บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

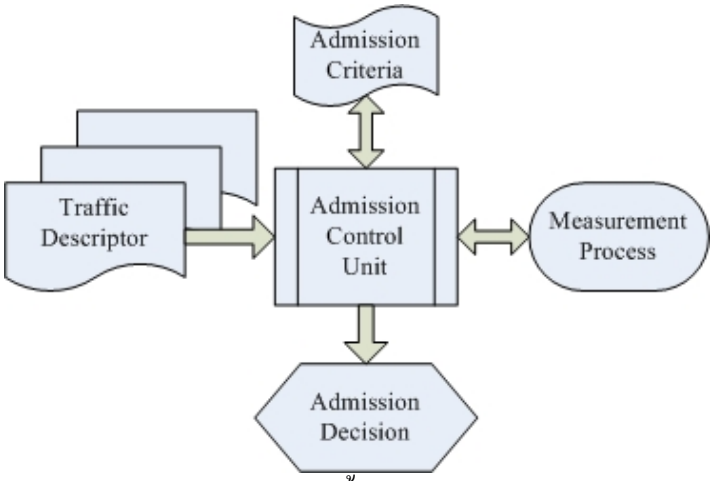
2.1 ส่วนควบคุมการอนุมัติการเชื่อมต่อ (Call Admission Control: CAC)

บางแอพพลิเคชันต้องการการรับประกันคุณภาพในการให้บริการจากระบบเครือข่ายซึ่งความต้องการคุณภาพในการให้บริการนี้อาจจะอยู่ในเทอมของแบนด์วิดธ์ขั้นต่ำหรือปริมาณการสูญหายของแพ็กเกตสูงที่สุดที่ยอมรับได้ ซึ่งเครือข่ายจะต้องมีอุปกรณ์ที่จะรองรับการเชื่อมต่อและสามารถจัดสรรหรือดูแลแบนด์วิดธ์ที่มีอยู่อย่างจำกัดของเครือข่ายให้สามารถบริการแก่การเชื่อมต่อที่ดำเนินการอยู่ตามระดับคุณภาพการให้บริการที่ตกลงกันเอาไว้ดังนั้นอุปกรณ์นี้อาจจะทำการปฏิเสธการเชื่อมต่อใหม่ในกรณีที่ทรัพยากรที่มีอยู่อาจจะไม่เพียงพอและการให้บริการแก่การเชื่อมต่อใหม่นั้นอาจเป็นสาเหตุให้เกิดผลกระทบต่อคุณภาพในการให้บริการแก่การเชื่อมต่อที่มีอยู่แล้วโดยกระบวนการตัดสินใจที่จะยอมรับหรือปฏิเสธการเชื่อมต่อที่มีการร้องขอต่อระบบเครือข่ายนี้ถูกเรียกว่า ส่วนควบคุมการอนุมัติการเชื่อมต่อ

โดยมากแล้วขั้นตอนการตัดสินใจยอมรับหรือปฏิเสธการร้องขอทรัพยากรของส่วนควบคุมการอนุมัติการเชื่อมต่อมักจะทำการตัดสินใจอย่างง่าย ๆ จากผลการคำนวณผลรวมของจำนวนแบนด์-วิดธ์ที่ใช้อยู่กับจำนวนแบนด์วิดธ์ที่การเชื่อมต่อใหม่ต้องการและถ้าผลรวมที่ได้มีค่ามากกว่าจำนวนแบนด์วิดธ์ทั้งหมดของเครือข่าย ก็จะทำการปฏิเสธการเชื่อมต่อนั้น

2.1.1 องค์ประกอบพื้นฐานของส่วนควบคุมการอนุมัติการเชื่อมต่อ

ส่วนควบคุมการอนุมัติการเชื่อมต่อจะมีองค์ประกอบที่สำคัญสามส่วนคือ ลักษณะทราฟฟิกเงื่อนไขในการอนุมัติ และกระบวนการวัดค่าพารามิเตอร์ ซึ่งทั้ง 3 องค์ประกอบจะมีความสัมพันธ์ตามรูปที่ 2.1



รูปที่ 2.1 ความสัมพันธ์ขององค์ประกอบพื้นฐานกับส่วนควบคุมการอนุมัติการเชื่อมต่อ

2.1.1.1 ลักษณะทราฟฟิก (Traffic Descriptor)

ลักษณะทราฟฟิกจะเป็นกลุ่มของค่าพารามิเตอร์ซึ่งเป็นคุณลักษณะของทราฟฟิกต้นกำเนิด เช่น ลักษณะทราฟฟิกของโทเก้นบัคเก็ต (token bucket) จะมีอัตราการใส่โทเก้นเท่ากับ r และมีขนาดของบัคเก็ตเท่ากับ b ซึ่งถ้าใช้โทเก้นบัคเก็ตเป็นแหล่งกำเนิดทราฟฟิก จะส่งจำนวนทราฟฟิกเท่ากับ r*t+b ตลอดช่วงระยะเวลา t ถ้าโทเก้นบัคเก็ตจะมีอัตราการส่งสูงสุดเท่ากับ p จะทำให้ระยะเวลาระหว่างแต่ละแพ็กเกตจะเท่ากับ 1/p

2.1.1.2 กระบวนการวัดค่าพารามิเตอร์ (Measurement Process)

อัลกอริทึมของส่วนควบคุมการอนุมัติการเชื่อมต่อต้องการค่าพารามิเตอร์เป็นอินพุท ซึ่งจะใช้ค่าพารามิเตอร์ที่ได้จากกระบวนการวัดเพื่อไปใช้ในการกำหนดเงื่อนไขในการอนุมัติการเชื่อมต่อ โดยส่วนใหญ่แล้วจะใช้จำนวนแบนด์วิดธ์ของเครือข่าย อัตราการมาถึงของแพ็กเกตเฉลี่ยของแพ็กเกต และความคับคั่งของข้อมูลในระบบเครือข่าย เป็นต้น

2.1.1.3 เงื่อนไขการอนุมัติ (Admission Criteria)

เงื่อนไขการอนุมัติคือ กฎซึ่งส่วนควบคุมการอนุมัติจัดตั้งการเชื่อมต่อที่ใช้ในการตอบรับหรือตอบปฏิเสธการร้องขอ เพื่อจัดสรรแบนด์วิดธ์ให้แก่ผู้ใช้งาน โดยการตัดสินใจของส่วนควบคุมการอนุมัติการเชื่อมต่อจะกระทำบนพื้นฐานของการประเมินผลกระทบที่มีต่อการเชื่อมต่ออื่น ๆ รวมถึงประสิทธิภาพการใช้งานแบนด์วิดธ์ของเครือข่ายด้วย

2.1.2 การพัฒนาส่วนควบคุมการอนุมัติการเชื่อมต่อ

โดยการพัฒนาส่วนควบคุมการอนุมัติการเชื่อมต่อจากนักวิจัยหลายท่านที่ผ่านมา จะมีแนวทางไปในด้านการลดปัญหาเรื่องขนาดของสถาปัตยกรรมแบบ IntServ และการแก้ปัญหาความเชื่อถือได้จากการรับประกันคุณภาพในการให้บริการของสถาปัตยกรรมแบบ DiffServ ซึ่งสามารถจำแนกได้เป็น 4 ประเภทตามลักษณะการควบคุมได้แก่ (1) ส่วนควบคุมการอนุมัติการเชื่อมต่อบนทุก ๆ เราเตอร์ตามเส้นทาง (2) ส่วนควบคุมการอนุมัติการเชื่อมต่อจากส่วนกลาง (3) ส่วนควบคุมการอนุมัติการเชื่อมต่อผ่านตัวแทนของระบบเครือข่าย และ (4) ส่วนควบคุมการอนุมัติการเชื่อมต่อที่มีการตัดสินใจโดยการวัดค่าพารามิเตอร์

2.1.2.1 ส่วนควบคุมการอนุมัติการเชื่อมต่อบนทุก ๆ เราเตอร์ตามเส้นทาง

ในประเภทนี้จะมีส่วนควบคุมการอนุมัติการเชื่อมต่อที่ทุก ๆ เราเตอร์ตามเส้นทางที่ต้องการจะสื่อสาร โดยในแต่ละเราเตอร์จะมีสิทธิ์ขาดในการตัดสินใจที่จะยอมรับหรือไม่ในการให้บริการแก่ผู้ที่ร้องขอ ซึ่งวิธีนี้จะสามารถรับประกันคุณภาพในการให้บริการได้อย่างดี

ก. สัญญาณโพรโตคอลสำรองทรัพยากร (RSVP Signaling)

