวอทิสเมนต์ในรูปแบบจักซ์ตา ด้วยวิธีนี้ จีไอเอสพีช่วยให้ทุกเพียร์บนระบบรู้ถึงดัชนีแฟ้มข้อมูลที่ใช้ร่วมกัน โดยดัชนีแฟ้มข้อมูลนี้เองก็ถูกแบ่งออกเป็นความรับผิดชอบของแต่ละเพียร์ กล่าวคือเป็นดัชนีแบบกระจาย เพียร์ที่มีค่าแฮชของเพียร์ไอดีใกล้เคียงกับค่าแฮชของคำหลัก ก็จะทำหน้าที่รับผิดชอบดัชนีคำหลักนั้นๆ ซึ่งจำนวนเพียร์ที่รับผิดชอบต่อดัชนีคำหลักหนึ่งๆ มีค่าปริยายเท่ากับ 5 สำหรับการทดลองในงานวิจัยนี้ การทำซ้ำความรับผิดชอบดัชนีเพื่อให้ระบบมีความน่าเชื่อถือ สังเกตว่าทุกเพียร์บนระบบต้องใช้ฟังก์ชันแฮชเดียวกัน ในงานวิจัยนี้เลือกใช้ SHA-1 ที่ให้ค่าแฮชขนาด 160 บิต

ตัวอย่างคำหลักคือ /peer2/data2.txt และ ตัวอย่างข้อมูลที่จัดเก็บคู่กับค่าแฮชของคำหลักคือ

```
<?xml version="1.0" encoding="UTF-8"?> <!DOCTYPE
jxta:ContentAdvertisement> <jxta:ContentAdvertisement> <name>
/peer2/data2.txt </name> <cid> md5:ff9cf2d690d888cb337f6bf4526b6130
</cid> <length> 5 </length> <metadata> <module> sharemanager
</module> <expire> 20051228-195218 </expire> <rate> 1 </rate>
<scheme> <name> jnushare </name> <content-type> text/litexml
</content-type> </scheme> </metadata> <address> jxta://uuid-
59616261646162614A78746150325033DE0C11F43E7A4B0C88E28B7BAADD733A03/CM
S/jxta-NetGroup </address> </jxta:ContentAdvertisement>
```

ผู้วิจัยดัดแปลงโปรแกรมประยุกต์ให้ทำหน้าที่รับผิดชอบต่อข้อมูลที่ตนรับผิดชอบดัชนีของข้อมูลนั้นด้วย โดยการดึงข้อมูลจากส่วนเชื่อมต่อมาจัดเก็บไว้ที่เนื้อที่จัดเก็บข้อมูลของตน

3.3.3.1 การดึงแฟ้มข้อมูลที่ต้องรับผิดชอบที่อยู่บนส่วนเชื่อมต่อ

เมื่อผู้ใช้กริดหรือโปรแกรมประยุกต์ต้องการจัดเก็บแฟ้มข้อมูลลงเนื้อที่เก็บข้อมูลเสมือนที่มีการจัดเก็บลงเนื้อที่ดิสก์ว่างของเพียร์ต่างๆ แฟ้มข้อมูลดังกล่าวจะถูกถ่ายโอนโดยส่วนไคลเอนต์มายังกริดโฮสต์ที่มีส่วนเชื่อมต่อ จากนั้นส่วนเชื่อมต่อจะเข้าสู่กระบวนการกระจายความรับผิดชอบของแฟ้มข้อมูลดังกล่าวไปยังเพียร์รับผิดชอบที่ใกล้เคียงตามโพรโทคอลจีไอเอสพี ให้สังเกตว่า ส่วนเชื่อมต่อและเพียร์ต่างๆ บนเครือข่ายเพียร์ทูเพียร์ไม่อาจรู้จักเพียร์ทุกเพียร์บนเครือข่ายเพียร์ทูเพียร์ได้ การเลือกเพียร์รับผิดชอบของส่วนเชื่อมต่อจึงเป็นการเลือกจากกลุ่มเพียร์ที่ส่วนเชื่อมต่อรู้จักในขณะนั้น

กระบวนการจัดเส้นทาง (Routing) ความรับผิดชอบของเพียร์ที่รับความรับผิดชอบมาก่อนไปยังเพียร์ที่เหมาะสมกว่าจึงเป็นสิ่งจำเป็น โพรโทคอลจีไอเอสพีที่ทุกเพียร์บนระบบร่วมใช้แฟ้มข้อมูลรองรับกระบวนการจัดเส้นทาง

เพื่อให้ความรับผิดชอบต่อแฟ้มข้อมูลถูกกระจายไปยังเพียร์ที่เหมาะสมและกระจายทั่วถึงบนระบบร่วมใช้แฟ้มข้อมูล ส่วนเชื่อมต่อจะทำหน้าที่กระจายความรับผิดชอบไปยังระบบร่วมใช้แฟ้มข้อมูลก่อน และใช้จำนวนหนึ่งสำหรับกระบวนการจัดเส้นทางความรับผิดชอบ ก่อนที่เพียร์ต่างๆ ที่ได้รับความรับผิดชอบไปแล้วจะเป็นผู้มาดึงแฟ้มข้อมูลที่ตนรับผิดชอบไปเอง ด้วยวิธีการเช่นเดียวกับที่ส่วนเชื่อมต่อดึงแฟ้มข้อมูลเมื่อมีการเรียกคืนแฟ้มข้อมูล

กระบวนการจัดเส้นทางนี้จะเกิดขึ้นเมื่อมีการค้นคืนตำแหน่งเพียร์รับผิดชอบแฟ้มข้อมูลใดๆ ด้วย หากการแสดงความรับผิดชอบของเพียร์ที่รับผิดชอบแฟ้มข้อมูลไม่ถูกส่งไปยังเพียร์ที่ทำการค้นคืน เช่นกรณีที่เพียร์ค้นคืนเป็นเพียร์ที่เพิ่งเข้าร่วมระบบ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย