

# บทที่ 1 บทนำ



## 1.1 ความเป็นมาและความสำคัญของปัญหา

ในระบบคอมพิวเตอร์แบบกระจาย (Distributed System) นั้นจะมีเครื่องคอมพิวเตอร์หลายๆเครื่อง เชื่อมต่อเข้าหากันผ่านทางเครือข่ายโดยเครื่องคอมพิวเตอร์ทุกเครื่องจะสามารถทำหน้าที่เป็นได้ทั้งผู้รับบริการ (Client) และผู้ให้บริการ (Server) ผู้รับบริการจะสามารถเรียกใช้งานบริการใดๆของผู้ให้บริการได้ โดยกระทำผ่านทางส่วนต่อประสาน (Interface) ของบริการนั้น ซึ่งจะถูกใช้เป็นเสมือนข้อตกลงของวิธีการ เรียกใช้งานบริการนั่นเอง

ปัจจุบันนี้สถาปัตยกรรมของระบบกระจายเชิงวัตถุได้ถูกพัฒนาขึ้นมาในหลายๆรูปแบบ งานวิจัยนี้ได้มุ่งความสนใจไปที่สถาปัตยกรรมระบบกระจายเชิงวัตถุแบบคอรีบา (CORBA)<sup>1</sup> [1] ซึ่งเป็นข้อกำหนดที่ใช้คลังจัดเก็บส่วนต่อประสาน (Interface Repository) ในการทำหน้าที่จัดเก็บและดูแลการใช้งานข้อมูล นิยามของส่วนต่อประสาน (Interface Definition) และข้อมูลความสัมพันธ์แบบซับไทป์ปิ้ง (Subtyping Relationship) ของส่วนต่อประสาน บริการหนึ่งในคอรีบาจะถูกใช้งานแทนที่อีกบริการหนึ่งได้ก็ต่อเมื่อบริการนั้นจะต้องมีส่วนต่อประสานที่เป็นซับไทป์ (Subtype) ของบริการเดิมเท่านั้น [1,2] โดยที่ความสัมพันธ์ของบริการแบบซับไทป์ปิ้งหมายถึงการที่บริการ 2 บริการใด ๆ มีความสัมพันธ์ในลักษณะที่เป็นแม่ลูก (Parent/Child Relationship) โดยบริการลูกจะทำการสืบทอด (Inheritance) คุณลักษณะของ ส่วนต่อประสานมาจากบริการแม่นั่นเอง การเปลี่ยนแปลงใดๆที่เกิดขึ้นกับส่วนต่อประสานของบริการลูก จะเป็นแบบเพิ่มขึ้น (Incremental Change) เท่านั้น กล่าวคือบริการลูกจะสามารถเพิ่มเติมคุณสมบัติใหม่ๆเข้าไปในส่วนต่อประสานโดยไม่ทำการลบหรือเปลี่ยนแปลงส่วนที่ได้รับสืบทอดมาจากส่วนต่อ ประสานของบริการแม่ ซึ่งวิธีนี้จะทำให้บริการลูกสามารถถูกใช้งานแทนที่บริการแม่ได้ แต่ถ้าหากการ เปลี่ยนแปลงของบริการที่เกิดขึ้นเป็นไปในลักษณะที่เป็นการเปลี่ยนแปลงรูปลักษณะของส่วนต่อประสาน (Interface Signature) ที่ได้รับสืบทอดมาแล้ว บริการใหม่นั้นจะไม่สามารถถูกใช้งานแทนที่บริการเดิมได้

ความสัมพันธ์แบบเท่าเทียมกัน (Equivalence Relationship) [3] จัดเป็นความสัมพันธ์ในอีกรูป แบบหนึ่งที่จะเกิดขึ้นระหว่างบริการใดๆในระบบคอมพิวเตอร์แบบกระจายเชิงวัตถุ ความสัมพันธ์ใน ลักษณะนี้จะกำหนดความสามารถที่เท่าเทียมกันของบริการ 2 บริการใดๆเพื่อระบุว่าบริการหนึ่งสามารถที่ จะทำงานแทนที่อีกบริการหนึ่งได้โดยพิจารณาจากความสามารถในการทำงาน (Functionality) ของ บริการเป็นหลักแทนที่จะพิจารณาจากรูปลักษณะของส่วนต่อประสาน ตัวอย่างเช่นบริการ A จะถือว่ามี

---

<sup>1</sup> CORBA ย่อมาจาก "Common Object Request Broker Architecture" เป็นข้อกำหนด (Specification) เกี่ยวกับ สถาปัตยกรรมระบบกระจายเชิงวัตถุ (Distributed Object Architecture) ของโอเอ็มจี (OMG)

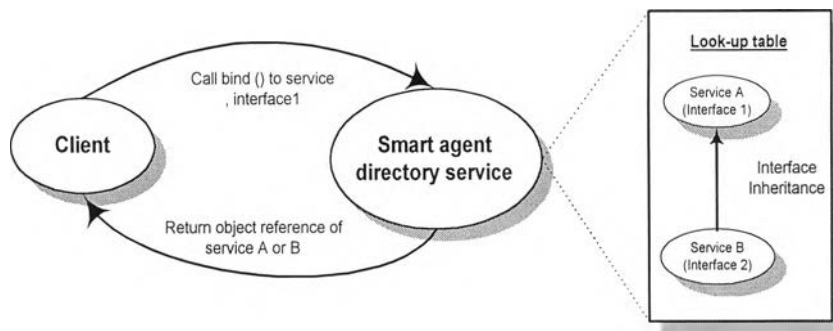
ความสัมพันธ์แบบเท่าเทียมกับบริการ B ได้ก็ต่อเมื่อการเรียกใช้งานบริการ A จะก่อให้เกิดผลลัพธ์เช่นเดียวกับการเรียกใช้งานบริการ B ทุกประการ ถึงแม้ว่าบริการ A จะมีส่วนต่อประสานที่แตกต่างกับส่วนต่อประสานของบริการ B ก็ตาม ความสัมพันธ์ในลักษณะนี้จะช่วยเพิ่มความยืดหยุ่น (Flexibility) ในการใช้งานโปรแกรมประยุกต์อีกด้วยทั้งนี้เนื่องจากผู้รับบริการจะได้รับบริการอื่นที่เท่าเทียมกัน ทดแทนเมื่อไม่มีบริการที่ต้องการหรือบริการที่ต้องการไม่พร้อมที่จะให้บริการในช่วงเวลาที่ขอใช้บริการ อย่างไรก็ตาม ข้อกำหนดของคอร์บาในปัจจุบันนี้ยังไม่สนับสนุนการทำงานแทนที่กันของบริการที่มีลักษณะความสัมพันธ์แบบเท่าเทียมกัน [1,4] ตลอดจนข้อมูลที่ถูกจัดเก็บอยู่ในคลังจัดเก็บส่วนต่อประสานของคอร์บาก็ยังไม่เพียงพอที่จะสนับสนุนการทำงานในลักษณะดังกล่าว

จากการศึกษากลไกการทำงานภายในของคอร์บาพบว่า ก่อนที่ผู้รับบริการจะสามารถเรียกใช้งานบริการใดๆได้ โปรแกรมของผู้รับบริการจะต้องทำการค้นหาตำแหน่งและสร้างการเชื่อมต่อ (Binding) ไปยังอินสแตนซ์ (Instance) ของบริการนั้นๆ ที่ทำงานอยู่ในโปรแกรมของผู้ให้บริการเสียก่อน ซึ่งถ้าออร์บ (ORB<sup>2</sup>) ในคอร์บาค้นหาและพบอินสแตนซ์ของบริการที่ต้องการแล้ว ออร์บจะทำการจัดส่งข้อมูลอ้างอิงของบริการ (Object Reference) ที่พบบackไปยังโปรแกรมผู้รับบริการเพื่อใช้งานต่อไป [5,6] โปรแกรมผู้รับบริการในคอร์บาสสามารถทำการค้นหาข้อมูลอ้างอิงของบริการได้หลายวิธี อาทิเช่นการเรียกใช้คำสั่ง bind เพื่อทำการค้นหาด้วยออร์บ [5,7] หรือเรียกใช้บริการค้นหาจากเทรดเดอร์ (Trader) [6,8] เป็นต้น

ในขั้นตอนของการค้นหาข้อมูลอ้างอิงของบริการ ถ้าโปรแกรมผู้รับบริการระบุเฉพาะชื่อส่วนต่อประสานของบริการ (Interface) ที่ต้องการแล้ว คอร์บาจะทำการค้นหาอินสแตนซ์ใดๆในระบบที่ถูกพัฒนาขึ้นตามส่วนต่อประสานนั้น ซึ่งถ้าหากไม่พบอินสแตนซ์ใดๆเลย คอร์บาจะทำการค้นหาและส่งกลับข้อมูลอ้างอิงของอินสแตนซ์ที่ถูกพัฒนาขึ้นตามส่วนต่อประสานที่มีความสัมพันธ์แบบซัพไทป์กับส่วนต่อประสานที่ระบุแทน (รูปที่ 1.1) ด้วยการทำงานดังกล่าวนี้ จึงอาจกล่าวได้ว่าคอร์บาสสามารถรองรับการแทนที่กันของบริการในระดับชนิดของบริการ (Type Level Substitution) ได้ [1,5]

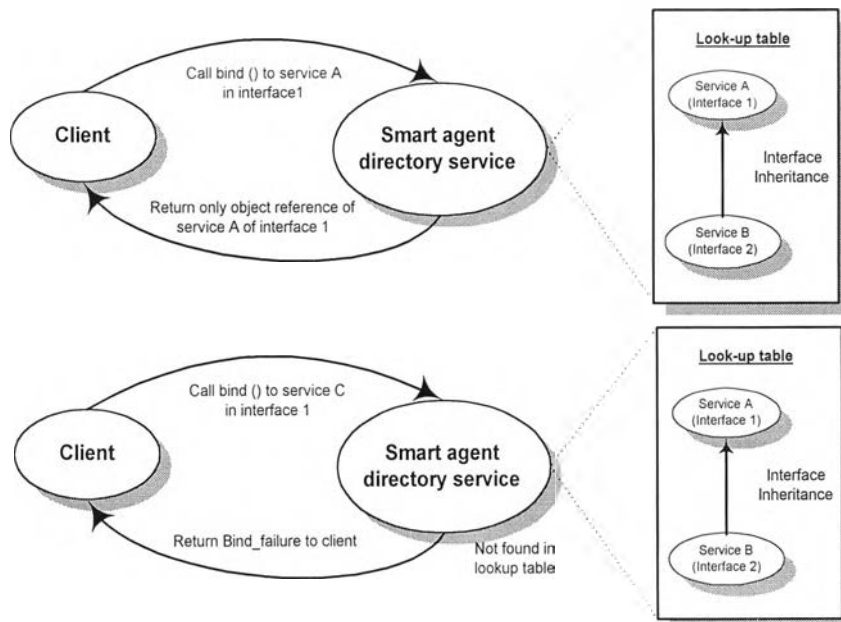
---

<sup>2</sup> ออร์บ (ORB) ย่อมาจาก "Object Request Broker" เป็นส่วนประกอบสำคัญที่ทำหน้าที่เป็นเสมือนเส้นทางรับส่งข้อมูล (Object Bus) ระหว่างวัตถุ (Object) ตามข้อกำหนดของคอร์บา



รูปที่ 1.1 การแทนที่กันของบริการในระดับชนิดของบริการ

อย่างไรก็ตาม ถ้าหากว่าโปรแกรมผู้รับบริการทำการระบุทั้งชื่อของส่วนต่อประสานและชื่อของอินสแตนซ์ที่ต้องการใช้งานในขั้นตอนของการค้นหาข้อมูลอ้างอิงของบริการแล้ว ในกรณีนี้คอร์บาก็จะทำการค้นหาเฉพาะอินสแตนซ์ที่มีชื่อและส่วนต่อประสานตามที่ระบุเท่านั้น ซึ่งถ้าหาไม่พบคอร์บาก็จะไม่ดำเนินการค้นหาอินสแตนซ์อื่นเพื่อใช้แทนที่อินสแตนซ์ที่ต้องการ (รูปที่ 1.2) ทั้งนี้เนื่องจากกลไกการทำงานของคอร์บายังไม่รองรับการทำงานแทนที่กันของบริการในระดับอินสแตนซ์ (Instance Level Substitution) [5,7]