โพรโตคอลสำรองทรัพยากร [2] [3] คือ สัญญาณที่ใช้ในการสร้างและบำรุงรักษาการสำรองทรัพยากรในเครือข่าย IntServ โดยมีวัตถุประสงค์เพื่อต้องการทรัพยากรและสำรองไว้ที่แต่ละเราเตอร์ตามเส้นทางที่จะไปสู่ปลายทาง โดยจะมีการทำงานดังนี้ (1) ผู้ส่งจะส่ง PATH message ไปยังผู้รับเพื่อบอกความต้องการคุณภาพในการให้บริการของการเชื่อมต่อและแต่ละเราเตอร์ตามเส้นทางที่จะต้องส่งต่อไปจนถึงปลายทาง (2) เมื่อได้รับ PATH message ผู้รับจะส่ง RESV message กลับไปยังผู้ส่งเพื่อทำการสำรองทรัพยากรของเครือข่ายไว้สำหรับการเชื่อมต่อ โดยแต่ละเราเตอร์ตามเส้นทางสามารถยอมรับหรือปฏิเสธการร้องขอนี้ถ้าไม่มีทรัพยากรเพียงพอ ถ้าเราเตอร์ปฏิเสธการร้องขอ ก็จะส่ง error message ไปยังผู้รับ และการติดต่อโดยใช้สัญญาณการสำรองทรัพยากรนี้จะสิ้นสุดลง แต่ถ้าตอบยอมรับเราเตอร์ก็จะมีการสำรองทรัพยากรและแบนด์วิดธ์ไว้ให้สำหรับการเชื่อมต่อนั้น

2.1.2.2 ส่วนควบคุมการอนุมัติการเชื่อมต่อจากส่วนกลาง

ในประเภทนี้จะมีการใช้ส่วนควบคุมการอนุมัติการเชื่อมต่อจากส่วนกลาง ซึ่งจะได้รับการติดต่อจากเราเตอร์ขอบอีกทีนึง โดยในประเภทนี้จะเป็นการพัฒนาเพื่อลดการใช้สัญญาณในการดูแลรักษาสถานะการรับประกันคุณภาพในการให้บริการ

ก. ส่วนควบคุมการอนุมัติการเชื่อมต่อที่มีส่วนต่อรองแบนด์วิดธ์
(Bandwidth Broker based Admission Control)

ส่วนต่อรองแบนด์วิดธ์ (Bandwidth Broker: BB) [17] จะเป็นการเอาสัญญาณการสำรองทรัพยากรที่เราเตอร์หลักจากสถาปัตยกรรมแบบ IntServ ออก โดยจะมีการเก็บและจัดการข้อมูลที่ศูนย์กลาง ซึ่งส่วนต่อรองแบนด์วิดธ์จะต้องเป็นเราเตอร์หรือซอฟต์แวร์แพ็กเกจที่ได้ติดตั้งไว้ในเราเตอร์/สวิตช์ของเครือข่าย

โมดูลหลักของส่วนต่อรองแบนด์วิดธ์คือ ส่วนควบคุมการอนุมัติการเชื่อมต่อและการหาเส้นทาง โดยการทำงานคือจะดูแลรักษาสถานะของคุณภาพในการให้บริการของเครือข่ายและรับผิดชอบเกี่ยวกับส่วนควบคุมการอนุมัติการเชื่อมต่อและการสำรองทรัพยากร โดยภายหลังจากการตัดสินใจที่จะยอมรับการเชื่อมต่อโดยพิจารณาตลอดเส้นทางไปจนถึงผู้รับแล้ว ส่วนต่อรองแบนด์วิดธ์จะใส่ข้อมูลลงในฐานข้อมูลเกี่ยวกับโครงสร้างของเครือข่าย การเชื่อมต่อ และสถานะการควบคุมคุณภาพในการให้บริการตามเส้นทางในทุก ๆ โหนด ซึ่งโดยปกติแล้วในหนึ่งเครือข่ายจะมีส่วนต่อรองแบนด์วิดธ์หนึ่งตัว

2.1.2.3 ส่วนควบคุมการอนุมัติการเชื่อมต่อผ่านตัวแทนของระบบเครือข่าย

ในประเภทนี้จะมีส่วนควบคุมการอนุมัติการเชื่อมต่อโดยใช้ตัวแทนของระบบเครือข่ายซึ่งส่วนใหญ่แล้วจะเป็นเราเตอร์ตัวแรกที่ผู้ร้องขอได้ส่งมายังเครือข่ายหรือที่เรียกว่า เราเตอร์ขอบขาเข้า โดยในเราเตอร์นี้จะมีอำนาจเต็มในการตัดสินใจที่จะรับหรือไม่รับการเชื่อมต่อจากผู้ร้องขอ และในประเภทนี้จะมีการพัฒนาเพื่อแก้ปัญหาจากสถาปัตยกรรม DiffServ โดยไม่มีการใช้สัญญาณในการรักษาคุณภาพในการให้บริการที่เราเตอร์หลักของเครือข่าย

ก. สถานะการเปลี่ยนแปลงของแพ็กเกต (Dynamic Packet State: DPS)

เทคนิคสถานะการเปลี่ยนแปลงของแพ็กเกต [18] คือ ข้อมูลสถานะของการเชื่อมต่อที่ถูกใส่เข้าไปที่ส่วนหัวของแพ็กเกต (เช่นอัตราการสำรองแบนด์วิดธ์) ซึ่งจำเป็นสำหรับสัญญาณการเชื่อมต่อและการจัดการสถานะของการควบคุมคุณภาพ โดยที่เราเตอร์ขอบขาเข้าจะเริ่มต้นจัดเตรียมข้อมูลสถานะ และที่เราเตอร์หลักจะทำการประมวลผลแพ็กเกตที่เข้ามาบนพื้นฐานของสถานะที่มากับแพ็กเกต และจะทำการอับเดทแล้วใส่ไว้ที่ส่วนหัวของแพ็กเกตแล้วส่งไปยังเราเตอร์ตัวถัด ไป

ในเทอมของส่วนควบคุมการอนุมัติการเชื่อมต่อจะใช้สัญญาณโพรโตคอลสำรองทรัพยากรเพื่อสื่อสารกันระหว่างผู้ส่งและผู้รับ แต่สัญญาณโพรโตคอลสำรองทรัพยากรนี้จะทำงานกับเราเตอร์ขอบเท่านั้น เริ่มแรกที่เราเตอร์ขอบขาเข้าเมื่อได้รับ PATH message ก็จะส่งตรงต่อไปยังเราเตอร์ขอบขาออกเลย ซึ่งเมื่อเราเตอร์ขอบขาออกได้รับ RESV message แล้วก็จะส่งตอบกลับมายังเราเตอร์ขอบขาเข้าเลย และเมื่อแต่ละโหนดระหว่างเส้นทางของการสื่อสารได้รับสัญญาณโพรโตคอลสำรองทรัพยากรนี้ก็จะทำการตรวจสอบอัตราการสำรองทรัพยากรที่โหนดนั้น ๆ ด้วย โดยจะใช้วิธีการคำนวณอย่างง่าย ๆ [18] และเมื่อการเชื่อต่อสิ้นสุดลงก็จะทำการปล่อยแบนด์วิดธ์กลับสู่เครือข่าย

ข. การรวมสัญญาณ IntServ (Aggregation in IntServ)

การรวมสัญญาณ [7] เป็นกลไกที่ใช้ในการลดจำนวนสัญญาณที่ส่งโดยสถาปัตยกรรมแบบ IntServ โดยส่วนควบคุมการอนุมัติการเชื่อมต่อในเทคนิคนี้จะกระทำการรวมกลุ่มของสัญญาณของแต่ละการเชื่อมต่อและรวมสัญญาณที่จำเป็นในการดูแลรักษาสถานะการสำรองทรัพยากรที่แต่ละเราเตอร์หลักด้วย

นอกเหนือจากนั้น จะมีส่วนควบคุมการอนุมัติการเชื่อมต่อที่เราเตอร์ขอบขาเข้าซึ่งจะทำการตัดสินใจบนพื้นฐานของจำนวนแบนด์วิดธ์ โดยจะอนุญาตให้ทำการปรับเปลี่ยนการสำรองทรัพยากรได้ แต่เราเตอร์ขอบขาเข้าจะใช้เวลาในการปรับการสำรองทรัพยากรที่เราเตอร์หลักมากกว่าวิธีการสำรองทรัพยากรของสถาปัตยกรรมแบบ IntServ

2.1.2.4 ส่วนควบคุมการอนุมัติการเชื่อมต่อที่มีการตัดสินใจโดยการวัดค่าพารามิเตอร์

ในประเภทนี้ส่วนใหญ่จะมีส่วนควบคุมการอนุมัติการเชื่อมต่อที่เราเตอร์ขาออกโดยจะมีการวัดค่าพารามิเตอร์ตามคุณภาพในการให้บริการที่ร้องขอมาจากแพ็กเกตร้องขอ ซึ่งจะนำผลที่ได้จากการวัดค่าพารามิเตอร์ไปใช้ในการตัดสินใจที่จะยอมรับหรือไม่ยอมรับการเชื่อมต่อนั้น

ก. ส่วนควบคุมการอนุมัติการเชื่อมต่อที่มีการวัดค่าที่เราเตอร์ขาออก (Measurement-Based Admission Control in Egress Router)

