การออกแบบกรอบการทำงานและสถาปัตยกรรม ของระบบบริการแสดงสถานะแบบรวมศูนย์

Design Framework and Architecture of Unified Presence Service

พีระพงษ์ บุญเลิศ

Peeraphong Boonlert

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา
วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์
มหาวิทยาลัยสงขลานครินทร์

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Computer Engineering
Prince of Songkla University
2553

ลิขสิทธิ์ของมหาวิทยาลัยสงขลานครินทร์

การออกแบบกรอบการทำงานและสถาปัตยกรรม ของระบบบริการแสดงสถานะแบบรวมศูนย์

Design Framework and Architecture of Unified Presence Service

พีระพงษ์ บุญเลิศ

Peeraphong Boonlert

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา
วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์
มหาวิทยาลัยสงขลานครินทร์

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Computer Engineering
Prince of Songkla University
2553

ลิขสิทธิ์ของมหาวิทยาลัยสงขลานครินทร์

ชื่อวิทยานิพนธ์ การออกแบบกรอบการทำงานและสถาปัตยกรรมของระบบบริการแสดงสถานะแบบรวมศูนย์

ผู้เขียน นายพีระพงษ์ บุญเลิศ

สาขาวิชา วิศวกรรมคอมพิวเตอร์

อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

.....
(รองศาสตราจารย์ ดร.สินชัย กมลภิวงศ์)

อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม

.....
(รองศาสตราจารย์ทศพร กมลภิวงศ์)

คณะกรรมการสอบ

.....ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.นิษฐิดา เอลซ์)

.....กรรมการ
(รองศาสตราจารย์ ดร.สินชัย กมลภิวงศ์)

.....กรรมการ
(รองศาสตราจารย์ทศพร กมลภิวงศ์)

.....กรรมการ
(ดร.เฉลิมพล ชาญศรีภิญโญ)

บัณฑิตวิทยาลัย มหาวิทยาลัยสงขลานครินทร์ อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษา ตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

.....
(รองศาสตราจารย์ ดร.เกริกชัย ทองหนู)
คณบดีบัณฑิตวิทยาลัย

ชื่อวิทยานิพนธ์	การออกแบบกรอบการทำงานและสถาปัตยกรรมของระบบบริการแสดงสถานะแบบรวมศูนย์
ผู้เขียน	นายพีระพงษ์ บุญเลิศ
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ปีการศึกษา	2552

บทคัดย่อ

ระบบการสื่อสารแบบรวมศูนย์ (Unified Communication and Collaboration: UCC) เป็นการรวบรวมบริการการสื่อสารในทุกรูปแบบเข้าไว้ด้วยกัน และทำให้ไอเอ็มเอส (IP Multimedia Subsystem) ถูกคาดว่าจะเป็นแกนหลักในการทำงานของระบบการสื่อสารแบบรวมศูนย์เพื่อให้บริการในทั้งโลกของโทรศัพท์เคลื่อนที่และโลกของอินเตอร์เน็ต ดังนั้นทำให้ระบบการสื่อสารแบบรวมศูนย์เกี่ยวข้องกับไอเอ็มเอส เพื่อที่จะให้บริการในอนาคตในทั้งสองโลกของการสื่อสาร และองค์ประกอบที่สำคัญในการพัฒนาระบบการสื่อสารแบบรวมศูนย์ คือ การบริการแสดงสถานะ (Presence Service) โดยในส่วนของการบริการแสดงสถานะ จะให้บริการการแจ้งให้ทราบถึงข้อมูลสถานะ (Presence Information) ของผู้ใช้จากอุปกรณ์ต่าง ๆ ในระบบไปยังผู้ที่ต้องการรับทราบข้อมูลดังกล่าว แต่อย่างไรก็ตามการบริการแสดงสถานะในปัจจุบันยังมีความสามารถไม่เพียงพอกับการตอบสนองที่จะทำให้ระบบระบบการสื่อสารแบบรวมศูนย์ ตัวอย่างเช่น ถ้าผู้ใช้มีแหล่งข้อมูลจำนวนมากที่ส่งมาจากการบริการการสื่อสารต่าง ๆ ในระบบ การบริการแสดงสถานะยังไม่มีวิธีการจัดสรรข้อมูลสถานะจากแหล่งต่าง ๆ ที่อาจจะเกิดการขัดแย้งของข้อมูลหรือมีรูปแบบของโปรโตคอลที่แตกต่างกัน นอกเหนือจากนี้ระบบยังจำเป็นต้องรองรับจำนวนบัญชีผู้ใช้ (User Account) ของผู้ใช้จากช่องทางการสื่อสารต่าง ๆ ซึ่งกรณีตัวอย่างนี้ถูกเรียกว่าแมสซีฟพีไอ (Massive PI) ซึ่งเป็นประเด็นของหัวข้อวิจัยนี้ โดยในวิทยานิพนธ์ฉบับนี้ได้เสนอ การแสดงสถานะแบบรวมศูนย์ (Unified Presence Service: UPS) เพื่อจัดการข้อมูลแสดงสถานะที่อยู่ในรูปแบบแมสซีฟพีไอในระบบระบบการสื่อสารแบบรวมศูนย์ โดยการแสดงสถานะแบบรวมศูนย์จะมีการให้บริการใน 3 ส่วนหลัก คือ การจัดการบัญชีรายชื่อและข้อมูลแสดงสถานะ (Large-Scale of PI), การรวบรวมข้อมูลแสดงสถานะ (PI Aggregation) และ การจำกัดสิทธิ์และการแสดงผลอย่างชาญฉลาด (Smart Distribution and Policy Control) ซึ่งกรอบการทำงาน (Framework) และสถาปัตยกรรม (Architecture) งานของระบบบริการแสดงสถานะแบบรวมศูนย์ได้แสดงรายละเอียดไว้ในวิทยานิพนธ์นี้ และได้แสดงคุณประโยชน์และข้อได้เปรียบของระบบการแสดงสถานะแบบรวมศูนย์ ซึ่งระบบอื่นไม่สามารถที่จะทำได้

คำสำคัญ: ระบบการสื่อสารแบบรวมศูนย์, ข้อมูลแสดงสถานะ, ระบบบัญชี, การบริการแสดงสถานะแบบรวมศูนย์, ไอเอ็มเอส

Thesis Title	Design Framework and Architecture of Unified Presence Service
Author	Mr. Peeraphong Boonlert
Major Program	Computer Engineering
Academic Year	2009

Abstract

Unified Communication and Collaboration (UCC) system is the converging service of all communications. IMS (IP Multimedia Subsystem) is expected to be a core service for both Internet and mobile phones. Hence, UCC certainly involves with IMS to provide a future service to both worlds. Presence service is a crucial component in UCC system. It provides notifications and presence information of users and any devices to a particular user who requires this service. However, a traditional presence service system is inadequate to satisfy rich features services of UCC. For example, if a user has a large number of presence information sent by many services in system, how a system gives all presence information to a user appropriately. This scenario is so called 'Massive PI'. There are several challenging issues on this topic. In this paper, we have proposed 'Unified Contact Service' (or UCS) to deliver a massive PI in a UCC environment. Unified contact service provides contact grouping, unified presence, and smart service selection. System framework and architecture of USC is presented. In this design framework, all related signalings and components are illustrated, then descriptions and functions are presented. We have shown some sample scenarios to demonstrate its benefits and advantages which cannot be delivered by those previous works. We believe that this service framework is significant for future services in a convergence world

Keywords: UCC, PI, unified contact, unified presence service, IMS

กิตติกรรมประกาศ

ขอขอบพระคุณรองศาสตราจารย์ ดร.สินชัย กมลภิวงศ์ อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก รองศาสตราจารย์ทศพร กมลภิวงศ์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม ที่ได้ให้คำปรึกษา ชี้แนะแนวทาง และให้ความรู้ในด้านต่าง ๆ รวมถึงการให้การสนับสนุนในเรื่องอุปกรณ์ในการทำวิจัย ตลอดจนช่วยตรวจและแก้ไขวิทยานิพนธ์ให้เป็นไปอย่างสมบูรณ์

ขอขอบพระคุณผู้ช่วยศาสตราจารย์ ดร.นิษฐิดา เอลซ์ ประธานกรรมการสอบวิทยานิพนธ์ และดร.เฉลิมพล ชาญศรีภิญโญ กรรมการสอบวิทยานิพนธ์ ที่ได้ให้คำแนะนำในการปรับปรุงวิทยานิพนธ์ให้สมบูรณ์ยิ่งขึ้น

ขอขอบพระคุณอาจารย์สุธน แซ่ว่องที่ได้ให้คำปรึกษา ชี้แนะแนวทาง และให้ความความรู้ รวมถึงข้อเสนอแนะต่าง ๆ ในการทำวิจัย ตลอดจนช่วยตรวจและแก้ไขวิทยานิพนธ์ให้เป็นไปอย่างสมบูรณ์

ขอขอบพระคุณคณาจารย์ และบุคลากรในภาควิชาวิศวกรรมคอมพิวเตอร์ทุกท่านที่ให้คำปรึกษาและความช่วยเหลือในระหว่างการทำวิทยานิพนธ์

ขอขอบพระคุณคณะวิศวกรรมศาสตร์ที่ได้สนับสนุนทุนการศึกษาในระดับปริญญาโทตลอดระยะเวลา 2 ปีการศึกษา

ขอขอบคุณเพื่อนนักศึกษาปริญญาโทสาขาวิชาวิศวกรรมคอมพิวเตอร์ทุกท่านที่ได้ให้คำแนะนำและเป็นกำลังใจมาโดยตลอด

พีระพงษ์ บุญเลิศ

สารบัญ

กิตติกรรมประกาศ 5
สารบัญ 6
สารบัญรูปภาพ 9
สารบัญตาราง 12
สัญลักษณ์คำย่อและตัวย่อ 13
บทที่ 1 บทนำ 1
1.1 ความสำคัญและที่มาของวิทยานิพนธ์ 1
1.2 วัตถุประสงค์ 3
1.3 ขอบเขตของการวิจัย 4
1.4 ขั้นตอนและวิธีการดำเนินการวิจัย 4
1.5 ประโยชน์ที่คาดว่าจะได้รับ 4
บทที่ 2 ทฤษฎีและหลักการ 5
2.1 Session Initiation Protocol (SIP) 5
2.1.1 คุณสมบัติของโปรโตคอล SIP 5
2.1.2 องค์ประกอบสำคัญในระบบ SIP (SIP Components) 7
2.2 คุณลักษณะของระบบการสื่อสารแบบรวมศูนย์ 8
2.3 การบริการแสดงสถานะในระบบอินเตอร์เน็ตปัจจุบัน 13
2.3.1 กรอบการทำงานและสถาปัตยกรรมของ Jabber/XMPP 13
2.3.2 กรอบการทำงานและสถาปัตยกรรมของ SIMPLE/SIP 15
2.3.3 การเปรียบเทียบระหว่างสถาปัตยกรรม SIP/SIMPLE และ XMPP/Jabber 18
2.4 IP Multimedia Subsystem (IMS) 19
2.4.1 ที่มาของระบบ IMS 19
2.4.2 ภาพรวมของสถาปัตยกรรม IMS 21
2.4.3 สถาปัตยกรรมระบบแสดงสถานะของ IMS 24
2.4.4 การทำงานร่วมกับสถาปัตยกรรมของ IMS 26

บทที่ 3 ทบทวนวรรณกรรม 28
3.1 การรวบรวมข้อมูลแสดงสถานะจากช่องทางการสื่อสารต่าง ๆ ให้เป็นข้อมูลแสดงสถานะแบบรวมศูนย์ (Presence Information Aggregation) 28
3.1.1 การรวบรวมข้อมูลแสดงสถานะที่เครื่องลูกข่าย 28
3.1.2 การรวบรวมข้อมูลแสดงสถานะที่เครื่องแม่ข่ายตามกฎที่ผู้ใช้กำหนดขึ้น 31
3.1.3 การรวบรวมข้อมูลแสดงสถานะที่เคลื่อนแม่ข่ายโดยใช้ตารางรวบรวม 33
3.1.4 สรุปการรวบรวมข้อมูลแสดงสถานะ (Aggregation) 39
3.1.5 การควบคุมการเข้าถึงข้อมูลแสดงสถานะ (Privacy Control) 40
3.1.6 สรุปการควบคุมการเข้าถึงข้อมูล 44
บทที่ 4 การออกแบบกรอบการทำงานและสถาปัตยกรรม ของระบบบริการแสดงสถานะแบบรวมศูนย์ 45
4.1 การวิเคราะห์ลักษณะของการบริการแสดงสถานะในระบบการสื่อสารแบบรวมศูนย์ .. 45
4.1.1 Massive Presence Information (Massive PI) 45
4.2 กรอบการทำงานของระบบบริการแสดงสถานะแบบรวมศูนย์ (Unified Presence Service: UPS) 47
4.2.1 ฟังก์ชัน Large-Scale of Presence Information 48
4.2.2 ฟังก์ชัน Presence Information Aggregation 49
4.2.3 ฟังก์ชัน Smart Distribution and Policy Control 51
4.3 สถาปัตยกรรมของระบบแสดงสถานะแบบรวมศูนย์ 55
4.3.1 แนวคิดในการออกแบบสถาปัตยกรรม 55
4.3.2 ภาพรวมของสถาปัตยกรรมของระบบบริการแสดงสถานะแบบรวมศูนย์ 56
4.3.3 Unified Server 58
4.3.4 Resource List Server (RLS) 62
4.3.5 Contact List and Trust Server 64
4.3.6 Presence Server 66
4.4 การทำงานร่วมกับสถาปัตยกรรมของ IMS 69
4.5 กระบวนการพื้นฐานที่เกี่ยวข้องกับระบบแสดงสถานะแบบรวมศูนย์ 70

4.5.1 เพิ่มและแก้ไขบัญชีรายชื่อ 70
4.5.2 กระบวนการ Publication, Subscription and Notification 73
4.5.3 การ Subscription and Notification ระหว่าง Internet Domain และ IMS Domain 75
4.6 สรุป 76
บทที่ 5 การพัฒนาและทดสอบระบบบริการแสดงสถานะแบบรวมศูนย์ 78
5.1 ขอบเขตและข้อจำกัดในการพัฒนาระบบบริการแสดงสถานะแบบรวมศูนย์ 78
5.2 การพัฒนาระบบบริการแสดงสถานะแบบรวมศูนย์ 79
5.2.1 ส่วนระบบบริการแสดงสถานะแบบรวมศูนย์ภายใน 79
5.2.2 ส่วนการเชื่อมต่อกับระบบ IMS 84
5.2.3 การพัฒนาการเชื่อมต่อระหว่างระบบบริการแสดงสถานะแบบรวมศูนย์กับระบบ IMS 85
5.3 การทดสอบระบบบริการแสดงสถานะแบบรวมศูนย์ 88
5.4 ผลการทดสอบ 92
5.5 สรุป 99
บทที่ 6 สรุปผลการดำเนินงาน 100
6.1 สรุปผลการวิจัย 100
6.1.1 การออกแบบกรอบการทำงานของระบบการแสดงสถานะแบบรวมศูนย์ 100
6.1.2 การออกแบบสถาปัตยกรรมของระบบการแสดงสถานะแบบรวมศูนย์ 101
6.1.3 การพัฒนาระบบบริการแสดงสถานะแบบรวมศูนย์เพื่อทดสอบการทำงาน 103
6.2 ข้อเสนอแนะ 103
6.2.1 การออกแบบข้อมูลบริบทและ Presence Ontology Model 103
6.2.2 การออกแบบสถาปัตยกรรมให้ระบบบริการแสดงสถานะแบบรวมศูนย์อยู่บนพื้นฐานของระบบ IMS 104
เอกสารอ้างอิง 105
ประวัติผู้เขียน 108

สารบัญรูปภาพ

รูปที่ 2-1 องค์ประกอบภายในระบบ SIP 7
รูปที่ 2-2 สถาปัตยกรรมของ Jabber 14
รูปที่ 2-3 องค์ประกอบพื้นฐานของการบริการแสดงสถานะของ SIP 16
รูปที่ 2-4 A Presence Data Model 17
รูปที่ 2-5 Presence Cofiguration 17
รูปที่ 2-6 Configuration of IM Subsystem Entities 21
รูปที่ 2-7 3GPP IMS Architecture Overview 22
รูปที่ 2-8 SIP based Presence Architecture in IMS 25
รูปที่ 3-1 สถาปัตยกรรมของ Universal Instant Messenger 29
รูปที่ 3-2 การทำงานภายในของ Aggregator และแสดงการการเปลี่ยนจากโปรโตคอล MSN ไปเป็นโปรโตคอล AOL 30
รูปที่ 3-3 รูปแบบ XML ของกฎที่ใช้ในการรวบรวมข้อมูลของนาย A จากกรณีตัวอย่าง 32
รูปที่ 3-4 สถาปัตยกรรมของระบบ User Presence Aggregator at Server 34
รูปที่ 3-5 แสดง Status Container ว่า มีกระบวนการ Aggregation ภายใน 35
รูปที่ 3-6 มุมมองระดับทั่วไปของระบบ R-U-In? Active-Oriented Social Networking 42
รูปที่ 3-7 Contextual Modeling ของ R-U-In? 43
รูปที่ 4-1แสดงถึงฟังก์ชันภายในของ Unified Presence Service เพื่อที่จะใช้จัดการกับปัญหา Massive PI 48
รูปที่ 4-2 แสดงหลักการของ PI Aggregation 49
รูปที่ 4-3 การแก้ไขปัญหาขัดแย้งของข้อมูลแสดงสถานะจากสองแหล่ง 50
รูปที่ 4-4 ตัวอย่างการแสดงสถานะที่ผู้ใช้ B และ C ได้รับแตกต่างกัน แม้ว่าข้อมูลแสดงสถานะจะมาจากแหล่งเดียวกัน 51
รูปที่ 4-5 แสดงความสัมพันธ์ตัวอย่างระหว่าง ผู้ใช้ A ผู้ใช้ B และ ผู้ใช้ C 52
รูปที่ 4-6 Amoeba Model ของผู้ใช้ A 53
รูปที่ 4-7 ความสัมพันธ์ระหว่าง Amoeba Model ของผู้ใช้ในระบบ 53
รูปที่ 4-8 Presence Ontology Model สำหรับฟังก์ชัน Smart Distribution and Policy Control 54
รูปที่ 4-9 ภาพรวมสถาปัตยกรรมของระบบการบริการแสดงสถานะแบบรวมศูนย์ 56
รูปที่ 4-10 การเชื่อมต่อกับองค์ประกอบต่างๆ ของ Unified Server 58
รูปที่ 4-11 การเชื่อมต่อกับองค์ประกอบต่างๆ ของ RLS 62

รูปที่ 4-12 RLS ดึงข้อมูลจาก Contact List and Trust Server 63
รูปที่ 4-13 การเชื่อมต่อกับองค์ประกอบต่างๆ ของ Contact List and Trust Server 65
รูปที่ 4-14 การเชื่อมต่อกับองค์ประกอบต่างๆ ของ Presence Server 66
รูปที่ 4-15 การทำงานร่วมกับสถาปัตยกรรม IMS 69
รูปที่ 4-16 ผู้ใช้ A ค้นหา Primary Account ของผู้ใช้ B 70
รูปที่ 4-17 การรับ Unified Contact ของผู้ใช้ A 72
รูปที่ 4-18 แสดงลำดับของการ Publication, Subscription and Notification 73
รูปที่ 4-19 รูปแบบของความสัมพันธ์ของ Bob, Claire และ Alice 74
รูปที่ 4-20 ลำดับสัญญาณของการ Subscription and Notification ระหว่าง Internet Domain และ IMS Domain 75
รูปที่ 5-1 ภาพรวมของระบบบริการแสดงสถานะแบบรวมศูนย์ที่ได้พัฒนาขึ้น 79
รูปที่ 5-2 ระบบทดสอบของบริการแสดงสถานะแบบรวมศูนย์ภายใน 80
รูปที่ 5-3 ลำดับการทำงานของ SIP Communicator 81
รูปที่ 5-4 Plug-in สำหรับการรวบรวมโปรโตคอลอื่นๆ 81
รูปที่ 5-5 ลำดับสัญญาณของ RLS 82
รูปที่ 5-6 ระบบ IMS 84
รูปที่ 5-7 การตั้งค่า Application Server Service Profile 86
รูปที่ 5-8 การตั้งค่า Trigger Point ของการทำ IFC 87
รูปที่ 5-9 การทดสอบส่วนที่ 1 PI Aggregation 88
รูปที่ 5-10 การทดสอบส่วนที่ 2 การทดสอบการทำงานของ RLS 89
รูปที่ 5-11 การทดสอบส่วนที่ 3 การ Subscription และ Publication ระหว่าง IMS และ SIP Network 90
รูปที่ 5-12 IMS Terminal แสดงสถานะของตนเองไปยัง SIP Network 91
รูปที่ 5-13 การร้องขอข้อมูลแสดงสถานะจาก IMS ไปยัง SIP Network 91
รูปที่ 5-14 ผลการทดสอบจากกรณีย่อยที่ 1 92
รูปที่ 5-15 ผลการทดสอบจากกรณีย่อยที่ 2 93
รูปที่ 5-16 ผลการทดลองจากกรณีย่อยที่ 3 94
รูปที่ 5-17 สัญญาณ SUBSCRIBE จากเครื่องลูกข่ายของ Oam 94
รูปที่ 5-18 สัญญาณ NOTIFY สัญญาณแรกจาก RLS ที่ส่งไปยังเครื่องลูกข่ายของ Oam 95
รูปที่ 5-19 บัญชีรายชื่อของ Oam จากส่วน Body ของสัญญาณ NOTIFY สัญญาณแรก 95

รูปที่ 5-20 SIP Communicator หลังจากรับ NOTIFY สัญญาณแรก 96
รูปที่ 5-21 SIP Communicator เมื่อรับ NOTIFY ที่มีข้อมูลการแสดงสถานะ..... 96
รูปที่ 5-22 การ Publication ในฝั่ง IMS Network 97
รูปที่ 5-23 การ Subscription และ Notification ในฝั่ง IMS Network..... 97
รูปที่ 5-24 การแสดงผลของเครื่องลูกข่ายของ Alice (Monster) 98
รูปที่ 6-1แสดงถึงฟังก์ชันภายในของ Unified Presence Service เพื่อที่จะใช้จัดการกับปัญหา Massive PI 100
รูปที่ 6-2 ภาพรวมของสถาปัตยกรรมของระบบแสดงสถานะแบบรวมศูนย์..... 101
รูปที่ 6-3 ภาพรวมของสถาปัตยกรรมแบบรวมศูนย์เมื่อรวมกับระบบโทรศัพท์เคลื่อนที่ยุคที่สาม 102

สารบัญตาราง

ตารางที่ 2-1 เปอร์เซ็นต์ข้อมูลการใช้งานบริการในระบบการสื่อสารแบบรวมศูนย์..... 12
ตารางที่ 2-2 เปรียบเทียบการเปลี่ยนระหว่าง XMPP ไปยัง SIP..... 15
ตารางที่ 2-3 การเปรียบเทียบระหว่าง SIP/SIMPLE และ XMPP/Jabber..... 19
ตารางที่ 2-4 IBCF Related Specification 27
ตารางที่ 3-1 แบ่งประเภทรายละเอียดของ Presence State ของระบบการรวบรวมของเอกสาร [6] 35
ตารางที่ 3-2 แสดงรายละเอียดของ User State โดยกำหนดเป็นค่าตัวเลข 36
ตารางที่ 3-3 แสดงรายละเอียดของ Phone State โดยกำหนดเป็นค่าตัวเลข 36
ตารางที่ 3-4 แสดงรายละเอียดของ Calendar State โดยกำหนดเป็นค่าตัวเลข 36
ตารางที่ 3-5 แสดงรายละเอียดของ Aggregated State โดยกำหนดเป็นค่าตัวเลขเป็นช่วง ๆ 37
ตารางที่ 4-1 ตารางเปรียบเทียบความสามารถของระบบ UPS กับระบบบริการแสดงสถานะในปัจจุบัน 77
ตารางที่ 5-1 ตารางข้อมูลแสดงสถานะของ Alice และผลที่คาดว่า Bob จะได้รับ 89

สัญลักษณ์คำย่อและตัวย่อ

HTTP	Hypertext Transfer Protocol
HSS	Home Subscriber Server
IETF	Internet Engineering Task Force
IMS	IP Multimedia Subsystem
PSTN	Public Switching Telephone Network
RFC	Request for Comments
RLS	Resource List Server
RTCP	Real-time Control Protocol
RTP	Real-time Transport Protocol
SDP	Session Description Protocol
SIMPLE	SIP for Instant Messaging and Presence Leveraging Extensions
SIP	Session Initiation Protocol
SS7	Signaling System 7
TCP	Transmission Control Protocol
UA	User Agent
UAC	User Agent Client
UAS	User Agent Server
UDP	User Datagram Protocol
UPS	Unified Presence Service
URI	Uniform Resource Identifier
XML	Extensible Markup Language
XMPP	Extensible Messaging and Presence Protocol

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มาของวิทยานิพนธ์

ระบบการสื่อสารแบบรวมศูนย์ (Unified Communication and Collaboration: UCC) เป็นระบบที่รวมบริการในการสื่อสารรูปแบบต่าง ๆ เข้าไว้ด้วยกัน เช่น VoIP, Voice and Video Conference, Instant Messaging, Voice Mail, E-Mail, SMS และ MMS เป็นต้น ทั้งนี้ ระบบการสื่อสารแบบรวมศูนย์ทำให้ผู้ใช้งานระบบสามารถทำการติดต่อสื่อสารได้ ทุกที่ ทุกเวลา ไม่ว่าอุปกรณ์ในการใช้งานจะแตกต่างกัน

โดยเครื่องมือสำคัญที่ทำให้การสื่อสารต่าง ๆ ในระบบการสื่อสารแบบรวมศูนย์สามารถหลอมรวมเข้าด้วยกันและให้บริการได้อย่างราบรื่นโดยไม่มีรอยต่อ คือ การบริการแสดงสถานะ (Presence Service) [1] [2] เนื่องจากการบริการแสดงสถานะจะมีการแจ้งข้อมูลสถานะของผู้ใช้ (Presence Information) เพื่อแสดงถึงความสามารถในการสื่อสารของผู้ใช้ที่มีในระบบว่าผู้ใช้มีความพร้อมที่จะรับการติดต่อการสื่อสารหรือไม่ ตัวอย่างเช่น ผู้ใช้ A ต้องการติดต่อไปผู้ใช้ B โดยผู้ใช้ B มีช่องทางในการสื่อสารเป็น VoIP, Instant Messaging และ Video Conference การบริการแสดงสถานะจะให้ข้อมูลสถานะของแต่ละช่องทางเป็น VoIP พร้อมรับการติดต่อ Instant Messaging ไม่ว่างและ Video Conference กำลังถูกใช้งาน ส่งข้อมูลดังกล่าวไปยังผู้ใช้ A เพื่อตัดสินใจในการเลือกช่องทางการติดต่อ จากตัวอย่างดังกล่าวเมื่อมองในมุมมองของผู้ใช้จะทำให้ผู้ใช้งานเสมือนว่าเห็นการสื่อสารจากอุปกรณ์ต่าง ๆ ถูกรวมมาไว้เป็นระบบเดียว

การบริการแสดงสถานะของระบบการสื่อสารแบบรวมศูนย์ จำเป็นต้องมีการจัดการข้อมูลจากช่องทางการสื่อสารต่าง ๆ เพื่อให้ข้อมูลแสดงสถานะเป็นข้อมูลแบบรวมศูนย์ ซึ่งมีรูปแบบการทำงานที่แตกต่างจากการบริการแสดงสถานะแบบทั่วไป (MSN, GTalk หรือ YaHoo Messenger) เนื่องจากช่องทางการสื่อสารและอุปกรณ์การสื่อสารต่าง ๆ จะให้ข้อมูลแสดงสถานะออกมาจำนวนมาก ซึ่งจะทำให้เกิดปัญหาความแตกต่างของโปรโตคอล เช่น ข้อมูลแสดงสถานะจาก GTalk ที่ทำงานบนโปรโตคอล XMPP ไม่สามารถแสดงผลบน MSN ได้ เนื่องจาก MSN มีรูปแบบโปรโตคอลเฉพาะตัว

นอกจากนี้ระบบการสื่อสารแบบรวมศูนย์มีข้อมูลแสดงสถานะที่มากกว่าผู้ใช้ออนไลน์หรือผู้ใช้ออฟไลน์ เช่น ข้อมูลสถานที่อยู่ อารมณ์หรือกิจกรรม เป็นต้น ซึ่งทำให้เกิดโอกาสที่ข้อมูลแสดงสถานะของผู้ใช้จากอุปกรณ์ต่าง ๆ มีความขัดแย้งกัน ตัวอย่างเช่น คอมพิวเตอร์ส่วนบุคคลแสดงสถานะของผู้ใช้คนหนึ่งว่าออนไลน์ที่สำนักงาน แต่ในขณะเดียวกันโทรศัพท์เคลื่อนที่กลับแสดงสถานะของผู้ใช้คนนั้นว่า เขากำลังขับรถ เป็นต้น ซึ่งจากตัวอย่างนี้แสดงให้เห็นว่าการจัดการระบบแสดงสถานะที่ไม่ดีพอจะส่งผลกระทบโดยตรงกับผู้ใช้งาน

นอกจากการจัดการข้อมูลแสดงสถานะที่ดีแล้ว ระบบการสื่อสารแบบรวมศูนย์จำเป็นจะต้องจัดการกับบัญชีรายชื่อ (User Account) ของผู้ใช้ที่ได้จากช่องทางการสื่อสารต่าง ๆ ตัวอย่างเช่น ผู้ใช้ A และผู้ใช้ B ทั้งคู่เป็นผู้ใช้ในระบบการสื่อสารแบบรวมศูนย์ ซึ่งแต่ละคนจะมีบัญชีรายชื่อผู้ใช้เป็นจำนวนมากจากช่องทางการสื่อสารต่าง ๆ เมื่อผู้ใช้ A ประสงค์จะรับรู้ข้อมูลแสดงสถานะของผู้ใช้ B ทำให้ผู้ใช้ A จำเป็นต้องเพิ่มบัญชีรายชื่อของผู้ใช้ B ทั้งหมด ถ้าระบบบริการแสดงสถานะไม่มีระบบในการจัดการกับบัญชีรายชื่อของผู้ใช้ อาจเป็นไปได้ว่าผู้ใช้ A มีการเพิ่มบัญชีรายชื่อของผู้ใช้ B ที่ผิดพลาดได้ เช่น ในบางช่องทางอาจจะมีการเพิ่มรายชื่อของบุคคลอื่นเข้าไป ทำให้ผู้ใช้ A รับทราบข้อมูลแสดงสถานะที่ผิดพลาดได้

ปัญหาที่ได้กล่าวมาทั้งหมดข้างต้น มีผลกระทบกับประสิทธิภาพของระบบการสื่อสารแบบรวมศูนย์ โดยจะเรียกปัญหาดังกล่าวว่า Massive Presence Information ด้วยเหตุผลนี้ทำให้หัวข้อวิจัยที่เกี่ยวข้องกับการบริการแสดงสถานะ เป็นหัวข้อวิจัยที่ได้รับความสนใจเป็นอย่างมาก [3] [4] [5] [6] [7] แต่งานวิจัยเหล่านั้นยังไม่สามารถที่จะสนับสนุนการทำงานในรูปแบบระบบการสื่อสารแบบรวมศูนย์ได้อย่างสมบูรณ์ เช่น กรอบการทำงาน SIMPLE (SIP for Instant Messaging and Presence Leveraging Extensions) [3] ได้กล่าวถึงระบบบริการแสดงสถานะ โดยใช้พื้นฐานของระบบแจ้งเตือน SIP [8] ที่สามารถทำให้ข้อมูลแสดงสถานะจากแหล่งต่าง ๆ ในระบบเป็นข้อมูลแบบรวมศูนย์ได้ [4] และมีกฎการป้องกันข้อมูลแบบพื้นฐาน (Privacy Control) [9] [10] แต่ระบบ SIMPLE ทำงานได้เพียงบนโปรโตคอล SIP เท่านั้น ซึ่งระบบการสื่อสารแบบรวมศูนย์จำเป็นต้องรองรับข้อมูลแสดงสถานะจากโปรโตคอลอื่น ๆ ด้วย นอกจากนี้การป้องกันข้อมูลแสดงสถานะของกรอบการทำงานแบบ SIMPLE ยังเป็นการป้องกันแบบพื้นฐานที่เป็นการลดการเข้าถึงข้อมูลเท่านั้น โดยไม่สามารถที่จะเพิ่มข้อมูลที่เป็นประโยชน์อื่น ๆ ตัวอย่างเช่น ผู้ใช้ A และผู้ใช้ B เป็นเพื่อนร่วมงาน และอยู่ในห้องทำงานเดียวกัน โดยในขณะนั้นผู้ใช้ A ข้อมูลแสดงสถานะของเขาแสดงว่า เขาไม่สามารถที่จะรับการติดต่อได้ แต่ผู้ใช้ B ในขณะนั้น พร้อมที่จะรับการติดต่อ ในเวลาต่อมามีบุคคลที่ต้องการจะติดต่อผู้ใช้ A ถ้าเป็นกฎแบบพื้นฐานของ SIMPLE จะแสดงข้อมูลได้เพียงว่า ผู้ใช้ A ไม่สามารถที่จะรับการติดต่อได้ในขณะนั้น แต่ถ้าระบบแสดงสถานะมีความฉลาดในการแสดงผลจะสามารถแสดงข้อมูลสถานะของผู้ใช้ B ไปให้กับบุคคลที่ต้องการจะติดต่อกับผู้ใช้ A เพื่อเพิ่มความาสามารถในการติดต่อไปยังผู้ใช้ A ซึ่งจากตัวอย่างนี้แสดงให้เห็นว่า ถ้าระบบการสื่อสารแบบรวมศูนย์มีการจัดการระบบกฎการป้องกันข้อมูลที่ดี จะทำให้ระบบสามารถที่จะเพิ่มประสิทธิภาพในการติดต่อระหว่างผู้ใช้ในระบบการสื่อสารแบบรวมศูนย์ได้ดียิ่งขึ้น

นอกเหนือจากปัญหา Massive Presence Information และกฎการป้องกันแบบพื้นฐาน ในปัจจุบันระบบการสื่อสารแบบรวมศูนย์ยังคงใช้คอมพิวเตอร์ส่วนบุคคลเป็นศูนย์กลาง [1] ซึ่งแนวโน้มของเทคโนโลยีของระบบการสื่อสารแบบรวมศูนย์จะมีการเชื่อมโยงกับระบบโทรศัพท์เคลื่อนที่ยุคที่สาม เพื่อตอบสนองการทำงานแบบไม่จำกัดเวลาและสถานที่ นอกจากนั้น

ระบบดังกล่าวมีอัตราการส่งข้อมูลที่สูงทำให้เหมาะสมกับการทำงานกับระบบการสื่อสารแบบรวมศูนย์ที่ต้องรองรับการบริการ Multimedia Communication โดยในระบบโทรศัพท์เคลื่อนที่ยุคที่สามมีบริการหนึ่งที่เรียกว่า IMS (IP Multimedia Subsystem) [11] ซึ่งเป็นการบริการที่เชื่อมโยงระหว่างโลกของอินเตอร์เน็ตกับโลกของโทรศัพท์เคลื่อนที่ ได้อย่างสมบูรณ์แบบและลงตัว จึงทำให้ระบบการบริการแสดงสถานะจำเป็นจะต้องเชื่อมต่อกับระบบโทรศัพท์เคลื่อนที่ เพื่อที่จะรับทราบและแสดงข้อมูลสถานะไปยังระบบโทรศัพท์เคลื่อนที่ได้

ดังนั้นวิทยานิพนธ์ฉบับนี้จึงนำเสนอกรอบการทำงานและสถาปัตยกรรมการบริการแสดงสถานะแบบรวมศูนย์ (Framework and Architecture of Unified Presence Service: UPS) เพื่อจัดการกับปัญหา Massive Presence Information ที่เกิดจากระบบการสื่อสารแบบรวมศูนย์ โดยกรอบการทำงานแบ่งเป็น 3 ส่วนคือ ส่วนที่หนึ่ง Large-Scale of Presence Information ทำหน้าที่ในการจัดการบัญชีรายชื่อและแหล่งข้อมูลของผู้ใช้ ส่วนที่สอง Presence Information Aggregation ที่จะทำหน้าในการรวบรวมข้อมูลแสดงสถานะจากแหล่งข้อมูลต่าง ๆ ให้เป็นข้อมูลแสดงสถานะแบบรวมศูนย์ และส่วนสุดท้าย คือ Smart Distribution and Policy Control ทำหน้าที่กระจายและปกป้องข้อมูลแสดงสถานะของผู้ใช้อย่างชาญฉลาด โดยการสร้างความสัมพันธ์ของผู้ใช้งานในระบบ ให้เป็นข้อมูลบริบท (Context Information) และมีส่วนในการตัดสินใจที่ชาญฉลาด (Presence Reasoner Module) เพื่อใช้แทนกฎการทำงานแบบเดิม ซึ่งจะทำให้การแสดงผลของระบบการบริการแสดงสถานะแบบรวมศูนย์เพิ่มความสามารถในการติดต่อระหว่างผู้ใช้ และสถาปัตยกรรมของระบบการบริการแสดงสถานะแบบรวมศูนย์ได้ออกแบบและสร้างขึ้นโดยการเพิ่มเติมความสามารถพื้นฐานที่อยู่บนระบบที่เป็นมาตรฐานของระบบแจ้งเตือน SIP [8] เพื่อที่จะรองรับกับกรอบการทำงานของระบบการบริการแสดงสถานะแบบรวมศูนย์ที่ได้ออกแบบไว้ และสถาปัตยกรรมของการบริการแสดงสถานะจะสามารถทำงานร่วมกับระบบ IMS เพื่อทำการเชื่อมโยงระบบการสื่อสารแบบรวมศูนย์กับโทรศัพท์เคลื่อนที่ได้

1.2 วัตถุประสงค์

1. เพื่อศึกษารูปแบบการทำงานการแสดงสถานะผู้ใช้ของการแจ้งเตือน เพื่อเป็นพื้นฐานในการออกแบบกรอบการทำงานของการบริการแสดงสถานะแบบรวมศูนย์
2. เพื่อออกแบบกรอบการทำงานการบริการแสดงสถานะแบบรวมศูนย์
3. เพื่อสร้างสถาปัตยกรรมของการบริการแสดงสถานะแบบรวมศูนย์ที่สอดคล้องกับกรอบการทำงานที่ได้ออกแบบไว้ โดยสถาปัตยกรรมดังกล่าวจะต้องอยู่บน Component มาตรฐานมากที่สุด และต้องรองรับการทำงานร่วมกับระบบ IMS

4. เพื่อทดสอบการทำงานของการบริการแสดงสถานะแบบรวมศูนย์ เพื่อแสดงให้เห็นถึงความสามารถที่ได้เพิ่มเติม ความเหมาะสมจากการออกแบบ และการทำงานร่วมกับระบบการติดต่อสื่อสาร IMS

1.3 ขอบเขตของการวิจัย

1. ออกแบบกรอบการทำงานและสถาปัตยกรรมการบริการแสดงสถานะแบบรวมศูนย์ที่เหมาะสมกับระบบการสื่อสารแบบรวมศูนย์ โดยสามารถที่จะทำงานร่วมกับ IMS ได้
2. พัฒนาระบบต้นแบบการบริการแสดงสถานะแบบรวมศูนย์ โดยจะแสดงถึงสถาปัตยกรรมของระบบ และลำดับของสัญญาณที่มีในระบบ
3. การทดสอบการทำงานของระบบการบริการการแสดงสถานะแบบรวมศูนย์ จะอยู่ในรูปแบบของการจำลองการทำงานขึ้นมาเพื่อใช้ในการทดสอบ