รูปที่ 1.2 การไม่รองรับการแทนที่กันของบริการในระดับอินสแตนซ์

งานวิจัย [9] ได้พยายามแก้ไขปัญหาลักษณะที่ทำงานแทนที่กันในระดับอินสแตนซ์ของบริการที่มีรุ่นต่างกันโดยได้ออกแบบแบบจำลองที่สนับสนุนความโปร่งใสต่อการเปลี่ยนรุ่นของบริการ (Type Evolution Transparency) ซึ่งทำให้บริการต่างรุ่นกันสามารถใช้งานแทนที่กันได้โดยบริการรุ่นใหม่ไม่ถูกจำกัดให้อยู่ในรูปแบบที่เป็นซับไทป์ของบริการรุ่นเก่าเท่านั้น ทั้งนี้งานวิจัย [9] จะอาศัยกลไกการแปลงคำร้องขอใช้บริการ (Service Request) ของบริการรุ่นเก่าเพื่อส่งต่อไปยังบริการรุ่นใหม่โดยใช้ตัวดำเนินการแปลง (Mapping operator) การทำงานแทนที่กันของบริการในงานวิจัยนี้เป็นการแทนที่กันในระดับอินสแตนซ์โดยที่อินสแตนซ์หนึ่งของบริการรุ่นใหม่สามารถทำงานแทนที่อินสแตนซ์หนึ่งของบริการรุ่นเดิมได้เท่านั้น ทำให้ผู้ทำการเปลี่ยนรุ่นของบริการ (Evolver) จะต้องกำหนดคู่ของอินสแตนซ์เก่าและใหม่ที่จะทำงานแทนที่กันเสียก่อน ทั้งนี้เนื่องจากสถานะ (State) ข้อมูล (Data) ตลอดจนพฤติกรรม (Behaviour) ของสองอินสแตนซ์ที่ต่างกันของบริการรุ่นเก่าที่ถึงแม้จะมีส่วนต่อประสานเดียวกันแต่มีการพัฒนา (Implementation) ที่แตกต่างกันจะส่งผลให้คำร้องขอใช้บริการที่ส่งมายังสองอินสแตนซ์นี้อาจต้องการการแปลงและการส่งต่อไปยังอินสแตนซ์ของบริการรุ่นใหม่ที่แตกต่างกันด้วย งานวิจัย [9] ได้จัดสร้างตัวดำเนินการแปลงให้มีชื่อของส่วนต่อประสานและชื่อของอินสแตนซ์ที่เหมือนกับอินสแตนซ์ของบริการรุ่นเก่าที่ถูกยกเลิกไปทุกประการ ทั้งนี้เพื่อใช้ดัก (Intercept) คำร้องขอใช้บริการจากโปรแกรมผู้รับบริการเดิมอย่างโปร่งใสก่อนที่จะเรียกใช้งานฟังก์ชันการแปลง (Mapping Function) เพื่อทำการแปลงคำร้องขอใช้บริการนั้นให้สอดคล้องกับส่วนต่อประสานของบริการรุ่นใหม่และจัดส่งคำร้องนั้นไปทำงานยังอินสแตนซ์ของบริการรุ่นใหม่ต่อไป

การที่ตัวดำเนินการแปลงถูกพัฒนาขึ้นโดยมีชื่อส่วนต่อประสานและชื่ออินสแตนซ์ที่เหมือนกับอินสแตนซ์ของบริการรุ่นเดิมทุกประการนั้นสืบเนื่องมาจากข้อสมมติฐานที่ว่าอินสแตนซ์ของบริการรุ่นเก่าจะถูกยกเลิกไปเมื่อผู้ให้บริการนำบริการรุ่นใหม่มาใช้แทนที่ การกระทำนี้จะส่งผลให้อินสแตนซ์ของตัวดำเนินการแปลงถูกเรียกใช้งานแทนอินสแตนซ์ของบริการรุ่นเดิมโดยอัตโนมัติไม่ว่าโปรแกรมผู้รับบริการจะค้นหาข้อมูลอ้างอิงของบริการด้วยวิธีใดก็ตาม ตัวอย่างเช่นถ้าต้องการแทนที่บริการ `bank_obj1` ของส่วนต่อประสาน `Bank` รุ่น 1.0 ด้วยบริการ `bank_obj1` ของส่วนต่อประสาน `Bank` รุ่น 2.0 แล้ว ตัวดำเนินการแปลงจะถูกสร้างขึ้นให้เป็นอินสแตนซ์ชื่อ `bank_obj1` ของส่วนต่อประสาน `Bank` รุ่น 1.0 เพื่อทำหน้าที่ดักคำร้องขอรุ่นเก่าแล้วจึงแปลงและส่งต่อคำร้องขอใช้บริการไปยังอินสแตนซ์ `bank_obj1` ของส่วนต่อประสาน `Bank` รุ่น 2.0 และเมื่อโปรแกรมผู้รับบริการทำการ `bind` ไปยังบริการ `bank_obj1` ของส่วนต่อประสาน `Bank` รุ่น 1.0 ด้วยรูปแบบใดก็ตาม ก็จะได้รับข้อมูลอ้างอิงของตัวดำเนินการแปลงนี้ไปใช้งานแทนโดยอัตโนมัติ การใช้งานตัวดำเนินการแปลงร่วมกับฟังก์ชันการแปลงนี้เป็นการสนับสนุนการแทนที่ของบริการในระดับอินสแตนซ์โดยอาศัยการปลอมตัวของตัวดำเนินการแปลงให้ทำงานเป็นบริการรุ่นเก่า ทำให้การเรียกใช้บริการของผู้รับบริการยังคงดำเนินตามแบบเดิมได้โดยไม่ต้องมีการเปลี่ยนแปลงใดๆ

งานวิจัย [4] ได้ทำการพัฒนาต่อจากงานวิจัย [9] โดยมุ่งประเด็นไปที่การหาแนวทางในการจัดการข้อมูลฟังก์ชันการแปลงใหม่โดยใช้การจัดเก็บลงในคลังจัดเก็บส่วนต่อประสานของคอร์บาเพื่อขยายขอบเขตของการใช้งานข้อมูลนี้ให้ไม่เพียงแต่ครอบคลุมการแปลงระหว่างบริการที่มีรุ่นต่างกันเท่านั้นแต่ยังรวมไปถึงการแปลงระหว่างบริการใดๆที่มีความสัมพันธ์แบบเท่าเทียมกันได้อีกด้วย อนึ่งการจัดเก็บข้อมูลเหล่านี้ลงในคลังจัดเก็บส่วนต่อประสานจะทำให้ข้อมูลถูกรวบรวมอย่างเป็นระเบียบและง่ายต่อการจัดการมากขึ้น