ส่วนควบคุมการอนุมัติการเชื่อมต่อในวิธีนี้ [19] [20] จะทำการตัดสินใจที่เราเตอร์ขาออกเท่านั้นโดยไม่มีการดูแลรักษาสถานะตลอดการเชื่อมต่อที่เราเตอร์หลักหรือที่เราเตอร์ขาออก โดยในการตัดสินใจจะทำบนพื้นฐานของการวัดค่าที่เราเตอร์ขาออกซึ่งเทคนิคหลักคือ การวัดบริการที่สามารถใช้ได้ตลอดเส้นทางตั้งแต่ต้นทางไปจนถึงปลายทางโดนใช้ระบบโมเดลกล่องดำ ซึ่งจะทำการวัดหรือควบคุมค่าตัวแปรที่มีผลกระทบต่อทราฟฟิกอื่น ที่เราเตอร์ขาออกจะทำการคำนวณอัตราการมาถึงและบริการที่ให้ได้ต่ำสุดและจะทำการกำหนดอัลกอริทึมของส่วนควบคุมการอนุมัติการเชื่อมต่อออกมาเพื่อทำการรับหรือปฏิเสธการจัดตั้งการเชื่อมต่อ ซึ่งถ้าบริการขั้นต่ำที่สามารถให้ได้มีเพียงพอและการรับประกันคุณภาพในการให้บริการมีครบตามความต้องการของการเชื่อมต่อนั้นก็จะทำการยอมรับและให้ทำการเชื่อมต่อได้และจะต้องไม่มีผลกระทบต่อการเชื่อมต่ออื่น ๆ ที่ได้ยอมรับไปแล้วในก่อนหน้านี้

ข. การตรวจผลของส่วนควบคุมการอนุมัติการเชื่อมต่อที่จุดปลายทาง (End-Point Admission Control through Probing)

ในกลไกนี้ [21] [22] จะทำการอนุมัติการเชื่อมต่อใหม่ที่โฮสต์ปลายทางหรือเราเตอร์ขาออกหรือเราเตอร์ขาเข้าโดยผ่านการวิเคราะห์ความคับคั่งของเครือข่ายในแต่ละเส้นทางของการเชื่อมต่อ โดยก่อนที่จะอนุญาตให้มีการเชื่อมต่อใหม่ได้ ผู้ส่งที่ส่งแพ็กเกตมาตามลักษณะทราฟฟิกที่ได้ร้องขอไว้จะถูกตรวจสอบค่าตัวแปรอัตราการสูญหายของแพ็กเกตและค่าเวลาหน่วงเป็นต้นที่ผู้รับ เพื่อทำการตรวจสอบระดับความคับคั่งของเครือข่าย ถ้ามีประสิทธิภาพตามคุณภาพในการให้บริการที่ร้องขอมาก็จะสามารถรับการเชื่อมต่อใหม่นี้ได้ แต่ถ้าประสิทธิภาพลดต่ำลงก็จะทำการปฏิเสธการเชื่อมต่อใหม่นี้ซึ่งกลไกนี้จะนำไปไว้ที่จุดปลายทางทำให้ไม่ต้องใช้สัญญาณโพรโตคอลหรือฟังก์ชันพิเศษที่จะส่งไปยังเราเตอร์หลักหรือเราเตอร์ขอบ

2.2 สถาปัตยกรรมการรับประกันคุณภาพในการให้บริการ (Quality of Service Architectures)

สถาปัตยกรรมที่สำคัญซึ่งกำหนดโดยคณะทำงานอินเทอร์เน็ต (Internet Engineering Task Force: IETF) เพื่อให้บริการในระดับที่แตกต่างกันของการเชื่อมต่ออินเทอร์เน็ตโพรโตคอล ซึ่งสามารถแบ่งตามการทำงานออกได้เป็น 3 ประเภท

- *The Integrated Services Architecture* เป็นการสำรองทรัพยากรตั้งแต่ตั้งทางไปจนถึงปลายทาง
- *The Differentiated Services Architecture* เป็นการรับประกันให้แต่ละแพ็กเกตที่แตกต่างกันไปตามประเภทของแพ็กเกตนั้น
- *Integrated Services over Differentiated Services* เป็นการรวมข้อดีของทั้งสองสถาปัตยกรรมไว้ด้วยกัน

2.2.1 นิยามของคุณภาพในการให้บริการ

คุณภาพในการให้บริการจะเป็นข้อตกลงระหว่างผู้ใช้บริการและผู้ให้บริการ ซึ่งข้อตกลงนั้นจะอยู่ในรูปของตัวแปรที่สามารถวัดค่าประสิทธิภาพได้ เช่น การสูญหายของข้อมูล (data loss) เวลาหน่วงของเครือข่ายสูงสุด (maximum network delay) ความแปรปรวนต่ำสุด (minimal jitter) แบนด์วิดธ์ต่ำสุด/สูงสุด (minimal/maximum bandwidth) เป็นต้น

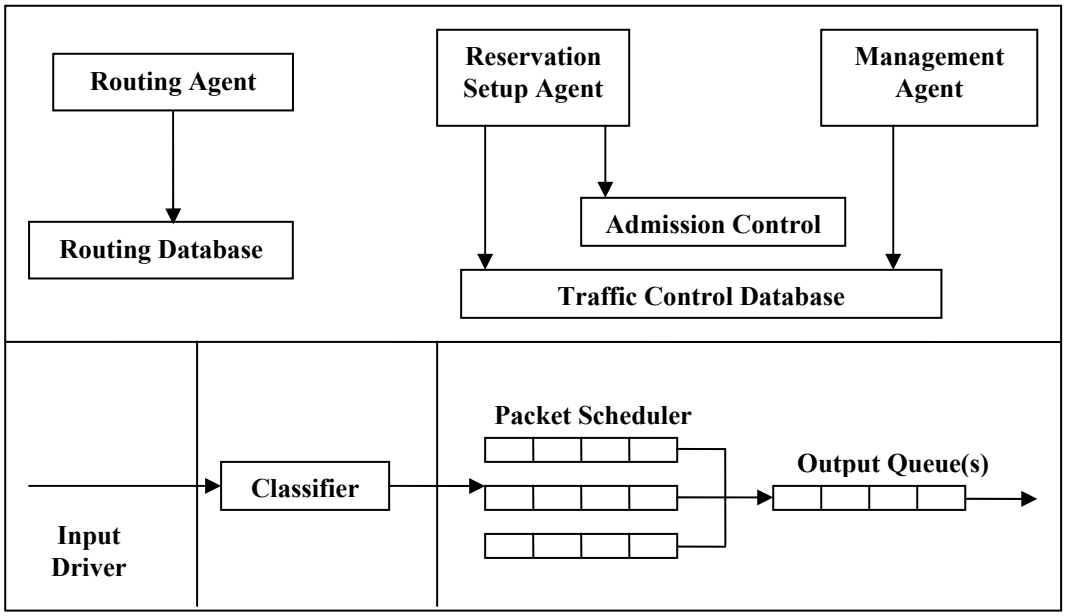
2.2.2 The Integrated Services Architecture

สถาปัตยกรรม The Integrated Services (IntServ: IS) [1] ได้ถูกกำหนดโดยคณะทำงาน IETF โดยเสนอให้มีการขยายการรองรับการสื่อสารนอกเหนือจากทราฟฟิกแบบ best-effort นั่นก็คือแอพพลิเคชั่นแบบ real-time ที่ใช้กันแพร่หลายบนอินเทอร์เน็ต ซึ่งอินเทอร์เน็ตแบบดั้งเดิมได้มีการรับประกันคุณภาพในการให้บริการแบบง่าย ๆ คือ การส่งข้อมูล best-effort แบบจุดต่อจุด

เนื่องจากมีเวลาที่สูญเสียไปในการหาเส้นทางและการสูญหายของแพ็กเกตจากความคับคั่งของข้อมูล ซึ่งทำให้แอพพลิเคชั่นแบบ real-time ไม่สามารถทำงานได้ดีบน best-effort อินเทอร์เน็ต ซึ่งการประชุมผ่านวิดิทัศน์ การแพร่ภาพวิดิทัศน์ และการประชุมผ่านเสียง เป็นต้น ต่างก็ต้องการการรับประกันแบนด์วิดธ์ที่ให้กับภาพและเสียงในคุณภาพที่ยอมรับได้ เพื่อความต้องการที่รองรับบริการต่าง ๆ เหล่านี้จึงจำเป็นที่จะต้องมีการปรับเปลี่ยนโครงสร้างหลักของอินเทอร์เน็ต เพื่อที่จะให้มีการควบคุมการเสียเวลาของแพ็กเกตและการอนุญาตใช้แบนด์วิดธ์ตั้งแต่ต้นทางไปจนถึงปลายทาง (end-to-end)