1.4 ขั้นตอนและวิธีการดำเนินการวิจัย

1. ศึกษาและทำความเข้าใจเกี่ยวกับการออกแบบการบริการแสดงสถานะแบบรวมศูนย์ของระบบการสื่อสารแบบรวมศูนย์จากบทความที่เกี่ยวข้อง
2. ศึกษาและทำความเข้าใจเกี่ยวกับสัญญาณ SIP และโปรโตคอลอื่น ๆ ที่เกี่ยวข้องกับการออกแบบการบริการแสดงสถานะแบบรวมศูนย์รวมทั้งจัดทำโครงร่างวิทยานิพนธ์
3. ออกแบบและจำลองสถาปัตยกรรมการบริการแสดงสถานะแบบรวมศูนย์
4. นำสถาปัตยกรรมที่ออกแบบและจำลองมาทำการทดสอบ
5. ปรับปรุงและตรวจสอบความถูกต้องของระบบที่ทำการออกแบบ
6. รวบรวมผลการทดสอบสรุปผลจัดทำรายงานฉบับสมบูรณ์และคู่มือการใช้งาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. รูปแบบกรอบการทำงานและสถาปัตยกรรมของการบริการแสดงสถานะแบบรวมศูนย์ที่เหมาะสมกับระบบการสื่อสารแบบรวมศูนย์
2. สถาปัตยกรรมการบริการแสดงสถานะแบบรวมศูนย์ที่ได้ทำการออกแบบจะช่วยให้นักพัฒนาสามารถพัฒนาโปรแกรมประเภทนี้ ได้ง่ายยิ่งขึ้น
3. ผลงานในระหว่างที่ทำวิทยานิพนธ์สามารถนำไปตีพิมพ์ลงในบทความทางวิชาการ

บทที่ 2
ทฤษฎีและหลักการ

บทที่ 2 กล่าวถึงทฤษฎีและหลักการพื้นฐานที่เกี่ยวข้องกับการทำวิทยานิพนธ์นี้ โดยเนื้อหาที่จะกล่าวถึงประกอบด้วยรายละเอียดของ SIP ซึ่งเป็นโปรโตคอลสำหรับจัดการและควบคุมการติดต่อสื่อสารผ่านเครือข่าย โดยกล่าวถึงลักษณะพื้นฐานและองค์ประกอบของ SIP ในส่วนต่อมา คือ คุณลักษณะของระบบการสื่อสารแบบรวมศูนย์ โดยจะอธิบายถึงส่วนประกอบภายในของระบบและชี้ให้เห็นว่าความสำคัญของระบบการบริการแสดงสถานะที่มีต่อระบบการสื่อสารแบบรวมศูนย์ ถัดมาเป็นการอธิบายถึงรายละเอียดการบริการแสดงสถานะในระบบอินเตอร์เน็ตว่ามีโปรโตคอลและหลักการทำงานพื้นฐานเป็นอย่างไร ต่อมาเป็นการอธิบายเทคโนโลยี IMS ซึ่งเป็นเทคโนโลยีที่ถูกนำมาใช้เป็นหนึ่งในช่องทางการสื่อสารของระบบการสื่อสารแบบรวมศูนย์ และส่วนสุดท้าย คือ การอธิบายถึงรายละเอียดการบริการแสดงสถานะที่อยู่บนพื้นฐานของเทคโนโลยี IMS เพราะระบบการสื่อสารแบบรวมศูนย์ จำเป็นต้องรวบรวมข้อมูลแสดงสถานะของผู้ใช้จากระบบ IMS

2.1 Session Initiation Protocol (SIP)

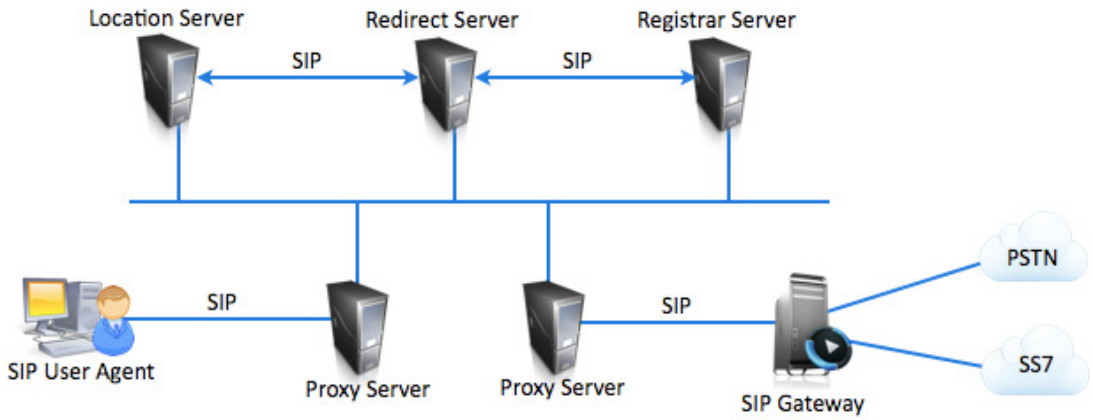
SIP [12] เป็นโปรโตคอลในระดับชั้น Application ที่ถูกพัฒนาขึ้นมาโดยองค์กร IETF โปรโตคอลถูกออกแบบให้ใช้ในการตกลงกันระหว่างคู่สนทนาหรือระหว่างผู้ใช้บริการ เพื่อทำการเริ่มต้นการโทร (Call Establishment), เปลี่ยนแปลงรูปแบบการโทร (Modification) และสิ้นสุดการโทรหรือจบการสนทนา (Call Termination) ดังนั้น SIP จึงมีหน้าที่รับผิดชอบในส่วนการส่งสัญญาณควบคุม (Control Signaling) ในช่วงก่อนการสนทนา ระหว่างการสนทนา และจบการสนทนาเท่านั้น นอกจากนี้การทำงานของ SIP เป็นไปตามมาตรฐาน ซึ่งถูกกำหนดอยู่ใน RFC 3261 [12] ทำให้เกิดความยืดหยุ่นและความสะดวกในการใช้งานแก่ผู้ใช้บริการมากยิ่งขึ้น สามารถที่จะนำไปประยุกต์ใช้ร่วมกับโปรโตคอลอื่น ๆ ได้

2.1.1 คุณสมบัติของโปรโตคอล SIP

1. การที่ SIP เป็นโปรโตคอลระดับ Application ซึ่งอยู่เหนือโปรโตคอลระดับ Transport ทำให้ SIP สามารถทำการส่งสัญญาณโดยเลือกใช้โปรโตคอลในระดับ Transport ได้ทั้งชนิด TCP (Transmission Control Protocol) และ UDP (User Datagram Protocol) ซึ่งขึ้นอยู่กับความต้องการของผู้ใช้งาน

2. รูปแบบสัญญาณที่นิยามตามมาตรฐาน SIP มีลักษณะเป็นข้อความ (Text-based) ที่มนุษย์สามารถอ่านเข้าใจได้ ถูกเรียกว่า SIP Message โดยรูปแบบและไวยากรณ์ของสัญญาณ มีลักษณะคล้ายกับรูปแบบสัญญาณโปรโตคอล HTTP (Hypertext Transfer Protocol) ทำให้ผู้พัฒนาสามารถทำการพัฒนาได้ง่าย อีกเหตุผลที่มีการใช้ SIP ในการแลกเปลี่ยนตลอดขั้นตอน ตั้งแต่การเริ่มสร้างการติดต่อสื่อสาร การเปลี่ยนแปลงการสื่อสาร และสิ้นสุดการสื่อสาร ก็เนื่องจาก SIP มีโครงสร้างที่ไม่ซับซ้อนเหมือนกับมาตรฐาน H.323 [13] ซึ่งเป็นอีกโปรโตคอลหนึ่งที่ถูกนำมาใช้ในระบบการติดต่อสื่อบนเครือข่ายเช่นกัน ดังนั้นจึงทำให้ SIP ได้รับความนิยมนำมาใช้งานเป็นมาตรฐานในการควบคุมระบบการติดต่อสื่อสารผ่านเครือข่าย และสามารถทำงานได้รวดเร็วกว่ามาตรฐาน H.323
3. มาตรฐาน SIP จะรับผิดชอบควบคุมขั้นตอนการส่งสัญญาณในส่วนการสร้างการติดต่อสื่อสาร ในระหว่างการสื่อสารเพื่อเปลี่ยนแปลงแก้ไขการติดต่อสื่อสารที่ใช้อยู่ และการสิ้นสุดการสื่อสารเท่านั้น ทำให้ในขั้นตอนการส่งข้อมูลสื่อประสม (Media Stream) เช่น เสียงหรือภาพจำเป็นต้องทำงานร่วมกับ RTP (Real-Time Transfer Protocol) อีกตัวอย่างหนึ่ง ในขั้นตอนการแลกเปลี่ยนความสามารถและข้อมูลสื่อประสม ก่อนทำการติดต่อสื่อสารในระบบนั้น (Media Capabilities Exchange) จะทำงานร่วมกับ SDP (Session Description Protocol) จากข้างต้นทำให้เห็นว่าการทำงานของ SIP นั้นมีความยืดหยุ่นและสามารถทำงานร่วมกับโปรโตคอลมาตรฐานอื่น ๆ เพื่อเพิ่มความสามารถในการจัดทำระบบการติดต่อสื่อสารโดยใช้ SIP ผ่านเครือข่ายได้
4. สถาปัตยกรรมตามมาตรฐาน SIP จะทำงานอยู่ในรูปแบบเครื่องลูกข่ายและแม่ข่าย (Client/Server) ซึ่งจะมีการอ้างอิงตัวเครื่องลูกข่าย (SIP Client, SIP User Agent) โดยชื่อที่ใช้อ้างอิงจะเรียกว่า SIP Uniform Resource Identifier (SIP URI) ทำให้เกิดความสะดวกและความยืดหยุ่นในการเรียกใช้บริการเพิ่มมากขึ้น เช่น sip:dommmy@open-ims.cnr.psu.ac.th ซึ่งมีลักษณะคล้ายกับ E-mail Address

2.1.2 องค์ประกอบสำคัญในระบบ SIP (SIP Components)



รูปที่ 2-1 องค์ประกอบภายในระบบ SIP

จากรูปที่ 2-1 เป็นองค์ประกอบภายในระบบ SIP ซึ่งเชื่อมต่อกันเป็นระบบการติดต่อสื่อสารผ่านเครือข่าย เนื่องจากโปรโตคอล SIP รองรับการทำงานแบบเครื่องลูกข่ายและแม่ข่าย ทำให้สามารถแบ่งได้ออกเป็น 2 ส่วนหลักใหญ่ ๆ ด้วยกัน คือ กลุ่มเครื่องลูกข่ายหรือ UA และกลุ่มเครื่องแม่ข่าย หรือ Network Servers

2.1.2.1 User Agents (UA)

ดังได้กล่าวข้างต้นแล้วว่า UA เป็นนิยามที่ใช้เรียกเครื่องลูกข่ายที่ทำการติดต่อสื่อสาร โดยความสามารถของ UA นั้นสามารถแบ่งแยกได้เป็น 2 รูปแบบด้วยกัน คือ

1. User Agent Client (UAC) เป็นเครื่องลูกข่าย ซึ่งทำหน้าที่ในการสร้างและส่งสัญญาณร้องขอ (SIP Request) ไปยังผู้ติดต่อปลายทาง ตัวอย่างเช่น INVITE, BYE, UPDATE เป็นต้น
2. User Agent Server (UAS) เป็นเครื่องลูกข่าย ซึ่งทำหน้าที่ในรับสัญญาณร้องขอจาก UAC มาประมวลผล และทำการสร้างสัญญาณตอบสนอง (SIP Response) ส่งกลับไปยังเครื่องลูกข่ายต้นทาง เช่น 200 OK, 302 Moved Temporarily, 404 Not Found เป็นต้น

โดยปกติแล้ว UA หนึ่ง ๆ จะทำหน้าที่เป็นทั้ง UAC และ UAS เนื่องจากในระบบ UA คนหนึ่ง ๆ จะมีการสร้างสัญญาณร้องขอการติดต่อสื่อสารไปยัง UA คนอื่น ๆ ซึ่งจะเรียกว่า ผู้โทร (Caller) และ UA จะต้องทำการรับสัญญาณร้องขอการติดต่อมาประมวลผล จากนั้นทำการสร้างสัญญาณตอบสนองไปยัง UA ต้นทาง ซึ่งจะเรียกว่าผู้ถูกเรียก (Callee) ดังนั้น UA ในระบบ

การติดต่อสื่อสารจึงต้องมีความสามารถหลัก ๆ ทั้งในเรื่องการสร้างสัญญาณร้องขอ ประมวลผลสัญญาณร้องขอ สร้างสัญญาณตอบสนอง และการส่งสัญญาณไปยัง UA เพื่อเริ่มต้นการติดต่อสื่อสารระหว่าง UA ด้วยกัน

2.1.2.2 Network Servers

เป็นนิยามที่ใช้เรียกกลุ่มของเครื่องแม่ข่ายที่ร่วมกันทำงานภายในระบบ โดยในระบบการติดต่อสื่อสารโดยใช้ SIP ผ่านเครือข่าย ไม่จำเป็นที่จะต้องมีเครื่องแม่ข่ายครบทุกส่วนเพื่อให้ระบบทำงานได้ เพียงแต่ถ้าหากมีครบทุกส่วนจะทำให้ระบบการติดต่อสื่อสารมีความสามารถเพิ่มขึ้น สำหรับเครื่องแม่ข่ายแต่ละตัว มีหน้าที่รับผิดชอบการทำงาน ดังนี้

1. Registrar Server ทำหน้าที่รับการลงทะเบียน (Register) จากผู้ใช้บริการ เพื่อใช้สำหรับเก็บข้อมูลผู้เข้าใช้บริการ ไม่ว่าจะเป็นชื่อข้อมูลส่วนตัว, SIP URI ที่ใช้ในการติดต่อ และข้อมูลสำคัญอื่น ๆ เช่น E-mail หรือโทรศัพท์ เป็นต้น
2. Proxy Server ทำหน้าที่เป็นตัวกลางในการส่งสัญญาณระหว่างผู้ใช้บริการด้วยกัน หรือระหว่างผู้ใช้บริการกับเครื่องแม่ข่าย โดย Proxy Server จะรับผิดชอบการส่งสัญญาณ (SIP Forwarding) ซึ่งในบางระบบ Proxy Server จะถูกรวมเข้ากับ Registrar Server เพื่อลดจำนวนเครื่องแม่ข่ายที่อยู่ในระบบลง
3. Redirect Server ทำหน้าที่ตอบสนองต่อสัญญาณ SIP ที่เข้ามา เพื่อทำการส่งสัญญาณ SIP ไปยังผู้รับที่ระบุอยู่ในสัญญาณ SIP ซึ่งโดยปกติจะทำงานร่วมกับ Proxy Server
4. Location Server ทำหน้าที่ในการเก็บข้อมูลผู้ใช้ ซึ่งจะเกี่ยวข้องกับตำแหน่งที่อยู่ของผู้ใช้บริการในระบบ โดยข้อมูลที่เก็บอยู่นั้น อาจจะถูกเรียกใช้จาก Proxy Server หรือ Redirect Server ในการส่งสัญญาณ SIP ไปยังผู้ใช้ ตามมาตรฐาน SIP อนุญาตให้ผู้พัฒนาสามารถทำการพัฒนา Location Server ร่วมอยู่ด้วยกันกับ SIP Server ได้
5. SIP Gateway ทำหน้าที่เชื่อมต่อกับระบบการติดต่อสื่อสารผ่านเครือข่ายกับระบบเครือข่ายอื่น ๆ เช่น ระบบเครือข่าย PSTN (Public Switching Telephony Network), ระบบ SS7 (Signaling System 7) หรือ IMS (IP Multimedia Subsystem) เป็นต้น ทั้งนี้เพื่อเพิ่มขีดความสามารถในสื่อสาร ทำให้สามารถติดต่อสื่อสารระหว่างกันไม่ว่าจะอยู่กันคนละเครือข่าย หรือใช้อุปกรณ์ที่แตกต่างกัน

2.2 คุณลักษณะของระบบการสื่อสารแบบรวมศูนย์

ระบบการสื่อสารแบบรวมศูนย์ คือ ระบบที่รวมการสื่อสารในทุกรูปแบบเข้าไว้เป็นหนึ่งเดียว โดยจะรวมการสื่อสารในทั้งแบบตามเวลาจริง (Real-Time) และไม่ตามเวลาจริง

(Non Real-Time) เข้าไว้อย่างลงตัว โดยไม่ทำให้ผู้ใช้รู้สึกถึงการเชื่อมของการสื่อสารในหลากหลายรูปแบบ [2] นอกเหนือจากบริการสื่อสารต่าง ๆ ระบบการสื่อสารแบบรวมศูนย์ยังมีส่วนประกอบอื่น ๆ ที่ทำให้ระบบรองรับการใช้งานผู้ใช้ให้ดียิ่งขึ้น โดยการบริการในระบบการสื่อสารแบบรวมศูนย์จะประกอบขึ้นด้วย

1. **ส่วนควบคุมการติดต่อสื่อสาร (Call Control Service)** คือ ส่วนที่สำคัญในระบบการติดต่อสื่อสารแบบรวมศูนย์ โดยจะเป็นโปรโตคอลที่ใช้ในการควบคุมการติดต่อสื่อสารของทั้งระบบ โดยจะประกอบโดยส่วนหลัก ๆ คือ ส่วนควบคุมระบบ Circuit-Switched เช่น โปรโตคอลที่ใช้กันทั่วไป TUP [ITU-T Recommendation Q.721] ISUP [ITU-T Recommendation Q.761] และ BICC [ITU-T Recommendation Q.1901] ต่อมาส่วนควบคุมในส่วนของ Packet-Switched ตัวอย่างเช่น SIP และส่วนควบคุมของ Web Services โดยจะทำหน้าที่ในการสร้างและจัดการช่องทางการติดต่อของระบบ
2. **การบริการแสดงสถานะ (Presence Service)** เป็นส่วนประกอบที่สำคัญมากที่สุด [1] [2] และเป็นพื้นฐานในการสร้างระบบการสื่อสารแบบรวมศูนย์ โดยบริการนี้จะทำหน้าที่ในการแจ้งและบ่งบอกสถานะของช่องทางการสื่อสารต่าง ๆ ของผู้ใช้ โดยระบบจะเก็บข้อมูลจากช่องทางการสื่อสารและอุปกรณ์ต่าง ๆ ในระบบและทำให้เป็นข้อมูลแสดงสถานะแบบรวมศูนย์ (Unified Presence Information) และแจ้งไปยังผู้ใช้หรือโปรแกรมประยุกต์อื่น ๆ เพื่อใช้ในการคิดหรือคำนวณว่าควรจะติดต่อไปยังช่องทางการสื่อสารใดของผู้ใช้ปลายทาง ซึ่งตัวอย่างที่เห็นได้ชัดเจนมากที่สุด คือ การทำงานในรูปแบบของ Instant Messenger ที่จะแสดงสถานะของเพื่อนในบัญชีรายชื่อว่า ออนไลน์ ออฟไลน์ พร้อมที่จะคุย หรือไม่ว่าง ทำให้ผู้ที่ได้รับทราบข้อมูลดังกล่าวสามารถที่จะเลือกได้ว่าควรจะติดต่อไปหรือไม่ จากเหตุผลที่ได้กล่าวมาทำให้เห็นได้ว่า การบริการแสดงสถานะจะส่งผลกระทบโดยตรงกับประสิทธิภาพของระบบ เช่น ถ้าระบบที่มีการแจ้งเตือนอย่างแม่นยำและถูกต้องก็จะทำให้ผู้ใช้ลดเวลาในการติดต่อสื่อสารลงไปได้ หรือถ้าไม่สามารถที่จะติดต่อได้ระบบก็จะบ่งบอกถึงสถานะ เช่น ไม่ว่าง ซึ่งจะทำให้ผู้ที่ต้องการจะติดต่อไม่ขัดข้องใจ แต่อย่างไรก็ดีการที่จะสร้างการบริการแสดงสถานะที่เหมาะสมกับระบบการสื่อสารแบบรวมศูนย์จำเป็นต้องคำนึงถึงความหลากหลายของช่องทางการติดต่อ ซึ่งระบบจำเป็นจะต้องจัดการ จำนวนเบอร์หรือที่อยู่ที่ใช้ในการติดต่อ (Contact Address) ที่มีจำนวนมากของผู้ใช้ หรือความหลากหลายของข้อมูลแสดงสถานะ (Presence Information) ไม่ว่าจะเป็นโปรโตคอลที่แตกต่างกัน โดยปัญหาข้างต้นจะถูกเรียกว่า Massive Presence Information (ซึ่งจะให้รายละเอียดในบทที่ 4) ทำให้การออกแบบระบบการบริการแสดงสถานะให้จัดการกับปัญหา Massive Presence Information ได้นั้น

จะทำให้ระบบการสื่อสารแบบรวมศูนย์มีประสิทธิภาพที่ดีมากขึ้น และฉลาดในการติดต่อระหว่างผู้ใช้ภายใน

3. **เมสเซนเจอร์ (Instant Messaging Service)** คือ หนึ่งในช่องทางการสื่อสารของระบบการสื่อสารแบบรวมศูนย์ ซึ่งจะทำงานในลักษณะของการส่งข้อความตามเวลาจริง (Real-Time Text based) ซึ่งตัวอย่างที่รู้จักกันแพร่หลายในปัจจุบัน เช่น AIM, YaHoo! Messenger, Window Live Messenger, MSN Messenger, GTalk และ Skype ซึ่งเป็นเมสเซนเจอร์ที่รู้กันทั่วไป (Public Messenger) ซึ่งไม่เหมาะสมสำหรับระบบที่เป็นองค์กร โดยเมื่อมองในเรื่องความปลอดภัยที่องค์กรนั้น ๆ ไม่สามารถควบคุมผู้ใช้งานได้ด้วยตนเอง แต่อย่างไรก็ตามถ้าระบบการสื่อสารแบบรวมศูนย์มีระบบ Policy และ Privacy ที่มีความฉลาดเพียงพอในการพิจารณาการส่งข้อมูลสถานะของผู้ใช้ไปยังเมสเซนเจอร์ดังกล่าวก็จะทำให้ระบบการสื่อสารแบบรวมศูนย์มีประสิทธิภาพที่เพิ่มจากช่องทางการสื่อสารที่มากขึ้นได้
4. **วอยส์โอเวอร์ไอพี (VoIP)** คือ หนึ่งในช่องทางการสื่อสารที่ใช้การส่งข้อมูลเสียงผ่านทางระบบเครือข่ายไอพี (IP Network) หรือแพกเก็ตสวิตซ์ (Packet-Switched) โดยข้อดีที่ทำให้วอยส์โอเวอร์ไอพีถูกเลือกมาเป็นหนึ่งในพื้นฐานของระบบการสื่อสารแบบรวมศูนย์ เนื่องจากประหยัดต้นทุนการโทรในระยะทางไกล และประหยัดในการเคลื่อนย้าย เนื่องจากเพียงแค่ย้ายตัวเครื่องลูกข่ายเท่านั้นไม่จำเป็นต้องย้ายสายสัญญาณเหมือนกับโทรศัพท์ขั้นพื้นฐาน นอกจากนี้ยังมีบริการเสริม เช่น วอยส์เมล์ การบันทึกข้อมูลการใช้ลงฐานข้อมูลที่จะระบุ ต้นสาย ปลายสาย และระยะเวลาการใช้งาน จากข้อดีเหล่านี้ ทำให้ระบบการสื่อสารแบบรวมศูนย์นำเอาวอยส์โอเวอร์ไอพีมาเป็นส่วนหนึ่งในช่องทางการสื่อสาร แต่ก็ยังมีปัญหาเล็กน้อยบางประการที่ทำให้วอยส์โอเวอร์ไอพีไม่สามารถที่จะทำงานแทนที่โทรศัพท์พื้นฐานได้ทั้งหมด เช่น ในกรณีที่ปลายสายมีแบนด์วิดท์ที่น้อยอาจจะทำให้เกิดข้อมูลที่ขาดตอนหรือศูนย์หายได้ ข้อเสียถัดมา คือ ไม่สามารถที่จะทำงานได้เมื่อเกิดปัญหาไฟฟ้าขัดข้องได้ และสุดท้าย คือ ไม่สามารถที่จะทำการหาเส้นทางการโทรของเลขหมายฉุกเฉิน (911) ได้ แต่อย่างไรก็ตามมีแนวคิดการแก้ปัญหาการโทรแบบฉุกเฉินแล้วจากเอกสาร [24] แต่จำเป็นต้องเพิ่มต้นทุนในการสร้างโครงข่ายเพิ่มขึ้นด้วย
5. **โทรศัพท์พื้นฐาน (PSTN)** มีข้อดีในการติดต่อภายในองค์กร เนื่องจากการติดต่อนั้นจะไม่มีต้นทุนและสามารถที่ทำการติดต่อแบบฉุกเฉิน หรืออาจจะติดต่อในสภาวะคับขัน เช่น เมื่อไฟฟ้าขัดข้อง แต่ข้อเสียของระบบดังกล่าว คือ ต้นทุนในการติดต่อในระยะทางไกล เช่น การติดต่อไปยังต่างประเทศ หรือการโทรข้ามเครือข่าย และยังมีต้นทุนในการติดตั้งและการเคลื่อนย้ายที่สูง นอกจากนี้การสร้างการบริการใหม่ ๆ จะทำได้ยาก เช่น อุปกรณ์นั้นไม่สามารถที่จะส่งข้อมูลสถานะของผู้ใช้งานได้ เป็นต้น จึงทำให้ระบบโทรศัพท์พื้นฐาน

มีความด้อยกว่าระบบวอยส์โอเวอร์ไอพี เมื่อมองในการเชื่อมโยงเข้าสู่ระบบการสื่อสารแบบรวมศูนย์

6. **ระบบการประชุม (Conferencing Service) และระบบเครื่องมือช่วยเหลือ (Collaborations Service)** คือ ระบบสามารถให้ผู้ใช้เชื่อมต่อการสื่อสารในรูปแบบเสียงและวีดีโอพร้อมกันได้จำนวนมากกว่าสองคน โดยในระบบการประชุมจะมีการบริการเครื่องมือช่วย เช่น Whiteboarding, File-Sharing, Document-Sharing, Recording และ Voting เป็นต้น ซึ่งเป็นประโยชน์อย่างมากสำหรับการสื่อสารแบบรวมศูนย์ เพราะทำให้ผู้ใช้ในระบบสามารถแลกเปลี่ยนความคิดเห็นและตอบโต้กันได้ในทันทีทันใด ทำให้ผู้ใช้ประหยัดต้นทุนไม่ว่าจะเป็นเวลาหรือค่าเดินทาง ซึ่งสามารถที่จะแบ่งการบริการการประชุมได้เป็น 2 กลุ่ม คือ กลุ่มโปรแกรมประยุกต์การประชุมผ่านระบบเครือข่าย (Application) และกลุ่มการประชุมผ่านระบบเครือข่ายโดยใช้โปรแกรมประยุกต์แบบเว็บ (Web Application) ซึ่งในแต่ละกลุ่มก็จะข้อดีข้อเสียและความเหมาะสมในงานแต่ละประเภทที่แตกต่างกัน เช่น เมื่อผู้ใช้ต้องการความชัดเจนของภาพและเสียงกลุ่มที่เหมาะสม คือ การใช้เป็นโปรแกรมประยุกต์ แต่ถ้าต้องการเพียงการตอบโต้กันเท่านั้นโดยไม่คำนึงถึงความชัดเจนของภาพและเสียง และสามารถที่จะผ่าน NAT และ Firewall เนื่องจากบางผู้ใช้ไม่ได้ใช้งานในระบบเน็ตเวิร์คเดียวกันทำให้การประชุมผ่านเว็บเหมาะสมกว่าการประชุมแบบโปรแกรมประยุกต์
7. **ระบบการเชื่อมต่อกับอุปกรณ์ที่สามารถเคลื่อนที่ได้ (Mobility Communication)** คือ การเชื่อมต่อไปยังระบบโทรศัพท์เคลื่อนที่ยุคที่สามและอนาคต เนื่องด้วยจุดเด่นของระบบโทรศัพท์เคลื่อนที่สามารถเข้าถึงผู้ใช้ได้ในวงกว้าง และในระบบโทรศัพท์เคลื่อนที่ยุคที่สามมีคุณสมบัติพื้นฐานในการทำงานแบบแพกเก็ตสวิตซ์ (Packet-Switched) ทำให้สามารถเชื่อมต่อกับโลกของอินเตอร์เน็ตได้ด้วยแบนด์วิดท์ขนาดสูง ทำให้เหมาะสมกับการทำการประชุมหรือโปรแกรมประยุกต์อื่นๆ เมื่อมองในมุมของระบบการสื่อสารแบบรวมศูนย์ นอกจากที่จะทำให้เข้าถึงผู้ใช้ได้ง่ายขึ้นแล้วก็ยังสามารถที่ติดต่อไปยังสถานที่ทำงานภาคสนาม เช่น ในกรณีที่ผู้ใช้ทำงานภาคสนามและต้องการขอคำปรึกษาก็สามารถที่จะทำการประชุมกับผู้ใช้งานที่อยู่ในสำนักงานใหญ่ได้ และอีกปัจจัยที่ทำให้ระบบโทรศัพท์เคลื่อนที่ยุคที่สามเหมาะสมกับระบบสื่อสารแบบรวมศูนย์เพราะมี IMS ที่เป็นโครงสร้างพื้นฐานในการพัฒนาโปรแกรมต่างๆ บนระบบโทรศัพท์เคลื่อนที่ยุคที่สาม โดยจะใช้การควบคุมการติดต่อ (Call Control) ของโปรโตคอล SIP และโปรโตคอลอื่นๆ ที่จะช่วยให้ผู้พัฒนาสามารถสร้างการบริการใหม่ๆ เช่น Conferencing Service, Presence Serviceหรือ IPTV Service เป็นต้น จึงทำให้การเชื่อมต่อกับระบบโทรศัพท์เคลื่อนที่ยุคที่สามเป็นพื้นฐานหนึ่งที่สำคัญต่อระบบการสื่อสารแบบรวมศูนย์

8. **อีเมล์ (E-mail)** คือ หนึ่งในรูปแบบการสื่อสารแบบไม่ตามเวลาจริง สำหรับระบบการสื่อสารแบบรวมศูนย์เป็นบริการที่มีการใช้งานมากที่สุดอีกบริการหนึ่ง เนื่องด้วยความสามารถในการส่งข้อความทิ้งไว้ในขณะที่ผู้ใช้ปลายทางไม่สามารถที่จะติดต่อได้ในขณะนั้น และยังสามารถส่งเอกสารต่าง ๆ ประกอบไปกับข้อความดังกล่าวได้ เช่น ภาพ, เอกสาร, ข้อมูลเสียงหรือภาพวีดีโอ เป็นต้น ทำให้อีเมล์เป็นบริการที่ผู้ใช้ในระบบมีการใช้กันมากที่สุดบริการหนึ่ง

จะเห็นได้ว่าการบริการการสื่อสารและบริการเครื่องมือช่วย (Collaboration Service) ที่แสดงไว้ข้างต้นนี้เป็นพื้นฐานที่ระบบการสื่อสารแบบรวมศูนย์ควรจะมี เพื่อให้บริการตอบสนองความต้องการพื้นฐานของผู้ใช้ โดยเมื่อเรียบเรียงลำดับความสำคัญและเปอร์เซ็นต์การใช้งานจากเอกสาร [25] สามารถที่จะแสดงได้ดังตารางที่ 2-1 นี้

ตารางที่ 2-1 เปอร์เซ็นต์ข้อมูลการใช้งานบริการในระบบการสื่อสารแบบรวมศูนย์

	คะแนนความสำคัญ	**เปอร์เซ็นต์การใช้งาน**	**หมายเหตุ**
การบริการแสดงสถานะ	5/5	81%	เปอร์เซ็นต์การใช้งานอ้างจาก [25]
การควบคุมการโทร	5/5	N/A	
เมสเซนเจอร์	4/5	81%	[25]
อี-เมล์	4/5	81%	[25]
การบริการการประชุมด้วยภาพและเสียง	4/5	74%	[25]
การสื่อสารแบบเคลื่อนที่ได้	4/5	N/A	
วอยส์โอเวอร์ไอพี	3/5	N/A	
ระบบโทรศัพท์ขั้นพื้นฐาน	1/5	N/A	

จากตาราตารางที่ 2-1 ชี้ให้เห็นว่าการบริการสำคัญเป็นอันดับหนึ่ง คือ การบริการแสดงสถานะ (Presence Service) เนื่องจากมีเปอร์เซ็นต์การใช้งานที่สูงและมีระดับความสำคัญเป็น 5/5 เพราะการบริการดังกล่าวทำให้ผู้ใช้รู้สึกได้ว่าสามารถมองเห็นการสื่อสารใน

ระบบเป็นหนึ่งเดียว โดยการบริการนี้สามารถบอกได้ว่าบุคคลที่ผู้ใช้ต้องการจะติดต่อนั้นมีความสามารถที่จะรับการสื่อสารในช่องทางใดบ้างในขณะนั้น หรือสามารถที่จะบ่งบอกว่าช่องทางที่เหมาะสมในการติดต่อกับผู้ใช้ปลายทาง ทำให้การบริการแสดงสถานะจะส่งผลกระทบโดยตรงต่อประสิทธิภาพของระบบการติดต่อสื่อสารแบบรวมศูนย์ ตัวอย่างเช่น เมื่อผู้ใช้ที่หนึ่งต้องการที่จะติดต่อไปยังผู้ใช้ที่สอง แต่เนื่องด้วยผู้ใช้ที่สองมีอุปกรณ์ที่ใช้ในการสื่อสารหลากหลายรูปแบบ ทำให้ผู้ใช้ที่หนึ่งต้องเลือกช่องในการสื่อสาร โดยอาจจะต้องโทรไปยังทุก ๆ ช่องทาง เพื่อที่จะติดต่อกับผู้ใช้ที่สอง แต่ถ้าระบบสามารถบ่งบอกได้ว่า ช่องทางที่เหมาะสมหรือบ่งบอกว่า ผู้ใช้ที่สองไม่สามารถที่ติดต่อได้ หรือไม่ว่างที่จะรับการติดต่อ ซึ่งจะทำให้ผู้ใช้ที่หนึ่งไม่จำเป็นที่จะต้องติดต่อไปยังทุก ๆ ช่องทางการสื่อสารของผู้ใช้ที่สอง และทำให้ผู้ใช้ที่หนึ่งไม่เกิดความไม่พอใจ เมื่อติดต่อผู้ใช้ที่สองไม่ได้ เพราะระบบได้แสดงสถานะว่าผู้ใช้ที่สองไม่ว่างหรือไม่สามารถติดต่อได้ในขณะนั้น จากเหตุผลข้างต้นทำให้การบริการบ่งบอกถึงสถานะและพฤติกรรมของผู้ใช้งานถึงได้คะแนนและเปอร์เซ็นต์การใช้งานมากกว่าบริการอื่น ๆ

2.3 การบริการแสดงสถานะในระบบอินเตอร์เน็ตปัจจุบัน

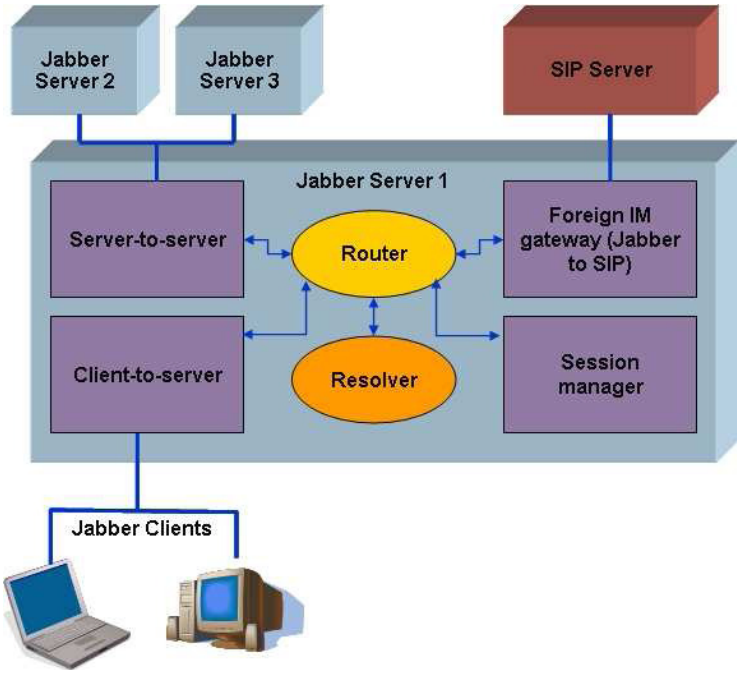
โดยกรอบการทำงานและสถาปัตยกรรมของการบริการแสดงสถานะในระบบอินเตอร์เน็ตปัจจุบันนั้นมี 2 รูปแบบมาตรฐาน โดยทั้งสองรูปแบบได้ถูกกำหนดเป็นรูปแบบพื้นฐานของการบริการแสดงสถานะและอินสแตนท์เมสเซจเจอร์ โดย IETF [14] คือ SIMPLE/SIP [3] และ Jabber/XMPP [15] โดยทั้งสองรูปแบบจะมีการทำงานที่แตกต่างกัน ซึ่งจะอธิบายรายละเอียดในหัวข้อที่ 2.3.1 และ 2.3.2

2.3.1 กรอบการทำงานและสถาปัตยกรรมของ Jabber/XMPP

กรอบการทำงานและสถาปัตยกรรมของ Jabber/XMPP เป็นที่น่าสนใจเพราะเป็นเทคโนโลยีโอเพนซอร์ซ (Open Source) สามารถที่หาข้อมูลทั้งหมดผ่านทางอินเตอร์เน็ต และมีผู้ให้บริการอินสแตนท์เมสเซนเจอร์อย่าง GTalk ที่ใช้สถาปัตยกรรมดังกล่าวในการทำงาน ตัวอย่างเช่น เมสเซนเจอร์, การบริการแสดงสถานะ และการทำวอยส์โอเวอร์ไอพี

สถาปัตยกรรมของ Jabber จะใช้โปรโตคอล Extensible Messaging and Presence Protocol (XMPP) ซึ่งเป็นโปรโตคอลที่ใช้ในการรับส่งข้อมูลระหว่าง 2 เครื่องลูกข่ายผ่านระบบเน็ตเวิร์ค ซึ่งจะแตกต่างกับ SIP ที่จะทำหน้าที่เป็นสัญญาณเพื่อใช้ในการควบคุมและสร้างการเชื่อมต่อก่อนที่จะรับส่งข้อมูล และข้อแตกต่างอีกข้อของ XMPP คือการใช้การเชื่อมต่อที่มีอยู่ก่อนแล้ว (Persistent Connection) ในการส่งข้อมูลการแสดงสถานะและอินสแตนท์เมสเซจเจอร์ โดยข้อมูลดังกล่าวจะอยู่บนพื้นฐานของสตรีมมิ่ง XML (Streaming XML) ที่จะไม่ใช่การส่ง XML ธรรมดาแต่จะเป็น XML ที่มีความสามารถเปลี่ยนแปลงได้ (Dynamic XML) ซึ่งข้อมูลที่

ส่งจะถูกเรียกว่า XML Stanzas [15] ที่จะนำข้อมูลข้อความและข้อมูลแสดงสถานะรับส่งระหว่างเครื่องลูกข่ายกับเครื่องแม่ข่าย Jabber และ Jabber จะทำหน้าที่ส่งต่อไปยังเครื่องลูกข่ายปลายทาง โดยสถาปัตยกรรมของ Jabber สามารถเขียนได้ดังรูปที่ 2-2



รูปที่ 2-2 สถาปัตยกรรมของ Jabber [15]

โดยเครื่องแม่ข่ายของ Jabber สามารถที่จะเชื่อมต่อกับเครื่องแม่ข่ายของ Jabber อื่น ๆ ดังรูปที่ 2-2 เพื่อที่จะทำให้เครื่องแม่ข่ายสามารถที่จะรองรับจำนวนผู้ใช้จำนวนมากขึ้น และ Jabber สามารถที่จะสร้างเกตเวย์ (Gateway) เพื่อที่จะเชื่อมต่อไปยังการสื่อสารที่อยู่บนพื้นฐานของโปรโตคอล SIP โดยการเปลี่ยนจากรูปแบบ XML ของ XMPP ไปเป็นสัญญาณ SIP ดังตารางที่ 2-2 และมีส่วนของ Router ทำหน้าที่ในการกระจายข้อมูลสทรีม XML ไปยังส่วนต่าง ๆ ภายในของ Jabber

ตารางที่ 2-2 เปรียบเทียบการเปลี่ยนระหว่าง XMPP ไปยัง SIP [15]