การแทนที่กันของบริการในระดับอินสแตนซ์ตามงานวิจัยที่ [4] และ [9] จะมีรูปแบบการแทนที่กันอยู่ 2 ลักษณะคือการแทนที่อินสแตนซ์ของบริการรุ่นเก่าที่ต้องการด้วยอินสแตนซ์ของบริการรุ่นใหม่ ซึ่งอินสแตนซ์ของบริการรุ่นใหม่อาจจะมีส่วนต่อประสานที่เหมือนหรือแตกต่างกับส่วนต่อประสานของบริการรุ่นเก่าที่ต้องการก็ได้ ทั้งนี้ผู้ทำการเปลี่ยนรุ่นของบริการจะเป็นผู้ทำหน้าที่กำหนดกฎการแปลงที่จะเกิดขึ้นภายในตัวดำเนินการแปลงและจัดเก็บข้อมูลนี้ลงในคลังจัดเก็บส่วนต่อประสานต่อไป อนึ่งถ้าหากผู้ให้บริการทำการขยายการแทนที่กันของบริการให้สามารถนำไปใช้ได้กับการแทนที่กันของบริการคู่ใดๆที่มีความสัมพันธ์แบบเท่าเทียมกัน โดยที่บริการทั้ง 2 นี้ไม่จำเป็นต้องเป็นรุ่นของกันและกันแล้ว ผู้ให้บริการจะต้องมั่นใจว่าบริการที่ทำงานแทนที่จะให้ผลการทำงานตลอดจนสถานะภายหลังการทำงานที่เหมือนกับบริการที่ผู้รับบริการต้องการอย่างแท้จริง ทั้งนี้บริการที่จะถูกแทนที่ได้ อาจเป็นบริการแบบไม่มีสถานะ (Stateless) หรืออาจเป็นบริการแบบมีสถานะ (Stateful) แต่สถานะต่างๆจะสามารถถูกจัดเก็บหรือเปลี่ยนแปลงให้สอดคล้องกับสถานะที่จะเกิดขึ้นภายหลังการทำงานของบริการที่ทำงานแทนที่ได้ โดยในที่นี้บริการแบบไม่มีสถานะจะหมายถึงบริการใดๆที่มีการรับข้อมูลเข้ามาเพื่อใช้ในการประมวลผลและจะส่งผลลัพธ์ที่ได้กลับไปยังผู้เรียกใช้บริการโดยไม่มีการจัดเก็บหรือเปลี่ยนแปลงสถานะและข้อมูลต่างๆภายหลังการทำงาน ตัวอย่างเช่นบริการคำนวณหาค่าอินทิเกรต (Integration) ของฟังก์ชันทางคณิตศาสตร์หนึ่งๆ ซึ่งบริการนี้จะรับค่าเริ่มต้นและค่าสิ้นสุดของตัวแปรที่ใช้ในการคำนวณและส่งผลการคำนวณนั้นกลับไปยังผู้เรียกใช้บริการ ส่วนบริการแบบมีสถานะจะหมายถึงบริการใดๆก็ตามที่มีการจัดเก็บหรือเปลี่ยนแปลงสถานะหรือข้อมูลต่างๆภายหลังการทำงานด้วย ตัวอย่างเช่นบริการการปรับปรุงยอดเงินของลูกค้าภายในฐานข้อมูลของระบบธนาคาร เป็นต้น

อย่างไรก็ดี จากการศึกษาแบบจำลองตามงานวิจัย [4] และ [9] จะพบข้อจำกัดขึ้นเมื่อต้องการขยายการแทนที่กันของบริการไปยังบริการที่มีความสัมพันธ์แบบเท่าเทียมกันโดยไม่เป็นรุ่นของกันและกัน โดยข้อจำกัดนี้จะเกิดขึ้นเนื่องจากผู้ให้บริการจะไม่สามารถใช้การปลอมตัวของตัวดำเนินการแปลงโดยสร้างตัวดำเนินการแปลงให้มีชื่อของส่วนต่อประสานและชื่อของอินสแตนซ์เหมือนกับอินสแตนซ์ของบริการที่จะถูกแทนที่เพื่อสร้างกลไกการดักจับคำร้องขอโดยอัตโนมัติตามรูปแบบที่ใช้ในงานวิจัยที่ [9] ได้ เพราะว่าอินสแตนซ์ของบริการที่จะถูกแทนที่อาจยังคงถูกใช้งานอยู่ในระบบโดยไม่ได้ถูกยกเลิกไป การตั้งชื่อใหม่ให้กับตัวดำเนินการแปลงจะไม่สามารถทำให้เกิดการแทนที่กันของบริการในระดับ

อินสแตนซ์ได้เสมอไป ทั้งนี้เพราะเมื่อใดก็ตามที่โปรแกรมผู้รับบริการทำการเรียกใช้คำสั่ง bind โดยเจาะจงชื่อของอินสแตนซ์แล้ว อินสแตนซ์ของตัวดำเนินการแปลงก็จะไม่ถูกเรียกใช้งาน

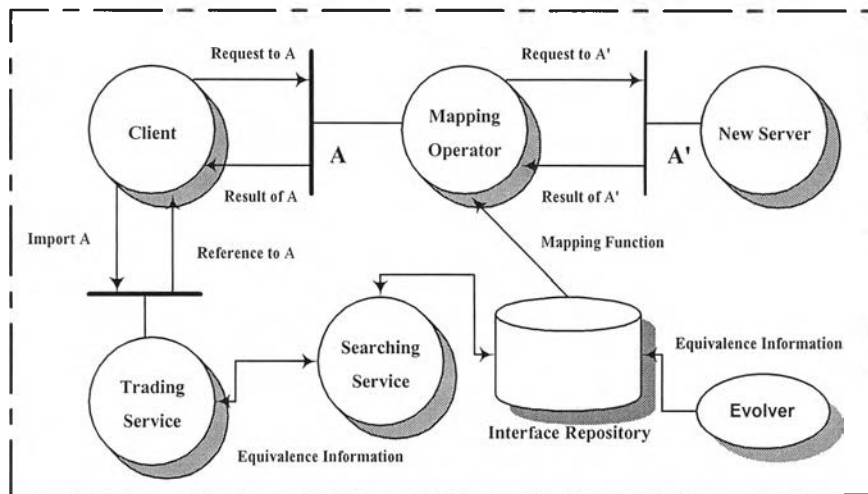
ด้วยปัญหาดังที่กล่าวถึงข้างต้น งานวิจัยนี้จึงมีเป้าหมายในการดำเนินงานเพื่อมุ่งพัฒนากลไกเพื่อสนับสนุนการแทนที่กันของบริการในระดับอินสแตนซ์ของคอร์บาให้สมบูรณ์ขึ้น โดยกลไกที่พัฒนาขึ้นจะแก้ไขข้อจำกัดดังกล่าวและจะส่งผลให้อินสแตนซ์ของตัวดำเนินการแปลงสามารถถูกเรียกใช้งานเพื่อทำการแปลงและส่งต่อคำร้องขอใช้บริการไปยังอินสแตนซ์ของบริการที่เท่าเทียมกันซึ่งอาจเป็นรุ่นหรือไม่เป็นรุ่นของบริการที่ต้องการและอาจมีส่วนต่อประสานที่แตกต่างออกไปได้โดยอัตโนมัติ ทั้งนี้ไม่ว่าผู้รับบริการจะค้นหาอินสแตนซ์ที่ต้องการผ่านทางออร์บของคอร์บาหรือบริการเทรเดเดอร์