2.2.2.1 โมเดล Integrated Services

เพื่อที่จะรองรับโมเดล IntServ อินเทอร์เน็ตเราเตอร์จึงจำเป็นที่จะต้องให้คุณภาพในการให้บริการที่เหมาะสำหรับแต่ละการเชื่อมต่อ ฟังก์ชันของเราเตอร์ต้องให้คุณภาพในการให้บริการที่แตกต่างกันเรียกว่า ส่วนควบคุมทราฟฟิก (Traffic Control) ซึ่งจะประกอบไปด้วย ส่วนควบคุมการอนุมัติ (Admission Control) ส่วนแยกประเภทของแพ็กเกต (Packet Classifier) ส่วนจัดลำดับของแพ็กเกต (Packet Scheduler) และโพรโตคอลสำรองทรัพยากร (Reservation Setup Protocol) ดังรูปที่ 2.2



รูปที่ 2.2 Implement Model [1]

2.2.2.1.1 ส่วนควบคุมการอนุมัติและนโยบาย

ส่วนควบคุมการอนุมัติของ IntServ จะประกอบไปด้วยอัลกอริทึม (Algorithm) ที่ใช้ในการตัดสินใจ ซึ่งเราเตอร์จะใช้ในการหาค่า ถ้าในเส้นทางที่จะไปสู่ปลายทางมีทรัพยากรเพียงพอกับคุณภาพในการให้บริการก็จะยอมรับการเชื่อมต่อใหม่ แต่ถ้าในเส้นทางนั้นไม่มีทรัพยากรเพียงพอที่จะรับการเชื่อมต่อเพิ่มขึ้นก็จะทำการตอบปฏิเสธการร้องขอนั้น ในกรณีที่ตอบรับการร้องขอใหม่ก็จะมีการสำรองแบนด์วิดธ์ไว้ที่เราเตอร์นั้น ๆ ซึ่งส่วนควบคุมการอนุมัติจะต้องมีในเราเตอร์ทุกตัวตลอดเส้นทางที่จะไปสู่ปลายทาง โดยจะทำหน้าที่ตัดสินใจตอบรับ/ตอบปฏิเสธเมื่อมีการร้องขอและสำรองแบนด์วิดธ์ให้แต่ละการร้องขอ

2.2.2.1.2 ส่วนแยกประเภทของแพ็กเกต

การจำแนกประเภทของแพ็กเกตเพื่อระบุแพ็กเกตของแต่ละโฮสต์ และเราเตอร์จะให้บริการในระดับที่ต่างกันเพื่อที่จะให้มีการควบคุมทราฟฟิกที่มีประสิทธิภาพโดยการจำแนกแต่ละแพ็กเกตที่เข้ามาไปสู่ในแต่ละคลาส (class) ทุกแพ็กเกตที่อยู่ในคลาสเดียวกันจะได้รับการบริการที่เหมือนกันจากส่วนจัดลำดับแพ็กเกต การเลือกคลาสจะขึ้นอยู่กับไอพีแอดเดรส (IP Address) ต้นทาง ไอพีแอดแดรสปลายทาง และหมายเลขช่องทาง (port number) ที่อยู่ในส่วนหัวของแพ็กเกต หรือหมายเลขการจำแนกประเภทซึ่งจะเพิ่มเข้าไปในแต่ละแพ็กเกต คลาสสามารถจำแนกไปตามการเชื่อมต่อได้ เช่น การเชื่อมต่อแบบวิดิทัศน์ทั้งหมดจากการประชุมผ่านวิดิทัศน์ ซึ่งมีหลายสมาชิกสามารถรวมได้เป็นหนึ่งประเภทของบริการแต่อาจจะเป็นไปได้ที่มีหนึ่งการเชื่อมต่อแยกไปเป็นหนึ่งประเภทของบริการได้

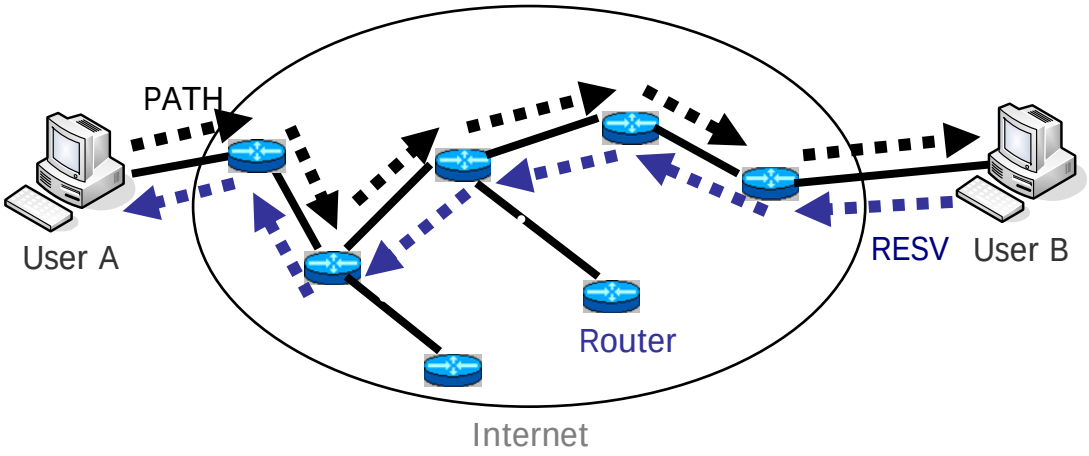
2.2.2.1.3 ส่วนจัดลำดับแพ็กเกต

ส่วนจัดลำดับแพ็กเกตจะมีหน้าที่จัดการส่งแพ็กเกตทั้งหลายของโฮสต์ต่อไปยังเราเตอร์บนพื้นฐานของระดับของบริการ การจัดการคิว (queue) และอัลกอริทึมการจัดลำดับที่แตกต่างกัน ส่วนจัดลำดับของแพ็กเกตจะทำให้แน่ใจว่าแพ็กเกตที่ส่งไปนั้นจะได้รับคุณภาพในการให้บริการตามที่กำหนดไว้ ซึ่งส่วนการจัดลำดับสามารถบริหารทราฟฟิกให้เป็นไปตามระดับของบริการโดยการจัดลำดับคิวในการรับส่งแพ็กเกต

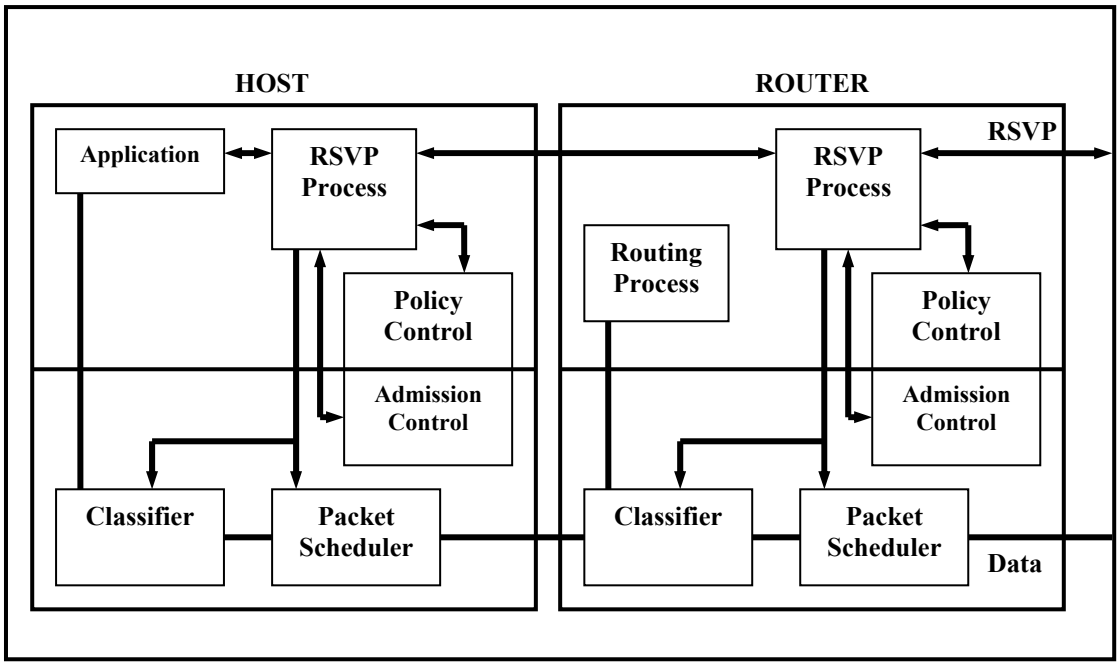
2.2.2.2 โพรโตคอลสำรองทรัพยากร (Resource Reservation Protocol)

โมเดล IntServ โดยการใช้โพรโตคอลสำรองทรัพยากร (Resource Reservation Protocol: RSVP) [2] [3] เพื่อเป็นการจองไว้ตามคุณภาพของบริการในเครือข่ายอินเทอร์เน็ตโพรโตคอล โดยโพรโตคอลสำรองทรัพยากรจะมีการสำรองทรัพยากรไปตามเส้นทางเดียวกันกับเส้นทางที่เชื่อมต่อ