XMPP element and Attribute	SIP Header and Content
<body/>	Body of MESSAGE
<subject/>	SUBJECT
<thread/>	Call-ID
From	From
To	To
Type	(no mapping)
Xml:Lang	Content-Language

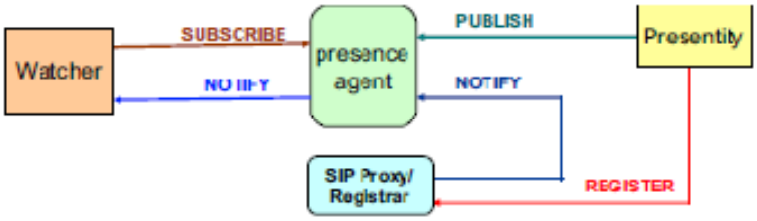
2.3.2 กรอบการทำงานและสถาปัตยกรรมของ SIMPLE/SIP [3]

SIMPLE เป็นสถาปัตยกรรมหนึ่งที่ออกแบบเพื่อรองรับการบริการแสดงสถานะผ่านระบบอินเตอร์เน็ต โดยใช้โปรโตคอล SIP เป็นสัญญาณพื้นฐานของสถาปัตยกรรม โดยสถาปัตยกรรม SIMPLE ได้ใช้การประยุกต์พื้นฐานของระบบการแจ้งเตือนของ SIP ให้สามารถทำงานการบริการแสดงสถานะรูปแบบใหม่ ๆ เช่น การรวบรวมข้อมูลแสดงสถานะจากแหล่งข้อมูลต่าง ๆ (Composition) กระบวนการป้องกันข้อมูล (Privacy Control) และการเก็บบัญชีรายชื่อของผู้ใช้ไว้ภายในระบบ (Resource List)

ก่อนจะทำความเข้าใจถึงสถาปัตยกรรมของ SIMPLE จำเป็นที่จะต้องเข้าใจถึงคำศัพท์เฉพาะที่ใช้ในการอธิบายเสียก่อน โดยคำศัพท์เฉพาะนี้จะถูกใช้ในการทำงานต่าง ๆ ของสถาปัตยกรรมของ SIMPLE มีดังต่อไปนี้

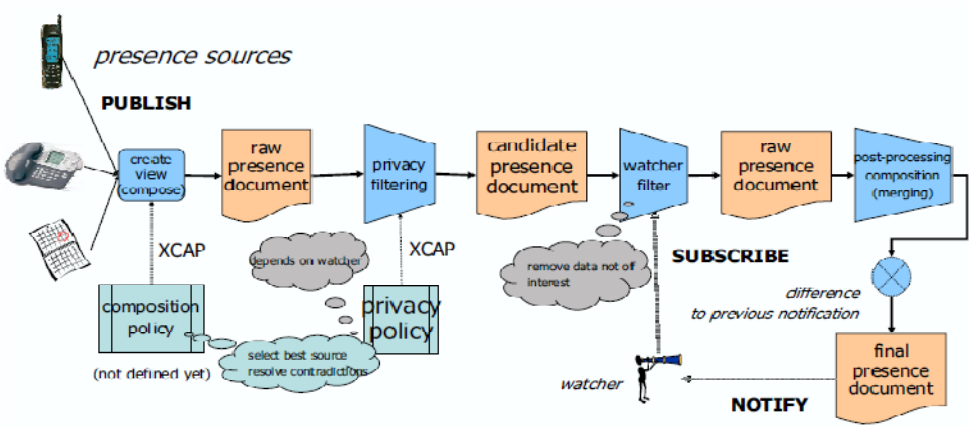
1. **Presentity** คือ บุคคลที่ต้องการแสดงข้อมูลสถานะ
2. **Watcher** คือ บุคคลหรือโปรแกรมที่ต้องการรับทราบสถานะของ Presentity
3. **Presence Server** คือ เครื่องแม่ข่ายที่ใช้ในการเก็บข้อมูลสถานะ และรองรับกระบวนการ Subscription, Publication และ Notification ให้กับ Presentity และ Watcher โดย Presence Server ทำหน้าที่คล้ายกับ SIP Proxy แต่จะรับจัดการเฉพาะสัญญาณ PUBLISH, SUBSCRIBE และ NOTIFY เท่านั้น
4. **Publication** คือ กระบวนการที่ Presentity ส่งข้อมูลแสดงสถานะของตนเองไปให้ Presence Agent (ซึ่งในที่นี้ คือ Presence Server) โดยผ่านทางสัญญาณ PUBLISH เพื่อรอกระบวนการ Subcription และ Notification ต่อไป

5. **Subscription** คือ กระบวนการที่ Watcher ส่งสัญญาณ SUBSCRIBE ไปยัง Presence Server เพื่อบ่งบอกถึงความต้องการของ Watcher ว่าต้องการจะรับทราบข้อมูลแสดงสถานะของ Presentity คนใด
6. **Notification** คือ กระบวนการที่ Presence Server ส่งข้อมูลแสดงสถานะของ Presentity ที่ถูกร้องขอไปให้ Watcher หลังจากรับกระบวนการ Subscription โดยที่ข้อมูลดังกล่าวจะถูกส่งใหม่ทุกครั้ง เมื่อ Presentity ได้เปลี่ยนแปลงข้อมูลแสดงสถานะ และการส่งข้อมูลจะถูกส่งผ่านทางสัญญาณ NOTIFY



รูปที่ 2-3 องค์ประกอบพื้นฐานของการบริการแสดงสถานะของ SIP [8]

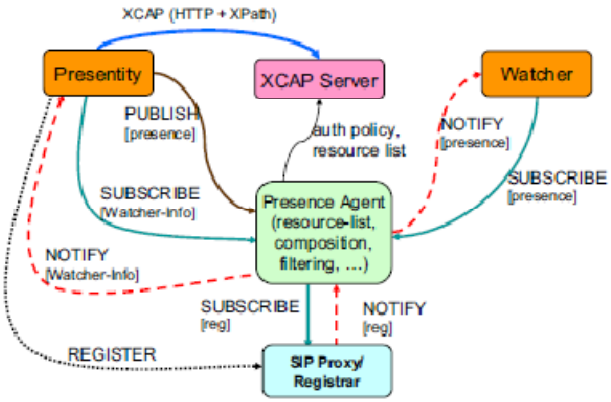
จากรูปที่ 2-3 แสดงถึงองค์ประกอบพื้นฐานที่ใช้ในกระบวนการบริการแสดงสถานะของ SIP เริ่มต้นโดย Presentity จะต้องลงทะเบียนตนเองเข้าสู่ระบบ ผ่านทางสัญญาณ REGISTER ต่อจากนั้น Presentity ที่ต้องการจะแสดงข้อมูลสถานะของตนเองให้ผู้ที่ต้องการรับทราบข้อมูลดังกล่าวไปยัง Presence Server ผ่านทางสัญญาณ PUBLISH โดยสัญญาณดังกล่าวจะส่งข้อมูลแสดงสถานะมากับส่วน Boby ของสัญญาณ ซึ่งเป็นรูปแบบข้อมูลแสดงสถานะจะอยู่ในรูปแบบ XML PIDF [16] ต่อมา Watcher ที่ต้องการรับทราบข้อมูลแสดงสถานะของ Presentity ที่เขาสนใจ Watcher ก็จะทำการร้องขอข้อมูลดังกล่าวผ่านทางสัญญาณ SUBSCRIBE ไปยัง Presence Server และข้อมูลของ Presentity ก็จะถูกส่งกลับไปยัง Watcher โดยผ่านทางสัญญาณ NOTIFY และจะส่งข้อมูลแสดงสถานะ PIDF ของ Presentity มาให้ จากกระบวนการพื้นฐานจะทำให้เห็นภาพรวมของการปฏิสัมพันธ์ระหว่างองค์ประกอบต่าง ๆ ภายในกระบวนการบริการแสดงสถานะพื้นฐานของ SIP แต่ยังไม่ได้อธิบายการลำดับการส่งของข้อมูลว่า จำเป็นต้องผ่านกระบวนการใดบ้าง จากรูปที่ 2-4 ได้อธิบายถึงรูปแบบของลำดับการส่งของข้อมูลแสดงสถานะ



รูปที่ 2-4 A Presence Data Model [17]

จากรูปที่ 2-4 เป็นการอธิบายถึงข้อมูลแสดงสถานะที่ต้องผ่านฟังก์ชันการทำงานใดบ้างในสถาปัตยกรรม SIMPLE โดยข้อมูลแสดงสถานะจะถูกส่งจากแหล่งข้อมูล (เช่น IP Phone แบบตั้งโต๊ะ, Wireless IP Phone หรือ ตารางเวลา เป็นต้น) ผ่านทางสัญญาณ PUBLISH ซึ่งข้อมูลแสดงสถานะจะมาจากหลายแหล่งข้อมูล จากนั้นจะเข้าสู่ฟังก์ชัน Pre-Compose เพื่อจะรวบรวมข้อมูลแสดงสถานะขั้นต้นก่อน แล้วส่งต่อไปยัง Privacy Filtering ที่ทำหน้าที่ในการคัดกรองข้อมูลที่ผู้แสดงสถานะไม่อยากให้ผู้มองสถานะรับรู้ และส่งต่อไปยังฟังก์ชัน Watcher Filter เพื่อคัดกรองข้อมูลแสดงสถานะที่ผู้มองสถานะต้องการ และสุดท้ายส่งต่อไปยังฟังก์ชัน Post-Composition เพื่อรวบรวมข้อมูลแสดงสถานะที่ได้ผ่านกระบวนการคัดกรองให้เป็นข้อมูลแบบรวมศูนย์

นอกจากกระบวนการ Publication, Notification และ Subscription แล้ว สถาปัตยกรรม SIMPLE ยังมีกระบวนการอื่น ๆ ที่เข้ามาประกอบ เพื่อที่จะทำให้การบริการดังกล่าวสมบูรณ์



รูปที่ 2-5 Presence Configuration [3]

จากรูปที่ 2-5 แสดงให้เห็นว่ามีกระบวนการเพิ่มเติมจากการทำงานพื้นฐานเมื่อเปรียบเทียบกับรูปที่ 2-3 คือ XCAP Server ที่จะทำหน้าในการเก็บข้อมูลบัญชีรายชื่อ (Resource List) และกฎการป้องกัน (Privacy Rule) และ Presence Agent จะมีฟังก์ชันการทำงานเพิ่มเติม คือ Resource List Function ที่จะอนุญาตให้ผู้ใช้ SUBSCRIBE เพียงหนึ่งครั้งไปยัง Resource List Server (RLS) และ RLS จะทำหน้าที่ SUBSCRIBE ตามบัญชีรายชื่อที่ผู้ใช้เก็บไว้ใน XCAP Server ทั้งหมดแทน เพื่อลดการใช้แบนด์วิดท์ระหว่างเครื่องลูกข่ายและเครื่องแม่ข่าย และสามารถที่จะเก็บบัญชีรายชื่อให้ผู้ใช้ เพื่อที่ให้ผู้ใช้สามารถที่จะใช้เครื่องลูกข่ายอื่น ๆ โดยไม่จำเป็นต้องเพิ่มรายชื่อใหม่เสมอ

นอกจากนี้สถาปัตยกรรม SIMPLE ยังมีการทำการรวบรวมข้อมูลจากแหล่งข้อมูลต่าง ๆ (Composition) และการทำ Privacy Controlling ซึ่งทำให้สถาปัตยกรรม SIMPLE มีการบริการที่มีความหลากหลายมากยิ่งขึ้น

2.3.3 การเปรียบเทียบระหว่างสถาปัตยกรรม SIP/SIMPLE และ XMPP/Jabber

จากตารางที่ 2-3 เห็นได้ว่า XMPP สามารถที่จะทำงานได้เทียบเคียงกับ SIP แต่ในการที่ XMPP จะทำงานได้เหมือน SIP จะต้องมีการเพิ่มความสามารถบางอย่างเข้าไป ตัวอย่างเช่น Jingle [19] ที่เพิ่มความสามารถ XMPP ให้ทำงาน Video Chat, Voice Chat และ File Transfer

นอกจากนี้ SIMPLE/SIP ได้ถูกเลือกให้เป็นพื้นฐานของการบริการการแสดงสถานะของระบบโทรศัพท์เคลื่อนที่ยุคที่สาม [15] ซึ่งช่องทางการสื่อสารดังกล่าวมีความจำเป็นในการทำระบบการสื่อสารแบบรวมศูนย์ เนื่องด้วยระบบโทรศัพท์เคลื่อนที่ยุคสาม สามารถที่จะให้บริการการส่งข้อมูลด้วยความเร็วสูง และที่ผู้ใช้สามารถเคลื่อนที่ได้ (Mobility Communication) ทำให้สามารถที่จะติดต่อกับผู้ใช้งานได้ทุกเวลาและทุกสถานที่

อีกประเด็นที่ทำให้ XMPP ไม่ถูกเลือกมาใช้งานในระบบการสื่อสารแบบรวมศูนย์ เพราะในปัจจุบัน SIP นั้นถูกเลือกเป็นโปโตคอลพื้นฐานสำหรับควบคุมและสร้างการบริการสื่อสารรูปแบบต่าง ๆ อยู่แล้ว เช่น VoIP หรือ Voice and Video Conferencing เป็นต้น ซึ่งหากมีการเลือก XMPP ระบบจะต้องสร้างฟังก์ชันหรือเกตเวย์ในการเชื่อมต่อข้อมูล จากความเหตุผลข้างต้นที่กล่าวมาทำให้ระบบ SIP/SIMPLE จึงถูกเลือกให้เป็นพื้นฐานการพัฒนาการบริการแสดงสถานะของระบบการสื่อสารแบบรวมศูนย์

ตารางที่ 2-3 การเปรียบเทียบระหว่าง SIP/SIMPLE และ XMPP/Jabber

	SIP/SIMPLE	**XMPP/Jabber**
การบริการการสื่อสาร	VoIP, Video and Voice Conference, Voice Mail, Presence, Instant Messaging	Presence, Instant Messaging
การบริการอื่น ๆ	Authentication, Authorization, SigComp, etc.	Authorization, Privacy List, Basic Communication Blocking
การทำงานกับระบบโทรศัพท์เคลื่อนที่	SIP/SIMPLE ถูกเลือกเป็นพื้นฐานของระบบ IMS	-
การทำงานร่วมกับระบบอื่น ๆ	Asterisk, 2G phone, PSTN, Web Conference, Web 2.0	SIP, HTTP

2.4 IP Multimedia Subsystem (IMS)

จากคุณสมบัติของการสื่อสารแบบรวมศูนย์ การที่ผู้ใช้สามารถรับการติดต่อได้ในทุก ๆ สถานการณ์ คือ ไม่ว่าจะเป็นสถานที่ใดไม่ว่าจะเป็นเวลาใดและไม่ว่าจะเป็นอุปกรณ์ใด จากหลักการดังกล่าว ทำให้ระบบการสื่อสารแบบรวมศูนย์จำเป็นที่จะต้องรองรับการทำงานแบบเคลื่อนที่ได้ จึงทำให้ระบบมีความจำเป็นต้องทำงานรวมกับเทคโนโลยีของโทรศัพท์เคลื่อนที่ยุคที่สาม ซึ่งเหมาะสมในการทำงานการรับส่งข้อมูลด้วยความเร็วสูง เช่น การประชุมผ่านทางภาพและเสียงเป็นต้น

โดยเทคโนโลยีของโทรศัพท์เคลื่อนที่ยุคที่สาม มีระบบที่สนับสนุนการทำงานในการพัฒนาการให้บริการด้านมัลติมีเดีย คือ IMS ซึ่งได้อธิบายรายละเอียดในหัวข้อถัดไป ถึงที่มาของเทคโนโลยี และสถาปัตยกรรมพื้นฐานของระบบ

2.4.1 ที่มาของระบบ IMS

ก่อนที่จะมีระบบ IMS ตลาดของ Circuit-Switched Voice เป็นรายได้เดียวที่ผู้บริการสามารถทำกำไรได้ แต่เนื่องด้วยเกิดการแข่งขันที่สูงขึ้น ทำให้ผู้ให้บริการยากที่จะทำกำไรจากการให้บริการดังกล่าวเพียงอย่างเดียว และการให้บริการของ Packet-Switched ยังไม่เป็นที่สนใจต่อผู้ใช้ เนื่องจากยังไม่มีการบริการที่น่าสนใจ ทำให้การใช้ในส่วนของ Packet-Switched จึงยังไม่สามารถที่จะทำกำไรให้กับผู้ให้บริการได้มากเพียงพอ

จากนั้นผู้ให้บริการจึงคิดวิธีการให้บริการ Packet-Switched ที่มีความน่าสนใจให้แก่ผู้ใช้ เพื่อที่จะให้ผู้ใช้มีความสนใจที่จะใช้บริการมากขึ้น และการให้บริการอินเตอร์เน็ตแบบ

เคลื่อนที่ก็เป็นที่สนใจกับผู้ใช้ จึงทำให้เกิดระบบ IMS ขึ้นมา โดยการออกแบบระบบ IMS เพื่อทำให้ระบบอินเตอร์เน็ตสามารถที่จะอยู่บนพื้นฐานของโทรศัพท์มือถือได้ โดยเป้าหมายของ IMS มีดังหัวข้อดังนี้

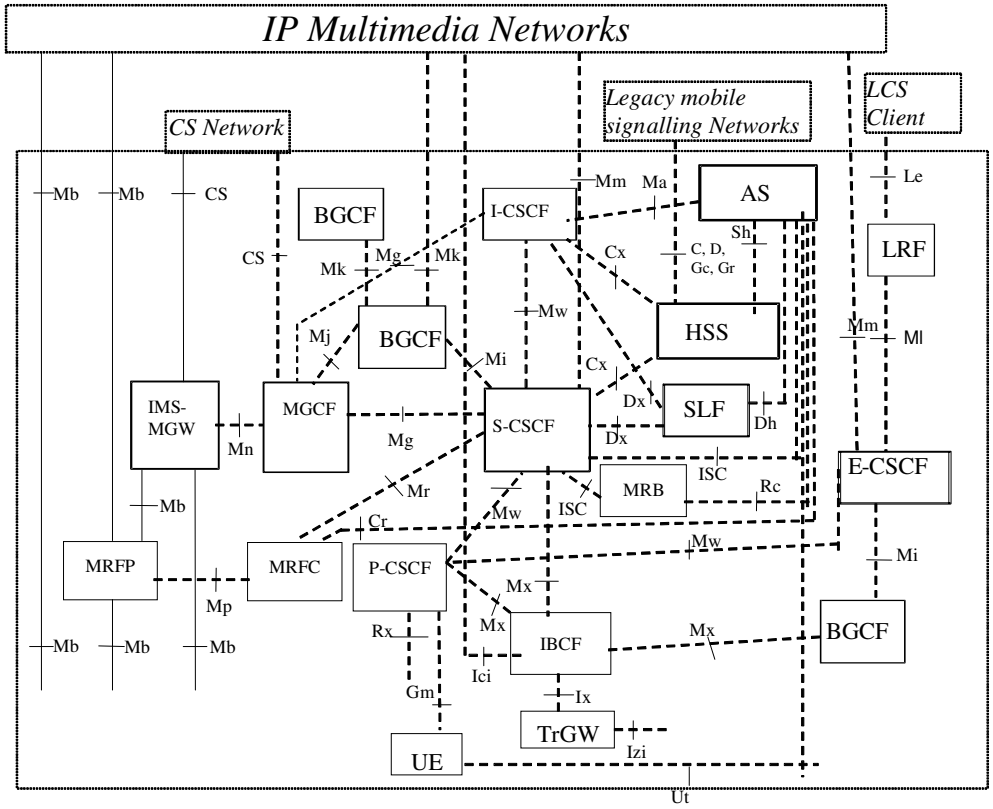
1. รวบรวมเทคโนโลยีต่าง ๆ ที่มีเข้ามาเป็นระบบเดียว
2. ทำให้อินเตอร์เน็ตแบบเคลื่อนที่เป็นจริง
3. สร้างสถาปัตยกรรมพื้นฐานสำหรับนักพัฒนา เพื่อให้นักพัฒนาสามารถที่จะใช้ระบบ IMS เป็นพื้นฐานในการสร้าการบริการใหม่ ๆ ในระบบ IMS
4. เพื่อทำกำไรจากการทำงาน Packet-Switched ให้มากยิ่งขึ้น

เมื่อมองถึงเป้าหมายของ IMS ที่กำหนดไว้ข้างต้น ทำให้ได้แนวทางการออกแบบกรอบการทำงานของ IMS [11] จำเป็นต้องมีความต้องการดังนี้

1. สนับสนุนการสร้าง IP Multimedia Session
2. สนับสนุนกระบวนการทำงานในการต่อรองสำหรับ Quality of Service
3. สนับสนุนการทำงานระหว่างสองระบบ คือ Internet และระบบ Circuit-Switched
4. สนับสนุนการทำ Roaming
5. สนับสนุนผู้ให้บริการสามารถที่จะควบคุมข้อกำหนดต่าง ๆ ที่ตกลงไว้กับผู้ใช้งานระบบได้อย่างดี
6. สนับสนุนการให้สร้างการบริการใหม่ ๆ ที่รวดเร็ว โดยไม่จำเป็นต้องมีมาตรฐานรองรับ
7. สนับสนุนการเข้าถึงระบบ IMS จากระบบเครือข่ายอื่น ๆ เช่น GPRS WLAN และ DSL เป็นต้น (Release 6)

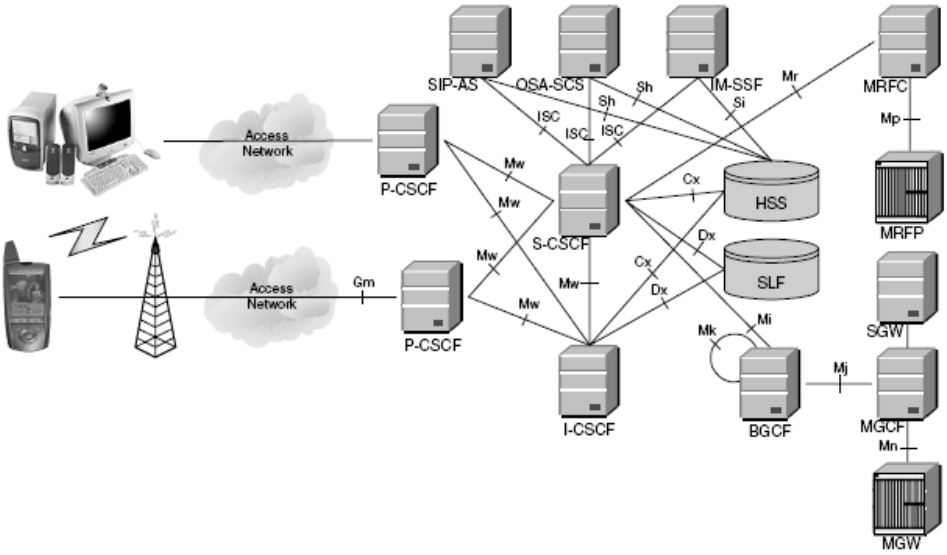
2.4.2 ภาพรวมของสถาปัตยกรรม IMS

ในหัวข้อนี้จะอธิบายถึงสถาปัตยกรรมและองค์ประกอบที่สำคัญของ IMS เพื่อแสดงให้เห็นการเชื่อมต่อระหว่างส่วนต่าง ๆภายในระบบ IMS



รูปที่ 2-6 Configuration of IM Subsystem Entities [20]

โดยจากรูปที่ 2-6 เป็นมาตรฐานที่กำหนดขึ้นโดย 3GPP เป็นฟังก์ชันมาตรฐานและการเชื่อมต่อระหว่างฟังก์ชันต่าง ๆ ทำให้ผู้พัฒนาสามารถที่จะรวบรวมหลาย ๆ ฟังก์ชันมาเป็นหนึ่งเครื่องแม่ข่าย หรือแบ่งหนึ่งฟังก์ชันเป็นหลาย ๆ เครื่องแม่ข่ายก็ได้ โดยทั่วไปหนึ่งฟังก์ชันจากรูปที่ 2-6 ที่ได้จากมาตรฐานของ 3GGP เกือบทั้งหมดจะสร้างเป็นหนึ่งเครื่องแม่ข่ายของ IMS แต่มีบางฟังก์ชันที่จำเป็นต้องแบ่งเป็นหลายเครื่องแม่ข่าย เนื่องด้วยจำเป็นต้องรองรับการใช้งานจากผู้ใช้จำนวนมาก เช่น ระบบฐานข้อมูลผู้ใช้ เมื่อพิจารณาจากฟังก์ชันพื้นฐาน สามารถเขียนรูปสถาปัตยกรรมของ IMS ขึ้นมาใหม่ในลักษณะของเครื่องแม่ข่ายและในส่วนของสัญญาณการเชื่อมต่อจะได้เป็นรูปที่ 2-7



รูปที่ 2-7 3GPP IMS Architecture Overview [21]

จากรูปที่ 2-7 ทางซ้ายจะเป็นเครื่องลูกข่าย IMS ซึ่งถ้าเป็นโทรศัพท์เคลื่อนที่จะมีคำศัพท์เฉพาะที่ใช้เรียกว่า User Equipment (UE) โดยจะเข้าใช้งานระบบ IMS ผ่านทาง Radio Link และนอกจาก UE แล้ว IMS ยังสนับสนุนการทำงานของเครื่องลูกข่ายแบบอื่น ๆ ด้วย เช่น PDA หรือคอมพิวเตอร์ส่วนบุคคลในการเชื่อมต่อกับระบบ IMS โดยอุปกรณ์ดังกล่าวจำเป็นต้องเชื่อมต่อผ่าน WLAN หรือ ADSL นอกจากนี้แล้วสถาปัตยกรรมของ IMS จะประกอบด้วยส่วนการทำงานหลัก ๆ ดังนี้

2.4.2.1 ฐานข้อมูลผู้ใช้ (HSS และ SLF)

HSS (Home Subscription Server) เป็นศูนย์กลางในการเก็บข้อมูลของผู้ใช้ โดยได้พัฒนาจาก HLR (Home Location Register) ของระบบ GSM โดย HSS จะเก็บข้อมูลทั้งหมดของผู้ใช้ โดยระบบ IMS อาจจะมี HSS ได้มากกว่าหนึ่งตัวได้ ในกรณีที่มีจำนวนผู้ใช้จำนวนมาก แต่จำเป็นต้องมี SLF (Subscriber Locator Function) ซึ่งเป็นฐานข้อมูลอย่างง่าย ที่ใช้ในการบ่งบอกที่อยู่ของข้อมูลหลักของผู้ใช้ใน HSS โดยทั้ง HSS และ SLF มีโปรโตคอลพื้นฐานในการเก็บและดึงข้อมูล คือ โปรโตคอล Diameter

2.4.2.2 Call/Session Control Function

CSCF (Call/Session Control Function) เป็นเครื่องแม่ข่าย SIP และทำหน้าที่ในการประมวลผลสัญญาณ SIP ใน IMS โดยแบ่งออกเป็น 3 ประเภท โดยแบ่งตามลักษณะฟังก์ชันที่มีในแต่ละ CSCF แบ่งได้ดังนี้

1. P-CSCF (Proxy-CSCF)

เป็นส่วนเชื่อมต่อในส่วนสัญญาณระหว่างระบบ IMS กับเครื่องลูกข่ายของ IMS โดย P-CSCF จะมีหน้าที่เป็น Outbound/Inbound SIP Proxy โดยทุก ๆ สัญญาณ SIP ที่ร้องขอมาจะผ่าน P-CSCF ทั้งหมด ซึ่ง P-CSCF จะส่งและตอบสนองสัญญาณ SIP ต่อไปยังส่วนประกอบอื่น ๆ ในระบบ IMS หรือกลับไปยังทิศทางของเครื่องลูกข่าย IMS

โดย P-CSCF ยังมีหน้าที่ในการตรวจสอบสัญญาณ SIP ว่าถูกต้องหรือไม่ ถ้าเป็นสัญญาณที่ไม่ตรงตามกฎของ SIP ก็จะไม่อนุญาตให้สัญญาณดังกล่าวเข้าสู่ระบบ IMS นอกจากนี้ P-CSCF ยังมีความสามารถในการบีบอัดข้อมูลให้มีขนาดเล็กเพื่อลดแบนด์วิดท์ในการทำงานระหว่างระบบ IMS และเครื่องลูกข่าย

ในการเข้าถึง P-CSCF จะสามารถทำได้โดยผ่านกระบวนการ DNS เนื่องด้วย P-CSCF จะมี URI ในการเข้าถึง ตัวอย่างเช่น “pcscf.open-ims.cnr.psu.ac.th” เมื่อเครื่องลูกข่ายที่ต้องการจะติดต่อไปยังระบบ IMS ก็จะนำเอา URI ของ P-CSCF มา Resolve กับ DNS ก็จะได้เป็น IP-Address ที่ DNS เก็บเอาไว้ และใช้ IP-Address ดังกล่าวในการติดต่อมายังระบบ IMS

2. I-CSCF (Interrogating-CSCF)

เป็นเครื่องแม่ข่าย SIP ที่ทำหน้าที่คล้ายกับ P-CSCF แต่ I-CSCF จะถูกใช้งานในตอนเริ่ม Session เท่านั้น โดยจะทำหน้าที่ในการตรวจสอบว่าเครื่องลูกข่ายได้รับอนุญาตให้ใช้งานระบบ IMS หรือไม่ โดย I-CSCF จะใช้ข้อมูลที่ได้จาก HSS ในการตัดสินใจ และถ้าเครื่องลูกข่ายได้รับอนุญาตก็จะส่งต่อสัญญาณไปให้ S-CSCF ต่อไป

โดยการเข้าถึง I-CSCF จะเหมือนกับกระบวนการของ P-CSCF แต่ของ I-CSCF จะอยู่ในส่วนของ Home Network เท่านั้น

3. S-CSCF (Service-CSCF)

เป็นเครื่องแม่ข่ายกลางที่ใช้ในการควบคุม Session โดยจะมีบทบาทคล้าย SIP Registrar โดยฟังก์ชันหลักของ S-CSCF จะประกอบด้วยสองส่วนหลัก ๆ คือ IFC (Initial Filter Criteria) และ Registration

ฟังก์ชัน Registration ทำหน้าที่ในการ Binding ที่อยู่ของผู้ใช้ (SIP URI) กับ IP-Address ของเครื่องลูกข่าย เนื่องจากผู้ใช้จะใช้ SIP URI ในการลงทะเบียนเข้าสู่ระบบ โดย S-CSCF จะจำ IP-Address ของเครื่องลูกข่ายที่ผู้ใช้ใช้ในการลงทะเบียน เพื่อที่จะสามารถส่งสัญญาณหรือข้อมูลไปยังผู้ใช้ได้ถูกต้อง

ฟังก์ชัน IFC ทำหน้าที่ในการตัดสินใจสัญญาณ SIP ที่ผ่านมาถึง S-CSCF ว่าควรจะส่งต่อยัง Application Server ใดในระบบ IMS โดยจะพิจารณาจากข้อมูลของผู้ใช้ใน HSS ว่าได้รับอนุญาตให้ใช้ Application Server ใดบ้าง และจะพิจารณาว่าผู้ให้บริการตั้งค่าว่า SIP สัญญาณดังกล่าวควรที่จะส่งต่อไปยัง Application Server ใด โดยการเข้าถึง S-CSCF จะเหมือนกับกระบวนการของ I-CSCF

2.4.2.3 Application Server

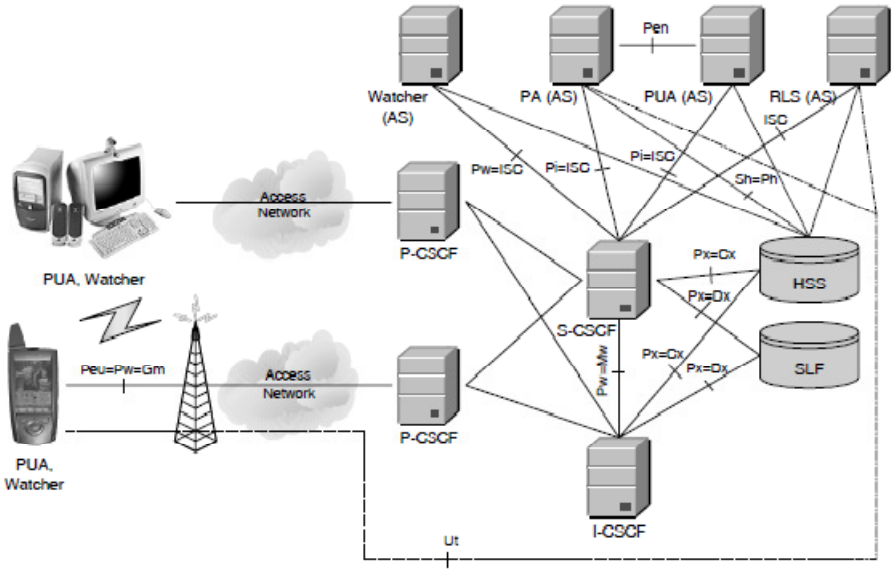
AS (Application Server) เป็นส่วนประกอบหนึ่งของระบบ IMS ที่ใช้ในการจัดการสัญญาณ SIP ที่มาจาก S-CSCF โดยจะเป็นจุดสุดท้ายของการให้บริการ ตัวอย่างของ Application Server เช่น Conference Service, IPTV, Presence Service และ RLS เป็นต้น

2.4.2.4 Media Resource Function

MRF (Media Resource Function) ใช้ในการบริการการส่งข้อมูลมัลติมีเดียของระบบ IMS โดย MRF แบ่งเป็น 2 ส่วน คือ ส่วนควบคุม และส่วนการส่งข้อมูล โดยส่วนควบคุมจะเรียกว่า MRFC (Media Resource Function Controller) ทำหน้าที่คล้ายกับ UA ใน SIP เพื่อจัดการสัญญาณ SIP ที่ S-CSCF หรือ Application Server ที่ส่งมาควบคุม MRF และส่วนที่สองคือ MRFP (Media Resource Function Processor) ใช้ในการจัดสรรข้อมูลมัลติมีเดียให้กับระบบ IMS โดยตัว MRFC จะควบคุม MRFP โดยใช้มาตรฐาน H.248

2.4.3 สถาปัตยกรรมระบบแสดงสถานะของ IMS

3GPP ได้นิยามการทำงานของการบริการแสดงสถานะของ IMS ไว้ในเอกสารอ้างอิง 3GPP TS 23.141 [22] โดยจะมีเพิ่มเติมความสามารถให้ Application Server ของ IMS



รูปที่ 2-8 SIP based Presence Architecture in IMS [21]

จากรูปที่ 2-8 ได้เขียนรูป Presence IMS Architecture ขึ้นมาใหม่ เพื่อให้เข้ากับฟังก์ชันที่ได้ระบุไว้ในเอกสารอ้างอิงจาก 3GPP TS 23.141 ของ IMS โดยรายละเอียดของรูปที่ 2-8 ได้มีการเพิ่มเติมการทำงานใน Application Server โดยอธิบายไว้ในหัวข้อดังนี้

1. Watcher คือ เครื่องแม่ข่ายที่ทำหน้าที่เป็นผู้ที่ต้องการจะรับรู้ข้อมูลแสดงสถานะ โดยในที่นี้จะใช้การ SUBSCRIBE ไปเพื่อขอข้อมูลสถานะจาก Presence Server และจะใช้ Diameter โปรโตคอลในการดึงข้อมูลการการใช้งานจากอุปกรณ์ที่ไม่สามารถจะแสดงสถานะตัวเองได้ เช่น PSTN, GSM Phone เป็นต้น
2. PUA (Presence User Agent) เครื่องแม่ข่ายในการดึงข้อมูลจากฐานข้อมูลและแหล่งต่าง ๆ เช่น ระบบ Circuit-Switched ที่ไม่สามารถจะแสดงข้อมูลมายัง Presence Server ได้ แล้วทำการ PUBLISH ข้อมูลดังกล่าวไปยัง Presence Server แทน
3. PA (Presence Agent) เป็น Application Server ที่อยู่ใน Home Network ทำหน้าที่คล้ายกับ Presence Server ในระบบ SIP ปกติ
4. RLS (Resource List Server [23]) เป็น Application Server โดยทำหน้าที่เพื่อลดแบนด์วิดท์ระหว่างเครื่องลูกข่ายกับระบบ IMS ในการทำกระบวนการ Subscription โดยเครื่องลูกข่ายไม่จำเป็นต้อง SUBSCRIBE ที่ละคนในบัญชีรายชื่อของที่เก็บไว้ใน XCAP Server แต่ RLS จะทำหน้าที่แทนผู้ใช้ในการ SUBSCRIBE และในการรวบรวม NOTIFY ของทั้งหมดส่งมาให้เครื่องลูกข่าย

ในส่วนต่อมา คือ การเชื่อมต่อ (Interface) ต่างๆ ในระบบ 3GPP จะระบุเป็นตัวอักษรสั้นๆ 2-3 ตัว โดยในที่นี้จะใช้ตัวอักษรขึ้นต้นด้วย “P” ซึ่งใช้สื่อความหมายถึงคำว่า Presence โดยอธิบายรายละเอียดแต่ละการเชื่อมต่อไว้ดังนี้

1. Pw เป็นส่วนการเชื่อมต่อระหว่าง Watcher กับ S-CSCF โดยจะใช้ SIP
2. Pi เป็นส่วนการเชื่อมต่อระหว่าง PUA กับ S-CSCF โดยจะใช้ SIP
3. Ph เป็นส่วนการเชื่อมต่อระหว่าง PA กับ HSS โดยจะใช้ Diameter
4. Px เป็นส่วนการเชื่อมต่อระหว่าง HSS และ SLF กับ S-CSCF โดยจะใช้ Diameter
5. Pen เป็นส่วนการเชื่อมต่อระหว่าง PUA กับ PA โดยจะใช้ SIP

2.4.4 การทำงานร่วมกับสถาปัตยกรรมของ IMS

เนื่องจากระบบการสื่อสารแบบรวมศูนย์ทำงานอยู่ในระบบที่เป็น Pure SIP ทำให้ต้องมองระบบ IMS เป็นระบบที่แยกออกไปจากระบบการสื่อสารแบบรวมศูนย์ โดยมาตรฐานของ 3GPP ได้พูดถึงการเชื่อมต่อระหว่าง IMS และ SIP Multimedia Network [26] [27] [28] เป็น Gateway ที่อยู่ในฝั่งของระบบ IMS เป็นตัวเชื่อมต่อ โดยแบ่งการทำงานเป็น 2 ส่วนคือ ส่วนของ Transport Layer ที่ต้องมี Interconnection Border Gateway Function (IBGF [29]) และส่วนของ Signaling Layer ต้องมีฟังก์ชัน Interconnection Border Control Function โดยมีพิจารณาการทำงานของการบริการแสดงสถานะแบบรวมศูนย์ นั้นเป็นการทำงานที่จะอยู่ในส่วน Signaling Layer จึงทำให้การทำงานของฟังก์ชัน IBCF ถูกนำมาวิเคราะห์ได้ดังนี้

IBCF เป็นฟังก์ชันที่ใช้ในฝั่งของ Control Plane โดยมีฟังก์ชันในการทำงานดังนี้

1. IMS-Application Level Gateway (ALG) [29]
2. Network Configuration Hiding (THIG) [30] [31]
3. Transport Plane Control (QoS) [32]
4. IWF [29]
5. Screening of SIP Signaling [29]

โดยมาตรฐานการทำงานของ IBCF จะอยู่ในเอกสารอ้างอิงจาก 3GPP และ TISPAN โดยสามารถสรุปรวมได้ดังตารางที่ 2-4 โดย IBCF คือเป็นจุดที่คอยรับสัญญาณ SIP ที่เข้าสู่ระบบ IMS จากระบบ อื่นๆ โดยข้อมูล IP Address และ Port ของ IBCF จะถูกเก็บไว้ใน DNS โดยเมื่อสัญญาณ SIP ที่มาจากระบบอื่นๆ สามารถที่จะ Resolve ที่อยู่ของ IBCF ได้

เมื่อสัญญาณ SIP เข้ามาสู่ IBCF โดยถ้าเป็นสัญญาณ SIP ดังกล่าวเป็น Session ใหม่ IBCF จะส่งต่อสัญญาณ SIP ดังกล่าวไปยัง I-CSCF ในส่วนสัญญาณขาออก (Outgoing) IBCF จะทำหน้าที่ในการเชื่อมต่อไปยังระบบที่ต้องการ