## 1.2 แนวคิดในการทำวิจัย

สืบเนื่องจากการที่กลไกตามแบบจำลองในงานวิจัย [4] และ [9] ยังมีข้อจำกัดอยู่เมื่อต้องการนำมาประยุกต์ใช้กับการแทนที่กันของบริการที่มีความสัมพันธ์แบบเท่าเทียมกัน ดังที่ได้กล่าวไปแล้วในหัวข้อที่ 1.1 งานวิจัยนี้จึงมีแนวความคิดที่จะเพิ่มเติมกลไกส่วนการค้นหาบริการแทนที่และส่วนการเรียกใช้ตัวดำเนินการแปลงเพื่อรองรับการแทนที่กันในระดับอินสแตนซ์ของบริการที่มีความสัมพันธ์แบบเท่าเทียมกัน ทั้งนี้ไม่ว่าผู้รับบริการจะค้นหาอินสแตนซ์ผ่านทางออร์บหรือเทรเดเดอร์ก็ตาม กลไกที่พัฒนาขึ้นจะใช้ข้อมูลที่ถูกจัดเก็บอยู่ในคลังจัดเก็บส่วนต่อประสานของงานวิจัย [4] เพื่อสร้างการทำงานดังรูปที่ 1.3

หากผู้รับบริการค้นหาอินสแตนซ์ผ่านทางออร์บ กลไกที่พัฒนาขึ้นในงานวิจัยนี้จะอยู่ในรูปของการใช้โปรแกรมพรีโพรเซสเซอร์ (Preprocessor) เพื่อทำการวิเคราะห์และแทรกคำสั่งการทำงานที่เหมาะสมลงในโปรแกรมของผู้รับบริการ (Client Program) ตามรูปที่ 1.4 โปรแกรมพรีโพรเซสเซอร์ที่พัฒนาขึ้นจะทำการแทรกคำสั่งการทำงานที่ใช้เป็นตัวสร้างกลไกการค้นหาอินสแตนซ์ของตัวดำเนินการแปลงจากข้อมูลที่ถูกจัดเก็บไว้ในคลังจัดเก็บส่วนต่อประสานของงานวิจัย [4] (รูปที่ 1.5) โดยเริ่มจากการตรวจสอบว่าโปรแกรมของผู้รับบริการเกิดความผิดพลาด (Bind Failure) ขึ้นจากการค้นหาอินสแตนซ์ของบริการใดๆในขณะรันไทม์ (Runtime) หรือไม่ การตรวจสอบความผิดพลาดนี้จะกระทำในรูปของการดักจับเอ็กซ์เซพชันของระบบ (System Exception) ซึ่งจะเกิดขึ้นเมื่อโปรแกรมของผู้รับบริการเรียกใช้คำสั่ง bind แล้วออร์บไม่สามารถค้นหาและส่งกลับข้อมูลอ้างอิงของบริการกลับไปยังโปรแกรมผู้รับบริการได้

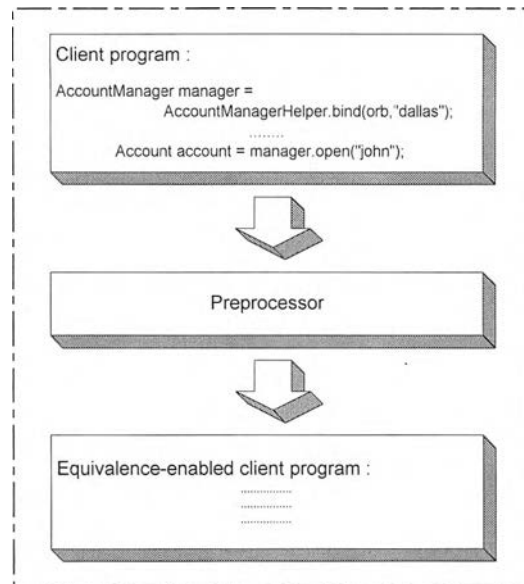
เมื่อเกิดเอ็กซ์เซพชันขึ้นจากการเรียกใช้คำสั่ง bind กลไกการทำงานของคำสั่งที่ถูกแทรกเข้าไปในโปรแกรมของผู้รับบริการจะทำการค้นหาบริการอื่นที่สามารถใช้งานแทนที่บริการที่ผู้รับบริการต้องการใช้งานโดยใช้การค้นหาข้อมูลจากคลังจัดเก็บส่วนต่อประสานและเมื่อพบบริการอื่นที่สามารถใช้งานแทนที่ได้ คำสั่งในโปรแกรมของผู้รับบริการจะพยายามสร้างการเชื่อมต่อไปยังตัวดำเนินการแปลงซึ่งทำงานร่วมกับบริการที่จะมาแทนที่นั้นผ่านทางออร์บของคอร์บาและนำข้อมูลอ้างอิงของตัวดำเนินการแปลงนั้นส่งกลับมายังโปรแกรมของผู้รับบริการเพื่อให้เรียกใช้งานในขั้นตอนอื่นๆต่อไป