จากรูปที่ 2.3 เมื่อผู้ใช้ A ต้องการส่งข้อมูลผ่านเครือข่าย IntServ ไปยังผู้ใช้ B ในขั้นแรกผู้ใช้ A จะต้องสร้างแพ็กเกตร้องขอที่ประกอบไปด้วยข้อมูลของผู้ส่ง ลักษณะทราฟฟิก และแบนด์วิดธ์ที่จะใช้ซึ่งจะเรียกว่า PATH message ซึ่งจะมีตัวแปรของทราฟฟิกอยู่ด้วย โดยจะส่งแพ็กเกตนี้ผ่านเครือข่ายไปตามเราเตอร์ต่าง ๆ ตามเส้นทางที่จะไปถึงผู้รับ B เมื่อเราเตอร์ได้รับ PATH message นี้ก็จะเก็บข้อมูลไว้แล้วส่งไปให้เราเตอร์ถัดไป เมื่อ PATH message มาถึง ผู้ใช้ B ก็จะสร้างแพ็กเกตตอบรับขึ้นมา ซึ่งจะเรียกว่า RESV message แล้วส่งกลับไปยังผู้ใช้ A โดยจะส่งไปในเส้นทางเดียวกันกับที่ส่ง PATH message มา เมื่อผู้ใช้ A ได้รับ RESV message ก็จะเริ่มทำการส่งข้อมูลไปยังผู้ใช้ B ได้ โดยในระหว่างที่กำลังส่งข้อมูลอยู่นั้น แต่ละเราเตอร์จะยังคงเก็บรักษาสถานะการรับประกันคุณภาพในการให้บริการตลอดเส้นทางของการสื่อสารระหว่างผู้ใช้ A และผู้ใช้ B ไว้จนสิ้นสุดการเชื่อมต่อ



รูปที่ 2.3 IntServ/RSVP Model



รูปที่ 2.4 Host-Router model in RSVP [2]

รูปที่ 2.4 เมื่อแอพพลิเคชั่นต้องการที่จะส่งแพ็กเกตโดยการสำรองทรัพยากร โพรโตคอล RSVP จะพยายามจัดตั้งการเชื่อมต่อตามคุณภาพในการให้บริการ RSVP จะรายงานแก่ส่วนจำแนกแพ็กเกตและส่วนจัดลำดับแพ็กเกตในแต่ละโหนดในการประมวลผลแพ็กเกตสำหรับการเชื่อมต่อ เมื่อแอพพลิเคชันได้ส่งแพ็กเกตข้อมูลไปยังส่วนจำแนกข้อมูลในโหนดแรก ก็จะทำการจำแนกการเชื่อมต่อนี้ไปสู่คลาสที่ให้บริการเฉพาะตามคุณภาพในการให้บริการ ซึ่งการเชื่อมต่อนี้จะจดจำไอพีแอดเดรสของผู้รับ ไอพีแอดเดรสผู้ส่ง และพอร์ต แล้วจะส่งต่อไปยังส่วนจัดลำดับแพ็กเกต โดยใน

ส่วนจัดลำดับแพ็กเกตจะทำการส่งแพ็กเกตต่อไปตามคลาสที่ให้บริการไปที่เราเตอร์ถัดไปจนถึงจุดหมายปลายทาง

2.2.2.3 คลาสของบริการ (Classes of Service)

โมเดล Integrated Services ได้ให้บริการในคลาสที่ต่าง ๆ กันโดยรวมเอาบริการที่กำหนดโดยคณะทำงาน IETF ไว้ในโมเดลนี้ด้วย ซึ่งก็คือ บริการแบบรับประกัน (Guaranteed Service) [4] และบริการแบบไม่รับประกัน (Controlled Service) [5]

IntServ ได้กำหนดบริการแบบไม่รับประกันจากพฤติกรรมของทราฟฟิกตั้งแต่ต้นทางไปจนถึงปลายทาง โดยมีการทดสอบประสิทธิภาพของบริการแบบ best-effort จากค่าพารามิเตอร์เมื่อเครือข่ายไม่มีภาระ ซึ่งระดับในการบริการที่สูงกว่า best-effort จะสามารถทำให้เครือข่ายเกิดความคับคั่งได้แต่แอพพลิชันที่ใช้บริการแบบนี้มีโอกาสสูงที่จะส่งแพ็กเกตได้สำเร็จ ซึ่งจำนวนแพ็กเกตที่ถูกละทิ้งควรจะเท่ากับอัตราความผิดพลาดในการส่งข้อมูล เวลาแฝงของเครือข่ายจะต้องมีค่าไม่มากกว่าค่าเวลาหน่วงที่ยอมรับได้ในการส่งแพ็กเกต

บริการแบบไม่รับประกันจะไม่มีการกำหนดตัวแปรที่ใช้ในการความคุมเวลาหน่วงหรือการสูญหายของแพ็กเกต แต่บริการนี้จะใช้กับแอพพลิเคชันแบบ real-time เช่น การกระจายภาพและเสียงแบบทิศทางเดียว ซึ่งไม่จำเป็นที่จะต้องควบคุมเวลาหน่วงโดยเฉพาะเจาะจง แต่ต้องมีการสำรองไว้ที่เครื่องผู้รับในการรับข้อมูลมาก่อนที่จะแสดงผล

บริการแบบรับประกันจะเป็นการสื่อสารสำหรับแอพพลิเคชันที่ต้องการการรับประกันทั้งแบนด์วิดธ์และค่าเวลาหน่วงสูงสุดที่ยอมรับได้จากเครือข่าย ในการรับประกันแบนด์วิดธ์ นั่นหมายความว่าจะต้องไม่มีการสูญหายของแพ็กเกตในระหว่างอยู่ในแถวคอยของส่วนสำรองข้อมูล ส่วนการรับประกันค่าเวลาหน่วงหมายถึงเวลาหน่วงของแพ็กเกตสูงสุดที่ยอมรับได้ในระหว่างส่งแพ็กเกตผ่านเครือข่ายตั้งแต่ต้นทางไปจนถึงปลายทาง

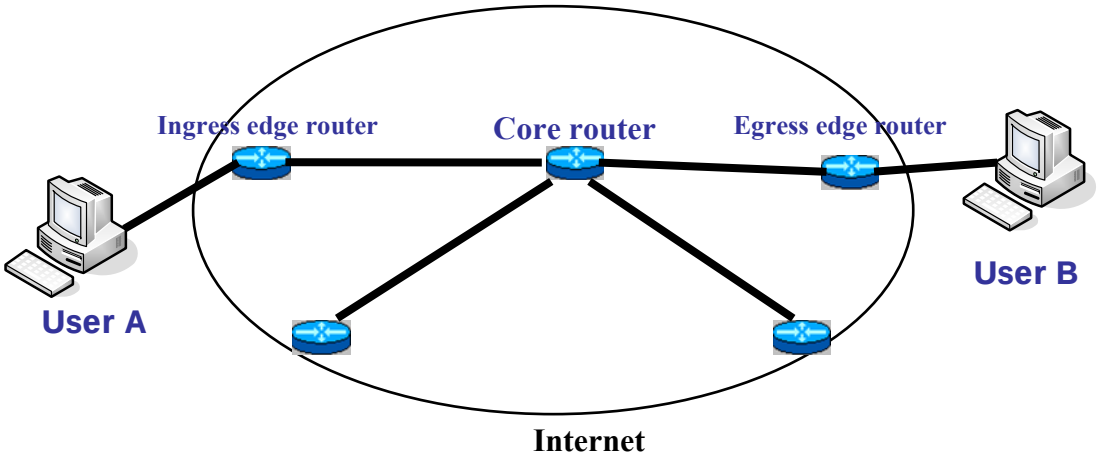
ปัญหาหลักของสถาปัตยกรรมแบบ IntServ คือ จะมีการเก็บรักษาสถานะการสำรองทรัพยากรไว้ที่ทุกเราเตอร์ตามเส้นทางของการเชื่อมต่อ ซึ่งเมื่อมีผู้ใช้งานจำนวนมากขึ้นจะทำให้เราเตอร์มีภาระอย่างหนักในการประมวลผล (เช่น การคำนวณ หน่วยความจำ) และดูแลรักษาสถานะการสำรองทรัพยากรของแต่ละการเชื่อมต่อ

2.2.3 Differentiated Services

สถาปัตยกรรม Differentiated Service (DiffServ) [6] ได้ขยายงานที่ทำจาก IntServ และ RSVP ซึ่งมีเป้าหมายพื้นฐานที่เหมือนกันคือ คุณภาพในการให้บริการและการบริการที่แตกต่างกันในเครือข่ายอินเทอร์เน็ตโพรโตคอล ซึ่งสถาปัตยกรรมแบบ IntServ มีการรับประกันสำหรับแต่ละการเชื่อมต่อ ส่วนสถาปัตยกรรม DiffServ ได้เปลี่ยนจากการรับประกันสำหรับแต่ละการเชื่อมต่อไป

เป็นการรับประกันตามระดับของบริการ ซึ่งจะเรียกว่า Class of Service (CoS) โดยการให้บริการแบบนี้จะทำให้เครือข่ายสามารถรองรับผู้ใช้งานจำนวนมากได้และไม่ใช้สัญญาณในการดูแลรักษาสถานะการสำรองทรัพยากรที่เราเตอร์ตามเส้นทาง