ตารางที่ 2-4 IBCF Related Specification [15]

Placement of the IBCF in IMS Architecture: ES 282 001 , ES 282 007
Reference Point: ES 282 007 , TS 183 021
Concept of Border Control: TS 23.228
ALG Functionality: TS 183 021 , TS 24.229
THIG Functionality: TS 24.229
Transport Plane Control: TS 183 017
IWF: TS 23.228 , TS 24.229
SIP Screening: TS 23.228

บทที่ 3
ทบทวนวรรณกรรม

บทนี้เป็นการการทบทวนวรรณกรรมที่เกี่ยวข้องกับการจัดทำวิทยานิพนธ์ โดยวรรณกรรมที่ได้ทำการตรวจสอบจะเกี่ยวข้องกับการออกแบบและสร้างการบริการการแสดงสถานะ (Presence Service) ในระบบอินเตอร์เน็ตและระบบการสื่อสารอื่น ๆ เช่น ระบบโทรศัพท์เคลื่อนที่ยุคที่สาม ระบบ Social Network เป็นต้น หรือเอกสารที่มีความใกล้เคียง ซึ่งสามารถที่จะอธิบายโดยแบ่งได้ออกเป็น 2 หัวข้อ คือ การรวบรวมข้อมูลแสดงสถานะ (Aggregation) ให้เป็นข้อมูลสถานะแบบรวมศูนย์ หัวข้อถัดมา คือ การกำหนดสิทธิการเข้าถึงและการแสดงข้อมูลแสดงถานะ โดยในแต่ละหัวข้อจะแสดงถึงข้อดีและข้อเสีย เพื่อทำให้เห็นถึงแนวทางที่เหมาะสมสำหรับการออกแบบกรอบการทำงานและสถาปัตยกรรมของระบบแสดงสถานะแบบรวมศูนย์ที่ดี

3.1 การรวบรวมข้อมูลแสดงสถานะจากช่องทางการสื่อสารต่าง ๆ ให้เป็นข้อมูลแสดงสถานะแบบรวมศูนย์ (Presence Information Aggregation)

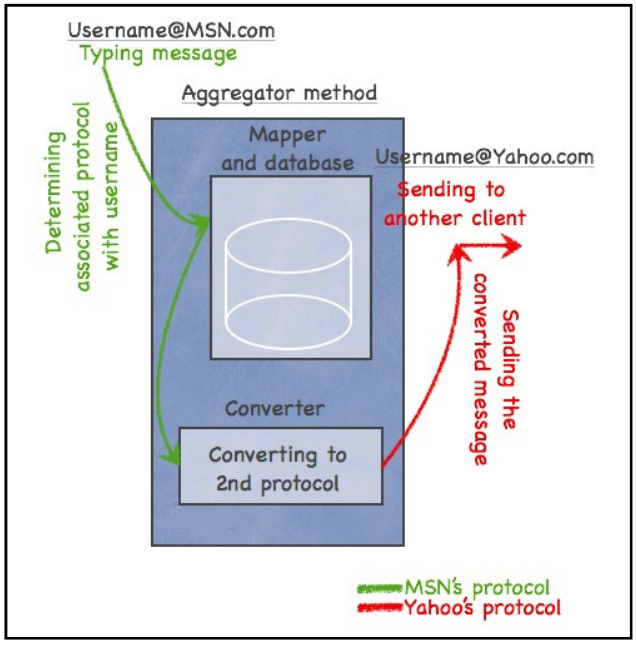
ฟังก์ชันรวบรวมข้อมูลแสดงสถานะให้เป็นข้อมูลแสดงสถานะแบบรวมศูนย์ คือฟังก์ชันที่ใช้ในการรวบรวมข้อมูลแสดงสถานะจากหลาย ๆ แหล่งข้อมูล เช่น จากอุปกรณ์สื่อสารต่าง ๆ จากโปรแกรมตัวช่วย (Collaboration) ให้เป็นข้อมูลแสดงสถานะแบบรวมศูนย์ ซึ่งเอกสารแต่ละฉบับ [4] [5] [6] จะมีกระบวนการรวบรวม การแก้ไขความขัดแย้งของข้อมูลและทำให้เป็นข้อมูลแสดงสถานะแบบรวมศูนย์ที่แตกต่างกัน โดยสามารถแบ่งฟังก์ชันการรวบรวมได้ 2 แบบ คือ การรวบรวมข้อมูลแสดงสถานะที่เครื่องลูกข่าย และการรวบรวมข้อมูลแสดงสถานะที่เครื่องแม่ข่าย

3.1.1 การรวบรวมข้อมูลแสดงสถานะที่เครื่องลูกข่าย

เอกสาร [5] เสนอแนวคิดในการสร้างให้ Aggregator Instant Messenger เพื่อที่จะเป็นตัวรวบรวม (Aggregator) ข้อมูลการแสดงสถานะของเมสเซนเจอร์ต่าง ๆ ให้เป็นหนึ่งเดียว โดยแสดงผลผ่านโปรแกรมเมสเซนเจอร์เดียวกัน แต่ในการติดต่อสื่อสารกันระหว่างเมสเซนเจอร์ต่าง ๆ (MSN, AOL, Gtalk เป็นต้น) ยังคงใช้โปรโตคอลของแต่ละเมสเซนเจอร์เหมือนเดิม

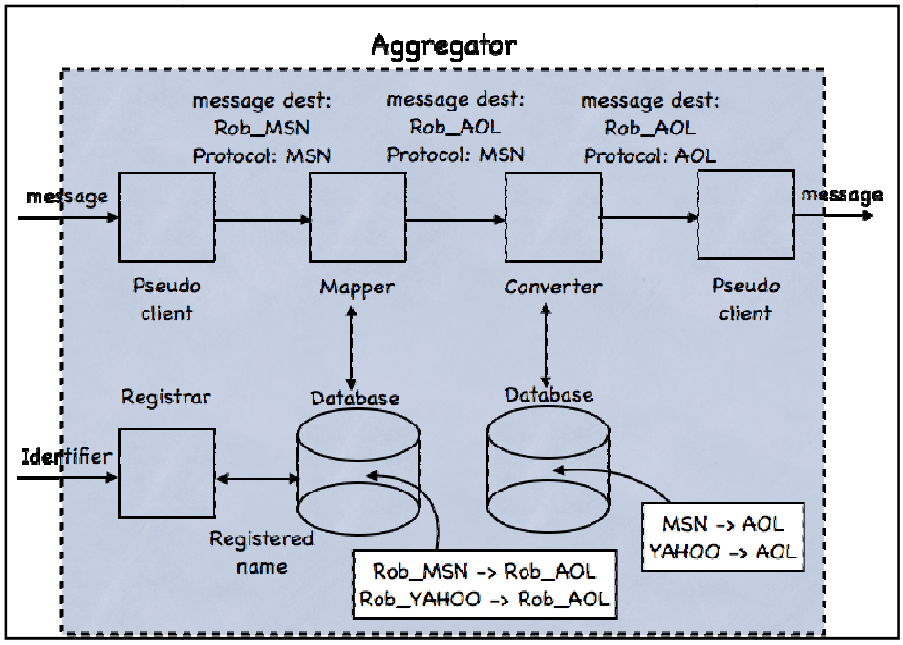
ต่อมาเอกสาร [5] เสนอ Universal Instant Messenger ที่จะสามารถรวบรวมข้อมูลการแสดงสถานะที่เครื่องลูกข่าย และสามารถติดต่อสื่อสารข้ามไปยังเมสเซนเจอร์ที่มี

โปรโตคอลแตกต่างกันได้ โดยไม่จำเป็นต้องลงทะเบียนทุก ๆ ระบบเมสเซนเจอร์ โดย Universal Instant Messenger จะทำการแปลงจากโปรโตคอลต้นทางให้เป็นโปรโตคอลปลายทางโดยฟังก์ชัน Converting ที่มีพื้นฐานอยู่บนการ Mapping Database โดยเมื่อวาดรูปจากแนวคิดของเอกสาร [5] สามารถเขียนรูปสถาปัตยกรรมของ Universal Instant Messenger ได้ดังรูปที่ 3-1



รูปที่ 3-1 สถาปัตยกรรมของ Universal Instant Messenger

จากรูปที่ 3-1 ได้แสดงถึงสถาปัตยกรรมของ Universal Instant Messenger โดยประกอบจาก 2 ส่วนหลัก ๆ คือ Mapper and Database และ Converter โดย Database จะเป็นฐานข้อมูลที่ใช้ในการเก็บความสัมพันธ์ระหว่างรายชื่อกับโปรโตคอลต่าง ๆ ในส่วนต่อมา คือ Mapper จะทำหน้าที่ในการเปลี่ยนรายชื่อของผู้ใช้จากระบบเมสเซนเจอร์ต้นทางเป็นรายชื่ออีกระบบเมสเซนเจอร์ปลายทาง โดยจะใช้ข้อมูลความสัมพันธ์จากฐานข้อมูลที่มีการกำหนดไว้ก่อน และในส่วนสุดท้าย คือ Converter จะทำหน้าที่ในการเปลี่ยนจากโปรโตคอลต้นทางเป็นโปรโตคอลปลายทาง โดยใช้ข้อมูลความสัมพันธ์จากฐานข้อมูลเช่นกัน โดยลำดับสัญญาณที่ผ่านส่วนต่าง ๆ ภายในระบบ Universal Instant Messenger เป็นดังรูปที่ 3-2



รูปที่ 3-2 การทำงานภายในของ Aggregator และแสดงการการเปลี่ยนจากโปรโตคอล MSN ไปเป็นโปรโตคอล AOL [5]

จากรูปที่ 3-2 จะเป็นกรณีตัวอย่าง โดย Mike ต้องการติดต่อไปยัง Rob เริ่มต้นโดย Mike ทำการ Identifier ตัวเองและสร้างข้อมูลความสัมพันธ์ของตัวเองที่มีกับผู้ใช้คนอื่น ๆ ไว้ในฐานข้อมูล เพื่อใช้ในขั้นตอนของการ Mapper และ Converter จากนั้นเมื่อ Mike เริ่มการติดต่อไปยัง Rob สัญญาณก็จะส่งไป Pseudo Client ที่จะทำหน้าในการเชื่อมต่อผู้ใช้กับ Aggregator และ Pseudo Client จะส่งสัญญาณต่อไปยัง Mapper ฟังก์ชัน ซึ่งพบว่าสัญญาณดังกล่าวต้องการติดต่อไปยัง Rob ซึ่งในขณะนั้น Mike จะเห็น Rob เป็นผู้ใช้ใน MSN เช่นเดียวกับเขา ต่อมา Mapper ฟังก์ชันก็จะตรวจสอบว่า Rob_MSN จะต้องเปลี่ยนเป็นปลายทางใด ซึ่งในที่นี้จะต้องเปลี่ยนเป็น AOL จากความสัมพันธ์ที่ Mike ได้สร้างไว้ในฐานข้อมูล ซึ่งจะเห็นได้ว่าต้องเปลี่ยนปลายทางเป็น Rob_AOL และส่งสัญญาณต่อมาให้ Converter เพื่อทำการเปลี่ยนจากโปรโตคอล MSN ให้เป็น AOL จากข้อมูลความสัมพันธ์ในฐานข้อมูล และส่งสัญญาณไปให้ Pseudo Client เพื่อที่ส่งไปสัญญาณดังกล่าวไปให้ Rob ต่อไป

ข้อดีที่ได้จากเอกสาร [5] คือ การเสนอการรวบรวมข้อมูลการแสดงสถานะจากผู้ให้บริการเมสเซนเจอร์ต่าง ๆ ไว้ที่เครื่องลูกข่าย โดยข้อดีสามารถแบ่งได้เป็น 2 ข้อ คือ เครื่องแม่ข่ายกลางของระบบไม่จำเป็นต้องเก็บข้อมูลของผู้ใช้ เช่น User Account และ Password เอาไว้เพื่อที่จะใช้ในการร้องขอข้อมูลแสดงสถานะไปยังผู้ให้บริการเมสเซนเจอร์ต่าง ๆ ทำให้ผู้ใช้ไม่จำเป็นต้องกังวลถึงความปลอดภัยของข้อมูลดังกล่าว

ข้อดีข้อที่สอง คือ ระบบ Universal Instant Messenger มีฟังก์ชันรวบรวมข้อมูลแสดงสถานะที่เครื่องลูกข่ายสำหรับการรวบรวมข้อมูลที่มาจากผู้ให้บริการต่าง ๆ จะทำให้เครื่องแม่ข่ายกลางลดภาระการทำงานในการเชื่อมต่อไปยังผู้ให้บริการเมสเซนเจอร์อื่น ๆ ได้ โดยพิจารณาได้จากสมการดังนี้

ให้ Y1 = จำนวนที่เครื่องแม่ข่ายกลางสร้างการเชื่อมต่อในกรณีที่มีการ Aggregator ที่เครื่องแม่ข่ายกลาง

Y2 = จำนวนที่เครื่องแม่ข่ายกลางสร้างการเชื่อมต่อในกรณีที่มีการ Aggregator ที่เครื่องลูกข่าย

n = เป็นค่าเฉลี่ยของจำนวนบัญชีรายชื่อเมสเซนเจอร์ต่าง ๆ ต่อผู้ใช้คนหนึ่ง

X = จำนวนผู้ใช้ของระบบทั้งหมด

จะได้ว่า $Y1 = (1+n)X$ ---------- (1)

$Y1 = X$ ---------- (2)

(1)/(2) $Y1/Y2 = (1+n)$

$Y1 = (1+n)Y2$

จากการพิจาณาสมการดังกล่าวฟังก์ชันรวบรวมข้อมูลแสดงสถานะที่เครื่องแม่ข่ายกลางมีการสร้างการเชื่อมต่อมากกว่าวิธีการรวบรวมข้อมูลแสดงสถานะที่เครื่องลูกข่ายถึง 1+n เท่า ซึ่งเมื่อพิจารณาว่ายิ่งมีจำนวนผู้ใช้มากขึ้นเท่าไรก็จะทำให้เครื่องแม่ข่ายกลางต้องรองรับการเชื่อมต่อเป็นจำนวนมากตามไปด้วย ด้วยเหตุนี้จึงทำให้การรวบรวมข้อมูลแสดงสถานะที่เครื่องลูกข่ายมีความเหมาะสมกว่าในการรวบรวมข้อมูลแสดงสถานะที่มีโปรโตคอลที่แตกต่างกันหรือมีกระบวนการเข้าถึงข้อมูลที่แตกต่างกัน

3.1.2 การรวบรวมข้อมูลแสดงสถานะที่เครื่องแม่ข่ายตามกฎที่ผู้ใช้กำหนดขึ้น

ในเอกสาร [4] ได้เสนอการแก้ไขและจัดการกับปัญหาข้อมูลแสดงสถานะที่มาจากแหล่งข้อมูลต่าง ๆ ในระบบ SIP ซึ่งทำให้เกิดปัญหาความขัดแย้งของข้อมูลจากแหล่งต่าง ๆ โดยจะส่งผลกระทบต่อผู้ที่ต้องการรับรู้สถานะ ทำให้เกิดความสับสนและไม่สามารถที่จะรับข้อมูลแสดงสถานะที่เป็นประโยชน์ในการตัดสินใจในการเลือกช่องทางในการติดต่อไปยังปลายทาง

เอกสาร [4] ได้นำเสนอวิธีประกอบข้อมูลแสดงสถานะขึ้นมาใหม่ (Composition Presence Information) เพื่อแก้ปัญหาข้อมูลที่เกิดความขัดแย้ง โดยที่ผู้ใช้สามารถระบุรูปแบบของการประกอบข้อมูลดังกล่าวด้วยตัวเอง โดยข้อมูลที่ใช้ในการรวบรวมจะเป็นข้อมูลแสดงสถานะของ SIP และจะรวบรวมข้อมูลการแสดงสถานะที่ Presence Server โดยมีฟังก์ชันรวบรวม (Compositor) คอยรวบรวมข้อมูลแสดงสถานะจากแหล่งต่าง ๆ เพื่อเป็นรูปแบบสำหรับผู้ใช้ โดย

รูปแบบของข้อมูลจะลบข้อมูลที่ขัดแย้งกันหรือข้อมูลที่ไม่เกี่ยวข้อง และสร้างข้อมูลการแสดงสถานะใหม่ไปยังผู้ใช้ นอกจากนี้ยังกำหนดวิธีการเข้าถึงและการสร้างกฎในการรวบรวมข้อมูลแสดงสถานะของผู้ใช้ โดยผ่านรูปแบบของ XML อย่างง่าย

```
<discard>
  <old-tuples \

      age="00:30:00.000" />
  <tuples-with-closed-contacts />
</discard>
<derive>
   <add sel= \
      '//person[place-type/car]'>
     <activities>
       <driving>
     </activities>
   </add>
</derive>
<resolve-conflicts>
   <conflict \
        element="activities">
     <merge />
   </conflict>
   <conflict element="place-type">
     <other-attribute \
       attribute=\
          'person/user-input'>
       <value>active</value>
       <value>idle</value>
     </other-attribute>
     <source-precedence>
       <source>
          reported current
       </source>
       <source>
          reported scheduled
       </source>
     </source-precedence>
   </conflict-element>
</resolve-conflicts>
```

รูปที่ 3-3 รูปแบบ XML ของกฎที่ใช้ในการรวบรวมข้อมูลของนาย A จากกรณีตัวอย่าง [4]

เอกสาร [4] ได้ยกตัวอย่างกรณีการใช้งาน คือผู้ใช้ A เป็นผู้แสดงสถานะของระบบ โดยผู้ใช้ A กำลังนั่งอยู่ในห้องทำงาน และกำลังรอที่จะเข้าประชุม และในขณะนั้นมีโทรศัพท์โทรเข้ามา ซึ่งเป็นการโทรที่สำคัญที่ผู้ใช้ A จำเป็นต้องรับสาย และทำให้ผู้ใช้ A จำเป็นต้องออกไปจากที่ทำงาน ผู้ใช้ A จึงได้บอกผู้เข้าร่วมประชุมคนอื่น จากนั้นก็ได้รีบออกจากสำนักงาน โดยไม่ได้ปิดเครื่องคอมพิวเตอร์ และผู้ใช้ A ได้ขับรถออกจากสำนักงาน

จากกรณีศึกษานี้จะเห็นได้ว่า ผู้ที่มองสถานะของผู้ใช้ A จะเห็นที่อยู่ของผู้ใช้ A ใน 3 สถานที่ คือ คอมพิวเตอร์ของผู้ใช้ A บอกว่ากำลังอยู่ที่ห้องทำงาน โทรศัพท์เคลื่อนที่ของผู้ใช้ A ระบุว่าอยู่ในรถ และสุดท้ายตารางเวลาของผู้ใช้ A ระบุว่ากำลังประชุมอยู่ที่ห้องประชุม แต่

ในขณะนั้นผู้ใช้ A กำลังขับรถ จากตัวอย่างดังกล่าวแสดงให้เห็นว่าข้อมูลแสดงสถานะของผู้ใช้ A มีความขัดแย้งกัน ซึ่งจะทำให้ผู้มองสถานะอาจจะได้รับข้อมูลที่คลาดเคลื่อนและทำให้เกิดความไม่แน่ใจในการเลือกช่องทางสำหรับติดต่อผู้ใช้ A จากตัวอย่างดังกล่าวถ้าผู้ใช้ A ได้กำหนดรูปแบบของกฎเอาไว้ดังรูปที่ 3-3 โดยเมื่อพิจารณากฎที่ผู้ใช้ A ได้กำหนดไว้ตัวรวบรวมข้อมูล (Compositor) จะได้รับข้อมูลที่แสดงว่าผู้ใช้ A อยู่ในรถทำให้ตัวรวบรวมก็จะเพิ่มข้อมูล Deriving Presence Information ลงไปว่าผู้ใช้ A กำลังขับรถอยู่ (Driving) เป็นหนึ่งในกิจกรรมของเขา ซึ่งในตอนนี้ระบบรวบรวมจะมีข้อมูลของกิจกรรม 3 อย่าง คือ เขากำลังขับรถ (ซึ่งเปลี่ยนได้จากการทำในขั้นตอน Driving) ผู้ใช้ A กำลังทำงานที่เครื่องคอมพิวเตอร์ และผู้ใช้ A กำลังประชุมอยู่จากข้อมูลตารางเวลา จากนั้นระบบก็จะทำการมองไปยังตัวแปลอื่นในกฎ ซึ่งในที่นี้จะเป็น Type-place และภายในระบุว่าเลือกข้อมูลที่ผู้ใช้ Active ในขณะนั้น ซึ่งจะทำให้ระบบตัดทิ้งข้อมูลกิจกรรมที่ระบุว่า ผู้ใช้ A กำลังทำงานอยู่ที่สำนักงานและกำลังประชุมทิ้งไป และเลือกว่าเขากำลังขับรถอยู่ไปแสดงผลไปยังผู้มองสถานะ

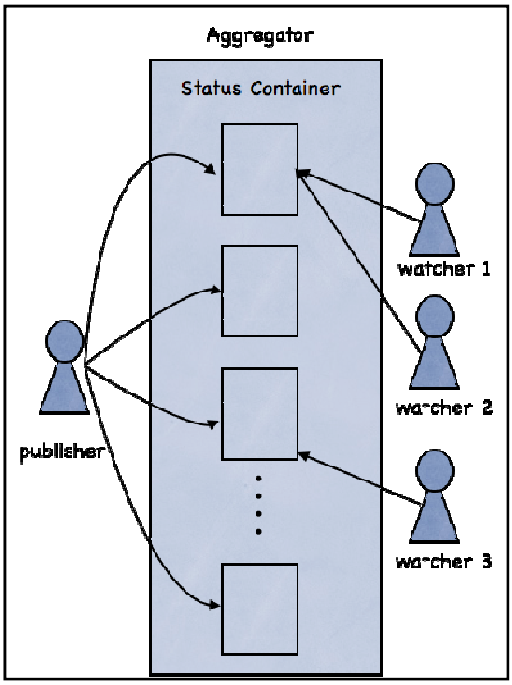
จากเอกสาร [4] เมื่อพิจารณาข้อดีข้อเสียของฟังก์ชันการรวบรวม (Composition) ที่ Presence Server โดยจะรวมข้อมูลจากหลาย ๆ แหล่งตามช่องทางต่าง ๆ ที่ทำให้ข้อมูลเป็นข้อมูลการแสดงสถานะแบบรวมศูนย์ เมื่อพิจารณาในแง่ดีจะเห็นได้ว่า มีกระบวนการเพิ่ม Deriving Presence Information ทำให้ผู้มองสถานะสามารถใช้ประโยชน์จากข้อมูลการแสดงสถานะได้ดีขึ้นและมีความแม่นยำมากขึ้น และข้อดีอีกข้อ คือ สามารถที่จะกำจัดข้อมูลที่ขัดแย้งกันได้ โดยใช้กฎที่ผู้แสดงสถานะกำหนดไว้ ทำให้ไม่เกิดความสับสนเมื่อผู้มองสถานะได้รับข้อมูลการแสดงสถานะแบบรวมศูนย์

แต่อย่างไรก็ตามเมื่อพิจารณาในมุมมองของระบบการสื่อสารแบบรวมศูนย์จะเห็นได้ว่า ฟังก์ชันการรวมจะจัดการได้เพียงแค่โปรโตคอล SIP เท่านั้น แต่ในระบบการสื่อสารแบบรวมศูนย์นั้น จำเป็นต้องจัดการโปรโตคอลที่แตกต่างกันจากช่องทางการสื่อสารในหลายรูปแบบ เช่น Instant Messenger ที่มีโปรโตคอลที่แตกต่างกัน ซึ่งจะเห็นได้ว่าในเอกสาร [4] ยังขาดการทำงานร่วมกับโปรโตคอลอื่น ๆ

3.1.3 การรวบรวมข้อมูลแสดงสถานะที่เคลื่อนแม่ข่ายโดยใช้ตารางรวบรวม

เอกสาร [6] ได้เสนอการรวบรวมข้อมูลแสดงสถานะที่เครื่องแม่ข่ายเช่นเดียวกับเอกสารที่ [4] แต่หลักการรวมเป็นข้อมูลแบบรวมศูนย์ จะใช้หลักการของตารางรวบรวม (Mapping Table) โดยตารางรวบรวมจะกำหนดเป็นช่วงของค่าตัวเลขจำนวนเต็ม ตัวอย่างเช่น สถานะ Busy กำหนดเป็นค่า 6000-6999 เพื่อพิจารณาค่าข้อมูลแสดงสถานะต่าง ๆ ที่ได้จากระบบการสื่อสาร และกำหนดค่าของข้อมูลแสดงสถานะจากแหล่งข้อมูลต่าง ๆ เป็นค่าจำนวนเต็มเช่นกัน แล้วนำค่าดังกล่าวจากแหล่งข้อมูลต่าง ๆ มาเปรียบเทียบ โดยค่าที่มีค่ามากที่สุดจะมี

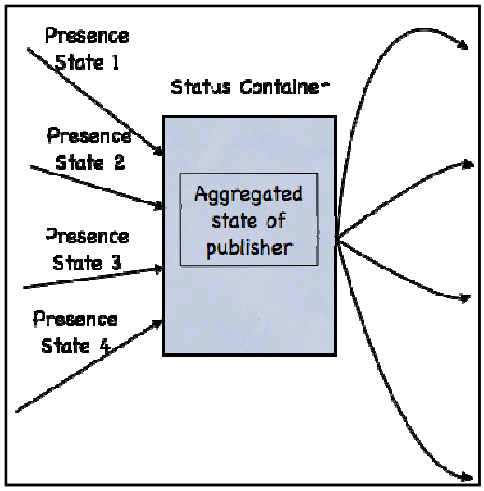
ความสำคัญในการแสดงผลมากที่สุด ซึ่งเครื่องแม่ข่ายดังกล่าวจะอนุญาตให้ผู้แสดงสถานะ (Publisher) ส่งข้อมูลการแสดงสถานะจาก Endpoint ต่าง ๆ ไปเก็บไว้ในเครื่องแม่ข่ายกลางได้ โดยสถาปัตยกรรมของระบบการรวบรวมสามารถเขียนได้ตามรูปที่ 3-4



รูปที่ 3-4 สถาปัตยกรรมของระบบ User Presence Aggregator at Server

จากรูปที่ 3-4 แสดงองค์ประกอบของระบบ User Presence Aggregator at Server จากเอกสารที่ [4] ภายใน Aggregator จะประกอบด้วยส่วนสำคัญ คือ Status Container ที่จะทำหน้าที่ในการเก็บข้อมูลการแสดงสถานะ (Presence State) ที่ได้จากผู้แสดงสถานะ (Publisher) และ Container จะอนุญาตให้ผู้แสดงสถานะเท่านั้นที่มีสิทธิ์ในการแก้ไขข้อมูลภายใน และจะมีกระบวนการที่เรียกว่า Access Control List ที่จะอนุญาตว่า Container ใดสามารถให้ผู้มองสถานะคนใดเข้าถึงได้ โดยผู้แสดงสถานะจะกำหนดสมาชิกในแต่ละ Container และผู้แสดงสถานะสามารถที่จะมี Container ได้หลาย ๆ Container ในกรณีการใช้งานจริงอาจจะแบ่ง Container ตามกลุ่มของบัญชีรายชื่อของผู้ใช้ เช่น เพื่อน เพื่อนร่วมงาน หรือครอบครัว เป็นต้น

นอกจากนี้ Container สามารถรับข้อมูลแสดงสถานะจากผู้แสดงสถานะได้พร้อมกันหลาย ๆ สถานะ แล้วทำการรวบรวมสถานะดังกล่าวเปลี่ยนจากข้อมูลการแสดงสถานะ (Presence State) ให้เป็นข้อมูลการแสดงสถานะแบบรวบรวม (Aggregation State) ดังรูปที่ 3-5 แล้วทำการส่งข้อมูลไปยังผู้มองสถานะ



รูปที่ 3-5 แสดง Status Container ว่า มีกระบวนการ Aggregation ภายใน

กระบวนการรวบรวม (Aggregation) เอกสาร [6] ได้แบ่งแยกประเภทของข้อมูลการแสดงสถานะเป็น 6 ประเภท คือ User, Machine, Phone, Calendar, Conference และ Generic ตามตารางที่ 3-1

ตารางที่ 3-1 แบ่งประเภทรายละเอียดของ Presence State ของระบบการรวบรวมของเอกสาร [6]

State type	**Description**
User	เป็นข้อมูลที่ผู้แสดงสถานะกำหนดด้วยตัวเองในช่วงเวลานั้น ๆ
Machine	ข้อมูลการใช้งานที่ได้จากอุปกรณ์
Phone	ข้อมูลการแสดงสถานะที่ได้จากอุปกรณ์โทรศัพท์ต่าง ๆ
Calendar	ข้อมูลที่ได้จากตารางการทำงานของผู้แสดงสถานะ
Conference	ข้อมูลที่ได้มาเมื่อผู้แสดงสถานะทำการสื่อสารกับกลุ่มบุคคลหรือข้อมูลที่ได้จาก Collaboration Session
Generic	ข้อมูลอื่น ๆ

โดยแต่ละประเภทของข้อมูลจะระบุว่ามีข้อมูลแบบใดบ้าง เช่น User State ที่ผู้แสดงสถานะจะใช้ในการบ่งบอกถึงความสมารถในการรับการสื่อสารของตนเองจะประกอบด้วย Online, Busy, Do Not Disturb, Be Right Back, Away และ Appear Offline และจะกำหนดค่า

ของสถานะต่าง ๆ เป็นตัวเลข โดยจะให้ค่าน้อยที่สุดเป็น Online และค่ามากที่สุดสำหรับ Appear Offline โดยค่ายิ่งมากแสดงว่าผู้ใช้นั้นยากที่จะรับการติดต่อ โดยค่าของสถานะต่าง ๆ ใน User State แสดงดังตารางที่ 3-2 สำหรับ State อื่น ๆ ก็จะมีค่าในลักษณะคล้าย ๆ กันดังตารางที่ 3-3 และตารางที่ 3-4

ตารางที่ 3-2 แสดงรายละเอียดของ User State โดยกำหนดเป็นค่าตัวเลข [3]

User State	Availability Value	Description
Online	3500	ผู้ใช้สามารถที่จะติดต่อได้
Busy	6500	ผู้ใช้ไม่ว่าง
Do Not Disturb	9500	ผู้ใช้ไม่อยากให้รบกวน
Be Right Back	12500	ผู้ใช้ไม่อยู่ในช่วงขณะนี้
Away	15500	ผู้ใช้ไม่อยู่ที่โต๊ะ
Appear Offline	18500	ผู้ใช้ต้องการแสดงว่า Offline

ตารางที่ 3-3 แสดงรายละเอียดของ Phone State โดยกำหนดเป็นค่าตัวเลข

Availability Value	Activity Token	Custom String	Description
6500	In a Call	In a 1 on 1	ผู้ใช้กำลังพูดคุยกับบุคคล
6750	In a Conference	In a Multiparty Conversation	ผู้ใช้กำลังพูดคุยกับกลุ่มบุคคล

ตารางที่ 3-4 แสดงรายละเอียดของ Calendar State โดยกำหนดเป็นค่าตัวเลข

Availability Value	Activity Token	Custom String	Description
3500	NULL	Free	ผู้แสดงสถานะไม่มีการประชุม
3500	NULL	Tentative	ผู้แสดงสถานะมีการประชุมแต่ผู้ใช้คนอื่น ๆไม่ตกลง
6500	In a Meeting	In a Meeting	ผู้แสดงสถานะมีการประชุมและผู้ใช้คนอื่นตกลง
3500	Out of Office	Out of The Office	ผู้แสดงสถานะไม่อยู่

จะเห็นได้ว่าค่า Availability Value ของแต่ละตารางที่มีสถานะที่ใกล้เคียงกัน เช่น “Busy” ของ User State “In a Call” ของตาราง Phone State และ “In a Meeting” ของ Calendar State ใช้ในการบ่งบอกว่า ผู้ใช้ไม่ว่างอยู่ในขณะนั้น ซึ่งทั้งหมดนี้เมื่อทำเป็น Aggregation State ซึ่งนำค่า Availability Value มาคิดเป็นช่วงของตัวเลขและพฤติกรรมของผู้ใช้ทำให้สามารถสร้างตารางของ Aggregation State ได้ดังตารางที่ 3-5

ตารางที่ 3-5 แสดงรายละเอียดของ Aggregated State โดยกำหนดเป็นค่าตัวเลขเป็นช่วง ๆ [3]

Aggregated Availability	**Default Sting**	**Availability Value Range**	**Description**
Online	Online	3000-3999	ผู้ใช้สามารถที่จะติดต่อได้
Busy	Busy	6000-6999	ผู้ใช้สามารถที่จะติดต่อได้แต่กำลังติดภารกิจอื่นอยู่
Do Not Disturb	Do Not Disturb	9000-9999	ผู้ใช้สามารถที่จะติดต่อได้แต่ไม่อยากโดนขัดจังหวะ
Temporarily Away	Temporarily Away	12000-12999	ผู้ใช้ไม่สามารถที่จะติดต่อได้ชั่วคราว
Away	Away	15000-15999	ผู้ใช้อาจจะสามารถที่จะติดต่อได้
Offline	Offline	18000-18999	ผู้ใช้ไม่สามารถที่จะติดต่อได้

หลักการทำงานของ Aggregator Container มีลักษณะดังนี้ ผู้แสดงสถานะจะส่งข้อมูลการแสดงสถานะต่าง ๆ (User State, Machine State, Calendar State, Phone State, Conference State และ Generic State) ไปยัง Container โดยระบบการรวบรวมข้อมูลแสดงสถานะจะคอยดูสถานะดังกล่าวจากอุปกรณ์ต่าง ๆ ของผู้แสดงสถานะ ถ้าผู้ใช้ได้ทำการแสดงสถานะของ User State เข้ามาที่ Container ระบบจะทำการบันทึกเวลาเอาไว้ และการพิจารณาการรวบรวมข้อมูลแสดงสถานะก็จะใช้เวลาของ User State ครั้งล่าสุดเป็นพื้นฐานในการพิจารณา ถ้ามีข้อมูลของ Presence State ต่าง ๆ ก่อนหน้านี้ก็ให้ลบทิ้งไป และจะพิจารณาข้อมูลที่เข้ามาใหม่กว่า User State และระบบรวบรวมข้อมูลแสดงสถานะก็จะสร้างข้อมูลแสดงสถานะแบบรวมศูนย์ขึ้นมา

สำหรับแต่ละ Container โดยผู้ที่มองสถานะจะได้ข้อมูลไปโดยกระบวนการ Subscribe เพื่อขอข้อมูลดังกล่าว และระบบก็จะทำส่งข้อมูลไปยังผู้มองสถานะโดยผ่านกระบวนการ Notify

โดยตัวอย่างในการคิดคำนวณค่า Availability Value ของ Presence State ให้เป็น Aggregated Availability ของ Aggregated State เช่น ในกรณีที่ Presence State ในที่นี้มี Phone State ที่บ่งบอกว่า In a Multiparty Conversation และ Calendar State บ่งบอกว่า In a Meeting ซึ่งทั้งสองข้อมูลเมื่อพิจารณาจากตารางที่ 3-5 จะอยู่ในช่วง 6000–6999 ทั้งคู่ทำให้สถานะเมื่อรวบรวมข้อมูลแสดงสถานะเป็น Busy แต่เนื่องด้วยค่า Availability Value ของ Phone State มีค่ามากกว่า Calendar State (6750>6500) ทำให้การแสดงค่า Custom String ซึ่งจะบ่งบอกรายละเอียดในการขยายความของ “Busy” ว่าเป็น In a Multiparty Conversation แทนที่จะเป็น In a Meeting ซึ่งระบบการรวบรวมข้อมูลแสดงสถานะจะยึดถือข้อมูลที่ใหม่กว่า User State และมีค่า Availability สูงที่สุด

เมื่อสรุปข้อมูลจากเอกสาร [6] จะทำให้เห็นข้อดีคือ การที่มีการกำหนดว่าข้อมูลการแสดงสถานะแต่ละข้อมูลเป็นค่าตัวเลขทำให้ง่ายต่อการทำเป็นข้อมูลแบบรวมศูนย์ เพราะเพียงพิจารณาว่าข้อมูลการแสดงสถานะใดมีค่ามากที่สุด และคิดเป็นสถานะใดในตารางการรวบรวมข้อมูลแสดงสถานะตารางที่ 3-5 และข้อดีอีกหนึ่งข้อ คือ การเข้าถึงข้อมูลที่เรียกว่า Access Control List ซึ่งเมื่อมองในมุมมองที่เป็น Rich Presence Feature การแบ่งเป็น Container ต่าง ๆ ทำให้สามารถที่จะเกิดมุมมองที่แตกต่างในแต่ละผู้ใช้ (Difference Point of View) เช่น ถ้าผู้มองสถานะคนที่หนึ่ง และผู้มองสถานะคนที่สอง Subscribe ไปยังคนละ Container ก็จะได้รับข้อมูลการรวบรวมข้อมูลแสดงสถานะที่แตกต่างกันจึงทำให้ผู้แสดงสถานะสามารถที่จะกำหนดได้ว่า ผู้มองสถานะของเขาแต่ละคนควรจะสื่อสารกับเขาอย่างไรที่จะเหมาะสมกับเขาในช่วงเวลานั้น ๆ

เมื่อพิจารณาในข้อเสียจะเห็นได้ว่าข้อมูลการสถานะบางอย่างอาจไม่เหมาะสมต่อการสร้างเป็นตัวเลข เช่น Location Presence ต่อระบบการรวบรวมข้อมูลแสดงสถานะแบบนี้ เช่น การบ่งบอกถึงสถานะของผู้ใช้โดยการตรวจสอบจาก Machine State ของอุปกรณ์ล่าสุดที่ผู้ใช้ได้ใช้งาน เช่น ผู้ใช้ได้ Login เข้าระบบโทรศัพท์เคลื่อนที่ ต่อมาผู้ใช้ได้ใช้คอมพิวเตอร์ที่สำนักงาน และต่อมาผู้ใช้ก็ขับรถออกไปข้างนอก จะเห็นได้ว่า Machine State ก็ระบุว่าผู้ใช้อยู่ที่โต๊ะทำงาน แทนที่จะระบุว่า ผู้ใช้ได้ขับรถไปแล้ว ซึ่งอาจจะทำให้เกิดความผิดพลาดในช่วงเวลาก่อนที่ Machine State ของคอมพิวเตอร์บ่งบอกสถานะว่า ผู้ใช้ได้ Away ซึ่งทำให้เกิดข้อผิดพลาดได้ และปัญหาความขัดแย้งของข้อมูลได้ เพราะระบบไม่มีกฎการรวบรวมข้อมูลแสดงสถานะ (Aggregation Policy) ตัวอย่างเช่น ผู้ใช้ได้ระบุไว้ใน Calendar ของเขาว่า ในช่วงเวลาดังกล่าวเขามีการประชุม แต่เนื่องด้วยการประชุมเลิกก่อนช่วงเวลาที่กำหนด ทำให้เขากลับไปยังโต๊ะทำงานของเขาก่อนแต่สถานะของ Calendar ที่ระบุไว้ว่า เขากำลังประชุมอยู่ทำให้สถานะเกิดความผิดพลาดได้

เมื่อมองในมุมมองของ Rich Presence Information แบบอื่น ๆ ในที่นี้ คือ ข้อมูลการแสดงสถานะแบบใหม่ ๆ เช่น อารมณ์ของผู้ใช้ในขณะนั้น ซึ่งยากที่จะคิดเป็นตัวเลข เช่น ถ้า

ผู้ใช้ในขณะนั้นได้ตั้งค่าว่า เขาเหนื่อย และต้องการพักผ่อน ผู้ที่มองสถานะของเขา ถ้ามีความสัมพันธ์กับเขาเป็นเพื่อนร่วมงานอาจจะมองสถานะของเขาในตอนนั้นเป็น Appear Offline แต่ถ้าเป็นเพื่อนที่ในชมรมกีฬาของเขาก็อาจจะมองเห็นว่า Online ได้ ซึ่งถ้าระบุเป็นตัวเลขก็จะทำให้ไม่สามารถที่จะแยกแยะข้อมูลดังกล่าวได้