รูปที่ 1.3 กลไกการทำงานของตัวดำเนินการแปลงและบริการเทรดเดอริในงานวิจัยนี้

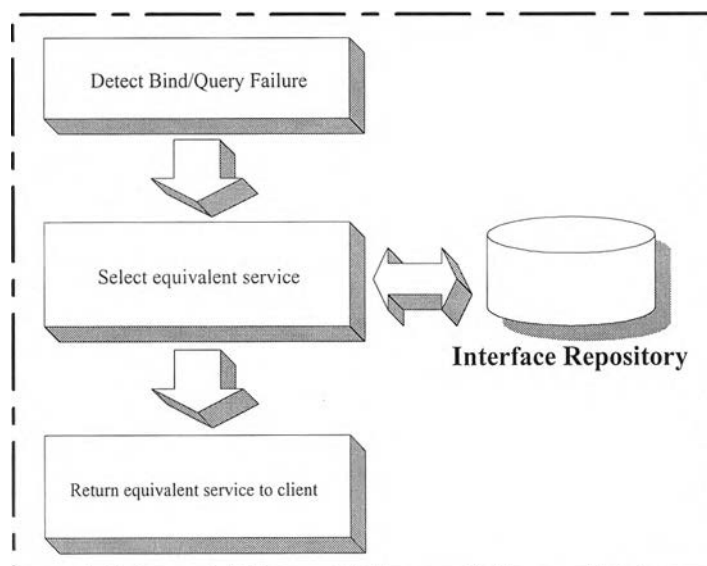
การแทรกคำสั่งลงในโปรแกรมของผู้รับบริการโดยใช้โปรแกรมพีโรเซสเซอร์จะสามารถแก้ไขปัญหาการแทนที่ของบริการที่มีความสัมพันธ์แบบเท่าเทียมกันได้ ในกรณีที่โปรแกรมของผู้รับบริการทำการค้นหาข้อมูลอ้างอิงของบริการใดๆผ่านทางออร์บด้วยการเรียกใช้คำสั่ง bind เท่านั้น วิธีการนี้จะไม่สามารถใช้ได้เมื่อโปรแกรมของผู้รับบริการไม่ได้เรียกใช้คำสั่ง bind แต่ทำการค้นหาอินสแตนซ์ของบริการโดยเรียกใช้บริการจากเทรดเดอริ ดังนั้นในงานวิจัยนี้จึงมีความจำเป็นที่จะต้องทำการเปลี่ยนแปลงกลไกการทำงานภายในของบริการเทรดเดอริให้รองรับการค้นหาข้อมูลความสัมพันธ์แบบเท่าเทียมกันจากคลังจัดเก็บส่วนต่อประสานของงานวิจัย [4] ด้วยเช่นกัน

งานวิจัยนี้จะใช้การเปลี่ยนแปลงบริการเทรดเดอริในส่วนของเมทอด query ภายในส่วนต่อประสาน Lookup ซึ่งเป็นส่วนต่อประสานที่ใช้ในการทำอิมพอร์ตโดยอิมพอร์ตเดอริให้สามารถทำการค้นหาข้อเสนอบริการ (Service Offer) ที่มีชนิดของบริการ (Service Type) และคุณสมบัติของบริการ (Service Properties) ที่เท่าเทียมและสอดคล้องกับชนิดและคุณสมบัติของบริการที่อิมพอร์ตเดอริระบุมาในเมทอด query เมื่อเทรดเดอริไม่สามารถค้นหาข้อเสนอบริการตามชนิดและเงื่อนไขที่อิมพอร์ตเดอริระบุมาได้ อนึ่ง กลไกการค้นหาที่จะถูกเพิ่มเข้าไปในบริการเทรดเดอริจะเป็นไปตามรูปที่ 1.5 เช่นเดียวกัน แต่ในขั้นตอนของการตรวจสอบความผิดพลาดและกระบวนการคัดเลือกข้อเสนอบริการที่เหมาะสมจะกระทำอยู่ภายในบริการเทรดเดอริแทน



รูปที่ 1.4 การแทรกคำสั่งลงในโปรแกรมผู้รับบริการด้วยโปรแกรมฟรีโพรเซสเซอร์

ผลที่ได้รับจากการทำงานแทนที่กันของบริการในระดับอินสแตนซ์ของงานวิจัยนี้ภายหลังจากที่โปรแกรมของผู้รับบริการได้รับข้อมูลอ้างอิงของตัวดำเนินการแปลงแล้วจะสอดคล้องกับกลไกในงานวิจัยที่ [9] โดยตัวดำเนินการแปลงจะทำหน้าที่แปลงคำร้องขอใช้บริการตามส่วนต่อประสานของบริการที่ต้องการเดิมให้สอดคล้องกับส่วนต่อประสานของบริการที่ถูกใช้งานแทนที่ตามรูปที่ 1.3



รูปที่ 1.5 รูปแบบการค้นหาอินสแตนซ์ของบริการที่เท่าเทียมกันในงานวิจัยนี้



การจะเรียกใช้โปรแกรมพีไอพีเอสเซอร์เพื่อขยายความสามารถของโปรแกรมผู้รับบริการให้รองรับการเรียกใช้ข้อมูลความสัมพันธ์แบบเท่าเทียมกันจากคลังจัดเก็บส่วนต่อประสานได้นั้น ผู้พัฒนาโปรแกรมของผู้รับบริการจะเรียกใช้โปรแกรมพีไอพีเอสเซอร์โดยทำการระบุทั้งชื่อแฟ้มข้อมูลที่จะถูกแทรกคำสั่งการทำงานตลอดจนชื่อแฟ้มข้อมูลที่จะถูกสร้างขึ้นใหม่เพื่อใช้จัดเก็บผลลัพธ์ภายหลังการแทรกคำสั่งการทำงานแล้ว โปรแกรมพีไอพีเอสเซอร์จะทำการแทรกคำสั่งและจัดสร้างแฟ้มข้อมูลผลลัพธ์เพื่อให้ผู้พัฒนาโปรแกรมของผู้รับบริการสามารถนำไปคอมไพล์และเรียกใช้งานได้ต่อไป