ส่วนประกอบหลักที่สำคัญของสถาปัตยกรรม DiffServ คือ Service Level Agreement (SLA) ซึ่ง SLA คือ ข้อตกลงของบริการระหว่างลูกค้าและผู้ให้บริการโดยจะมีการเจาะจงรายละเอียดของทราฟฟิก และการตอบสนองของบริการในการส่งต่อแพ็กเกตที่ลูกค้าควรจะได้รับ ซึ่งลูกค้าอาจจะเป็นกลุ่มผู้ใช้งานหรือเป็นกลุ่ม DiffServ อื่น ๆ โดยผู้ให้บริการมั่นใจว่าทราฟฟิกของลูกค้าจะได้รับคุณภาพในการให้บริการตามสัญญา SLA ที่ได้ตกลงกันไว้ ฉะนั้นผู้ให้บริการของเครือข่ายจะต้องมีการกำหนดนโยบายในการให้บริการที่เหมาะสมและสามารถวัดประสิทธิภาพในการรับประกันคุณภาพในการให้บริการได้ ซึ่งเครือข่าย DiffServ จะต้องมีการจำแนกประเภทของแพ็กเกตข้อมูลจากลูกค้า จึงมีการปรับเปลี่ยนแพ็กเกตให้มีขอบเขตพิเศษขึ้นมาเพื่อใช้ในการเก็บรหัสจำนวนหนึ่งซึ่งเรียกว่า DS field (DiffServ field) โดยกำหนดให้ DS field จะต้องมีอยู่ในทุก ๆ แพ็กเกตซึ่งจะใช้เป็นตัวบ่งบอกถึงระดับในการให้บริการที่ได้กำหนดไว้ตามข้อตกลง

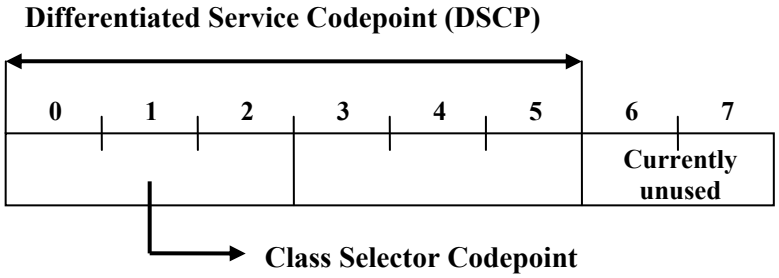


รูปที่ 2.5 DiffServ Model

2.2.3.1 DiffServ (DS) field

ส่วนสำคัญของสถาปัตยกรรมแบบ DiffServ คือ DS field ซึ่งจะใช้เป็นตัวบ่งบอกถึงระดับที่แพ็กเกตนั้น ๆ จะได้รับการบริการจากเราเตอร์ในเครือข่าย โดย DS field จะมีขนาด 8 บิต โดยจะใช้พื้นที่ของ TOS (Type of Services) ในส่วนหัวของแพ็กเกต โดยจะมีการจำแนกระดับในการให้บริการให้แก่ทุก ๆ แพ็กเกตที่เข้ามายังเครือข่ายที่เราเตอร์ตัวขอบขาเข้า (ingress edge router)

จะใช้พื้นที่ขนาด 6 บิต ของ DS field ในการกำหนดรหัสของระดับในการให้บริการ ซึ่งค่าของขอบเขตนี้จะเรียกว่า DiffServ Codepoint (DSCP) ตามรูปที่ 2.6 โดยแต่ละรหัสของ DSCP จะมีความหมายของ PHB (อธิบายในหัวข้อถัดไป) ที่แตกต่างกันไป ซึ่ง 2 บิต ที่เหลือยังไม่ใช้ในขณะนี้ โดยจะกำหนดเป็น Currently Unused (CU) field



รูปที่ 2.6 DiffServ (DS) codepoint

DSCP field จะมีขนาด 6 บิตโดยจะแบ่งเป็น 3 กลุ่ม ไปใช้ในการกำหนดรหัสและการจัดการร่วมกัน กลุ่มที่ 1 จะมีทั้งหมด 32 รหัส ซึ่งจะกำหนดให้เป็นมาตรฐาน กลุ่มที่ 2 จะมีทั้งหมด 16 รหัส ซึ่งใช้สำรองในการทดลอง และกลุ่มที่ 3 จะมี 16 รหัส ซึ่งจะเป็นค่าเริ่มต้นในการทดลองหรือใช้เป็นมาตรฐาน ตามตารางที่ 2.1

ตารางที่ 2.1 DiffServ (DS) codepoint [8]

Pool	Codepoint Space	Assignment Policy
1	XXXXX0	Standard Action
2	XXXX11	EXP/LU
3	XXXX01	EXP/LU or Standard Action

2.2.3.2 Per-Hop Behavior (PHB)

Per-Hop Behavior ถูกกำหนดเป็นคุณลักษณะที่แสดงถึงพฤติกรรมการให้บริการแพ็กเกตที่เข้ามายังเราเตอร์ภายในเครือข่ายแบบ DiffServ ซึ่งความสำคัญของ PHB คือการให้บริการที่สร้างขึ้นเพื่อให้มีลักษณะในการปฏิบัติต่อแพ็กเกตเหมือนกันสำหรับแพ็กเกตที่อยู่ในระดับเดียวกันในที่ทุก ๆ เราเตอร์

ในแต่ละโหนดจะมีการเลือกใช้ PHB ซึ่งจะได้มาจากรหัส DSCP ที่กำหนดมาในแต่ละแพ็กเกต โดยเราเตอร์ในเครือข่ายแบบ DiffServ จะต้องมีฟังก์ชันที่จะใช้ในการควบคุมแพ็กเกตในแต่ละระดับ โดยการกำหนดลักษณะของ PHB จะกำหนดในรูปของตัวแปรที่สามารถวัด

ประสิทธิภาพได้ เช่น การกำหนดความสำคัญของแพ็กเกตที่อยู่ในแถวคอย ความยาวของแถวคอย หรืออัลกอริทึมในการละทิ้งแพ็กเกต ในปัจจุบันได้มีการกำหนดลักษณะของ PHB เป็น 3 ประเภท คือ Class Selector (CS) PHB [8] Expedited Forwarding (EF) PHB [9] และ Assured Forwarding (AF) PHB [10]

2.2.3.2.1 Class Selector PHB

Class Selector PHB จะเป็นระดับในการให้บริการที่ไม่มีการรับประกัน ซึ่งจะกำหนดค่า DSCP เท่ากับ “XXX000” โดยปกติบริการประเภทนี้จะเหมาะกับแอพพลิเคชันประเภทเว็บบราวเซอร์

2.2.3.2.2 Expedited Forwarding PHB

Expedited Forwarding PHB จะเป็นลักษณะที่กำหนดให้ค่าสูญหายของแพ็กเกตต่ำ ค่าเวลาแฝงต่ำ และมีบริการรับรองแบนด์วิดธ์ตั้งแต่ต้นทางไปจนถึงปลายทาง ซึ่งการสูญหายของแพ็กเกต ค่าเวลาแฝง และการชนกันของแพ็กเกตทั้งหมดนี้จะเกิดขึ้นในช่วงของแถวคอยในขณะที่กำลังเดินทางในเครือข่าย โดยการที่จะทำให้มีค่าเวลาแฝงและการชนกันของแพ็กเกตที่ต่ำคือ การทำให้ทราฟฟิกที่ไหลผ่านเราเตอร์โดยไม่ต้องมีการรอคอยในคิวหรือมีเพียงเล็กน้อย ซึ่งการที่จะทำให้ไม่มีการอคอยจะต้องให้มีอัตราการมาถึงสูงสุดของแพ็กเกตมีค่าน้อยกว่าอัตราการไหลออกของแพ็กเกต โดยบริการประเภทนี้จะเหมาะสำหรับแอพพลิเคชันประเภท real-time โทรศัพท์ผ่านอินเทอร์เน็ตโพรโตคอล และการประชุมผ่านวิดิทัศน์ เป็นต้น และใช้รหัส DSCP เท่ากับ “101110”

2.2.3.2.3 Assured Forwarding PHB

จะเป็นบริการที่มีการบริการในระดับต่าง ๆ ไว้ด้วยกันโดยจะใช้ข้อตกลงเงื่อนไขของทราฟฟิก (Traffic Conditioning Agreement: TCA) เพื่อบ่งบอกระดับของการรับประกันในการให้บริการต่อแพ็กเกตที่แตกต่างกันในประเภทนี้ ซึ่งในปัจจุบันมีการแบ่งประเภทไว้ 4 ประเภท โดยแต่ละประเภทจะแบ่งระดับความสำคัญได้อีก 3 ระดับ ตามตารางที่ 2.2 แพ็กเกตในระดับที่หนึ่งจะได้รับบริการโดยทันทีและไม่ขึ้นกับคลาส AF ในระดับอื่น ๆ ซึ่งเราเตอร์ DiffServ จะต้องจัดสรรทรัพยากรให้แต่ละคลาส AF ตามระดับที่ได้ตกลงกันไว้ในการให้บริการต่อแพ็กเกต