ข้อเสียอีกอย่างหนึ่ง คือ การออกแบบระบบของเอกสาร [6] การรวบรวมข้อมูลแสดงสถานะยังเป็นระบบแบบปิดที่ไม่ได้มีแนวคิดในการรวบรวมข้อมูลแสดงสถานะข้อมูลการแสดงสถานะของระบบหรือโปรโตคอลอื่น ๆ เช่น การรวบรวมข้อมูลจาก Instant Messenger จากผู้บริการอื่น ๆ ข้อมูลตารางเวลาจากผู้บริการอื่น ๆ ทำให้ไม่สามารถครอบคลุมการทำงานทั้งหมดของระบบการสื่อสารแบบรวมศูนย์

3.1.4 สรุปการรวบรวมข้อมูลแสดงสถานะ (Aggregation)

จากเอกสาร [4][5][6] ทำให้สามารถสรุปได้ว่า ฟังก์ชันการรวบรวมข้อมูลแสดงสถานะ จะสามารถเกิดได้ที่เครื่องแม่ข่ายตรงกลางและที่เครื่องลูกข่าย ซึ่งจะมีข้อดีและข้อเสียที่แตกต่างกัน และถ้าใช้ให้เหมาะสมกับสถานการณ์จะทำให้เกิดประโยชน์สูงสุด

โดยระบบที่มีฟังก์ชันการรวบรวมข้อมูลแสดงสถานะ ที่เครื่องลูกข่าย [5] มีจุดเด่นในการรวบรวมข้อมูลแสดงสถานะ จากระบบอื่น ๆ ที่ใช้โปรโตคอลแตกต่างกัน หรือกระบวนการในการเข้าถึงข้อมูลการแสดงสถานะที่แตกต่าง เพราะการที่จะให้เครื่องแม่ข่ายรับหน้าที่ในการรวบรวมข้อมูลแสดงสถานะ เครื่องแม่ข่ายจำเป็นต้องมีการสร้าง Gateway เพื่อใช้ในการเข้าถึงข้อมูลการแสดงสถานะของระบบอื่น ๆ เป็นเหตุให้เครื่องแม่ข่ายจำเป็นต้องสร้างการเชื่อมต่อจำนวนมาก จึงเป็นการดีกว่า ถ้าจะแบ่งภาระดังกล่าวไปให้เครื่องลูกข่ายแทน และนอกจากเรื่องของการเชื่อมต่อ ยังมีประเด็นในการที่ข้อมูลการแสดงสถานะจากผู้ให้บริการต่าง ๆ โดยการเข้าถึงข้อมูลจำเป็นต้องมีกระบวนการเข้าถึงที่แตกต่างกัน โดยถ้าไม่ผ่านกระบวนการปกติของผู้ให้บริการนั้นเป็นเรื่องที่ยากจะเป็นไปได้ เพราะแต่ละผู้ให้บริการจำเป็นต้องปิดบังข้อมูลกระบวนการทำงานของตนเอง จึงจำเป็นต้องใช้ Username และ Password ของผู้ใช้ในการขอข้อมูลต่าง ๆ ซึ่งจะเป็นการไม่เหมาะสมอย่างมาก ที่เครื่องแม่ข่ายจะเก็บข้อมูลดังกล่าวเอาไว้ เพราะจะทำให้ผู้ใช้อาจจะสูญเสียความปลอดภัยและความเป็นส่วนตัว

การที่มีฟังก์ชันรวบรวมข้อมูลแสดงสถานะ ในฝั่งของเครื่องแม่ข่ายจะมีลักษณะที่เครื่องแม่ข่ายจะมีตัวรวบรวมข้อมูลแสดงสถานะ (Compositor หรือ Aggregator) ที่คอยรวบรวมข้อมูลแสดงสถานะ จากแหล่งต่าง ๆ ดังในเอกสาร [4] [6] และผู้แสดงสถานะจะมีกฎที่ใช้ในการกำหนดว่า จะรวบรวมข้อมูลแสดงสถานะให้ข้อมูลต่าง ๆ ที่ได้มาจากตัวรวบรวมข้อมูลแสดงสถานะให้กลายเป็นข้อมูลชุดเดียว

โดยข้อดีของเอกสาร [4] คือ การแก้ไขปัญหาความขัดแย้งของข้อมูลต่าง ๆ เช่น ในกรณีของข้อมูลสถานที่อยู่ (Location Presence) เมื่อมองในมุมมองของระบบการสื่อสารแบบรวมศูนย์ ที่จะมีช่องทางการบริการการสื่อสารเป็นจำนวนมากให้กับผู้ใช้ ทำให้ข้อมูลการแสดงสถานะมาจากหลาย ๆ ช่องทางการสื่อตามไปด้วย ซึ่งจะเป็นการเหมาะกว่าที่จะมีการทำการรวบรวมข้อมูลดังกล่าวภายในก่อนแล้วค่อยส่งไปยังผู้มองสถานะ เพราะอาจจะทำให้ผู้มองสถานะจะต้องสร้างการเชื่อมต่อมากเกินไปในการรับข้อมูลการแสดงสถานะของแต่ละผู้แสดงสถานะที่เขาสนใจ แต่การรวบรวมข้อมูลแสดงสถานะ ไว้ที่เครื่องแม่ข่าย อาจจะมีปัญหาได้ในกรณีที่ต้องการข้อมูลจากระบบการสื่อสารที่มีโปรโตคอลหลักที่แตกต่างกันทำให้ตัวรวบรวมข้อมูลแสดงสถานะ (Compositor หรือ Aggregation) นั้นต้องทำการสร้างการเชื่อมต่อ และแปลงข้อมูลดังกล่าวให้เป็นโปรโตคอลที่สามารถใช้งานได้ ซึ่งจะเป็นเพิ่มภาระที่มากเกินไปสำหรับเครื่องแม่ข่าย เมื่อมีผู้ใช้ที่มากขึ้นในระบบ

3.1.5 การควบคุมการเข้าถึงข้อมูลแสดงสถานะ (Privacy Control)

การควบคุมการเข้าถึงข้อมูลแสดงสถานะ คือ ส่วนที่คอยควบคุมความเป็นส่วนตัวและความปลอดภัยของข้อมูลการแสดงสถานะของผู้ใช้ เพราะข้อมูลบางอย่าง เช่น ข้อมูลที่อยู่ ข้อมูลอารมณ์ อาจจะไม่เหมาะสมต่อการแสดงต่อบุคคลที่ไม่มีความสนิทสนมกับเจ้าของข้อมูล หรือเป็นข้อมูลที่มีเปิดเผยออกไปแล้วจะทำให้เจ้าข้อมูลขาดความเป็นส่วนตัว หรือข้อมูลกิจกรรมที่เจ้าของข้อมูลอยากแสดงกับบุคคลเฉพาะกลุ่ม ตัวอย่างเช่น ผู้แสดงสถานะออกไปเล่นฟุตบอล เจ้าของข้อมูลต้องการอยากจะแสดงให้บุคคลที่เป็นเพื่อนของผู้แสดงสถานะรับทราบ ไม่ใช่เจ้านายหรือเพื่อนร่วมงานได้รับรู้ จึงทำให้การควบคุมการเข้าถึงข้อมูลมีความสำคัญเป็นอย่างมากต่อระบบการบริการแสดงสถานะ

3.1.5.1 การควบคุมโดยกฎขั้นพื้นฐาน

โดยก่อนที่จะกล่าวถึงการควบคุมที่มีความซับซ้อน จะขออธิบายกฎการควบคุมขั้นพื้นฐานที่ระบบแสดงสถานะของ SIP [9] [10] ใช้งานอยู่ในปัจจุบัน กฎโดยทั่วไปมักจะเป็นกฎที่ตั้งขึ้น โดยจะใช้เพียงการจำกัดสิทธิในการเข้าถึงข้อมูลการแสดงสถานะของผู้มองสถานะ แต่อย่างไรก็ตามได้มีการพัฒนากระบวนการดังกล่าว เพื่อให้เหมาะสมกับการแสดงข้อมูลสถานะให้มีความลาดเพิ่มขึ้น โดยระบบที่เคยมีการนำเสนอ คือ [9][10] จะอนุญาตให้ผู้แสดงสถานะสามารถกำหนดการเข้าถึงข้อมูลการแสดงสถานะของผู้มองสถานะ โดยสามารถที่จะระบุได้ว่า ถ้าผู้แสดงสถานะอยู่ในสถานที่ ๆ กำหนดไว้ในกฎของผู้แสดงสถานะ การควบคุมควรที่จะกำหนดการเข้าถึงเป็นอย่างไร หรืออาจจะระบุเป็นช่วงเวลาของวันนี้ที่กำหนดไว้ ตัวอย่างเช่น นาย A เป็นผู้แสดงสถานะ โดยเขากำหนดไว้ว่าช่วงเวลาหลังเลิกงาน (17:00–8:00) จะทำให้ผู้มองสถานะของเขาไม่

สามารถที่จะเห็นสถานะบางอย่างได้ เช่น ข้อมูลของสถานที่อยู่ ทำให้เขาสามารถที่จะกำหนดการเข้าถึงเป็นช่วงของเวลาได้ โดยกฎที่ถูกออกแบบและสร้างขึ้น จะเป็นกฎแบบอย่างง่ายและสามารถที่จะกำหนดระดับความเป็นส่วนตัวกับความปลอดภัยได้ แต่อย่างไรก็ดีกฎดังกล่าวเป็นกฎที่ใช้ในการลดระดับการเข้าถึงเท่านั้น ซึ่งจะไม่สามารถที่จะเพิ่มระดับได้ เช่น ถ้าลดการเข้าถึงข้อมูลการแสดงสถานะใดแล้วในขณะนั้นก็จะไม่สามารถเพิ่มได้อีก ซึ่งจะทำให้ไม่สามารถที่จะกำหนดระดับการเข้าถึงข้อมูลของผู้มองสถานะแต่ละคนให้แตกต่างกันได้

3.1.5.2 การควบคุมโดยใช้กฎขั้นสูง

จากเอกสาร [7] ได้นำเสนอการรวบรวมข้อมูลการแสดงสถานะแบบหลากหลาย (Rich Presence) เช่น ที่อยู่ในขณะนั้น ความสามารถที่จะรับการติดต่อ กิจกรรมที่สนใจ หรืออารมณ์ในขณะนั้นจากระบบ Social Network ต่างๆ เนื่องด้วยระบบ Social Network ในขณะนี้มีผู้ใช้งานเป็นจำนวนมาก และถูกพัฒนาขึ้นด้วยเทคโนโลยี Web 2.0 ทำให้ข้อมูลต่างๆ ที่ผู้ใช้แสดงใน Social Network สามารถที่จะถูกจัดเก็บเป็นข้อมูลบริบท (Context Information) ได้ดี และเอกสาร [7] ได้เสนอการรวบรวมการสื่อสารต่างๆ เช่น ระบบ IMS [11], Parlay-X [33] และ Web 2.0 เพื่อสร้างมูลค่าให้กับผู้ให้บริการทั้ง Social Network และโทรศัพท์เคลื่อนที่

ข้อมูลบริบทของผู้ใช้ที่ได้มาจาก Social Network ต่างๆ ได้ถูกนำมาใช้ประโยชน์เพื่อทำเป็นฟังก์ชัน Social Search ซึ่งฟังก์ชันดังกล่าวจะช่วยให้ผู้ใช้สามารถที่จะค้นหาเพื่อนที่มีกิจกรรมต่างๆ ตามที่ผู้ใช้สนใจ หรืออาจจะเป็นเพื่อนของเพื่อน (Friend of Friend) ที่ไม่ต้องอยู่ในรายชื่อของผู้ใช้ก็ได้ โดยใช้เทคนิคของ Context-Awareness Computing โดยใช้ Context Model และ Reasoning ที่ได้ออกแบบเอาไว้ และทำการเสนอกิจกรรมต่างๆ ที่ตรงกับความต้องการของผู้ใช้แบบทันทีทันใด นอกจากนี้ยังสร้างการเชื่อมต่อการสื่อสารกับบุคคลที่ผู้ใช้สนใจผ่านทางข้อความสั้นหรือ Instant Message เป็นต้น

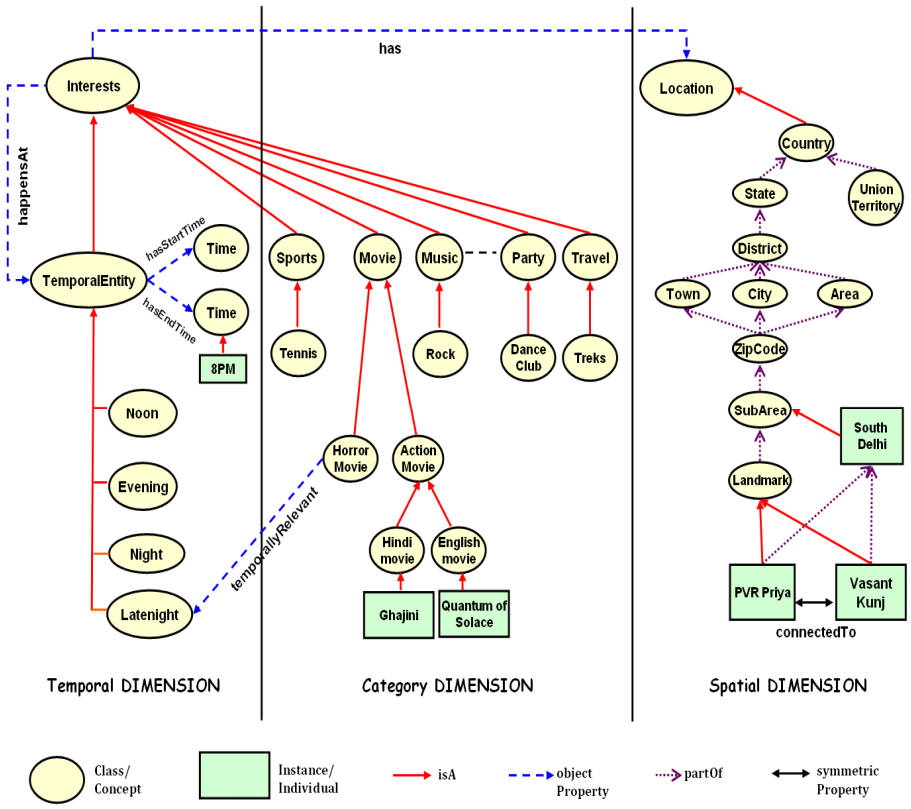
รูปที่ 3-6 มุมมองระดับทั่วไปของระบบ R-U-In? Active-Oriented Social Networking [7]

โดยเอกสาร [7] ได้ออกแบบฟังก์ชันการทำงานเป็นดังรูปที่ 3-6 โดยได้แสดงภาพรวมของระบบ R-U-In? ซึ่งประกอบด้วย 3 ระดับ (Layer) คือ Interaction Management Layer, Knowledge Management Layer, Active Lifecycle Management Layer ซึ่งแต่ระดับมีการทำงานดังนี้

1. R-U-In? Interaction Management Layer จะทำหน้าที่เป็นส่วนเชื่อมต่อระหว่างผู้ใช้และระบบ R-U-In? โดยผู้ใช้จะติดต่อกับระบบโดยใช้การสื่อสารแบบเดิม เช่น โปรแกรมประยุกต์บนโทรศัพท์เคลื่อนที่ Instant Messaging หรือ Social Network เพื่อเชื่อมต่อเข้าสู่ระบบ โดยระดับการทำงานนี้จะทำหน้าที่เป็น Gateway ที่คอยเชื่อมต่อกับผู้ใช้ เพื่อให้ผู้ใช้ อัพเดทข้อมูลการแสดงสถานะเข้าสู่ระบบ ค้นหาและสร้างกิจกรรมต่าง ๆ ที่ผู้ใช้สนใจ และคอยรับข้อมูลที่อัพเดทจากระบบแบบตามเวลาจริง และจะใช้ Context-Awareness Computing เพื่อที่จะตั้งเป็นค่า Preference ใช้กับผู้ใช้ในการกำหนดช่องทางการสื่อสาร ตัวอย่าง เช่น กำหนดเวลา 9.00-18.00 ให้ติดต่อผ่าน SMS และเวลานอกจากนั้นให้ติดต่อผ่านทาง MSN เป็นต้น
2. R-U-In? Knowledge Management Layer จะเป็นส่วนที่ใช้ในการพิจารณาการทำ Matchmaking Process จากข้อมูลบริบทของผู้ใช้โดยพิจารณาจาก Contextual Modeling และ Reasoning โดยจะแบ่งเป็น 2 ส่วนหลักคือ R-U-In? Contextual Modeling และ Matchmarking Methodology โดยในส่วนแรกจะเป็นลักษณะของ Contextual Modeling

ที่ใช้ในการพิจารณาจากรูปที่ 3-7 โดยได้แบ่งเป็น 3 Tag ที่ใช้ในการพิจารณาคือ Location Description, Category Description และ Time Description โดยผู้ใช้สามารถที่จะระบุโดยการใช้ 3 Tag ดังกล่าวประกอบขึ้น เพื่อใช้ในการค้นหากิจกรรมที่ผู้ใช้สนใจ เช่น Major, Movie, 7 PM โดย Contextual Modeling จะอธิบายว่าสิ่งที่ผู้ใช้ระบุข้อมูลสามารถที่จะพิจารณาได้อย่างไร โดย Contextual Model ที่เอกสาร [7] ได้นำเสนอไว้มีลักษณะดังรูปที่ 3-7 Contextual Modeling ของ R-U-In? ในส่วนต่อมา Matchmaking Methodology เป็นส่วนพิจารณาโดยจะมีกระบวนการ 2 ขั้นตอนคือ 1. รับการพิจารณาข้อมูลบริบท (สถานที่, ประเภท, เวลา) และขั้นตอนที่ 2. คือการนำข้อมูลบริบทดังกล่าวมาใช้ตัดสินใจโดยใช้ Contextual Modeling ที่ได้กำหนดไว้ข้างต้น

3. R-U-In? Lifecycle Management Layer เป็นเลเยอร์ที่ใช้ในการควบคุมและทำงานในส่วนของการสร้างและจัดตารางเวลาของกิจกรรมต่าง ๆที่จะเกิดขึ้นในระบบโดย Lifecycle Management Layer จะจัดการดูแลในส่วนของการเก็บข้อมูลต่าง ๆ เช่น Preference ของผู้ใช้ ช่องทางการสื่อสารต่าง ๆ ของผู้ใช้ กิจกรรมของผู้ใช้ ข้อมูลบริบทของผู้ใช้ (สถานที่, ประเภท, เวลา) รายชื่อผู้ใช้ และกฎที่ใช้ในแต่ละกิจกรรม



รูปที่ 3-7 Contextual Modeling ของ R-U-In? [7]

จากเอกสาร [7] จะเห็นได้ถึงข้อดีในการใช้ Context-awareness Computing มาใช้ในการพิจารณาการกระจายข้อมูลการแสดงสถานะ จะเห็นได้ชัดเจนถึงความยืดหยุ่นจากตัวอย่าง ในการที่จะระบบพิจารณาสามารถให้ข้อมูลของบุคคลที่ไม่ได้อยู่ในรายชื่อของผู้ใช้โดยที่ยังคำนึงถึงความเป็นส่วนตัวของบุคคลดังกล่าวได้ด้วยจากจุดนี้เอง เมื่อมองในมุมของการสื่อสารแบบรวมศูนย์นั้น Contextual Modeling ของเอกสาร [7] ยังมองในมุมมองของข้อมูลบริบทที่ได้จาก Social Network เพียงอย่างเดียว ซึ่งถ้าเป็นข้อมูลบริบทที่จะได้จากการทำงานของการสื่อสารแบบรวมศูนย์ควรจะมีมากขึ้น เช่น ข้อมูลการแสดงสถานะของช่องทางการสื่อสารต่างๆ ซึ่งจะทำให้การออกแบบ Contextual Modeling มีรูปแบบที่ซับซ้อนมากกว่า และการนำไปประยุกต์ใช้อาจจะไม่เหมือนกัน เช่น การนำไปใช้ในการคำนวณหาเส้นทางในการใช้ช่องทางการสื่อสารที่ฉลาด (Smart Routing) เป็นต้น

3.1.6 สรุปการควบคุมการเข้าถึงข้อมูล

จากเอกสาร [5] [7] [9] [10] จะสามารถแบ่งการทำงานได้เป็น 2 ลักษณะ คือ กฎการเข้าถึงและกระจายข้อมูลแบบพื้นฐาน (Basic Policy Rules) และ กฎการเข้าถึงและกระจายข้อมูลแบบขั้นสูง (Advance Policy Rules)

จากกฎแบบพื้นฐาน [9][10] จะพบปัญหาเมื่อผู้ใช้ต้องการเพิ่มรูปแบบของข้อมูลการแสดงสถานะแบบใหม่เข้าสู่ระบบ จำเป็นต้องเปลี่ยนแปลงกฎที่มีอยู่เดิม ซึ่งอาจทำให้ยากต่อการแก้ไข และยากในการเพิ่มรูปแบบการทำงานใหม่ๆ แต่เมื่อนำเทคนิคของ Context-awareness Computing มาประยุกต์ใช้ จากเอกสาร [7] ใช้จะเห็นได้ว่า ระบบจะเหมาะสมกับการรองรับการกระจายข้อมูลที่มีหลากหลายมิติขึ้น ตัวอย่างเช่น การรับข้อมูลการแสดงสถานะบุคคลที่ไม่ได้อยู่ในรายชื่อของตนเอง เมื่อมองในมุมของระบบการสื่อสารแบบรวมศูนย์ จะทำให้เพิ่มความสามารถในการติดต่อไปยังปลายทางได้มากยิ่งขึ้น ซึ่งส่งผลให้ประสิทธิภาพสูงขึ้นตามไปด้วย แต่การออกแบบให้ Ontology Model นั้นครอบคลุมความต้องการของผู้ใช้ทั้งหมด อาจจะยังเป็นเรื่องที่ยาก ซึ่งระบบอาจจะจำเป็นต้องอนุญาตให้ผู้ใช้สร้าง Ontology Model สำหรับตนเอง หรือระบบอาจจะออกแบบเฉพาะเจาะจงสำหรับองค์กรต่างๆ แต่อย่างไรก็ดีด้วยหลักการของ Context-awareness Computing ทำให้ง่ายต่อการแก้ไขรูปแบบ Contextual Modeling ให้เข้ากับองค์กรที่มีกระบวนการที่แตกต่างกันได้ง่ายกว่ารูปแบบเดิมๆ จึงทำให้การสร้างกฎการกระจายและการเข้าถึงข้อมูลที่อยู่บนพื้นฐานของเทคนิค Context-awareness Computing ส่งผลให้ระบบการสื่อสารแบบรวมศูนย์นั้นมีความยืดหยุ่น และง่ายต่อการพัฒนาการบริการใหม่ๆ

บทที่ 4
การออกแบบกรอบการทำงานและสถาปัตยกรรม ของระบบบริการแสดงสถานะแบบรวมศูนย์

บทนี้จะกล่าวถึงการออกแบบกรอบการทำงาน (Framework) และสถาปัตยกรรม (Architecture) ของระบบบริการแสดงสถานะแบบรวมศูนย์ (UPS) โดยกรอบการทำงานดังกล่าวมีการจัดการปัญหารูปแบบ Massive PI ซึ่งเป็นรูปแบบของข้อมูลแสดงสถานะที่มาจากระบบการสื่อสารแบบรวมศูนย์ โดยนำกรอบการทำงานนี้มาเป็นแนวทางในการออกแบบสถาปัตยกรรม เพื่อที่จะทำให้สถาปัตยกรรมดังกล่าวสามารถจัดการกับปัญหา Massive PI ได้ และสถาปัตยกรรมของระบบบริการแสดงสถานะแบบรวมศูนย์สามารถอธิบาย โดยการแสดงรูปขององค์ประกอบ (Network Component) ต่าง ๆ และลำดับของสัญญาณ (Signaling Flow) นอกจากนี้ยังมีการออกแบบการเชื่อมต่อกับระบบโทรศัพท์เคลื่อนที่ยุคที่สาม ซึ่งในที่นี้ คือ ระบบ IMS

4.1 การวิเคราะห์ลักษณะของการบริการแสดงสถานะในระบบการสื่อสารแบบรวมศูนย์

ในหัวข้อนี้จะอธิบายให้เห็นถึงรูปแบบข้อมูลแสดงสถานะที่มีลักษณะเฉพาะตัวของระบบการสื่อสารแบบรวมศูนย์ จำเป็นจะต้องอธิบายให้เห็นว่าลักษณะของ Massive PI และความจำเป็นที่จะต้องแบ่งฟังก์ชันของกรอบการทำงานของระบบบริการแสดงสถานะแบบรวมศูนย์เป็นไปตามฟังก์ชันฟังก์ชันดังกล่าว

4.1.1 Massive Presence Information (Massive PI)

เนื่องด้วยระบบการสื่อสารแบบรวมศูนย์ประกอบขึ้นด้วยการบริการการสื่อสารหลากหลายรูปแบบ ยกตัวอย่างเช่น VoIP, Voice and Video Conference, Calendar, GSM Phone และ 3G Phone ซึ่งการสื่อสารที่กล่าวมานี้ทำให้ระบบการสื่อสารแบบรวมศูนย์จำเป็นต้องมีข้อมูลการแสดงสถานะที่เข้าสู่ระบบเป็นจำนวนมาก โดยสามารถแบ่งประเภทข้อมูลได้ดังนี้

1. บัญชีรายชื่อของผู้ใช้ เนื่องด้วยระบบการสื่อสารแบบรวมศูนย์ประกอบด้วยการบริการการสื่อสารในหลายรูปแบบ จึงทำให้ระบบการสื่อสารแบบรวมศูนย์จำเป็นที่ต้องอนุญาตให้ผู้ใช้มีบัญชีรายชื่อได้หลากหลาย
2. โปรโตคอลที่แตกต่างกันของแต่ละช่องทางการสื่อสารและการบริการ เช่น Google Calendar, MSN, GTalk, GSM Phone, 3G Phone เป็นต้น

3. อุปกรณ์ต่าง ๆ ที่มีอยู่ในระบบการสื่อสารแบบรวมศูนย์ของผู้ใช้ ตัวอย่างเช่น Mobile Phone, PC, SIP Phone และ PSTN เป็นต้น

รูปแบบของข้อมูลการแสดงสถานะข้างต้นถูกเรียกว่า Massive PI ระบบบริการแสดงสถานะในปัจจุบันไม่สามารถที่จัดการกับรูปแบบ Massive PI ได้อย่างมีประสิทธิภาพและเหมาะสมสำหรับการทำงานในระบบการสื่อสารแบบรวมศูนย์ โดยตัวอย่างที่จะแสดงต่อไปนี้เป็นตัวอย่างบางส่วนที่ระบบบริการแสดงสถานะในปัจจุบันไม่สามารถจัดการข้อมูลในรูปแบบ Massive PI ได้ ซึ่งทำให้ประสิทธิภาพของการสื่อสารในระบบการสื่อสารแบบรวมศูนย์โดนลดทอนลงได้

ตัวอย่างที่ 1 ในกรณีที่ Alice มีบัญชีรายชื่อจำนวนมากจากอุปกรณ์และการบริการต่าง ๆ ในระบบการสื่อสารแบบรวมศูนย์ เช่น SIP Phone, IM Address, Mobile Phone เป็นต้น และ Bob ต้องการที่จะเพิ่มรายชื่อ Alice เพื่อที่จะรับทราบข้อมูลสถานะของ Alice ในระบบการสื่อสารแบบรวมศูนย์

Bob จำเป็นจะต้องรู้ว่า Alice นั้นมีบัญชีรายชื่ออะไร แล้วทำการเพิ่มบัญชีรายชื่อเข้าสู่ระบบการสื่อสารแบบรวมศูนย์ด้วยตัวเอง ซึ่งในการใช้วิธีดังกล่าว Bob อาจเพิ่มรายชื่อที่ผิดของ Alice เข้าไป ซึ่งจะทำให้ Bob รับรู้สถานะที่ผิดพลาดได้ และทำให้การติดต่อกับ Alice โดนลดประสิทธิภาพลงไป จากตัวอย่างที่ 1 จะเห็นได้ว่าถ้าระบบบริการแสดงสถานะไม่มีการจัดการบัญชีรายชื่อที่ดีพออาจจะส่งผลให้ Bob ติดต่อไปยังบุคคลที่ไม่ใช่เป้าหมายในการติดต่อของเขาได้

ตัวอย่างที่ 2 เนื่องด้วยระบบการสื่อสารแบบรวมศูนย์นั้นจะต้องรองรับการทำงานของข้อมูลการแสดงสถานะรูปแบบใหม่ ๆ ที่ผู้ใช้จะสร้างขึ้นมา เพื่อที่จะรองรับกับการบ่งบอกสถานะของตน ทำให้ผู้ที่ต้องการจะติดต่อผู้ใช้สามารถจะติดต่อได้อย่างแม่นยำ เนื่องด้วยระบบปัจจุบันของ SIP การสร้างข้อมูลแสดงสถานะรูปแบบใหม่ ๆ เข้าสู่ระบบจำเป็นต้องแก้ไขทั้งในส่วนของกฎการปกป้องและการเข้าถึงข้อมูล (Policy Rule) และ Presence Server เพื่อรองรับการทำงานของข้อมูลการแสดงสถานะแบบใหม่ ๆ ทำให้เห็นว่า การทำงานแบบนี้ไม่ยืดหยุ่นเพียงพอต่อผู้ใช้ในระบบการสื่อสารแบบรวมศูนย์ ทำให้ไม่สามารถที่จะสร้างรูปแบบเฉพาะในการติดต่อของแต่ละบุคคลได้

ตัวอย่างที่ 3 ระบบการสื่อสารแบบรวมศูนย์จะให้ข้อมูลการแสดงสถานะเป็นจำนวนมากและมีความหลากหลายของโปรโตคอลที่เข้าสู่ระบบ โดยระบบจำเป็นจะต้องมีวิธีการให้ลำดับความสำคัญของข้อมูลการแสดงสถานะแต่ละชนิดและมีวิธีการที่รวบรวมและเชื่อมต่อกับข้อมูลแสดงสถานะที่มีโปรโตคอลที่แตกต่างกันอย่างเหมาะสม

ตัวอย่างที่ 4 ระบบการสื่อสารแบบรวมศูนย์ต้องเชื่อมโยงกับการบริการการสื่อสารหลากหลายรูปแบบ เช่น IMS, PSTN, SIP Phone, Calendar Service, Voice mail และ E-mail ดังนั้นจึงต้องมีวิธีการเชื่อมต่อสู่ระบบดังกล่าว เพื่อที่จะนำข้อมูลการแสดงสถานะในระบบ

ที่กล่าวไว้ข้างต้นเข้าสู่ระบบบริการแสดงสถานะของระบบการสื่อสารแบบรวมศูนย์ โดยต้องคำนึงถึงรูปแบบการเชื่อมต่อพื้นฐานของแต่ละระบบด้วย

ตัวอย่างที่ 5 การแจ้งเตือนในระบบ SIP แบบเดิมการพิจารณาเพื่อให้ข้อมูลการแสดงสถานะอะไรบ้างแก่ผู้ใช้งานจะเป็นแบบ 1 ต่อ 1 ซึ่งหมายถึงผู้ใช้จะไม่สามารถใช้ประโยชน์จากบุคคลอื่นรอบข้างได้ เช่น ในกรณีที่ Bob เป็นหัวหน้างานในสำนักงานแห่งหนึ่ง Bob มีเลขานุการคือ Alice ซึ่งในขณะนั้น Bob กำลังยุ่งและไม่สามารถที่จะรับการติดต่อได้ จากจุดนี้แสดงให้เห็นว่าบุคคลที่ควรจะได้รับการติดต่อสื่อสารแทน Bob ควรจะเป็น Alice ซึ่งระบบจะต้องมี กฎการปกป้องและการเข้าถึงข้อมูลที่สามารถพิจารณา 1-To-Many ซึ่งจะทำให้ระบบการสื่อสารแบบรวมศูนย์จะเพิ่มความสามารถในการติดต่อได้มากยิ่งขึ้น

ตัวอย่างที่ 6 ระบบ SIP ในปัจจุบันไม่สามารถที่จะแจกจ่ายข้อมูลการแสดงสถานะที่เหมาะสมไปยังกลุ่มหรือบุคคลที่ต้องการได้ เช่น บุคคลที่มีความสัมพันธ์กับคนที่แสดงสถานะเป็นเพื่อนควรจะส่งข้อมูลเป็นอย่างไร หรือบุคคลที่เป็นเพื่อนร่วมงานเป็นอย่างไร ซึ่งทำให้ระบบการสื่อสารแบบรวมศูนย์จะต้องมีความฉลาดมากพอในการตัดสินใจว่า ข้อมูลการแสดงสถานะที่เหมาะสมกับแต่ละบุคคลเป็นเช่นไร

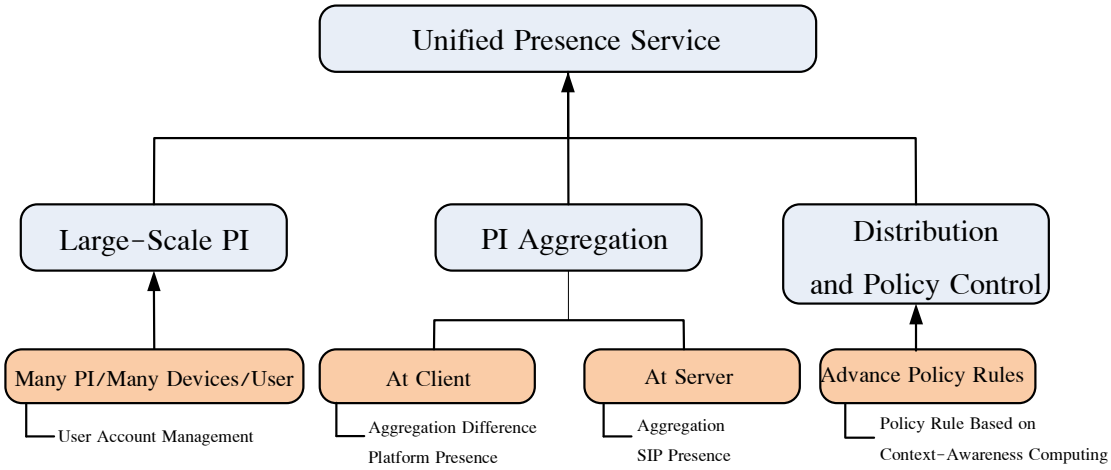
จากตัวอย่างปัญหาข้างต้นและเอกสารที่ได้อธิบายไว้ในบทที่ 3 แสดงให้เห็นว่าระบบบริการแสดงสถานะจำเป็นจะต้องมีการจัดการกับข้อมูลแสดงสถานะที่มีประสิทธิภาพเพียงพอ เพื่อที่จะทำให้ระบบการสื่อสารแบบรวมศูนย์ไม่โดนลดทอนประสิทธิภาพการสื่อสารของผู้ใช้ในระบบ นอกจากนี้ถ้าระบบบริการแสดงสถานะมีความฉลาดในการแสดงข้อมูลก็จะยิ่งเพิ่มความสามารถการสื่อสารระหว่างผู้ใช้ได้มากยิ่งขึ้น

4.2 กรอบการทำงานของระบบบริการแสดงสถานะแบบรวมศูนย์ (Unified Presence Service: UPS)

จากรูปแบบ Massive PI ของระบบการสื่อสารแบบรวมศูนย์ ซึ่งจำเป็นต้องมีการจัดการกับบัญชีรายชื่อของผู้ใช้ การรวบรวมข้อมูลแสดงสถานะให้เป็นข้อมูลแบบรวมศูนย์ และกฎการเข้าถึงข้อมูลและการแสดงผลที่ฉลาด เพื่อจะเพิ่มประสิทธิภาพการทำงานของระบบการสื่อสารแบบรวมศูนย์

วิทยานิพนธ์นี้จึงนำเสนอ กรอบการทำงานของระบบบริการแสดงสถานะแบบรวมศูนย์ ซึ่งประกอบด้วย 3 ฟังก์ชัน คือ ฟังก์ชัน Large-scale Presence Information ที่ใช้จัดการกับบัญชีรายชื่อในระบบการสื่อสารแบบรวมศูนย์ ฟังก์ชัน Presence Information Aggregation ที่ใช้ในการจัดการการรวบรวมข้อมูลแสดงสถานะจากช่องทางการสื่อสารต่าง ๆ ให้เป็นข้อมูลที่แสดงผลในรูปแบบรวมศูนย์ และฟังก์ชัน Smart Distribution and Policy Control ที่จัดการการเข้าถึงข้อมูลและการแสดงผลที่เพิ่มรูปแบบการนำเสนอข้อมูลแบบใหม่ ๆ เพื่อที่จะทำให้ผู้ใช้

สามารถรับทราบว่าจะต้องติดต่อไปยังผู้ใช้ปลายทางในช่องใดได้อย่างเหมาะสม โดยกรอบการทำงานของระบบการบริการแสดงสถานะแบบรวมศูนย์สามารถที่จะเขียนได้ดังรูปที่ 4-1



รูปที่ 4-1แสดงถึงฟังก์ชันภายในของ Unified Presence Service เพื่อที่จะใช้จัดการกับปัญหา Massive PI

4.2.1 ฟังก์ชัน Large-Scale of Presence Information

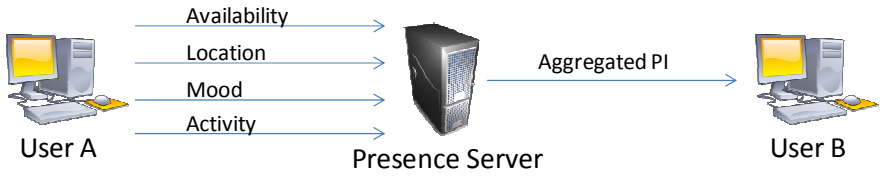
ฟังก์ชัน Large-Scale PI คือ ฟังก์ชันที่ใช้จัดการรูปแบบของบัญชีรายชื่อและข้อมูลการแสดงสถานะของผู้ใช้ เนื่องด้วยผู้ใช้แต่ละบุคคลจะมีข้อมูลการแสดงสถานะที่มากกว่าหนึ่ง และมีอุปกรณ์การสื่อสารที่มากกว่าหนึ่ง ด้วยเหตุนี้ทำให้ผู้ใช้จะมีบัญชีรายชื่อจำนวนมาก จึงจำเป็นต้องมีระบบที่คอยจัดการบัญชีรายชื่อดังกล่าวที่ได้จากช่องทางการสื่อสารต่าง ๆ โดยในการเก็บข้อมูลบัญชีรายชื่อของผู้ใช้ ระบบบริการแสดงสถานะแบบรวมศูนย์ได้ออกแบบ Unified Server ไว้ให้ผู้ใช้สามารถเพิ่มหรือลบข้อมูลของบัญชีรายชื่อของผู้ใช้ โดยบัญชีรายชื่อดังกล่าวจะรวมถึง URI ของ SIP, MSN, G-Talk หรือ Skype เป็นต้น นอกจากนี้ผู้ใช้จะกำหนดความสามารถในการเข้าถึงบัญชีรายชื่อแต่ละรายชื่อเอาไว้เป็นข้อมูลบริบท คือ Relationship และ Trust Level (จะอธิบายรายละเอียดของข้อมูลบริบทไว้ในหัวข้อ 4.2.3) เพื่อกำหนดว่าบุคคลที่จะสามารถเข้าถึงบัญชีรายชื่อแต่ละบัญชีจำเป็นต้องมีระดับ Relationship และ Trust Level เป็นเท่าไรจึงจะสามารถเข้าถึงบัญชีรายชื่อดังกล่าวได้ทั้งหมด

นอกจากการเข้าถึงบัญชีรายชื่อแล้วฟังก์ชัน Large-Scale Presence Information ยังออกแบบให้รองรับการสร้างข้อมูลแสดงสถานะรูปแบบใหม่ ๆ ของข้อมูลแสดงสถานะ โดยการสร้าง Plug-in เข้ากับ Terminal Platform ตัวอย่างเช่น ผู้พัฒนาได้ทำการสร้างการ Plug-in เพื่อเชื่อมต่อกับ GPS Location โดยข้อมูลดังกล่าวจะถูกแปลงให้อยู่ในรูป XML RPIDF [7] เหตุผลที่เลือก XML RPIDF เนื่องจากรูปแบบของ XML จะเป็นรูปแบบ Tree Structure ซึ่งง่ายต่อการ