การทดสอบผลการทำงานของงานวิจัยนี้จะใช้การทดสอบ 2 ส่วนโดยส่วนแรกจะใช้การสร้างสถานการณ์ให้โปรแกรมผู้รับบริการทำการค้นหาข้อมูลอ้างอิงของบริการโดยใช้คำสั่ง bind ของออร์บและส่วนที่สองจะใช้การเรียกใช้บริการจากเทอร์ตเตอร์ ในการทดสอบส่วนแรกจะพัฒนาโปรแกรมผู้รับบริการให้เชื่อมต่อไปยังบริการโดยใช้รูปแบบต่างๆของคำสั่ง bind จากนั้นตรวจสอบผลการทำงานของโปรแกรมเมื่อไม่มีอินสแตนซ์ของบริการที่ต้องการนั้นทำงานอยู่ว่าโปรแกรมผู้รับบริการที่ไม่ผ่านการแทรกคำสั่งการทำงานจะไม่สามารถเรียกใช้บริการนั้นได้ ในขณะที่โปรแกรมของผู้รับบริการที่ผ่านการแทรกคำสั่งการทำงานโดยโปรแกรมพีไอพีเอสเซอร์จะสามารถค้นหาข้อมูลจากคลังจัดเก็บส่วนต่อประสานและเชื่อมต่อไปยังอินสแตนซ์ของตัวดำเนินการแปลงเพื่อเรียกใช้บริการอื่นแทนที่ได้

การทดสอบในส่วนที่สองจะเน้นไปที่การทำงานของบริการเทอร์ตเตอร์ที่ผ่านการแก้ไขให้สามารถนำข้อมูลความสัมพันธ์แบบเท่าเทียมกันภายในคลังจัดเก็บส่วนต่อประสานมาใช้ในการคัดเลือกข้อเสนอบริการเพื่อส่งกลับไปให้โปรแกรมของผู้รับบริการ ทั้งนี้ผู้วิจัยจะใช้การทดสอบโดยทำการพัฒนาโปรแกรมของผู้รับบริการให้อิมพอร์ตบริการจากเทอร์ตเตอร์ โดยจะกำหนดเงื่อนไขของบริการที่ต้องการตามเมทอด query ให้เป็นบริการที่ไม่มีข้อเสนอบริการอยู่ในฐานข้อมูลของเทอร์ตเตอร์ จากนั้นตรวจสอบว่าเทอร์ตเตอร์ที่ผ่านการแก้ไขแล้วจะต้องสามารถค้นหาข้อมูลของบริการที่มีความสัมพันธ์แบบเท่าเทียมกันกับบริการที่ต้องการพร้อมจัดส่งข้อมูลอ้างอิงของตัวดำเนินการแปลงที่อยู่ในข้อเสนอบริการกลับไปยังโปรแกรมผู้รับบริการเพื่อเรียกใช้งานบริการนั้นๆต่อไปได้ ในขณะที่บริการเทอร์ตเตอร์ที่ไม่ได้รับการแก้ไขจะไม่สามารถกระทำได้

### 1.3 วัตถุประสงค์ของการวิจัย

เพื่อออกแบบและพัฒนากลไกที่รองรับการทำงานแทนที่กันในระดับอินสแตนซ์ของบริการโดยสามารถทำให้โปรแกรมผู้รับบริการที่ต้องการใช้งานอินสแตนซ์ของบริการหนึ่งสามารถใช้งานอินสแตนซ์ของอีกบริการหนึ่ง ซึ่งอาจมีส่วนต่อประสานที่แตกต่างกันแต่มีการทำงานที่เท่าเทียมกันได้ไม่ว่าโปรแกรมของผู้รับบริการจะทำการค้นหาอินสแตนซ์ของบริการผ่านทางออร์บของคอร์บาหรือเรียกใช้บริการจากเทอร์ตเตอร์ก็ตาม

#### 1.4 ขอบเขตของการวิจัย

1. ผู้ให้บริการจะเป็นผู้กำหนดความสัมพันธ์ของบริการและเป็นผู้รับผิดชอบในการกำหนดข้อมูลในตัวดำเนินการแปลงใช้ในการแปลงคำร้องขอใช้บริการจากบริการหนึ่งๆให้สอดคล้องกับส่วนต่อประสานของอีกบริการหนึ่งที่ถูกใช้แทนที่ โดยข้อมูลนี้จะถูกจัดเก็บไว้ภายในคลังจัดเก็บส่วนต่อประสานที่รองรับความสัมพันธ์แบบเท่าเทียมกัน

2. โปรแกรมพีโรเซสเซอร์ต้นแบบที่พัฒนาขึ้นจะทำงานอยู่บนสภาพแวดล้อมคอร์บาของผู้ผลิตซอฟต์แวร์รายเดียวเท่านั้นคือใช้งานกับซอฟต์แวร์วิสิโบรกเกอร์ (VisiBroker) ของบริษัทอินไพร์ส (Inprise) ทั้งนี้เนื่องจากลักษณะการพัฒนาโปรแกรมเพื่อทำการค้นหาบริการยังมีความแตกต่างกันอยู่บ้างระหว่างซอฟต์แวร์คอร์บาของผู้ผลิตแต่ละราย

3. โปรแกรมพีโรเซสเซอร์ต้นแบบจะจำกัดภาษาคอมพิวเตอร์ที่ใช้ในการพัฒนาโปรแกรมของผู้รับบริการอยู่ที่ภาษาเดียวเท่านั้นคือภาษาจาวา (Java)

4. การเปลี่ยนแปลงกลไกการทำงานของเทรดเดอร์จะทำการเปลี่ยนแปลงที่คำสั่งการทำงานภายในของโปรแกรมเทรดเดอร์ ซึ่งงานวิจัยนี้จะใช้โปรแกรมเทรดเดอร์ที่เป็นพีริแวลและเป็นของผู้ผลิตซอฟต์แวร์เพียงรายเดียวคือโปรแกรมออร์บาคัสเทรดเดอร์ (ORBacus Trader) เป็นต้น

5. งานวิจัยนี้จะใช้ข้อมูลภายในคลังจัดเก็บส่วนต่อประสานประกอบการทำงานของกลไกที่จะพัฒนาขึ้นเพื่อสร้างการแทนที่กันในระดับอินสแตนซ์ของบริการแต่เนื่องจากงานการพัฒนาค้างจัดเก็บส่วนต่อประสานที่รองรับความสัมพันธ์แบบเท่าเทียมกันยังอยู่ระหว่างการดำเนินงาน ซึ่งถ้าแล้วเสร็จไม่ทัน งานวิจัยนี้จะทำการจำลองคลังจัดเก็บส่วนต่อประสานขึ้นเองโดยให้มีข้อมูลและการดำเนินงานที่เพียงพอต่อการทำวิจัยเท่านั้น

#### 1.5 ขั้นตอนและวิธีดำเนินการวิจัย

1. ศึกษาแนวคิด กลไกการทำงานตลอดจนวิธีการพัฒนาโปรแกรมบนสถาปัตยกรรมระบบกระจายเชิงวัตถุคอร์บา