ตารางที่ 2.2 Assured Forwarding (AF) Classes [10]

Drop Precedence	AF Class 1	AF Class 2	AF Class 3	AF Class4
Low	001010	010010	011010	100010
Medium	001100	010100	011100	100100
High	001110	010110	011110	100110

2.2.4 Integrated Services over Differentiated Services

ในสังคมสื่อสารมีความเห็นตรงกันที่จะต้องการดัดแปลงโมเดล IntServ หรือโมเดล DiffServ เพื่อให้สามารถรองรับบริการของเครือข่ายได้หลากหลายรูปแบบ ซึ่ง RFC 2998 [11] ได้แนะนำให้รวมสถาปัตยกรรมทั้งสองไว้ด้วยกันซึ่งก็คือ IntServ over DiffServ Framework โดยจะให้ IntServ ทำงานควบคุมกลไกที่เราเตอร์ขอบ (Edge router) ของเครือข่าย และ DiffServ จะทำงานควบคุมกลไกที่เราเตอร์หลัก (Core router)

IntServ over DiffServ Framework [12] [13] [14] สามารถจัดสรรการสำรองทรัพยากรได้ทั้งแบบคงที่ (statically) และแบบเปลี่ยนแปลงได้ (dynamically) วิธีการสำรองทรัพยากรแบบคงที่จะมีการกำหนดระดับในการให้บริการแก่ลูกค้าที่เราเตอร์ขอบขาเข้า และทรัพยากรของเครือข่ายจะถูกสำรองแบบคงที่ตามระดับในการให้บริการที่ได้ตกลงกันไว้ ที่แต่ละเราเตอร์ขอบจะประกอบไปด้วยตารางที่ใช้ในการสำรองแบนด์วิดธ์ให้แก่ประเภทของบริการต่าง ๆ โดยเราเตอร์ขอบจะสามารถทำการตัดสินใจอนุมัติการเชื่อมต่อได้ซึ่งเมื่อเปรียบเทียบกับสถาปัตยกรรมแบบ DiffServ แล้ว วิธีนี้จะรักษาคุณภาพในการให้บริการแก่ผู้ใช้งานโดยการใช้ส่วนควบคุมการอนุมัติการเชื่อมต่อที่ขอบของเครือข่าย ส่วนวิธีการสำรองทรัพยากรแบบเปลี่ยนแปลงได้จะมีการใช้สัญญาณ RSVP ในเริ่มแรกจะใช้ RSVP เพื่อสำรองแบนด์วิดธ์ไว้ก่อนแล้วจะมีการนำมาใช้กับส่วนควบคุมอนุมัติการเชื่อมต่อและการสำรองแบนด์วิดธ์ตั้งแต่ต้นทางไปสู่ปลายทาง โดยในเราเตอร์หลักจะยังคงมีการทำงานแบบ DiffServ เช่นกัน

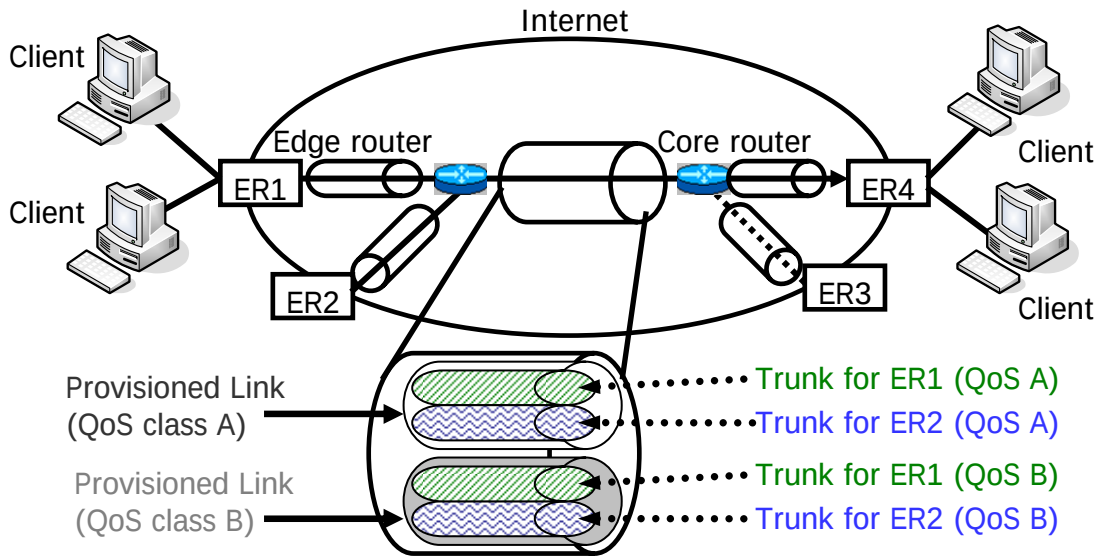
IntServ over DiffServ Framework ได้มีการพัฒนาอย่างแพร่หลายในขณะนี้ซึ่งได้พัฒนาไปในหลายทิศทาง โดยในวิทยานิพนธ์ฉบับนี้ได้เลือกเอาสถาปัตยกรรมแบบ IntServ over DiffServ ของ Mei Yang [15] มาเป็นเครือข่ายมาตรฐานที่ใช้ในการทดลองและนำมาเปรียบเทียบกับวิธีการที่นำเสนอ (อธิบายในบทที่ 3) ซึ่งเครือข่ายนี้จะมีหลักการทำงานดังต่อไปนี้

2.2.4.1 สถาปัตยกรรมพื้นฐาน

ในสถาปัตยกรรมนี้จะกำหนดให้มีเราเตอร์ 2 ประเภทในเครือข่ายคือ เราเตอร์ขอบ (Edge Router) และเราเตอร์หลัก (Core Router) โดยเราเตอร์ขอบจะทำหน้าที่เป็นส่วนควบคุมการตัดสินใจและทำการเชื่อมโยงแต่ละการเชื่อมต่อไปเป็นบริการตามประเภทที่แตกต่างกันและส่งแพ็กเกตไปยังเครือข่าย เราเตอร์หลักจะทำหน้าที่เป็นเราเตอร์แบบ DiffServ ซึ่งให้บริการตามประเภททราฟฟิกในระดับบริการที่แตกต่างกัน

สถาปัตยกรรมนี้ได้ให้บริการตามประเภทของบริการต่าง ๆ และมีส่วนควบคุมการอนุมัติที่ขอบของเครือข่ายโดยปราศจากสัญญาณที่ส่งไปเป็นช่วงที่เราเตอร์ มีการจัดการแบ่งแบนด์วิดธ์เป็นลำดับชั้น (Hierachical) โดยในแต่ละสายสื่อสาร physical link จะแบ่งได้เป็นหลาย Provision Link (PL) ซึ่งหนึ่ง PL จะใช้สำหรับทราฟฟิกประเภทเดียวเท่านั้น ในแต่ละ PL จะสามารถแบ่งได้เป็น

หลาย trunk ซึ่งหนึ่ง trunk จะใช้สำหรับหนึ่งเราเตอร์ขอบ โดย trunk จะรองรับการเชื่อมต่อจากเราเตอร์ขอบที่เป็นแหล่งกำเนิดเดียวกันโดยไม่สนใจจุดหมายปลายทาง การเชื่อมต่อผ่าน PL เดียวกันในเครือข่ายแต่มีแหล่งกำเนิดของเราเตอร์ขอบที่แตกต่างกันจะใช้ trunk ต่างกัน โดยจะมีลักษณะการทำงานตามรูปที่ 2.7



รูปที่ 2.7 IntServ over DiffServ Architecture

2.2.4.2 กระบวนการควบคุมการจัดตั้งการเชื่อมต่อ

เราเตอร์ขอบจะมีส่วนควบคุมการอนุมัติการเชื่อมต่อโดยไม่ต้องใช้สัญญาณในดูแลรักษาสถานะการเชื่อมต่อไปที่เราเตอร์หลัก โดยจะมีการสร้างตารางในการเก็บข้อมูลขึ้นมา 2 ตารางคือ ตาราง Virtual IP path (VIP) และตาราง trunk ซึ่ง VIP คือ เส้นทางของแต่ละการเชื่อมต่อจากเราเตอร์ขอบที่เป็นต้นทางไปจนถึงเราเตอร์ขอบที่เป็นจุดปลายทาง ในตาราง VIP จะบันทึกเส้นทางทั้งหมดที่มาจากเราเตอร์ขอบขาเข้าไปจนถึงเราเตอร์ขอบขาออกอื่น ๆ ทั้งหมด โดยในตารางจะสามารถกำหนดหรือสร้างกลไกการหาเส้นทางไว้ [16] ในตาราง VIP จะประกอบไปด้วย VIP ID ID ของเราเตอร์ปลายทาง ประเภทของทราฟฟิก รายการ ID ของ trunk ที่ประกอบเป็นเส้นทางของ VIP ตามตารางที่ 2.3