กำหนดการเข้าถึง และสามารถสร้างความแตกต่างในการมองเห็นข้อมูลแสดงสถานะ หลังจากได้ข้อมูลดังกล่าว Terminal Platform นั้นจะทำการสร้างสัญญาณ SIP PUBLISH และเพิ่มข้อมูลแสดงสถานะที่อยู่ในรูปแบบ XML ไปใน Body ของสัญญาณดังกล่าว และส่งไปยัง Presence Server จึงทำให้ระบบการบริการแสดงสถานะแบบรวมศูนย์มีความยืดหยุ่นต่อรูปแบบข้อมูลแสดงสถานะในรูปแบบใหม่ ๆ

4.2.2 ฟังก์ชัน Presence Information Aggregation

ฟังก์ชัน Presence Information Aggregation คือ ฟังก์ชันที่ใช้ในการรวบรวมข้อมูลแสดงสถานะของผู้ใช้จากแหล่งข้อมูลต่าง ๆ เช่น อุปกรณ์ โปรโตคอล หรือรูปแบบกระบวนการทำงานที่มีความแตกต่างกัน เพื่อทำให้เป็นข้อมูลแสดงสถานะแบบรวมศูนย์ ดังรูปที่ 4-2 แสดงให้เห็นตัวอย่างโดย ผู้ใช้ A มีความสามารถในการแสดงข้อมูลสถานะของตนเองให้ผู้ที่ต้องการรับรู้ข้อมูลได้ถึง 4 ข้อมูล คือ Availability, Location, Mood และ Activity โดยข้อมูลแสดงสถานะทั้งหมดถูกส่งไปหา Presence Server ที่มีความสามารถในการรวบรวมข้อมูลแสดงสถานะ โดยจะนำข้อมูลแสดงสถานะทั้ง 4 ของผู้ใช้ A มาประมวลผลตามค่า User Preference ที่ผู้ใช้ A กำหนดไว้ แล้วจึงส่งข้อมูลแสดงสถานะแบบรวมศูนย์ไปให้กับผู้ใช้ B ได้รับทราบ ซึ่งจะทำให้ผู้ใช้ B สามารถที่จะเข้าใจสถานะของผู้ใช้ A ได้ง่ายขึ้น

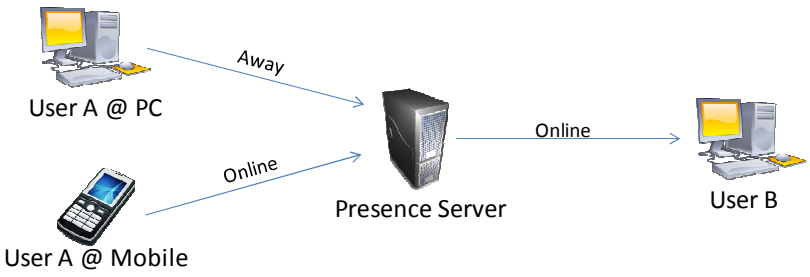


รูปที่ 4-2 แสดงหลักการของ PI Aggregation

ระบบจะอนุญาตให้ผู้ใช้ตั้งค่า User Preference ไว้เพื่อที่จะแก้ไขปัญหาในการขัดแย้งของข้อมูลแสดงสถานะ ดังตัวอย่างรูปที่ 4-3 ผู้ใช้ A ส่งข้อมูลแสดงสถานะของตนเองมาจาก 2 แหล่งที่มาดังนี้

1. PC ของผู้ใช้ A ส่ง Availability: “ไม่อยู่หน้าเครื่อง” มาให้ Presence Server
2. Mobile ของผู้ใช้ A ส่ง Availability: “ออนไลน์” มาให้ Presence Server

เมื่อ Presence Server ทำการรวบรวมข้อมูลแสดงสถานะให้เป็นแบบรวมศูนย์จากข้อมูลของผู้ใช้ A แล้วได้ออกมาเป็น Online จึงค่อยส่งข้อมูลแสดงสถานะไปบอกผู้ใช้ B ว่าผู้ใช้ A มีสถานะ Online เพราะว่า ผู้ใช้ A ยังมีช่องทางที่สามารถติดต่อได้อยู่ผู้ใช้ B จึงควรเห็นว่าผู้ใช้ A ยังออนไลน์อยู่ดังรูปที่ 4-3



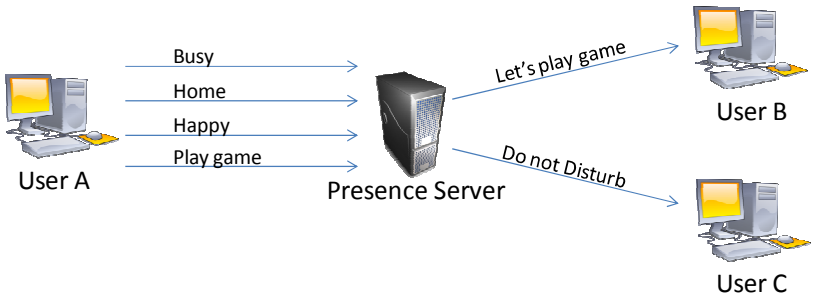
รูปที่ 4-3 การแก้ไขปัญหาขัดแย้งของข้อมูลแสดงสถานะจากสองแหล่ง

นอกจากการแก้ไขปัญหาความขัดแย้งของข้อมูลแสดงสถานะยังมีกระบวนการในการเพิ่มข้อมูลแสดงสถานะแบบพิเศษ (Deriving Presence Information) โดยข้อมูลดังกล่าวได้มาจากที่ผู้ใช้กำหนด ตัวอย่างเช่น ถ้าผู้ใช้ตั้งค่าข้อมูลแสดงถานะเป็น Idle ในช่วงเวลา 21.00-6.00 ให้เพิ่มข้อมูลกิจกรรมเป็น Sleeping ซึ่งจะเห็นได้ว่าข้อมูลดังกล่าวจะทำให้ผู้รับทราบไม่ทำการติดต่อไปยังผู้ใช้

ระบบบริการแสดงสถานะถูกออกแบบเพื่อรวบรวมข้อมูลแสดงสถานะใน 2 ส่วนด้วยกันคือ ส่วนของการรวมข้อมูลแสดงสถานะที่เครื่องลูกข่ายและส่วนของการรวมข้อมูลแสดงสถานะที่เครื่องแม่ข่ายกลาง โดยการรวบรวมข้อมูลแสดงสถานะที่เครื่องลูกข่ายใช้ในการรวบรวมข้อมูลแสดงสถานะจากระบบอื่น ๆ (Difference Platform) เช่น ข้อมูลแสดงสถานะจาก Public Instant Messenger (MSN, YaHoo, GTalk เป็นต้น) เนื่องด้วยข้อดีในเรื่องของลดการเชื่อมต่อได้มากกว่าแบบการสร้าง Gateway เชื่อมต่อกับระบบอื่น ๆ และคำนึงถึงความเป็นส่วนตัวของผู้ใช้ที่จำเป็นต้องบอกข้อมูลของ User Name และ Password ให้กับระบบ ส่วนการรวบรวมข้อมูลแสดงสถานะที่เครื่องแม่ข่ายใช้สำหรับการรวบรวมข้อมูลแสดงสถานะที่อยู่ในรูปแบบ SIP

การเพิ่มความยืดหยุ่นให้กับฟังก์ชันการรวบรวมข้อมูลแสดงสถานะ โดยการใช้ Context-awareness Computing มาประยุกต์เข้ากับการพิจารณาการรวบรวมข้อมูลแสดงสถานะระบบจะสามารถเพิ่มเติมการทำงานแบบใหม่ ๆ ของการรวบรวมข้อมูลแสดงสถานะ โดยการใช้ข้อมูลบริบท Relationship และ Trust Level (ที่จะอธิบายในหัวข้อฟังก์ชัน Smart Distribution and Policy Control) เข้ามาพิจารณาดังรูปที่ 4-4 เมื่อผู้ใช้ A กำหนดให้ผู้ใช้ B มี Relationship เป็นเพื่อน และผู้ใช้ C เป็นเพื่อนร่วมงาน จะเห็นได้ว่าในช่วงเวลาขณะที่ผู้ใช้ A กำลังเล่นเกมส์อยู่และเป็นเวลาส่วนตัว การรวบรวมข้อมูลแสดงสถานะจะให้ความสำคัญกับการเล่นเกมส์ของผู้ใช้ A

มากกว่าสถานะ Busy เมื่อผู้ใช้ B เป็นผู้มองสถานะของผู้ใช้ A ระบบจะแสดงว่าผู้ใช้ A กำลังเล่นเกมส์ แต่ในทางกลับกันผู้ใช้ C จะรับทราบว่าผู้ใช้ A ไม่อยากให้รบกวน ดังนั้นข้อมูลสถานะที่แสดงแก่ผู้ใช้ B และ C จะแตกต่างกันตามความสัมพันธ์ของผู้ใช้ทั้งสอง โดยผู้ใช้ B เห็นสถานะของ A ว่ากำลังเล่นเกมส์ และผู้ใช้ C เห็นสถานะของ A ว่าไม่ว่างและห้ามไม่ให้รบกวน



รูปที่ 4-4 ตัวอย่างการแสดงสถานะที่ผู้ใช้ B และ C ได้รับแตกต่างกัน แม้ว่าข้อมูลแสดงสถานะจะมาจากแหล่งเดียวกัน

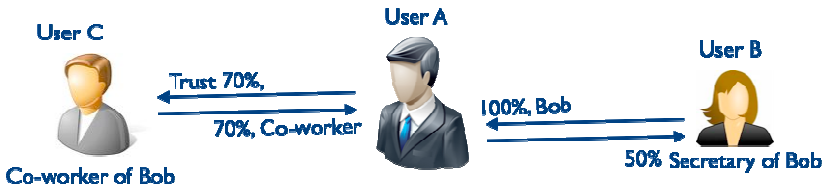
จากเหตุผลข้างต้นซึ่งแสดงให้เห็นถึงตัวอย่างการพิจารณาแบบ Context-awareness Computing ส่งผลให้ผู้ใช้ในระบบได้รับข้อมูลแสดงสถานะที่ต้องการแตกต่างกันตามความสัมพันธ์ของผู้ใช้งานในระบบ และผู้ใช้งานเหล่านั้นไม่ถูกรบกวนจากข้อมูลแสดงสถานะที่ไม่ต้องการ

4.2.3 ฟังก์ชัน Smart Distribution and Policy Control

ฟังก์ชัน Presence Information Distribution and Policy Control คือ ฟังก์ชันที่ใช้ในการจำกัดการเข้าถึงข้อมูลการแสดงสถานะที่มีความอ่อนไหว (Sensitive Information) ของผู้ใช้ เช่น ข้อมูลตำแหน่งที่อยู่ผู้ใช้ อารมณ์ หรือ กิจกรรมที่กำลังทำอยู่ในขณะนั้น โดยระบบบริการแสดงสถานะในปัจจุบัน [10] [11] สามารถจำกัดการเข้าถึงข้อมูลดังกล่าวได้ และยังมีความสามารถในการแบ่งช่วงเวลาในการเข้าถึงข้อมูลแสดงสถานะได้ แต่ความสามารถดังกล่าวที่ระบบแสดงสถานะปัจจุบันสามารถทำได้ เป็นเพียงการจำกัดสิทธิในการเข้าถึงข้อมูลแสดงสถานะที่ต้องการจะปกป้องเท่านั้น ความสามารถดังกล่าวจึงไม่สามารถนำมาใช้ในระบบการสื่อสารแบบรวมศูนย์ได้

เมื่อมองในมุมมองของระบบการสื่อสารแบบรวมศูนย์ที่ต้องการระบบการจำกัดการเข้าถึงและการแสดงข้อมูลจะต้องอยู่ในรูปแบบ Person-Centric [12] คือ ระบบไม่ได้บอกเพียงว่าข้อมูลที่แสดงมีอะไรบ้างและต้องกำหนดระดับการเข้าถึงเป็นอย่างไร แต่ระบบจำเป็นต้องบอกข้อมูลแสดงสถานะที่เหมาะสมและช่องทางที่เป็นไปได้มากที่สุดที่จะสามารถติดต่อไปยัง

บุคคลที่ผู้ใช้ต้องการได้ โดยที่ระบบจะไม่ทำให้บุคคลที่ได้รับการติดต่อรู้สึกถึงความไม่ปลอดภัยกับการแสดงข้อมูลดังกล่าว โดยตัวอย่างของการทำงานรูปแบบดังกล่าว คือ ผู้ใช้ A เป็นผู้แสดงสถานะ โดยขณะนั้นผู้ใช้ A มีข้อมูลแสดงสถานะเป็น Busy, At Office, Meeting และ Tired และผู้ใช้ B เป็นผู้แสดงสถานะเช่นกัน โดยมีข้อมูลแสดงสถานะคือ Available, At Office และ Happy และ ผู้ใช้ C เป็นผู้ที่ต้องการจะติดต่อกับผู้ใช้ A และบุคคลทั้งสามมีความสัมพันธ์กันดังรูปที่ 4-5 แสดงความสัมพันธ์ตัวอย่างระหว่าง ผู้ใช้ A ผู้ใช้ B และ ผู้ใช้ C



รูปที่ 4-5 แสดงความสัมพันธ์ตัวอย่างระหว่าง ผู้ใช้ A ผู้ใช้ B และ ผู้ใช้ C

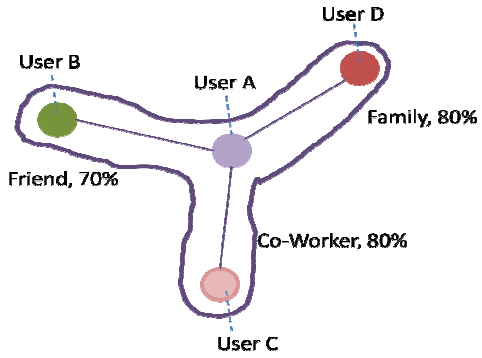
ความสัมพันธ์จากรูปที่ 4-5 ระบบจะพิจารณาข้อมูลที่ควรจะส่งไปให้ผู้ใช้ C รับทราบเป็น Busy, Meeting, At Office และเพิ่มข้อมูลของช่องทางในการที่จะสามารถติดต่อกับผู้ใช้ A ได้โดยการนำข้อมูลแสดงสถานะและช่องทางการสื่อสารของผู้ใช้ B เพิ่มเข้ามา Please Contact My Secretary: IM:Alice@Office1.com, Available ทำให้ผู้ใช้ C สามารถที่จะติดต่อผู้ใช้ A ได้ผ่านทางผู้ใช้ B

ระบบที่จะทำงานในรูปแบบดังกล่าวได้ จำเป็นต้องสร้างให้ผู้ใช้ภายในระบบมีความสัมพันธ์กัน และระบบต้องมีการพิจารณาที่มีความฉลาด ซึ่งจะสอดคล้องกับการทำงานในรูปแบบของ Context-awareness Computing ที่จะต้องมีข้อมูลบริบท ซึ่งในที่นี้คือความสัมพันธ์ระหว่างผู้ใช้ และ Ontology Model อธิบายความสัมพันธ์ของข้อมูล ด้วยเหตุนี้ระบบบริการแสดงสถานะแบบรวมศูนย์จึงเลือกใช้การทำงานในรูปแบบ Context-awareness Computing เพื่อรองรับการทำงานในส่วนของ Smart Distribution and Policy Control

4.2.3.1 Amoeba Model และ Presence Ontology Model

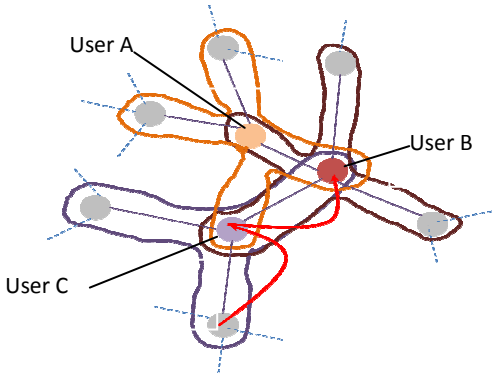
วิทยานิพนธ์นี้เสนอรูปแบบความสัมพันธ์ของผู้ใช้ภายในระบบ ซึ่งอยู่ในรูปแบบของ Amoeba Model ซึ่งใช้แสดงความสัมพันธ์ของผู้ใช้ในระบบบริการแสดงสถานะแบบรวมศูนย์ โดยเป็น Relationship และ Level of Trust ซึ่ง Relationship จะเป็นตัวบ่งบอกถึงความสัมพันธ์ระหว่างผู้ใช้ เช่น Co-worker, Friend, Family หรือ Master Student เป็นต้น ผู้ใช้สามารถที่จะสร้างรูปแบบความสัมพันธ์ตามที่ผู้ใช้ต้องการ และส่วนที่สอง คือ Level of Trust จะเป็นร้อยละของความสัมพันธ์ ที่จะบ่งบอกถึงความเชื่อใจของผู้ใช้ที่มีต่อบุคคลต่าง ๆ ตามความสัมพันธ์

ตัวอย่างรูปที่ 4-5 เมื่อเขียนเป็น Amoeba Model ผู้ใช้ A มีค่า Relationship กับผู้ใช้ C เป็น Co-Worker และมีความเชื่อใจหรือ Level of Trust เป็น 70% เมื่อเขียนภาพความสัมพันธ์ของผู้ใช้ในระบบ จะได้ดังรูปที่ 4-6



รูปที่ 4-6 Amoeba Model ของผู้ใช้ A

ความสัมพันธ์ของ Amoeba Model สามารถที่จะทำให้ระบบบริการแสดงสถานะแบบรวมศูนย์สร้างความเชื่อมต่อของผู้ใช้ในระบบเป็นกลุ่มใหญ่ โดยขาของ Amoeba Model จะเชื่อมต่อไปเรื่อย ๆ ซึ่งจะคล้ายกับการทำงานของระบบ Social Network ดังรูปที่ 4-7 ทำให้ระบบบริการแสดงสถานะแบบรวมศูนย์มีความสามารถที่จะใช้งานข้อมูลการแสดงสถานะของบุคคลที่อยู่นอกบัญชีรายชื่อของผู้ใช้ได้ (Friend of Friend)

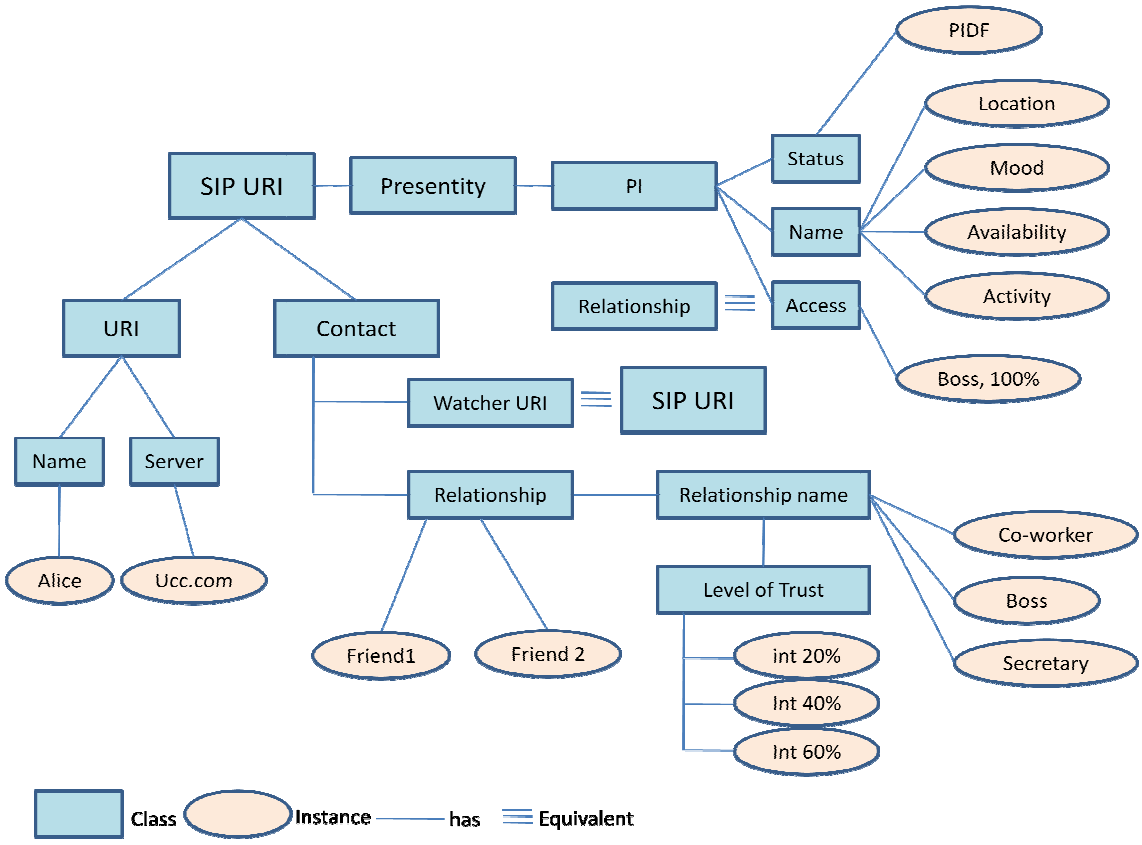


รูปที่ 4-7 ความสัมพันธ์ระหว่าง Amoeba Model ของผู้ใช้ในระบบ

เมื่อพิจารณาความสามารถดังกล่าวอาจจะทำให้บุคคลที่สามที่ถูกใช้ข้อมูลแสดงสถานะอาจจะเกิดความไม่ปลอดภัยได้ แต่เนื่องด้วยค่า Level of Trust ที่แสดงถึงระดับความเชื่อใจมีค่าสูงและความสัมพันธ์ที่เหมาะสมกับสถานการณ์ (ดังกรณีตัวอย่างในรูปที่ 4-5) ทำให้

ปัญหาดังกล่าวสามารถที่จะถูกจัดการได้ ด้วยเหตุผลนี้จึงทำให้ผู้ใช้ในระบบบริการแสดงสถานะแบบรวมศูนย์มีความสามารถที่จะติดต่อไปยังปลายทางได้มากยิ่งขึ้น

เมื่อพิจารณาส่วนของ Ontology Model นอกเหนือจากการทำงานเหมือนระบบการปกป้องแบบทั่วไป [10] [11] ระบบบริการแสดงสถานะแบบรวมศูนย์จำเป็นต้องเพิ่มการค้นหาข้อมูลแสดงสถานะข้ามไปยังบุคคลที่ 3 หรือ 4 ต่อไปตามที่ระบบหรือผู้ใช้ต้องการ โดยรูปแบบของ Presence Ontology Model ที่จะพิจารณาตัดสินใจจะพิจารณาข้อมูล 5 ส่วน คือ Relationship, Level of Trust, Presence Information Type, Watcher และ Presentity โดยสามารถวาดรูปของ Presence Ontology Model ได้ดังรูปที่ 4-8



รูปที่ 4-8 Presence Ontology Model สำหรับฟังก์ชัน Smart Distribution and Policy Control

4.3 สถาปัตยกรรมของระบบแสดงสถานะแบบรวมศูนย์

ในหัวข้อนี้จะอธิบายถึงแนวคิดการออกแบบและภาพรวมของสถาปัตยกรรมระบบบริการแสดงสถานะแบบรวมศูนย์ โดยจะอธิบายถึงรายละเอียดของแนวคิดในการออกแบบและการทำงานของส่วนที่มีการเพิ่มเติมความสามารถและการเชื่อมต่อ (Interface) ขององค์ประกอบต่าง ๆ (Network Components) ในระบบ

4.3.1 แนวคิดในการออกแบบสถาปัตยกรรม

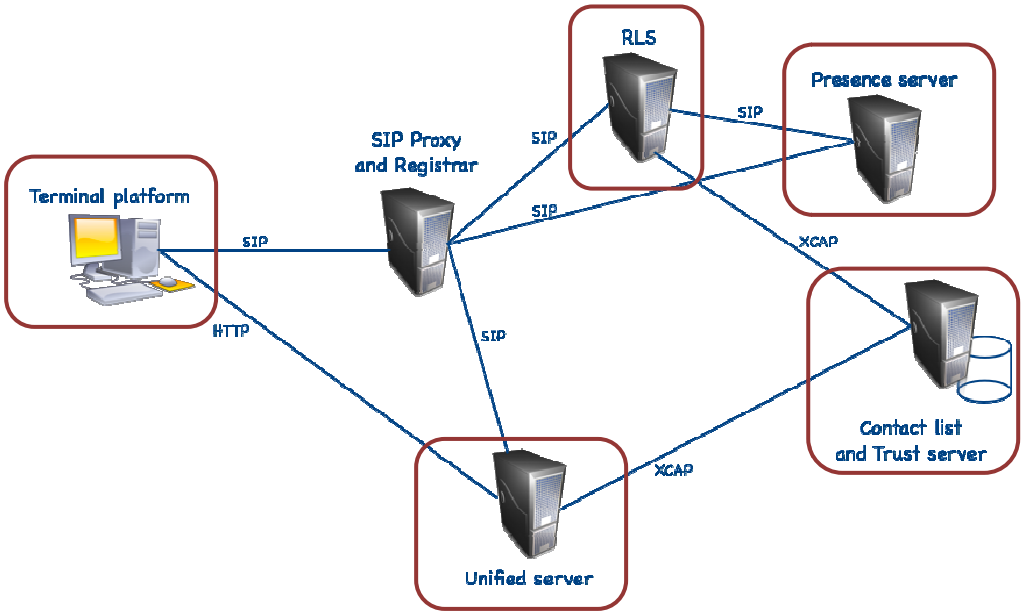
เพื่อให้เข้าใจถึงแนวความคิดของการออกแบบสถาปัตยกรรมระบบบริการแสดงสถานะแบบรวมศูนย์ ในหัวข้อนี้จะอธิบายถึงปัญหา ข้อเสียของระบบบริการแสดงสถานะโดยใช้ SIP แบบเดิมที่ถูกเลือกเป็นพื้นฐานสำหรับการสร้างสถาปัตยกรรมของระบบบริการแสดงสถานะแบบรวมศูนย์ จากความรู้ในบทที่ 2 ที่ได้อธิบายเกี่ยวข้องกับระบบบริการแสดงสถานะโดยใช้ SIP (SIMPLE) สถาปัตยกรรมแบบนี้มีข้อดีอยู่หลายประการ แต่อย่างไรก็ตามการที่ระบบการสื่อสารแบบรวมศูนย์ต้องพึ่งพาระบบบริการแสดงสถานะดังกล่าวในการแสดงสถานะของผู้ใช้ ซึ่งยังคงประสบกับปัญหาหรือข้อเสียที่เกี่ยวข้องกับ Massive PI ดังนี้

1. สถาปัตยกรรม SIMPLE ยังไม่มีการจัดการกับบัญชีรายชื่อของผู้ใช้ที่ไม่ใช่ SIP URI ซึ่งจะส่งผลให้ระบบบริการแสดงสถานะแสดงผลที่ผิดพลาดได้ เช่น ผู้ใช้อาจจะเพิ่มข้อมูลบัญชีรายชื่อที่ไม่ใช่บุคคลที่ผู้ใช้ต้องการ
2. สถาปัตยกรรม SIMPLE ยังไม่รองรับกับรูปแบบข้อมูลแสดงสถานะที่มีโปรโตคอลอื่น ๆ (Non-SIP) ซึ่งอาจจะทำให้ผู้ใช้ไม่สามารถที่จะรับข้อมูลจากช่องทางการสื่อสารหรืออุปกรณ์ที่ไม่ได้ทำงานอยู่บนพื้นฐานของ SIP ได้
3. สถาปัตยกรรม SIMPLE มีเพียงการจำกัดสิทธิ์การเข้าถึงข้อมูลแสดงสถานะของผู้ใช้เท่านั้น ซึ่งจะไม่สามารถเพิ่มเติมสิทธิ์ดังกล่าวได้ เช่น ผู้ใช้มีสิทธิ์ในการปิดกั้นข้อมูลแสดงสถานะเท่านั้น โดยไม่สามารถที่จะใช้ข้อมูลจากเพื่อนหรือบุคคลรอบข้างของผู้ใช้ได้ ดังตัวอย่างในรูปที่ 4-5

จากปัญหาข้างต้นทำให้ระบบบริการแสดงสถานะของสถาปัตยกรรม SIMPLE ที่จะใช้เป็นพื้นฐานในการออกแบบสถาปัตยกรรมของระบบบริการแสดงสถานะแบบรวมศูนย์จึงยังไม่เหมาะสมกับกรอบการทำงานที่ออกแบบไว้เพื่อแก้ไขปัญหาของ Massive PI ดังนั้นวิทยานิพนธ์นี้ได้มีการออกแบบเพิ่มเติมความสามารถในองค์ประกอบต่าง ๆ และได้ออกแบบองค์ประกอบใหม่ ที่จะใช้ในการจัดการปัญหาดังกล่าวให้เป็นไปตามกรอบการทำงานที่ได้อธิบายไว้ในหัวข้อ 4.2 ซึ่งจะแสดงรายละเอียดของการออกแบบสถาปัตยกรรมไว้ในหัวข้อ 4.3.2 และการออกแบบ

สถาปัตยกรรมดังกล่าวจะต้องอยู่บนพื้นฐานขององค์ประกอบที่เป็นมาตรฐานที่มีในปัจจุบันให้มากที่สุด (Exist Standard Components)

4.3.2 ภาพรวมของสถาปัตยกรรมของระบบบริการแสดงสถานะแบบรวมศูนย์



รูปที่ 4-9 ภาพรวมสถาปัตยกรรมของระบบการบริการแสดงสถานะแบบรวมศูนย์

จากรูปที่ 4-9 ได้แสดงองค์ประกอบต่างๆ ในสถาปัตยกรรม ได้แก่ Terminal Platform, SIP Proxy and Registrar Server, Unified Server, Resource List Server (RLS), Presence Server, และ Contact List and Trust Server และแสดงให้เห็นถึงส่วนที่มีการเพิ่มเติมความสามารถให้สถาปัตยกรรมเหมาะสมกับการทำงานสำหรับระบบการสื่อสารแบบรวมศูนย์ (ในกรอบสี่เหลี่ยม) โดยภาพรวมของสถาปัตยกรรมสามารถอธิบายได้ดังนี้

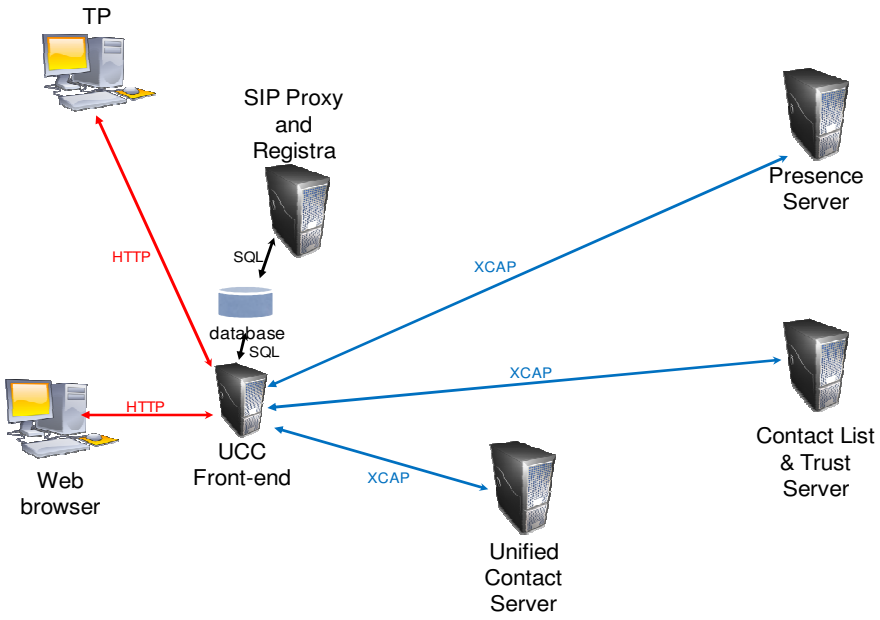
1. Terminal Platform คือ เครื่องลูกข่ายลูกข่ายของระบบการสื่อสารแบบรวมศูนย์ ซึ่งจะถูกใช้งานโดยผู้ใช้ทั่วไป โดยมีความสามารถในการรับค่า Username, Password เพื่อระบุตัวตนของผู้ใช้งาน ปรับแต่งค่าต่างๆ แสดงบัญชีรายชื่อของผู้ที่ต้องการจะติดต่อ ซึ่งสามารถแสดงผล Alias ของผู้ใช้งานคนอื่นๆ พร้อมทั้งแสดงสถานะของผู้ใช้งาน (Presence) ในรูปแบบของข้อมูลแสดงสถานะแบบรวมศูนย์ และมีความสามารถในการเชื่อมต่อกับผู้ใช้คนอื่นได้ทั้งในรูปแบบของการส่งข้อความภาพ เสียง ตัวอักษรและความสามารถในการแสดงผลห้องประชุมพร้อมทั้งอุปกรณ์อำนวยความสะดวกต่างๆ

2. SIP Proxy and Registrar server เปรียบเสมือนประตูที่ใช้สำหรับเชื่อมต่อ Terminal Platform เข้าสู่ระบบสื่อสารแบบรวมศูนย์ โดย SIP Proxy & Registrar Server จะทำหน้าที่ในการรับสัญญาณ SIP เริ่มการเชื่อมต่อ แล้วส่งต่อไปให้องค์ประกอบอื่น ๆ ในระบบการสื่อสารแบบรวมศูนย์ เพื่อประมวลผลสัญญาณ SIP ต่อไป
3. Unified Server จะเป็นเครื่องแม่ข่ายที่ถูกเพิ่มเติมจากพื้นฐานของ SIP ซึ่งประกอบด้วย 2 ส่วน คือ UCC Front-end ทำหน้าที่เป็นส่วนเชื่อมต่อระหว่างระบบการสื่อสารแบบรวมศูนย์และ TP โดยใช้สำหรับการตั้งค่าและอ่านข้อมูลต่าง ๆ ของระบบการสื่อสารแบบรวมศูนย์ โดย UCC Front-end จะสามารถรองรับการเชื่อมต่อได้ทั้งแบบ Web Based และ Application Based อีกส่วนประกอบ คือ Unified Contact Server ทำหน้าที่ในการเก็บค่า Alias ของผู้ใช้งานระบบการสื่อสารแบบรวมศูนย์ ซึ่งค่า Alias เกิดจากแนวคิดที่ว่าผู้ใช้งานหนึ่งคนสามารถมีบัญชีรายชื่อได้หลายรูปแบบ เช่น การติดต่อผู้ใช้คนหนึ่งสามารถติดต่อผ่านทาง Hotmail, G-mail, Facebook, Mobile Phone หรือ SIP Phone เป็นต้น โดย Unified Contact Server ทำหน้าที่ในการรวม Alias ต่าง ๆ ของผู้ใช้เหล่านั้นเอาไว้ในชื่อของบุคคล ๆ เดียว และแสดงผลเป็นบัญชีรายชื่อแบบรวมศูนย์ (Unified Contact)
4. Resource List Server (RLS) เป็นองค์ประกอบที่ใช้เพื่อเพิ่มประสิทธิภาพการทำงานของระบบบริการแสดงสถานะแบบรวมศูนย์ ในแง่ของการลดปริมาณแบนด์วิดท์ที่จะเกิดขึ้นที่ระบบ โดย RLS จะช่วยในการเก็บบัญชีรายชื่อและสอบถามข้อมูลแสดงสถานะของผู้ใช้แต่ละคนในบัญชีรายชื่อให้โดยอัตโนมัติ และนอกเหนือจากการทำงานพื้นฐานของ RLS ที่ได้กล่าวไว้ข้างต้น วิทยานิพนธ์นี้ได้ออกแบบให้ RLS ของระบบมีความสามารถเพิ่มเติมในการส่งข้อมูลบริบท (Relationship และ Level of Trust) ไปยัง Presence Server เพื่อจะให้ Presence Server ใช้ข้อมูลเพื่อตัดสินใจในการจำกัดสิทธิ์และแสดงข้อมูลแสดงสถานะที่เหมาะสมกับผู้ใช้
5. Presence Server ทำหน้าที่ในการเก็บค่าข้อมูลแสดงสถานะและแสดงผลไปยังผู้ใช้ และมีความสามารถในการกำหนดสิทธิของการเข้าถึงของข้อมูลแสดงสถานะได้อย่างชาญฉลาด (Smart Policy) โดย Presence Server จะมี Modules ที่จะใช้พื้นฐานการทำงานของระบบ Context-awareness Computing ในการตัดสินใจในการกำหนดสิทธิ์ และการแสดงผล
6. Contact list and Trust Server (XCAP Server [34]) โดยพื้นฐานจะทำหน้าที่สำหรับเก็บบัญชีรายชื่อของผู้ใช้ แต่ในระบบบริการแสดงสถานะแบบรวมศูนย์ได้เพิ่มเติมความสามารถของ XCAP Server ให้สามารถเก็บข้อมูลบริบท (Amoeba Model) ซึ่งข้อมูลดังกล่าวจะถูกแทรกไว้ในแต่ละบัญชีรายชื่อเดิมของ XCAP Server ของ SIP โดยข้อมูลบัญชีรายชื่อจะถูกเก็บไว้ในรูปแบบของ XML

ในหัวข้อถัดไปจะอธิบายถึงรายละเอียดขององค์ประกอบที่มีการเพิ่มเติมจากระบบการแจ้งเตือนของ SIP โดยประกอบด้วย Unified Server, RLS, Contact List and Trust Server และ Presence Server

4.3.3 Unified Server

วิทยานิพนธ์นี้ได้ออกแบบ Unified Server เพิ่มเติมจากระบบการแจ้งเตือนของ SIP เพื่อใช้ในการจัดการปัญหาที่เกี่ยวข้องกับบัญชีรายชื่อผู้ใช้ โดย Unified Server ประกอบด้วย 2 ส่วน คือ UCC Front-end ที่จะทำหน้าที่ในการรับค่า User Preference ที่เกี่ยวข้องกับบัญชีรายชื่อของผู้ใช้ และ Unified Contact Server ที่จะทำหน้าที่ในการจัดการบัญชีรายชื่อแบบรวมศูนย์ตามลำดับ



รูปที่ 4-10 การเชื่อมต่อกับองค์ประกอบต่างๆ ของ Unified Server

รูปที่ 4-10 แสดงถึงการเชื่อมต่อของ UCC Front-end และ Unified Server กับองค์ประกอบอื่นๆ ในระบบบริการแสดงสถานะแบบรวมศูนย์ โดยรูปแบบของการทำงานจะอยู่ในลักษณะของ Web Application ซึ่ง UCC Front-end จะมีหน้าที่ในการ สอบถามเพิ่มเติม หรือแก้ไขข้อมูลที่ผู้ใช้ป้อนเข้ามาผ่านทาง UCC Front-end โดยองค์ประกอบที่ UCC Front-end จะติดต่อใช้งานด้วย มีดังนี้

1. Account Database เป็น ฐานข้อมูลที่ใช้ในการเก็บชื่อผู้ใช้และรหัสผ่าน สำหรับการติดต่อในส่วนนี้จะทำให้ UCC Front-end สามารถทำ Authentication โดยใช้ชื่อผู้ใช้และรหัสผ่านชุดเดียวกับการใช้งาน SIP
2. Contact List and Trust Server ในการติดต่อส่วนนี้จะทำให้ผู้ใช้สามารถจะเก็บบัญชีรายชื่อของผู้ติดต่อที่ผู้ใช้ต้องการจะรับทราบสถานะ โดยแต่ละรายชื่อผู้ใช้สามารถที่จะระบุข้อมูลบริบทที่แสดงถึง Relationship และ Level of Trust ได้เป็นรายบุคคล
3. Presence Server ในการติดต่อกับ Presence Server เป็นการเปิดช่องทางให้ผู้ใช้ สามารถกำหนดค่าเริ่มต้นต่าง ๆ ที่ใช้กำหนดการเข้าถึงข้อมูลแสดงสถานะที่จะแสดงให้ผู้ใช้คนอื่นได้รับทราบ
4. TP จะติดต่อกับ UCC Front-end โดยผ่านโปรโตคอล HTTP เป็นหลัก ซึ่ง TP ในที่นี้เป็น Web Browser เมื่อผู้ใช้ทำการติดต่อเข้ามาผ่านทางหน้าเว็บ เพื่อทำการแก้ไขข้อมูลพื้นฐานของผู้ใช้