2. ศึกษาโครงสร้างและวิธีการพัฒนาโปรแกรมเพื่อเรียกใช้งานข้อมูลภายในคลังจัดเก็บส่วนต่อประสานที่รองรับความสัมพันธ์แบบเท่าเทียมกันของบริการ

3. ศึกษากลไกการทำงานภายในของโปรแกรมเทรดเดอร์และวิธีการพัฒนาโปรแกรมผู้รับบริการให้สามารถเรียกใช้บริการจากเทรดเดอร์ได้

4. จัดหา ติดตั้งเครื่องคอมพิวเตอร์ อุปกรณ์และซอฟต์แวร์ที่จำเป็นสำหรับการทำวิจัย

5. ออกแบบและพัฒนาโปรแกรมพีโรเซสเซอร์ที่สามารถแทรกคำสั่งเรียกใช้ข้อมูลจากคลังจัดเก็บส่วนต่อประสานที่รองรับความสัมพันธ์แบบเท่าเทียมกันของบริการลงในโปรแกรมของผู้รับบริการเพื่อสร้างกลไกการทำงานแทนที่กันระหว่างอินสแตนซ์ของบริการที่เท่าเทียมกัน

6. แก้ไขโปรแกรมเทอร์มเดอริให้มีกลไกในการนำข้อมูลจากคลังจัดเก็บส่วนต่อประสานที่รองรับความสัมพันธ์แบบเท่าเทียมกันมาใช้ในการแทนที่อินสแตนซ์ของบริการที่ต้องการด้วยอินสแตนซ์ของบริการอื่นที่ทำงานได้เท่าเทียมกัน

7. ทดสอบการใช้งานโปรแกรมฟรีโพรเซสเซอร์และโปรแกรมเทอร์มเดอริที่พัฒนาขึ้นว่าสามารถช่วยให้เกิดความโปร่งใสในการเปลี่ยนแปลงการทำงานของโปรแกรมผู้รับบริการให้ไปใช้งานบริการอื่นที่เท่าเทียมกันกับบริการที่ต้องการได้

8. สรุปผลการวิจัยและข้อเสนอแนะ

9. จัดทำรายงานวิทยานิพนธ์

## 1.6 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้โปรแกรมฟรีโพรเซสเซอร์ต้นแบบที่สามารถใช้ประโยชน์จากข้อมูลภายในคลังจัดเก็บส่วนต่อประสานที่รองรับความสัมพันธ์แบบเท่าเทียมกันของคอร์บาเพื่อสนับสนุนให้เกิดการเปลี่ยนแปลงการทำงานของโปรแกรมผู้รับบริการให้ไปใช้งานบริการอื่นที่เท่าเทียมกับบริการที่ต้องการได้อย่างโปร่งใสและมีความยืดหยุ่นมากขึ้น

2. ได้โปรแกรมเทอร์มเดอริที่ผ่านการแก้ไขให้รองรับการค้นหบริการอื่นที่เท่าเทียมกันกับบริการที่ต้องการจากคลังจัดเก็บส่วนต่อประสานที่รองรับความสัมพันธ์แบบเท่าเทียมกันได้

## 1.7 เนื้อหาของวิทยานิพนธ์

เนื้อหาของวิทยานิพนธ์ฉบับนี้จะแบ่งออกเป็น 6 บทและภาคผนวกอีก 2 บท โดยในบทที่ 1 ซึ่งได้กล่าวถึงมาแล้วนั้นจะเป็นรายละเอียดเกี่ยวกับบทนำ ในบทที่ 2 จะกล่าวถึงทฤษฎีและงานวิจัยต่างๆที่เกี่ยวข้องและจำเป็นสำหรับการทำวิจัยนี้ ในบทที่ 3 จะกล่าวถึงรายละเอียดเกี่ยวกับการออกแบบและพัฒนาโปรแกรมฟรีโพรเซสเซอร์ ซึ่งเป็นกลไกสำคัญที่จะช่วยทำให้คอร์บาเกิดการแทนที่กันของบริการที่มีความสัมพันธ์แบบเท่าเทียมกันในระดับอินสแตนซ์ งานวิจัยนี้จะใช้ทั้งโปรแกรมฟรีโพรเซสเซอร์ร่วมกับการเปลี่ยนแปลงการทำงานของบริการเทอร์มเดอริเพื่อให้คอร์บารองรับการแทนที่ของบริการที่เท่าเทียมกันได้อย่างสมบูรณ์ รายละเอียดเกี่ยวกับการแก้ไขบริการเทอร์มเดอริจะอยู่ในบทที่ 4 ส่วนบทที่ 5 ซึ่งเป็นบททดสอบจะกล่าวถึงรายละเอียดเกี่ยวกับการทดสอบการใช้งานโปรแกรมฟรีโพรเซสเซอร์ต้นแบบในการแทรกคำสั่งการทำงานลงในโปรแกรมของผู้รับบริการตลอดจนการทดสอบบริการเทอร์มเดอริที่ผ่านการแก้ไขโปรแกรมแล้วตามรายละเอียดในบทที่ 4 ในบทนี้จะแสดงตัวอย่างในการทดสอบทั้งสิ้น 2 ตัวอย่าง ในบทที่ 6 จะเป็นบทสรุปผลการวิจัยรวมทั้งข้อเสนอแนะซึ่งควรนำไปปรับปรุงให้ดียิ่งขึ้น ในส่วนของภาคผนวกทั้ง 2 บทนั้น ภาคผนวก ก. จะมีรายละเอียดเกี่ยวกับการพัฒนาโปรแกรมประยุกต์บนสถาปัตยกรรมระบบกระจายเชิงวัตถุคอร์บาด้วยการใช้ซอฟต์แวร์วิสิโบลโคเกอร์ โดยจะกล่าวถึง

ลำดับขั้นตอนและคำสั่งต่างๆที่จะถูกเรียกใช้งานในการพัฒนาโปรแกรมเช่นคำสั่ง `idl2java` สำหรับใช้ในการคอมไพล์เพิ่มข้อมูลไอดีแอลเป็นต้น ส่วนในภาคผนวก ข. จะมีรายละเอียดเกี่ยวกับการใช้งานซอฟต์แวร์เจทีบีและจาวาซีซีในการสร้างจาวาพาร์สเซอร์ที่ใช้ในโปรแกรมพีพีเอสเซอร์ของงานวิจัยนี้