ตารางที่ 2.3 VIP table

VIP ID	Destination Edge Router	Traffic Class	List of Trunks

ในตาราง trunk จะบันทึก trunk ทั้งหมดที่เป็นของเราเตอร์ขอบ และข้อมูลการใช้งานแบนด์วิดธ์ ซึ่งภายในตาราง trunk จะประกอบไปด้วย Trunk ID, Provision Link ID จำนวนแบนด์-วิดธ์ที่ได้สำรองไว้ให้ trunk และจำนวนแบนด์วิดธ์ที่ใช้อยู่ในขณะนั้นของ trunk ตามตารางที่ 2.4

ตารางที่ 2.4 Trunk table

Trunk ID	Provisioned Link ID	Reserved Bandwidth	Bandwidth Being Used

เมื่อมีการร้องขอใหม่มาถึงที่เราเตอร์ขอบตัวแรกของเครือข่าย จะทำการตรวจสอบ VIP ที่เราเตอร์ขอบปลายทางว่ามีอยู่แล้วในตารางหรือไม่ (ได้มีการแยกประเภทของทราฟฟิกไว้เรียบร้อยแล้วก่อนทำการตรวจสอบเราเตอร์ขอบจากตาราง VIP) ต่อมาเราเตอร์ขอบขาเข้าจะทำการตรวจสอบแบนด์วิดธ์ที่เหลืออยู่ของแต่ละ trunk จากตาราง trunk ถ้ามีแบนด์วิดธ์เพียงพอที่จะรองรับการร้องขอก็จะทำการตอบยอมรับการร้องขอนั้นและทำการสำรองแบนด์วิดธ์ตั้งแต่ต้นทางไปจนถึงปลายทางสำหรับการร้องขอนั้น และจะอับเดทค่าในช่องแบนด์วิดธ์ที่กำลังใช้สำหรับทุก ๆ trunk แต่ในกรณีที่มีแบนด์วิดธ์ไม่เพียงพอที่จะรองรับการเชื่อมต่อใหม่เพิ่มได้ ก็จะทำการตอบปฏิเสธกลับไปยังผู้ใช้งาน และถ้าผู้ใช้งานยังคงต้องการที่จะติดต่อผ่านเครือข่ายก็จะต้องทำการสร้างแพ็กเกตร้องขอการเชื่อมต่อขึ้นมาใหม่และส่งมาร้องขอจัดตั้งการเชื่อมต่อที่เครือข่ายซ้ำ ๆ จนกว่าจะได้รับการตอบยอมรับ เมื่อการเชื่อมต่อได้สิ้นสุดลง แบนด์วิดธ์ที่เคยใช้อยู่ก็จะถูกปลดปล่อยกลับสู่เครือข่ายและก็จะทำการจัดสรรให้การเชื่อมต่ออื่นแล้วก็จะมีการปรับปรุงตาราง trunk ที่มีผลเปลี่ยนแปลง

2.2.4.3 กระบวนการทำงานของ Trunk

ในงานนี้ได้มีการทำการปรับเปลี่ยนทราฟฟิกที่กำหนดแบนด์วิดธ์ให้กับ trunk ในช่วงเวลาสั้น ๆ โดยเราเตอร์ขอบที่เป็นจุดกำเนิดจะสามารถขอแบนด์วิดธ์เพิ่มให้แก่ trunk หรือทำการปล่อยแบนด์วิดธ์ที่ไม่ได้ใช้ของ trunk ซึ่งแบนด์วิดธ์นี้จะทำการปรับเปลี่ยนภายใต้ provision link ซึ่งได้กำหนดเป็นคงที่ กระบวนการทำงานของ trunk จะเป็นดังต่อไปนี้

ขั้นที่ 1 การจัดการตาราง Provisioned Link

ตาราง Provisioned Link ได้มีการบำรุงประสิทธิภาพการใช้งานแบนด์วิดธ์ของ provision link ในเครือข่าย ซึ่งจะแสดงตามตารางที่ 2.5 ในตารางจะประกอบไปด้วย Provisioned Link ID, Physical Link ID จำนวนแบนด์วิดธ์ที่สำรองไว้ จำนวนแบนด์วิดธ์ที่ใช้ในขณะนั้น และประเภทของทราฟฟิกที่ provision link นั้นรองรับ

ตารางที่ 2.5 Provisioned Link table

Provisioned Link ID	Physical Link ID	Reserved Bandwidth	Bandwidth Being Used	Traffic Class

ขั้นที่ 2 การปล่อยแบนด์วิดธ์ที่ไม่ได้ใช้งานของ Trunk

เราเตอร์ขอบที่เป็นจุดกำเนิดจะมีการตรวจสอบ trunk เป็นช่วง ๆ ตามตารางที่ 2.4 และมีการหาค่าประสิทธิภาพการใช้งานของ trunk ถ้าประสิทธิภาพการใช้แบนด์วิดธ์ของ trunk ต่ำกว่าค่าที่กำหนดเอาไว้ (lower threshold) เราเตอร์ขอบจะทำการปรับปรุงตาราง trunk และส่ง control message เพื่อทำการปรับปรุงตาราง provisioned link

ขั้นที่ 3 ความต้องการแบนด์วิดธ์เพิ่มของ Trunk

เมื่อมีการร้องขอมาถึงที่เราเตอร์ขอบ จะมีการกำหนด VIP ให้แก่การเชื่อมต่อจากตามตารางที่ 2.3 และจะมีการตรวจสอบตาราง trunk ตามตารางที่ 2.4 ถ้าพบว่ามีการใช้งานแบนด์วิดธ์ของแต่ละ trunk เกินกว่าค่าสูงสุดที่กำหนดไว้ (upper threshold) เราเตอร์ขอบจะส่ง control message เพื่อขอแบนด์วิดธ์เพิ่มจาก provision link

ตามที่ได้อธิบายในขั้นที่ 2 การปล่อยแบนด์วิดธ์ของ trunk ที่ไม่ใช้แล้วและในขั้นที่ 3 การขอแบนด์วิดธ์เพิ่ม จะมีการใช้ control message 3 ประเภทคือ Trunk Reconfiguration Request Message (TRRequest), Trunk Reconfiguration Release Message (TRRelease) และ Trunk Reconfiguration Confirm Message (TRConfirm) ซึ่งทั้ง 3 ประเภทจะมีรูปแบบของแพ็กเกตที่เหมือนกันตามรูปที่ 2.8 ซึ่งรูปแบบของแพ็กเกตจะประกอบไปด้วย ประเภทของแพ็กเกต (เช่น TRRequest TRRelease และ TRConfirm) ID ของเราเตอร์ขอบต้นกำเนิด ID ของเราเตอร์ขอบปลายทาง VIP ID ความยาวของ control message และจะมีชุดข้อมูลของ trunk ID จำนวนแบนด์-วิดธ์ที่จะร้องขอเพิ่มหรือจำนวนแบนด์วิดธ์ที่จะปล่อย

Message Type	Source Edge Router ID	Destination Edge Router ID	VIP ID	Length	(Trunk i, Bdw_requested / Bdw_released)	(Trunk j, Bdw_requested / Bdw_released)

รูปที่ 2.8 TRRequest / TRRelease / TRConfirm Message Format

ซึ่ง TRRequest Message จะมีอัลกอริทึมในการทำงานตามรูปที่ 2.9 และ TRRelease จะมีอัลกอริทึมการทำงานตามรูปที่ 2.10

1. Genarate a TRConfirm message.
2. Copy each (trunk I, Bdw_requested or Bdw_released) pair tuple in received TRRequest to the TRConfirm.
3. For trunk I specified in the TRRequest,
3.1 Examine if provisioned link PL_i that trunk I belongs to is in X's provisioned link table;
If yes, then do 3.2
Else set the Bdw_requested field of trunk I in TRConfirm message to 0 (0 indicates that X cannot make a decision for this request.)
3.2 Examine if Provisioned link PL_i has sufficient bandwidth available for reconfiguration;
If (the reserved bandwidth on PL_i – bandwidth being used on PL_i) ≥ Bdw_requested for trunk i
Then increase the bandwidth for trunk I, update the provisioned link table by
Bandwidth being used on PL_i = bandwidth being used on PL_i + Bdw_requested for trunk i;
And set the bandwidth requested field for trunk i in TRConfirm message to Bdw_requested (i.e. requested bandwidth has been assign).
Else set the bandwidth requested field for trunk i in TRConfirm message to – 1 (i.e. the bandwidth reconfiguration request for the trunk has failed, and no bandwidth has been assign.)
4. Send the TRConfirm message to the edge router which initiated the TRRequest message.

รูปที่ 2.9 TRRequest Message Process Algorithm

For trunk i specified in the TRRelease message,
1. Exanmin if provisioned link PL_i that trunk I belongs to is in X's provisioned link table.
2. If yes, update the provisioned link table by
bandwidth being used on PL_i = bandwidth being used on PL_i – Bdw_released for trunk i.
3. If no, ignore the TRRelease message.

รูปที่ 2.10 TRRelease Message Process Algorithm