4.3.3.1 รายละเอียดการทำงานของ UCC Front-end

UCC Front-end ได้ออกแบบให้เป็นจุดเชื่อมต่อผู้ใช้ ซึ่งพัฒนาในรูปแบบของ Web Application Server ที่มีหน้าที่ในการติดต่อกับผู้ใช้เพื่อที่จะรับข้อมูลต่างจากผู้ใช้ ซึ่งจะมีส่วนประกอบสำหรับรับข้อมูลต่าง ๆ ดังนี้

4.3.3.1.1 ข้อมูล Sign Up

เป็นส่วนที่ใช้สำหรับการสมัครเพื่อที่จะใช้งานระบบการสื่อสารแบบรวมศูนย์ ซึ่งจะต้องมีการกรอกข้อมูลต่าง ๆ ดังนี้

1. Sip Account ที่ต้องการใช้งาน ซึ่งเป็นบัญชีรายชื่อหลักในการใช้งานความสามารถต่าง ๆ ที่มีอยู่ในระบบการสื่อสารแบบรวมศูนย์
2. ข้อมูลส่วนตัวของผู้ใช้ เช่น ชื่อ-สกุล อายุ อาชีพ ฯลฯ
3. Unified Display Name คือ ชื่อที่ใช้แทนชื่อจริงของตนเอง ซึ่ง Unified Display Name ถูกนำไปแสดงอยู่ในบัญชีรายชื่อของผู้ใช้ที่กำลังติดต่อ ซึ่งจะแสดงขึ้นมาแทนที่ชื่อของ Alias

4.3.3.1.2 ข้อมูล Account Profile

เป็นส่วนที่ใช้สำหรับการแก้ไขและเพิ่มเติมข้อมูลของ Alias ที่ผู้ใช้ใช้งานอยู่ เพื่อที่จะส่งต่อข้อมูลดังกล่าวไปให้กับ Unified Contact Server เพื่อใช้ในการทำ Unified Contact โดยจะมีข้อมูลที่สามารถเพิ่มและแก้ไขได้ดังต่อไปนี้

1. Alias คือ ช่องทางการติดต่อที่สามารถใช้ในการติดต่อหาผู้ใช้ได้ เช่น E-mail เบอร์โทรศัพท์ เป็นต้น
2. Service คือ ระบบของการบริการที่ Alias นั้น ๆ ใช้งานอยู่ เพื่อที่จะใช้ในการติดต่อหาผู้ใช้ได้ถูกต้องตามระบบที่ Alias นั้น ๆ ใช้งาน เช่น A@hotmail.com เป็น Alias ที่ผู้ใช้ใช้งานกับโปรแกรม MSN Messenger ดังนั้นจึงมี Service เป็น MSN ซึ่งหมายถึง Alias a@hotmail.com ใช้งานอยู่บนระบบ MSN และ A@gmail.com เป็น Alias ที่ผู้ใช้ใช้งานกับโปรแกรม Google GTalk ดังนั้นจึงมี Service เป็น GTalk ซึ่งหมายถึง Alias a@hotmail.com ใช้งานอยู่บนระบบ GTalk
3. Priority คือ ลำดับความสำคัญในการเลือกการติดต่อ โดยระบบจะเลือกติดต่อช่องทางที่มี Priority สูงก่อนเสมอ และทุก Alias จะต้องมี Priority ของตัวเอง ซึ่งค่า Priority นี้จะต้องให้ผู้ใช้เป็นผู้กำหนดเนื่องจากผู้ใช้จะสามารถเลือกได้ว่า อยากให้ผู้อื่นติดต่อเข้ามาหาด้วยช่องทางใดมากที่สุด ซึ่งก็คือ ช่องทางที่มีค่า Priority ที่สูงที่สุดนั่นเอง
4. Relationship and Level of Trust คือ ความสัมพันธ์และระดับความน่าเชื่อถือขั้นต่ำที่ Unified Contact Server ใช้ในการคำนวณสิทธิ์การเข้าถึง Alias นั้น ๆ ซึ่ง Unified Contact Server จะนำข้อมูลของความสัมพันธ์และระดับความน่าเชื่อถือที่กำหนดนี้ไปเป็นค่าในการคำนวณสิทธิ์ในการเข้าถึง Alias ของผู้ที่ต้องการจะเพิ่มรายชื่อ
5. Share Tag เป็นตัวระบุว่า Alias นั้น ๆ มีการใช้งานร่วมกับผู้อื่นหรือไม่ เพื่อให้ผู้ที่ติดต่อต้องการเข้ามาได้รับรู้ก่อนว่า ในการติดต่อครั้งนั้น ๆ อาจจะไม่ได้พบกับผู้ใช้ที่ต้องการติดต่อโดยทันที เช่น ในกรณีของโทรศัพท์บ้าน เป็นต้น

4.3.3.1.3 ข้อมูล Contact List

เป็นส่วนที่ใช้สำหรับการแก้ไข และเพิ่มเติมข้อมูลของบัญชีรายชื่อของผู้ใช้ ซึ่งจะเก็บ Alias และข้อมูลของบุคคลที่ผู้ใช้ทำการติดต่อด้วย เพื่อที่จะส่งต่อข้อมูลไปให้กับ Contact List and Trust Server ที่จะทำหน้าที่ในการจัดเก็บข้อมูลดังกล่าว โดยจะมีข้อมูลดังต่อไปนี้

1. Unified Contact คือ ช่องทางการติดต่อที่ผู้ใช้ใช้ในการติดต่อหาผู้อื่น โดยจะเก็บข้อมูลไว้ในรูปแบบของ Unified Contact ซึ่งจะต้องมี Primary Contact (Sip Address)ในการอ้างอิงถึง Alias อื่น ๆ ที่เป็นเสมือน Contact ย่อย ๆ ของ Primary Contact
2. Relationship and Level of Trust คือ ความสัมพันธ์ และค่าความน่าเชื่อถือ ซึ่งใช้ในการคำนวณสิทธิ์ในการเข้าถึงข้อมูล Unified Contact ของผู้ใช้ และคำนวณสิทธิ์ในการรับทราบข้อมูลแสดงสถานะแต่ละอย่างของผู้ใช้

4.3.3.2 Unified Contact Server

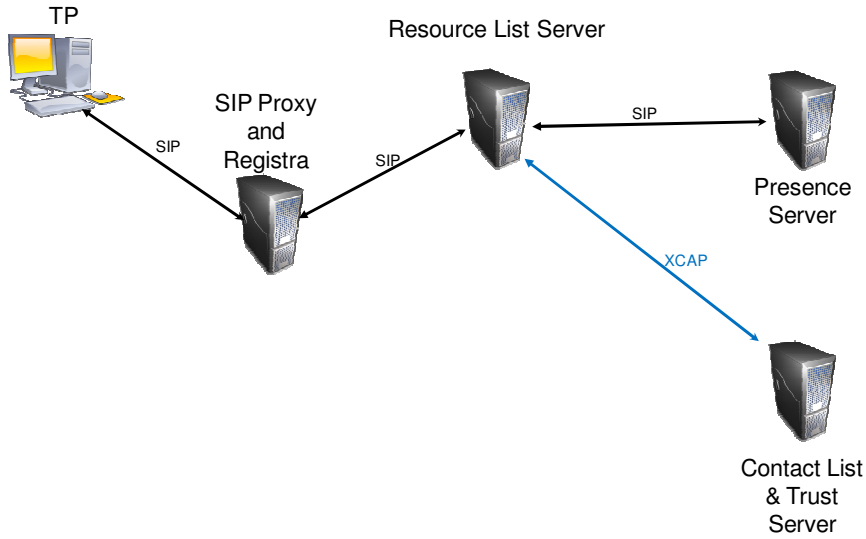
ในปัจจุบันการติดต่อสื่อสารระหว่างกันมีอยู่มากมายหลากหลายรูปแบบ ทำให้เกิดความยุ่งยากในหลายด้าน เมื่อแต่ละบุคคลมีการใช้เทคโนโลยีในการสื่อสารที่หลากหลายและแตกต่างกันออกไป ทำให้ผู้ที่ต้องการจะติดต่อด้วยไม่สามารถทราบได้ว่า ควรจะใช้ช่องทางใดในการติดต่อจึงจะเป็นทางช่องที่ดีที่สุด จึงทำให้เกิดปัญหาและทำให้เสียเวลาในการติดต่อ วิทยานิพนธ์นี้ได้มีการนำ Unified Contact มาใช้ในการแก้ไขปัญหาต่าง ๆ ทำให้ผู้ที่จะติดต่อไม่จำเป็นต้องสนใจว่า จะใช้ช่องทางใดในการติดต่อ เพียงแค่เลือกชื่อของผู้ที่ต้องการจะติดต่อ แล้วทำการสั่งเริ่มการติดต่อ (Person Centric) โดยที่โปรแกรมจะทำการตัดสินใจเลือกช่องทางที่ดีที่สุดให้โดยอัตโนมัติ ดังนั้นจึงต้องมี Server ที่มีความสามารถในการรับรู้ได้ว่า ผู้ใช้แต่ละคนนั้นใช้งานช่องทางใดบ้างในการติดต่อ เพื่อที่จะให้ TP ใช้เป็นแหล่งอ้างอิงว่าผู้ที่จะต้องทำการติดต่อด้วยมีช่องทางในการติดต่อใด

4.3.3.2.1 การทำงานของ Unified Contact Server

1. มีความสามารถในการเก็บรวบรวม Alias และรายละเอียดต่าง ๆ ของ Alias
2. มีความสามารถในการค้นหาว่าผู้ใช้แต่ละคนมี Alias ใดบ้าง
3. มีความสามารถในการค้นหาว่า Alias ที่ระบุไว้เป็นของผู้ใช้คนใด
4. มีความสามารถในการรับการติดต่อเพื่อร้องขอข้อมูล Unified Contact จาก TP หรือ Web Browser
5. มีความสามารถในการร้องขอข้อมูลของ Relationship and Level of Trust จาก Contact List and Trust Server เพื่อที่จะนำมาใช้ในการคำนวณสิทธิ์ของผู้ที่เข้ามาขอข้อมูล Unified Contact
6. มีความสามารถในการประมวลผลว่า ผู้ที่เข้ามาทำการร้องขอข้อมูล Unified Contact ของผู้ใช้ มีสิทธิ์ที่จะเข้าถึงข้อมูลได้มากหรือน้อยเพียงใด โดยจะใช้หลักการในการคำนวณสิทธิ์ของผู้ที่ร้องขอข้อมูลดังนี้ ตรวจสอบว่าผู้ที่ทำการร้องขอข้อมูล Unified Contact ของผู้ใช้มีความสัมพันธ์และค่าความน่าเชื่อถือเป็นอย่างไร ต่อมานำค่าความสัมพันธ์ที่ได้มาไปเปรียบเทียบกับความสัมพันธ์ที่กำหนดไว้ในแต่ละ Alias ว่าตรงกันหรือไม่ เมื่อความสัมพันธ์ถูกต้องแล้วจึงเปรียบเทียบค่าความน่าเชื่อถืออีกครั้งว่า มีความน่าเชื่อถือมากเพียงพอหรือไม่ ซึ่งถ้ามีมากเพียงพอก็จะกำหนดได้ว่า ผู้ที่ร้องขอมามีสิทธิ์เพียงพอที่จะได้รับข้อมูล Alias นั้น ๆ ไปได้ และคำนวณซ้ำอีกครั้งกับ Alias อื่น ๆ จนครบทุก Alias ของผู้ใช้ แล้วจึงค่อยทำการส่งเฉพาะรายชื่อของ Alias ที่ผู้ร้องขอมีสิทธิ์ที่จะรับรู้ได้กลับไป

4.3.4 Resource List Server (RLS)

นอกเหนือจากการทำงานพื้นฐานของ RLS วิทยานิพนธ์นี้ได้ออกแบบให้ RLS มีกระบวนการทำงานเพิ่มเติมเพื่อการนำส่งข้อมูลบริบทต่าง ๆ ที่ใช้ในการทำงานของ Presence Server โดยรายละเอียดการทำงานของ RLS ดังรูปที่ 4-11



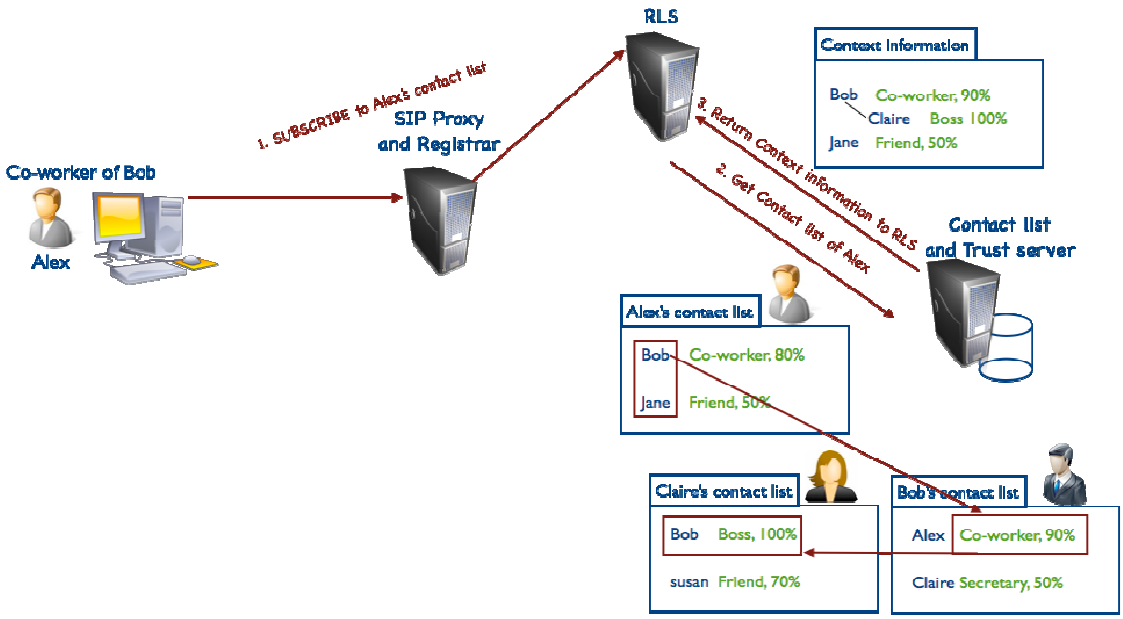
รูปที่ 4-11 การเชื่อมต่อกับองค์ประกอบต่าง ๆ ของ RLS

จากรูปที่ 4-11 แสดงว่า TP มีการติดต่อเพื่อร้องขอข้อมูลแสดงสถานะจาก Presence Server โดยในการร้องขอข้อมูลแต่ละครั้งจะมีการติดต่อผ่าน Resource List Server หรือ RLS ซึ่งรายละเอียดในการติดต่อระหว่างผู้ใช้งานกับ RLS และ RLS กับ Presence Server มีดังนี้

1. TP ทำการติดต่อเพื่อเข้ามาใช้งานในระบบผ่านทาง SIP Proxy และเมื่อมีการตรวจสอบและยืนยันตัวตนแล้วก็จะแจ้งการติดต่อไปยัง RLS เพื่อให้ทำการติดต่อไปยัง Presence Server โดยการทำงานดังกล่าวจะผ่านทางโปรโตคอล SIP
2. ก่อนที่ RLS จะติดต่อไปยัง Presence Server จะต้องมีการติดต่อไปยัง Contact List and Trust Server เพื่อรับทราบบัญชีและข้อมูลบริบทของผู้ใช้งาน โดยการทำงานจะผ่านทางโปรโตคอล XCAP
3. เมื่อ RLS ได้บัญชีและข้อมูลบริบทมาแล้วก็ทำการติดต่อไปยัง Presence Server เพื่อขอข้อมูลสถานะของผู้ใช้งานคนอื่น ๆ ที่อยู่ในบัญชีรายชื่อที่ได้จาก Contact List and Trust Server โดยการทำงานจะผ่านทางโปรโตคอล SIP

4. เมื่อ RLS ได้ข้อมูลมาแล้วก็จะทำการรวมข้อมูลของ TP แต่ละ TP ให้เป็นข้อมูลชิ้นเดียว และทำการส่งไปยังผู้ใช้งานที่เรียกขอข้อมูล โดยการทำงานดังกล่าวจะผ่านทางโปรโตคอล SIP
5. ผู้ใช้งานที่เรียกขอข้อมูลก็จะอ่านข้อมูลที่ได้รับมากจาก RLS และแสดงค่าเหล่านั้นต่อไป

นอกเหนือจากการทำงานในรูปแบบปกติของ RLS ระบบบริการแสดงสถานะแบบรวมศูนย์ได้ออกแบบให้ RLS สามารถที่จะนำส่งข้อมูลบริบท คือ Relationship และ Level of Trust โดยการดึงข้อมูลที่เก็บไว้ใน Contact List and Trust server ผ่านทางกระบวนการรับบัญชีรายชื่อของ RLS แบบปกติ จากนั้นจึงเพิ่มข้อมูลบริบทดังกล่าวไว้ในส่วนของ Body ของสัญญาณ SUBSCRIBE ที่ส่งไปยัง Presence Server โดยข้อมูล Relationship และ Trust จะเป็นข้อมูลของบุคคลที่ถูกมองสถานะดังรูปที่ 4-12



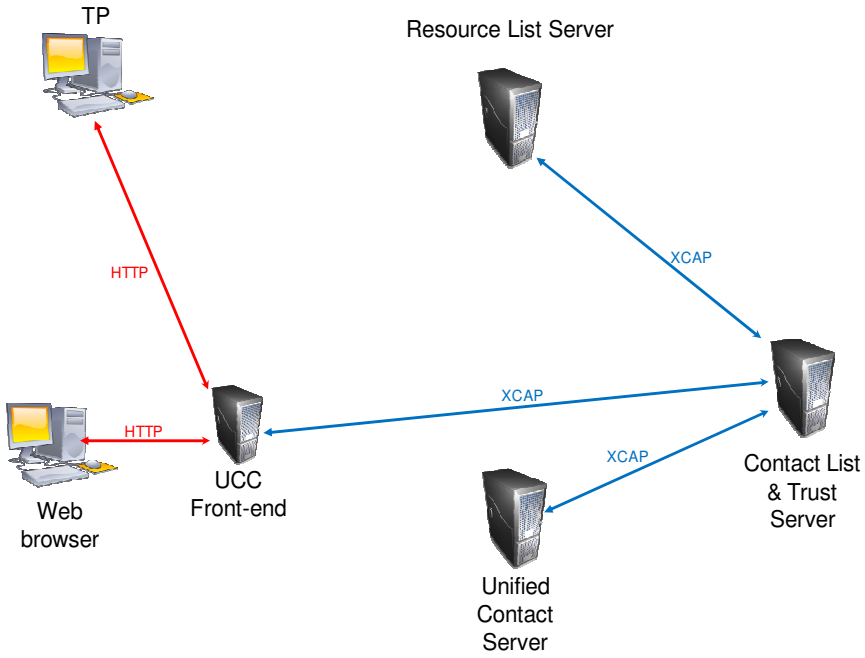
รูปที่ 4-12 RLS ดึงข้อมูลจาก Contact List and Trust Server

จากรูปที่ 4-12 ได้แสดงตัวอย่างการทำงาน โดย Alex เป็นผู้มองสถานะของ Bob ซึ่ง Claire เป็นบุคคลที่มีความสัมพันธ์กับ Bob บุคคลทั้งสามมีชุดบัญชีรายชื่อที่เก็บอยู่ใน Contact List and Trust Server โดยบัญชีรายชื่อได้แทรกข้อมูลบริบท (Relationship และ Level of Trust) ลงไป โดยกระบวนการในรูปที่ 4-12 จะมีลำดับขั้นตอนดังนี้

1. Alex ได้ทำการ SUBSCRIBE ที่ระบุว่า เป็นการ SUBSCRIBE ไปยังบัญชีรายชื่อของ Alex ซึ่งเมื่อ SIP Proxy ได้รับสัญญาณ SUBSCRIBE ก็จะส่งต่อสัญญาณไปให้ RLS เพื่อดำเนินการต่อไป
2. เมื่อ RLS ได้รับสัญญาณ SUBSCRIBE แล้ว RLS ก็จะดึงข้อมูลบัญชีรายชื่อของ Alex และดึงข้อมูลบริบทของบุคคลที่ Alex ต้องการ SUBSCRIBE ที่ได้ระบุ Relationship และ Level of Trust ที่มีต่อ Alex ซึ่งในที่นี้ Bob ถูกกำหนดเป็น Co-Worker, 90% และ RLS ถูกกำหนดให้สามารถดึงข้อมูลบุคคลที่อยู่ใน Relationship ของ Bob และมี Level of Trust ที่มีค่าสูงต่อ Bob ทำให้ RLS ดึงข้อมูลของ Claire ที่มีความสัมพันธ์กับ Bob ส่งไปเป็น Sub Context Information ที่ Bob มีต่อ Alex
3. เมื่อได้ข้อมูลบริบททั้งหมดที่กล่าวมาแล้ว RLS ก็จะส่งข้อมูลบริบทไปพร้อมกับ สัญญาณ SUBSCRIBE โดยเพิ่มเข้าไปใน Body ของสัญญาณ แล้วส่งไปให้ Presence Server ตัดสินใจต่อไป

4.3.5 Contact List and Trust Server

Contact List and Trust Server คือ XCAP Server ที่เก็บข้อมูลบริบทเพิ่มเติม โดยปกติแล้ว XCAP Server จะทำหน้าที่ในการเก็บบัญชีรายชื่อของผู้ใช้สำหรับกระบวนการทำงานของ RLS เท่านั้น แต่ในวิทยานิพนธ์นี้ได้ออกแบบให้ XCAP Server เก็บข้อมูลบริบทเพิ่มเติม โดยการแทรกข้อมูลดังกล่าวเข้าไปในบัญชีรายชื่อปกติ เพื่อบ่งบอกความสัมพันธ์และระดับความเชื่อใจ (Amoeba Model) และ RLS จะนำข้อมูลดังกล่าวส่งไปให้ Presence Server ผ่านทางสัญญาณ SUBSCRIBE เพื่อให้ Presence Server ใช้ในกระบวนการ Presence Reasoner Modules ต่อไป ซึ่งรูปแบบการเชื่อมต่อและการทำงานของ Contact List and Trust Server มีดังนี้



รูปที่ 4-13 การเชื่อมต่อกับองค์ประกอบต่าง ๆ ของ Contact List and Trust Server

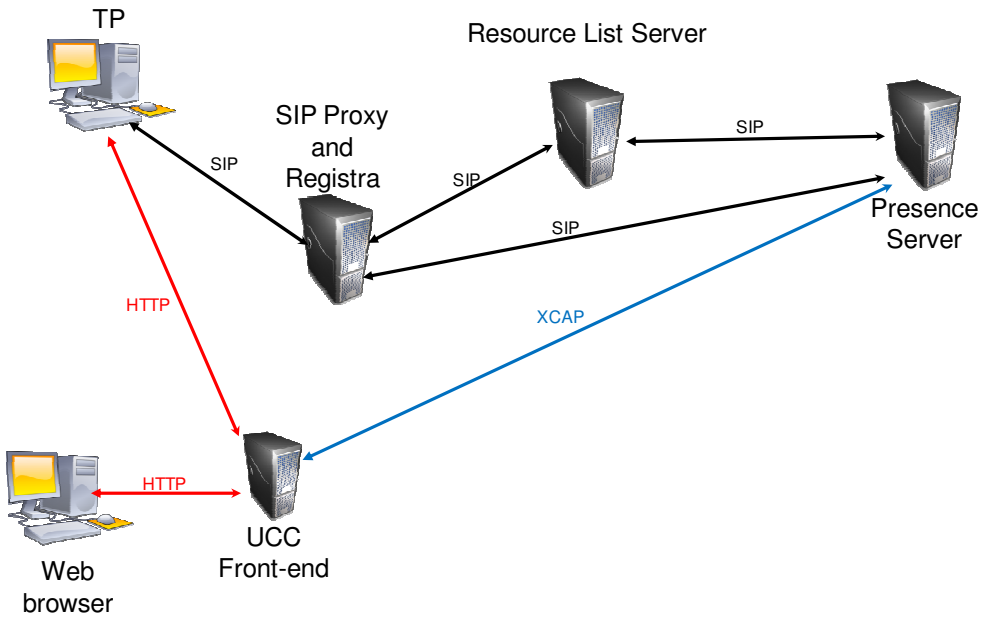
จากรูปที่ 4-13 แสดงการเชื่อมต่อของ Contact List and Trust Server กับองค์ประกอบต่าง ๆ ของระบบบริการแสดงสถานะแบบรวมศูนย์ ซึ่งสามารถแบ่งการเชื่อมต่อได้ 3 ทาง คือ RLS, UCC Front-end และ Unified Contact Server โดยใช้การเชื่อมต่อแบบ XCAP โปรโตคอลเหมือนกัน แต่มีลักษณะการใช้งานที่แตกต่างกัน

1. การใช้งานกับ RLS แบ่งเป็นสองแบบ คือ การเข้าถึงรายชื่อของผู้ใช้ที่เป็นคนร้องขอ (Private Contact List) และการเข้าถึงข้อมูลรายชื่อร่วมกัน (Share Contact List) ในแบบที่หนึ่ง คือ RLS จะดึงข้อมูล (ผ่านสัญญาณ XCAP GET) รายชื่อและข้อมูลบริบทของผู้ใช้จาก Contact List and Trust Server เพื่อนำรายชื่อและข้อมูลบริบทเหล่านั้นมา SUBSCRIBE ในแบบที่สอง คือ RLS จะดึงข้อมูลระดับความน่าเชื่อถือของผู้ใช้ที่มีต่อรายชื่อที่ใช้ร่วมกันเอาไว้ก่อน จากนั้นระบบจะตัดสินใจว่าจะให้บัญชีรายชื่อเหล่านั้นหรือไม่
2. การใช้งานกับ Unified Server แบ่งเป็นสองแบบ คือ การทำงานผ่านทาง UCC Front-end และการทำงานผ่านทาง Unified Contact Server ในแบบที่หนึ่งเป็นการใช้งานผ่านทาง UCC Front-end ระบบจะอนุญาตให้ผู้ใช้ (Terminal Platform และ Web Browser) สามารถแก้ไขบัญชีรายชื่อ ระดับความเชื่อใจ ที่อยู่ของผู้ใช้ และค่า User Preference ที่ผู้ใช้กำหนดขึ้นต่อข้อมูลแสดงสถานะผ่านทาง UCC Front-end ส่วนในแบบที่สองเป็น

การใช้งานกับ Unified Contact Server เป็นกรณีที่ผู้ใช้ต้องการจะเพิ่มหรือลบบัญชีรายชื่อ นอกจากนี้ Unified Contact Server มีหน้าที่ในแสดงข้อมูลบัญชีรายชื่อในรูปแบบของ Unified Contact โดย Unified Contact Server จะดึงข้อมูล Alias จาก Contact list and Trust Server

4.3.6 Presence Server

นอกเหนือจากการทำงานแบบปกติของ Presence Server ซึ่งเป็นเครื่องแม่ข่ายที่ทำหน้าที่เก็บข้อมูลแสดงสถานะจากกระบวนการ Publication และการแสดงผลข้อมูลผ่านทางกระบวนการ Subscription และ Notification แต่ในวิทยานิพนธ์นี้ออกแบบให้ Presence Server สามารถเพิ่มรูปแบบการแสดงผลแบบใหม่ๆ โดยการแก้ไขกระบวนการ Subscription จาก RLS ให้สามารถนำส่งข้อมูลความสัมพันธ์ (Amoeba Model) ของผู้แสดงสถานะและผู้มองสถานะ เพื่อให้ Presence Server ใช้ข้อมูลความสัมพันธ์ดังกล่าวในการพิจารณาข้อมูลแสดงสถานะที่จะถูกส่งไปให้ผู้มองสถานะ ตัวอย่างของความสามารถใหม่ๆ เช่น การแสดงข้อมูลแสดงสถานะที่ไม่ได้อยู่ในบัญชีรายชื่อของผู้ใช้ หรือการติดต่อโดยผ่านตัวแทน เป็นต้น โดยรูปแบบการทำงานของ Presence Server จะแสดงการเชื่อมต่อ และ Modules การทำงานภายในดังนี้



รูปที่ 4-14 การเชื่อมต่อกับองค์ประกอบต่างๆ ของ Presence Server

จากรูปที่ 4-14 แสดงการเชื่อมต่อของ Presence Server ซึ่งติดต่อไปยังองค์ประกอบอื่น ๆ 3 ช่องทาง คือ Unified Server, SIP Proxy และ RLS โดยการติดต่อแต่ละช่องทางมีรายละเอียดดังนี้

1. การทำงานร่วมกับ Unified Server เนื่องจากระบบอนุญาตให้ผู้ใช้สามารถที่จะกำหนดสิทธิในการเข้าถึงข้อมูลแสดงสถานะของตนได้ โดยผู้ใช้งานระบบสามารถกำหนดสิทธิผ่านทาง UCC Front-end ซึ่งทำงานเชื่อมต่ออยู่กับ Presence Server โดย Presence Server จะทำหน้าที่ในการดึงข้อมูลสิทธิที่ส่งมาผ่านทาง UCC Front-end และเมื่อผู้ใช้งานต้องการทราบถึงสิทธิต่าง ๆ ที่ได้มีการกำหนดไว้ก็สามารถที่จะทำการร้องขอไปยัง Presence Server เพื่อดึงข้อมูลออกมาแสดงทาง UCC Front-end ได้ สำหรับข้อมูลสิทธิถูกจัดเก็บอยู่ในรูปแบบของ XML ดังนั้นโปรโตคอลที่ใช้ในการติดต่อระหว่าง Presence Server และ UCC Front-end จะเป็นโปรโตคอล XCAP ที่มีความยืดหยุ่นและเหมาะสมในการดึงหรือเก็บข้อมูลที่อยู่ในรูปแบบ XML
2. การทำงานร่วมกับ SIP Proxy เป็นการทำงานทางตรงโดยไม่จำเป็นต้องผ่าน RLS เพื่อสนับสนุนการทำงานของ TP ที่ไม่รองรับการทำงานแบบ RLS โดยโปรโตคอลที่ใช้ในการเชื่อมต่อ คือ SIP
3. การทำงานร่วมกับ RLS ถูกนำมาใช้ เพื่อลดภาระหน้าที่ของ SIP Proxy ในการส่งสัญญาณที่จะเกิดขึ้นระหว่าง TP และ Presence Server โดย RLS จะทำหน้าที่ในการร้องขอข้อมูลแสดงสถานะแทน TP ตามบัญชีรายชื่อที่ TP ได้เก็บเอาไว้ในระบบ โดยโปรโตคอลที่ใช้ในการเชื่อมต่อ คือ SIP

การทำงานภายในของ Presence Server สามารถแบ่งได้เป็น 3 ส่วน คือ Presence Module ทำหน้าที่ในการจัดการกับข้อมูลแสดงสถานะทั่วไป ส่วนต่อมา Presence Reasoner Module ทำหน้าที่ในการประมวลผลสิทธิการเข้าถึงข้อมูลแสดงสถานะจากข้อมูลบริบทที่ได้รับมาจาก RLS และส่วนสุดท้ายคือ PI Aggregation Module ทำหน้าที่ในการรวบรวมข้อมูลแสดงสถานะของ SIP

4.3.6.1 Presence Module

Presence Module เป็นส่วนการทำงานของ Presence Server ที่ทำหน้าที่ในการจัดการกับข้อมูลแสดงสถานะแบบทั่วไป ซึ่งจะต้องมีฟังก์ชันการทำงานดังนี้

1. มีความสามารถในการจัดการกับสัญญาณ PUBLISH ซึ่งเป็นสัญญาณที่ TP จะใช้ในการบอกข้อมูลแสดงสถานะของผู้ใช้มายัง Presence Server โดยจะต้องนำข้อมูลแสดงสถานะ

ที่อยู่ภายใน Body ของสัญญาณ PUBLISH มาเก็บไว้ภายใน Database เพื่อที่จะส่งต่อให้กับผู้ที่มาร้องขอข้อมูลแสดงสถานะต่อไป

2. มีความสามารถในการจัดการกับสัญญาณ SUBSCRIBE ซึ่งเป็นสัญญาณที่ TP หรือ RLS ใช้ในการร้องขอข้อมูลแสดงสถานะของผู้อื่น โดย Presence Server จะต้องจดจำรายชื่อของผู้ที่ร้องขอข้อมูลแสดงสถานะไว้ เพื่ออัพเดตข้อมูลแสดงสถานะเหล่านั้นให้แก่ผู้ที่ทำการร้องขอ
3. มีความสามารถในการส่งสัญญาณ NOTIFY ซึ่งเป็นสัญญาณที่ใช้ในการส่งต่อข้อมูลแสดงสถานะไปให้กับผู้ที่ได้ทำการร้องขอไว้

4.3.6.2 Presence Reasoner Module

Presence Reasoner Module เป็นส่วนการทำงานของ Presence Server ซึ่งทำหน้าที่ในการประมวลผลสิทธิการเข้าถึงข้อมูลแสดงสถานะของผู้ที่ทำการร้องขอ โดยมีฟังก์ชันการทำงานดังนี้

1. มีความสามารถในการรับรู้ถึงข้อมูลบริบทความสัมพันธ์ของผู้ใช้ (Amoeba Model) ที่มากับกระบวนการ Subscription ของ RLS
2. มีความสามารถนำข้อมูลบริบทดังกล่าวในข้อที่ 1 มาตัดสินใจกับ Policy Based on Context-awareness เพื่อหาข้อมูลแสดงสถานะที่เหมาะสมกับผู้ร้องขอข้อมูล โดยพิจารณาจาก Relationship และ Level of Trust ที่แสดงถึงความสัมพันธ์และระดับความเชื่อใจ
3. มีความสามารถที่จะนำข้อมูลแสดงสถานะที่ไม่ปรากฏในบัญชีรายชื่อของผู้ร้องขอได้ เพื่อใช้ในกรณีที่ไม่สามารถติดต่อกับผู้ใช้ได้ ระบบก็จะนำรายชื่อที่เป็นตัวแทนของผู้ใช้ส่งไปให้ผู้ร้องขอแทน ทำให้เป็นการเพิ่มช่องทางการสื่อสารไปยังปลายทางได้มากยิ่งขึ้น

4.3.6.3 PI Aggregation Module

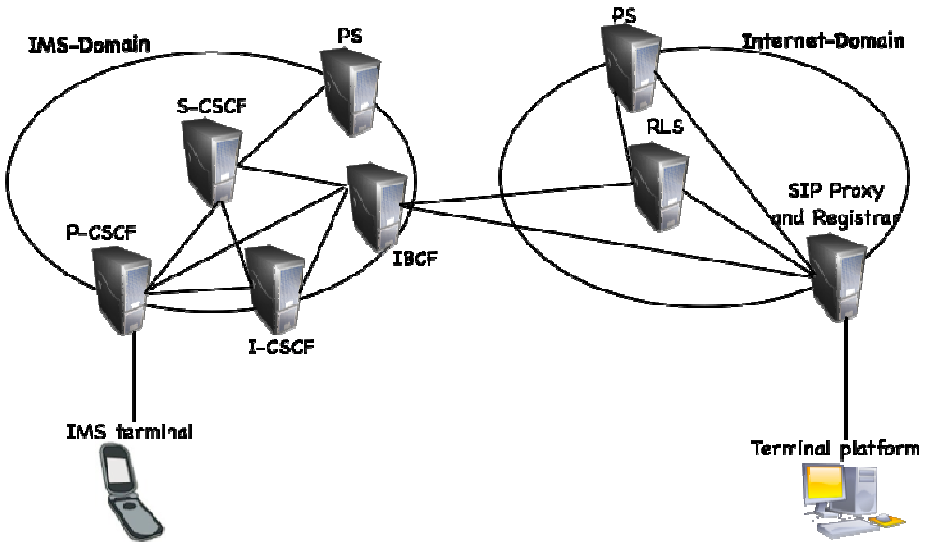
หลังจากได้ข้อมูลแสดงสถานะในกระบวนการ Presence Reasoner Module ซึ่งเป็นข้อมูลแสดงสถานะที่ยังไม่เป็นข้อมูลชุดเดียวกันและอาจจะยังมีความขัดแย้งของข้อมูล จึงจำเป็นต้องทำข้อมูลดังกล่าวให้เป็นข้อมูลแสดงสถานะแบบรวมศูนย์ โดยมีฟังก์ชันการทำงานดังนี้

1. มีความสามารถรวบรวมข้อมูลแสดงสถานะที่มากกระจายอยู่จำนวนมากให้เป็นข้อมูลแบบรวมศูนย์ โดยที่ข้อมูลดังกล่าวจะต้องเป็น SIP เท่านั้น

2. มีความสามารถเพิ่มข้อมูลที่เป็นประโยชน์เข้าไปแทนที่ข้อมูลที่ไม่เป็นประโยชน์ โดยข้อมูลเหล่านั้นได้จากการที่ผู้ใช้ตั้งค่าไว้ เช่น ในเวลา 21.00-6.00 ถ้าข้อมูลแสดงสถานะของผู้ใช้เป็น Idle ให้เพิ่มข้อมูลกิจกรรมของผู้ใช้ให้เป็น Sleeping เป็นต้น

4.4 การทำงานร่วมกับสถาปัตยกรรมของ IMS

จากบทที่ 2 ได้กล่าวไว้ถึงความสำคัญของระบบ IMS เพื่อทำให้ระบบการสื่อสารแบบรวมศูนย์สามารถทำงานได้ครอบคลุมและผู้ใช้สามารถติดต่อได้ง่าย ทำให้ระบบบริการแสดงสถานะแบบรวมศูนย์จำเป็นต้องดึงข้อมูลแสดงสถานะจากระบบของ IMS เพื่อแสดงผลให้กับผู้ใช้งานระบบการสื่อสารแบบรวมศูนย์



รูปที่ 4-15 การทำงานร่วมกับสถาปัตยกรรม IMS

จากรูปที่ 4-15 แสดงการเชื่อมต่อของระบบบริการแบบรวมศูนย์กับระบบ IMS โดยแนวคิดในการออกแบบของวิทยานิพนธ์นี้ จะมองระบบ IMS เป็นส่วนที่แบ่งแยกจากระบบการสื่อสารแบบรวมศูนย์ ซึ่งจะเรียกระบบ IMS ว่า IMS Domain และระบบการสื่อสารแบบรวมศูนย์ว่า Internet Domain ในการเชื่อมต่อระหว่าง 2 ระบบจะใช้ IBCF (Interconnection Border Control Function) เป็น Gateway ในการเชื่อมต่อสู่ IMS Domain โดย IBCF จะคอยรับและส่งสัญญาณ SIP จาก Internet Domain ดังแสดงรายละเอียดการเชื่อมต่อในหัวข้อ 4.5.3

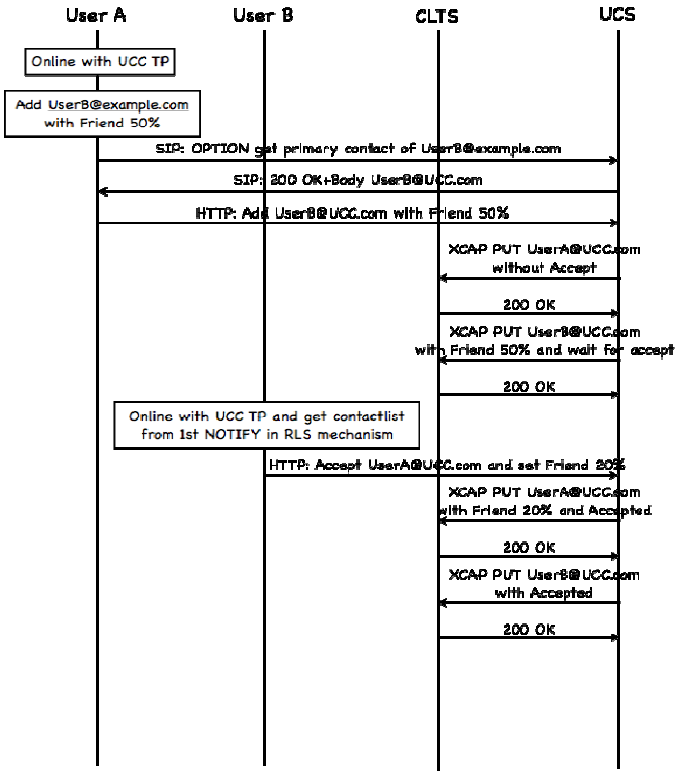
4.5 กระบวนการพื้นฐานที่เกี่ยวข้องกับระบบแสดงสถานะแบบรวมศูนย์

ในหัวข้อนี้จะอธิบายถึงกระบวนการทำงานพื้นฐานที่เกี่ยวข้องกับระบบบริการแสดงสถานะแบบรวมศูนย์ โดยแสดงถึงลำดับสัญญาณที่ผ่านองค์ประกอบต่าง ๆ ของระบบบริการแสดงสถานะแบบรวมศูนย์ และจะอธิบายองค์ประกอบที่มีการเพิ่มเติมจากระบบบริการสถานะของ SIP แบบธรรมดา ซึ่งแบ่งเป็น 3 กรณี คือ การเพิ่มและแก้ไขบัญชีรายชื่อ การ Publication, Subscription and Notification และกรณีศึกษาสุดท้าย การ Subscription and Notification ระหว่าง Internet Domain และ IMS Domain

4.5.1 เพิ่มและแก้ไขบัญชีรายชื่อ

กระบวนการในการเพิ่มและการแก้ไขบัญชีรายชื่อมี 2 กระบวนการ คือ การค้นหา Primary Account และการรับ Unified Contact

4.5.1.1 การค้นหา Primary Account



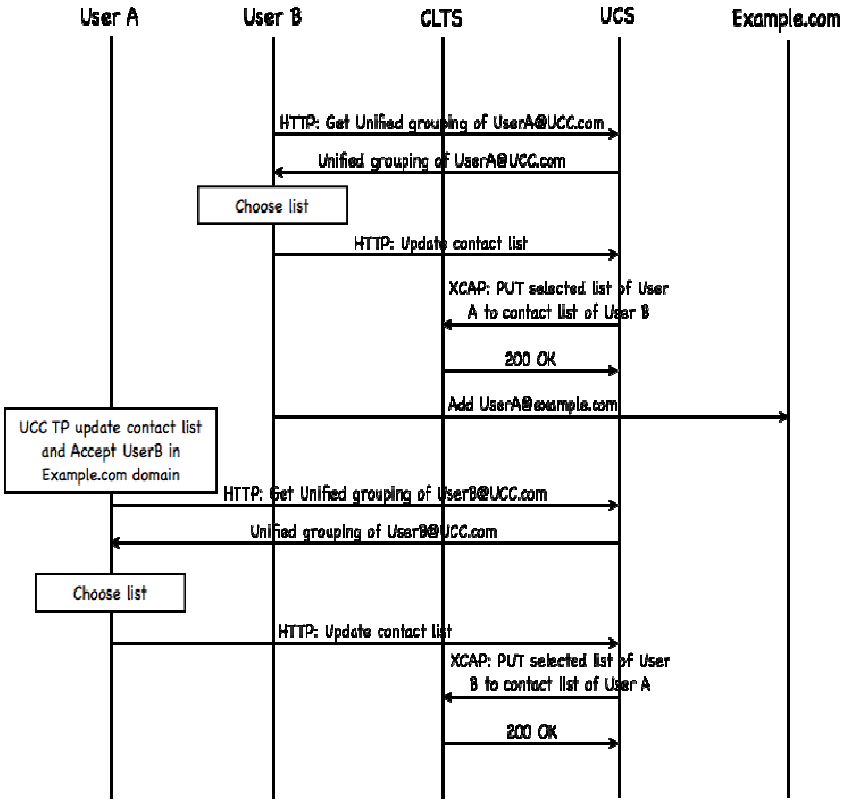
รูปที่ 4-16 ผู้ใช้ A ค้นหา Primary Account ของผู้ใช้ B

จากรูปที่ 4-16 เป็นลำดับการทำงานที่ผู้ใช้ A ต้องการจะเพิ่มผู้ใช้ B เข้าสู่บัญชีรายชื่อ โดยมีขั้นตอนการทำงานดังนี้

1. ผู้ใช้ A ทำการออนไลน์ด้วย TP ของ UCC และสั่งเพิ่มรายชื่อผู้ใช้ B โดยเพิ่ม UserB@example.com และระบุ Relationship และ Level of Trust ว่า Friend:50%
2. TP จะทำการค้นหา Primary Account ของผู้ใช้ A โดยการใช้สัญญาณ SIP OPTION ซึ่งใส่ Keyword ไปในส่วน Body ของสัญญาณดังกล่าวเป็น UserB@example.com และส่งสัญญาณดังกล่าวไปให้ Unified Server
3. เมื่อ Unified Server ได้รับสัญญาณ OPTION ก็จะทำการค้นหาข้อมูลของผู้ใช้ B แล้วตอบ Primary Account ของผู้ใช้ B ผ่านทางสัญญาณ SIP 200 OK โดยใส่ข้อมูลเป็น UserB@ucc.com ซึ่งเป็น Primary Account ของผู้ใช้ B
4. เมื่อ TP ของผู้ใช้ A ได้รับ Primary Account ของผู้ใช้ B TP ของผู้ใช้ A จะส่งสัญญาณ HTTP เพื่อเพิ่มรายชื่อ B ไปยัง Unified Server ด้วย Primary Account ที่ได้มาและข้อมูล Relationship และ Level of Trust ที่ได้กำหนดไว้ในตอนต้น
5. เมื่อ Unified Server ได้รับ HTTP Add ก็จะทำการเพิ่ม Primary Account ของผู้ใช้ A ลงไปในบัญชีรายชื่อของผู้ใช้ B และก็จะทำการเพิ่ม Primary Account ของผู้ใช้ B ลงไปในบัญชีรายชื่อของผู้ใช้ A โดยผ่านสัญญาณ XCAP PUT ไปให้กับ Contact List and Trust Server แต่การเพิ่มดังกล่าวจะระบุถึงการ Waiting for Accept
6. เมื่อผู้ใช้ B ได้เข้าสู่ระบบ UCC ผ่านทาง UCC TP ผู้ใช้ B จะได้รับการแจ้งเตือนถึงผู้ที่ได้ทำการเพิ่มรายชื่อของตน ซึ่งในที่นี้คือผู้ใช้ A
7. ถ้าผู้ใช้ B ไม่ต้องการที่จะรับการเพิ่มรายชื่อของผู้ใช้ A TP ก็ส่งสัญญาณ HTTP เพื่อลบข้อมูลที่ Unified Server เพิ่มไว้ แต่ถ้าผู้ใช้ B อนุญาตให้ผู้ใช้ A เพิ่มข้อมูลได้ TP ของผู้ใช้ B จะส่งสัญญาณ HTTP Accept ไปยัง Unified Server โดยระบุ Relationship และ Level of Trust
8. เมื่อ Unified Server ได้รับข้อมูลยืนยันการอนุญาตก็จะทำการเพิ่ม ข้อมูล Relationship และ Level of Trust ที่ผู้ใช้ B ระบุสำหรับผู้ใช้ A และระบุว่า Accepted เช่นเดียวกัน Unified Server ก็จะยืนยันการตอบรับของ B ไปยัง A ด้วย

4.5.1.2 การรับ Unified Contact ของผู้ใช้

กระบวนการรับ Unified Contact ของผู้ใช้ คือ กระบวนที่ผู้ใช้จะรับกลุ่มบัญชีรายชื่อของบุคคลที่ผู้ใช้เพิ่มรายชื่อ



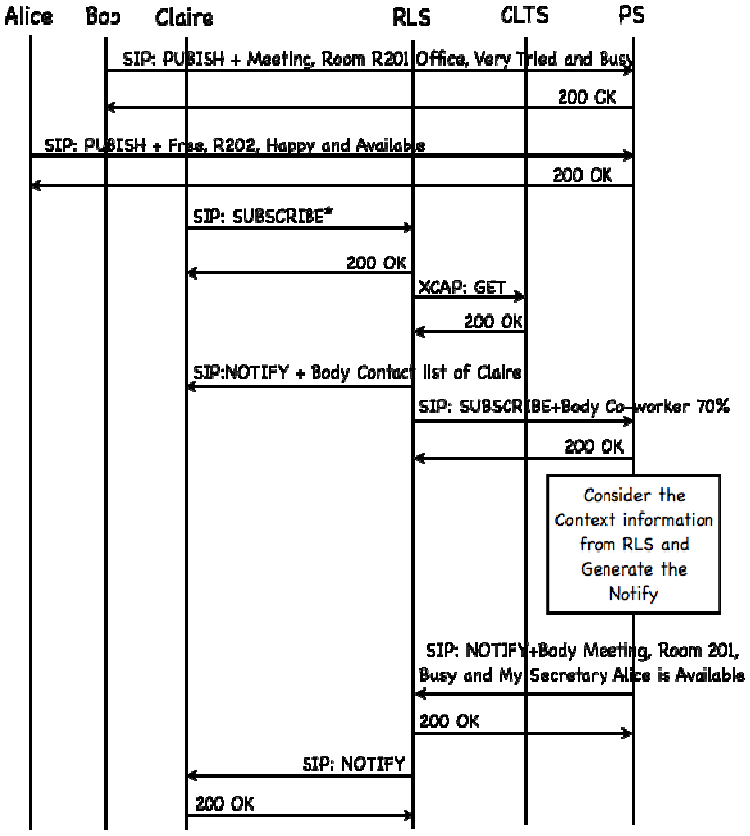
รูปที่ 4-17 การรับ Unified Contact ของผู้ใช้ A

จากรูปที่ 4-17 เป็นลำดับของกระบวนการรับกลุ่มบัญชีรายชื่อที่ได้จากการใช้ Primary Account ในการค้นหา ซึ่งจะเรียกว่าเป็นกระบวนการรับ Unified Contact โดยกระบวนการนี้ทำให้ผู้ใช้เพิ่มข้อมูล Alias ได้อย่างถูกต้อง ซึ่งมีลำดับการทำงานดังนี้

1. เมื่อผู้ใช้ B ได้รับ Primary Account ของผู้ใช้ A ก็จะส่งสัญญาณ HTTP GET เพื่อร้องขอข้อมูล Unified Contact ของผู้ใช้ A ไปยัง Unified Server จากนั้น Unified Server จะตอบกลับผู้ใช้ B ด้วย Unified Contact ของผู้ใช้ A โดย Unified Contact จะเป็นกลุ่มของบัญชีรายชื่อของผู้ใช้ A ซึ่งได้จากการคำนวณ Relationship และ Level of Trust ของ Unified Server

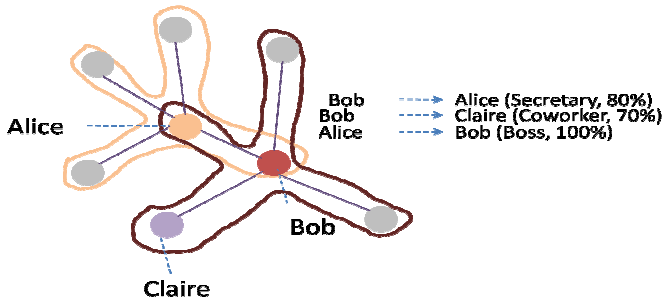
2. เมื่อผู้ใช้ B ได้รับข้อมูล Unified Contact ก็จะเลือกบัญชีรายชื่อที่ผู้ใช้ B ที่ต้องการ จากนั้น TP ของผู้ใช้ B จะส่งสัญญาณ HTTP Update ไปยัง Unified Server เพื่อเพิ่มข้อมูลของบัญชีรายชื่อต่าง ๆ ไว้ในบัญชีรายชื่อของผู้ใช้ B ต่อไป
3. Unified Server จะทำการเพิ่มรายชื่อของ B โดยการใช้ XCAP PUT
4. ถ้า Unified Contact ของผู้ใช้ A มีบัญชีรายชื่อที่ไม่ปรากฏในระบบเดียวกัน เช่น MSN หรือในตัวอย่างจะเป็น UserA@example.com TP ก็จะส่งสัญญาณที่มีรูปแบบเฉพาะเพื่อเพิ่มข้อมูลในระบบนั้น ๆ
5. เมื่อผู้ใช้ A อัพเดทบัญชีรายชื่อ ก็จะพบว่า ผู้ใช้ B ได้ยอมรับการติดต่อแล้ว และทำกระบวนการในขั้นตอน 1-4 อีกครั้ง เพื่อที่จะเพิ่มข้อมูลบัญชีรายชื่อต่าง ๆ ลงไปในบัญชีรายชื่อของผู้ใช้ A เช่นกัน

4.5.2 กระบวนการ Publication, Subscription and Notification



รูปที่ 4-18 แสดงลำดับของการ Publication, Subscription and Notification

จากรูปที่ 4-18 จะแบ่งกระบวนการออกเป็น 2 ส่วน คือ Publication และ Subscription and Notification โดยจากตัวอย่างนี้ Bob และ Alice จะเป็นผู้แสดงสถานะ และ Claire เป็นผู้ที่ต้องการรู้สถานะของ Bob ซึ่งบุคคลทั้งสามมีความสัมพันธ์กันดังนี้ Bob ตั้งค่าความสัมพันธ์กับ Alice เป็น Secretary และให้ระดับความเชื่อใจเป็น 80% Alice ตั้งค่าความสัมพันธ์กับ Bob เป็น Boss และให้ระดับความเชื่อใจเป็น 90% และ Bob ตั้งค่าความสัมพันธ์กับผู้ที่มองสถานะของตน Claire เป็น Co-worker 70% ซึ่งสามารถเขียนความสัมพันธ์ของ Bob และ Alice ตามรูปแบบของ Amoeba Model ได้ดังรูปที่ 4-19



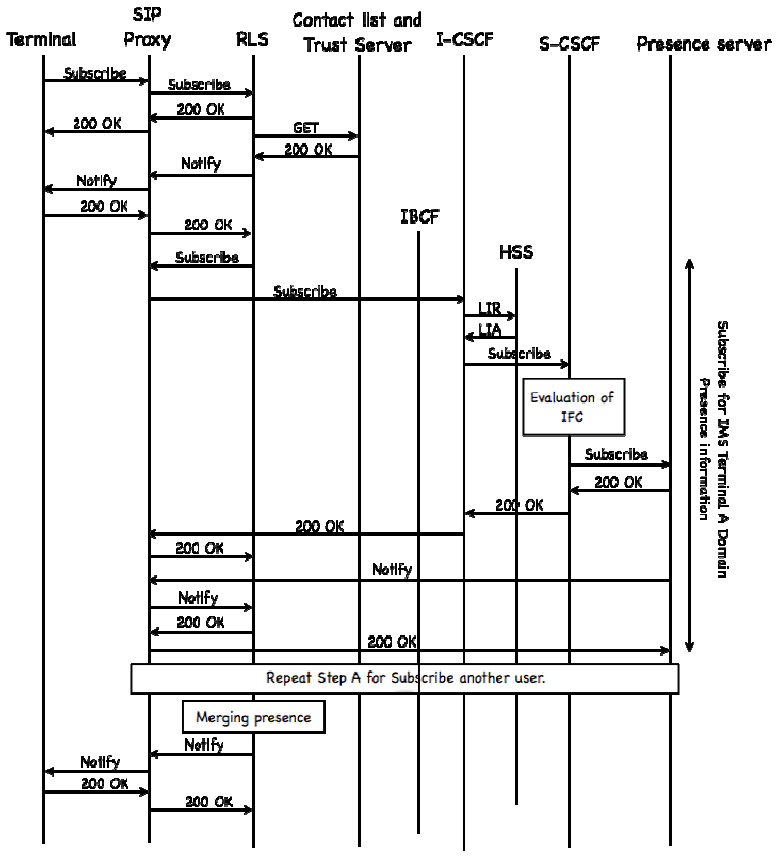
รูปที่ 4-19 รูปแบบของความสัมพันธ์ของ Bob, Claire และ Alice

โดยกรณีตัวอย่างนี้แสดงให้เห็นถึงข้อดีของระบบบริการแสดงสถานะแบบรวมศูนย์ที่สามารถนำข้อมูลแสดงสถานะของบุคคลรอบข้าง Bob มาใช้ประโยชน์ได้ โดยที่ Claire ไม่จำเป็นต้องมีบัญชีรายชื่อของบุคคลนั้น ๆ อยู่ในบัญชีรายชื่อของตนเอง (ซึ่งในที่นี้ คือ Alice) การที่จะเกิดกรณีนี้ได้ก็ต่อเมื่อ Bob ต้องแสดงข้อมูลสถานะที่ไม่สามารถรับการติดต่อได้ และ Alice จำเป็นต้องมีความเชื่อใจต่อ Bob ในระดับสูง และค่า Relationship ระหว่าง Bob และ Alice ต้องเป็นความสัมพันธ์ที่สอดคล้องกับ Claire จากตัวอย่างนี้ความสัมพันธ์ของ Alice และ Claire ที่มีกับ Bob คือ Co-worker Secretary และ Boss ซึ่งแสดงว่า เป็นผู้ร่วมงานกัน โดยอธิบายลำดับของสัญญาณที่ผ่านระบบบริการแสดงสถานะแบบรวมศูนย์ได้ดังนี้

1. เริ่มต้นโดยที่ Bob และ Alice ทำการ Publication ข้อมูลแสดงสถานะของตนเองไปยัง Presence Server ผ่านสัญญาณ SIP PUBLISH
2. Claire ต้องการจะติดต่อ Bob จึงต้องการทราบข้อมูลแสดงสถานะของ Bob จึงทำกระบวนการ Subscription โดยส่งสัญญาณ SUBSCRIBE ที่ระบุถึงการทำงานแบบ RLS ไปให้ RLS
3. RLS ก็จะดึงข้อมูลบัญชีรายชื่อและข้อมูลบริบทที่เกี่ยวข้องกับ Bob โดยข้อมูลบริบทดังกล่าวคือ Amoeba Model ของ Bob

4. จากนั้น RLS ก็จะทำการสร้างสัญญาณ SUBSCRIBE ใหม่ส่งไปให้ Presence Server โดยสัญญาณดังกล่าวจะเพิ่มข้อมูลบริบทลงไปในส่วน Boby ของสัญญาณ SUBSCRIBE เพื่อส่งให้ Presence Server ตัดสินใจว่า ข้อมูลใดบ้างที่เหมาะสมกับ Claire
5. เมื่อ Presence Server ได้พิจารณาข้อมูลบริบทก็จะทำการสร้างสัญญาณ NOTIFY เพื่อตอบกลับไปยัง RLS โดยข้อมูลนั้นจะระบุข้อมูลการแสดงสถานะของ Alice ไปด้วย
6. RLS ก็จะรวบรวมสัญญาณ NOTIFY ทั้งหมดของ Claire แล้วส่งต่อไปให้ Claire เป็นอันเสร็จสิ้นกระบวนการ

4.5.3 การ Subscription and Notification ระหว่าง Internet Domain และ IMS Domain



รูปที่ 4-20 ลำดับสัญญาณของการ Subscription and Notification ระหว่าง Internet Domain และ IMS Domain

จากรูปที่ 4-20 เป็นลำดับของสัญญาณในการที่ TP ในฝั่ง Internet Domain ต้องการจะรับทราบข้อมูลจาก Presence Server ในฝั่งของ IMS Domain โดยจะมีลำดับขั้นตอนดังนี้

1. เมื่อ UCC TP เข้าสู่ระบบก็จะเริ่มกระบวนการ SUBSCRIBE ไปยัง RLS
2. RLS จะร้องขอข้อมูลบัญชีรายชื่อจาก Contact List and Trust Server ซึ่งข้อมูลบัญชีรายชื่อที่ได้จาก Contact List and Trust Server เป็นรายชื่อจาก IMS Domain ตัวอย่างเช่นเช่น dommmy@ims.cnr.ac.th
3. RLS จะทำการ SUBSCRIBE ไปยัง IMS ผ่านทาง IBCF โดยที่ RLS จะรู้ที่อยู่ของ IBCF ได้ด้วยกระบวนการทำงานของ DNS โดย DNS จะเก็บข้อมูล IP Address และ Port ของ IBCF ของระบบ IMS ไว้
4. เมื่อ IBCF ได้รับสัญญาณ SUBSCRIBE ก็จะส่งสัญญาณต่อไปยัง I-CSCF เนื่องจากเป็นการสร้าง Session ใหม่ระหว่าง UCC TP และ Presence Server ใน IMS Domain
5. เมื่อ I-CSCF ได้รับสัญญาณ SUBSCRIBE ก็จะปรึกษากับ HSS ผ่านทางสัญญาณ LIR (Diameter Protocol) เพื่อสอบถามว่า บุคคลที่ SUBSCRIBE เข้ามาได้รับอนุญาตหรือไม่ HSS จะตอบกลับผ่านทางสัญญาณ LIA เพื่อยืนยันการอนุญาตดังกล่าว
6. ถ้า I-CSCF อนุญาตให้สัญญาณดังกล่าวผ่านได้ ก็จะส่งสัญญาณ SUBSCRIBE ต่อไปให้ S-CSCF ซึ่งจะคอยพิจารณาว่าควรจะส่งสัญญาณ SUBSCRIBE ไปยัง Application Server ใด โดยใช้กระบวนการ IFC (Initial Filter Criteria) โดยในที่นี้จะทำให้รู้ได้ว่าต้องส่งสัญญาณไปให้กับ Presence Server ในฝั่ง IMS
7. Presence Server ก็จะส่งสัญญาณ NOTIFY เพื่อบ่งบอก ข้อมูลแสดงสถานะกลับไปให้ UCC TP ต่อไป

4.6 สรุป

จากเนื้อหาในบทที่ 4 เป็นการออกแบบสถาปัตยกรรมของระบบการแสดงสถานะแบบรวมศูนย์ โดยได้สรุปถึงคุณลักษณะเฉพาะตัวของรูปแบบข้อมูลแสดงสถานะที่ได้จากระบบการสื่อสารแบบรวมศูนย์ และได้ยกตัวอย่างที่ทำให้เห็นถึงปัญหาที่ระบบการแสดงสถานะในปัจจุบันยังไม่สามารถจัดการได้ ดังนั้นจึงจำเป็นต้องออกแบบระบบการแสดงสถานะที่เหมาะสม โดยวิทยานิพนธ์นี้นำเสนอระบบการแสดงสถานะแบบรวมศูนย์ หรือ Unified Presence Service (UPS) ที่ประกอบด้วย 3 ฟังก์ชัน คือ Large-Scale of PI, PI Aggregation และ Smart Distribution and Policy Control ที่แก้ปัญหา และเพิ่มความสามารถใหม่ๆ ทำให้ระบบการสื่อสารแบบรวมศูนย์สามารถที่จะเพิ่มประสิทธิภาพในการรับรู้สถานะของปลายทางจากอุปกรณ์หลายๆ ชนิด และยังเพิ่มความสามารถในการติดต่อไปยังปลายทางได้มากยิ่งขึ้น จากตัวอย่างการทำงานในรูปแบบใหม่ๆ เช่น การแสดงข้อมูลของบุคคลนอกรายชื่อ หรือ การติดต่อผ่านตัวแทน เป็นต้น

จากแนวคิดการออกแบบกรอบการทำงานของระบบบริการแสดงสถานะแบบรวมศูนย์ ได้นำกรอบการทำงานไปออกแบบสถาปัตยกรรมของระบบบริการแสดงสถานะแบบรวมศูนย์ดังรูปที่ 4-9 โดยได้อธิบายรายละเอียดการทำงานภายในขององค์ประกอบต่าง ๆ และได้ออกแบบการเชื่อมต่อสู่ระบบ IMS โดยอ้างอิงจากมาตรฐานของ 3GPP และ TISPAN ทำให้สามารถที่จะเชื่อมระบบแสดงสถานะระหว่าง Internet Domain และ IMS Domain ได้

เมื่อเขียนตารางเปรียบเทียบความสามารถของระบบบริการแสดงแบบรวมศูนย์ที่ได้ออกแบบไว้กับงานอื่น ๆ สามารถเปรียบเทียบได้ดังตารางที่ 4-1

ตารางที่ 4-1 ตารางเปรียบเทียบความสามารถของระบบ UPS กับระบบบริการแสดงสถานะในปัจจุบัน

	Unified Contact	**PI Aggregation**	**Policy control and Distribution**	**Interworking with IMS**
SIMPLE [3]	Weak	Smart	Basic	N/A
Jabber [15]	Weak	N/A	Basic	N/A
R-U-In? [7]	Smart	N/A	Smart	Basic
Buddyspace [35]	Weak	N/A	Basic	N/A
UPS	Smart	Smart	Smart	Basic

จากตารางที่ 4-1 ได้เปรียบเทียบระบบแสดงสถานะในปัจจุบันและอธิบายไว้ในบทที่ 2 และ 3 โดยเปรียบเทียบจากปัญหาที่ระบบบริการแสดงสถานะจำเป็นต้องจัดการ ได้แก่ Unified Contact, PI Aggregation Policy Control and Distribution และ Interworking with IMS จะเห็นได้ว่า ระบบการแสดงสถานะแบบรวมศูนย์ที่ได้ออกแบบจะมีความยืนหยุ่นในการสร้างรูปแบบการทำงาน ใหม่ ๆ ได้มากกว่าระบบแบบอื่น ๆ และยังสามารถทำงานร่วมกับระบบ IMS ได้

บทที่ 5
การพัฒนาและทดสอบระบบบริการแสดงสถานะแบบรวมศูนย์

บทนี้จะเป็นการกล่าวถึงการพัฒนาระบบบริการแสดงสถานะแบบรวมศูนย์ ตามการออกแบบในบทที่ 4 เพื่อใช้ในการทดสอบการทำงาน โดยรายละเอียดที่จะกล่าวถึงนั้นจะเริ่มต้นด้วย ขอบเขตและข้อจำกัดในการพัฒนาระบบบริการแสดงสถานะแบบรวมศูนย์ ซึ่งในการพัฒนาจะใช้พื้นฐานของเครื่องลูกข่ายและเครื่องแม่ข่ายที่เป็น Open Source สำหรับใช้ในการต่อยอดเพื่อพัฒนาให้เป็นระบบบริการแสดงสถานะแบบรวมศูนย์ นอกจากนี้จะกล่าวถึงส่วนที่ใช้ในการทดสอบกับระบบที่ได้พัฒนาขึ้น ซึ่งจะแสดงผลลัพธ์ของการทดสอบของแต่ละส่วน และสรุปการพัฒนาระบบและผลการทดสอบ

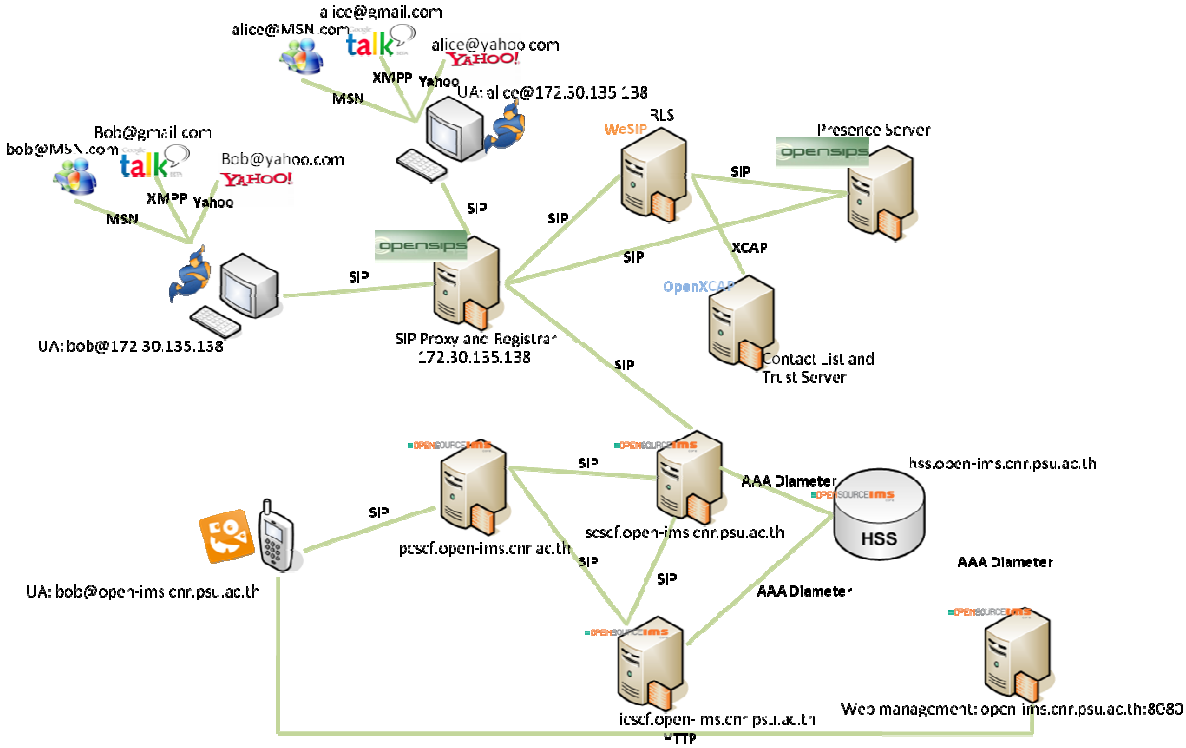
5.1 ขอบเขตและข้อจำกัดในการพัฒนาระบบบริการแสดงสถานะแบบรวมศูนย์

ในหัวข้อนี้จะอธิบายถึงขอบเขตและข้อจำกัดในการพัฒนาระบบบริการแสดงสถานะแบบรวมศูนย์ โดยระบบที่พัฒนาขึ้นเป็นการพัฒนาต่อยอดจากระบบ Open Source พื้นฐาน เพื่อใช้ในการทดสอบและพิสูจน์การทำงานที่ได้ออกแบบไว้ในบทที่ 4 ซึ่งการพัฒนาระบบจะมีขอบเขตและข้อจำกัดดังนี้

1. ระบบเครื่องแม่ข่ายและเครื่องลูกข่ายจะพัฒนาต่อยอดจาก Open Source ทั้งหมด
2. การพัฒนาระบบเครื่องแม่ข่ายประกอบด้วย 2 ส่วน คือ OpenIMSCore [38] เพื่อใช้สำหรับการพัฒนาการเชื่อมต่อกับระบบ IMS และ OpenSIPS [36] เพื่อใช้สำหรับการพัฒนาระบบบริการแสดงสถานะแบบรวมศูนย์
3. การพัฒนาเครื่องแม่ข่ายของระบบบริการแสดงสถานะแบบรวมศูนย์จะใช้เทคโนโลยี WeSIP [37] เพื่อใช้เป็นเครื่องมือช่วยในการพัฒนา OpenSIPS ที่เป็นพื้นฐาน ให้สามารถทำงานเป็นตามการออกแบบในบทที่ 4
4. ระบบเครื่องแม่ข่ายของระบบบริการแสดงสถานะแบบรวมศูนย์ที่พัฒนาขึ้นจะไม่มีส่วนของ Unified Server โดยให้ถือว่าผู้ใช้ได้กำหนดค่า User Preference และบัญชีรายชื่อต่างๆ เอาไว้แล้ว
5. เครื่องลูกข่ายของระบบบริการแสดงสถานะแบบรวมศูนย์จะใช้ SIP Communicator โดยจะมีการพัฒนา SIP Communicator ให้สามารถทำงานได้ตามการออกแบบในบทที่ 4

5.2 การพัฒนาระบบบริการแสดงสถานะแบบรวมศูนย์

ในหัวข้อนี้จะอธิบายการพัฒนาระบบบริการแสดงสถานะแบบรวมศูนย์ โดยภาพรวมของระบบแสดงดังรูปที่ 5-1

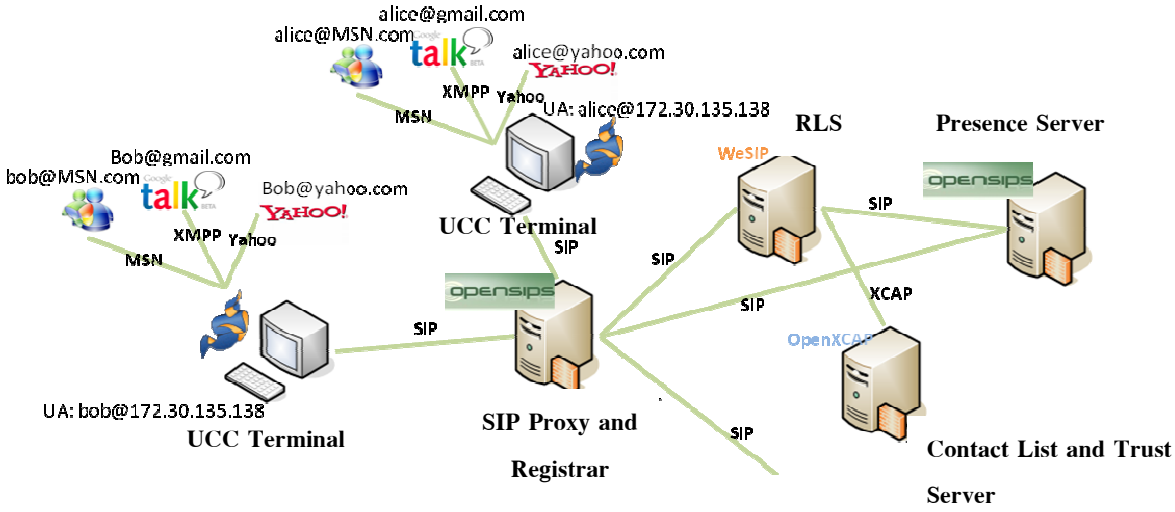


รูปที่ 5-1 ภาพรวมของระบบบริการแสดงสถานะแบบรวมศูนย์ที่ได้พัฒนาขึ้น

จากรูปที่ 5-1 แสดงให้เห็นองค์ประกอบต่างๆ และการเชื่อมต่อขององค์ประกอบ โดยแบ่งออกเป็น 2 ส่วน คือ ส่วนระบบบริการแสดงสถานะแบบรวมศูนย์ภายในและส่วนการเชื่อมต่อกับระบบ IMS โดยจะอธิบายแต่ละส่วนดังนี้

5.2.1 ส่วนระบบบริการแสดงสถานะแบบรวมศูนย์ภายใน

ในส่วนของระบบบริการแสดงสถานะแบบรวมศูนย์ภายในจะเป็นการพัฒนาระบบ เพื่อใช้ในการทดสอบกระบวนการแสดงผลและการร้องขอข้อมูลแสดงสถานะเท่านั้น แต่จะไม่มีพัฒนาส่วนของ Unified Server เนื่องจากได้ถือว่าระบบได้รับค่า User Preference ของผู้ใช้ไว้แล้ว



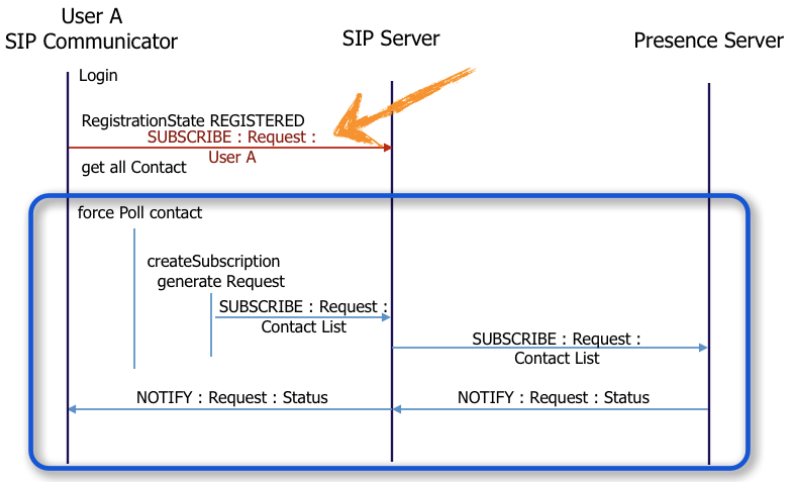
รูปที่ 5-2 ระบบทดสอบของบริการแสดงสถานะแบบรวมศูนย์ภายใน

จากรูปที่ 5-2 แสดงองค์ประกอบที่ใช้ในการทดสอบระบบบริการแสดงสถานะแบบรวมศูนย์ภายใน โดยมีรายละเอียดของแต่ละองค์ประกอบดังนี้

5.2.1.1 UCC Terminal

UCC Terminal จะใช้ SIP Communicator [41] เป็นเครื่องลูกข่ายเนื่องด้วย SIP Communicator เป็น Open Source ที่สามารถแก้ไขการทำงานได้ โดยในที่นี้ได้มีการแก้ไขใน 2 กระบวนการ คือ การทำงานร่วมกับ RLS ที่ได้พัฒนาขึ้น และพัฒนาฟังก์ชันการรวบรวมข้อมูลแสดงสถานะที่เครื่องลูกข่าย

โดยการทำงานร่วมกับ RLS จะเห็นได้ว่า เครื่องลูกข่ายจะต้องมีความสามารถที่จะ SUBSCRIBE โดยที่ระบุ Header เป็น Eventlist ได้ ทำให้จำเป็นต้องพัฒนาการทำงานของ SIP Communicator เพื่อให้รองรับการทำงานดังกล่าว และในระบบบริการแสดงสถานะแบบรวมศูนย์ผู้ใช้งานสามารถเก็บรายชื่อผู้ติดต่อไว้ที่เครื่องแม่ข่าย ในที่นี้กำหนดให้มีรายชื่อผู้ติดต่ออยู่ที่ Contact List and Trust Server จึงจำเป็นต้องปรับเปลี่ยนลำดับทำงานของ SIP Communicator ให้มีการ SUBSCRIBE เพียงครั้งเดียวหลังจากการ Registration ขั้นตอนการทำงานแสดงได้ดังรูปที่ 5-3 โดยสิ่งที่เพิ่มเติมเข้าไป คือการเพิ่มเติมสัญญาณ SUBSCRIBE เมื่อเสร็จสิ้นกระบวนการ Registration



รูปที่ 5-3 ลำดับการทำงานของ SIP Communicator

สำหรับการพัฒนาฟังก์ชันการรวบรวมข้อมูลแสดงสถานะที่เครื่องลูกข่าย SIP Communicator จะเป็นการพัฒนา Plug-in เพื่อรองรับการใช้งานกับโปรโตคอลต่าง ๆ แสดงได้ดังรูปที่ 5-4

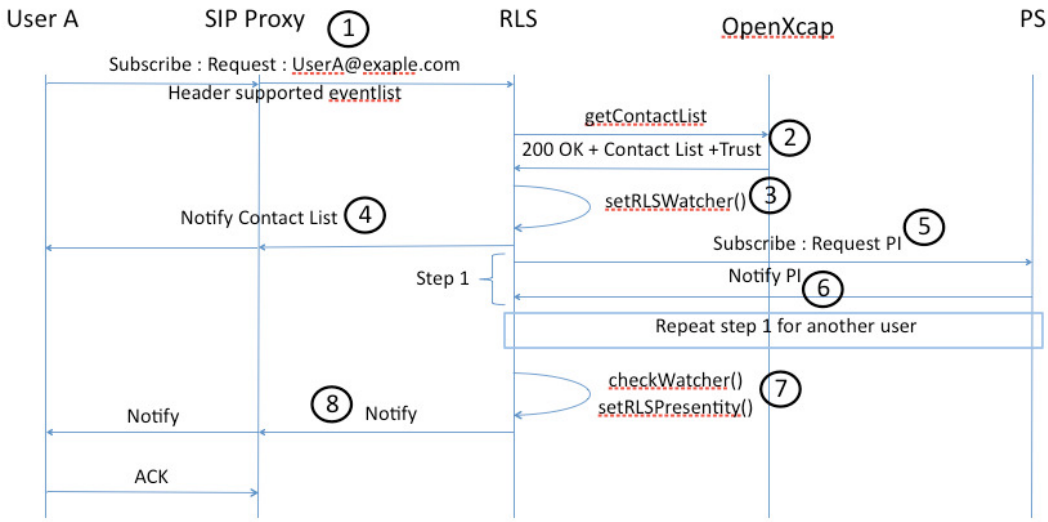
รูปที่ 5-4 Plug-in สำหรับการรวบรวมโปรโตคอลอื่น ๆ

5.2.1.2 SIP Proxy and Registrar

SIP Proxy and Registrar ทำหน้าที่เป็น Outbound/Inbound Proxy Server ในระบบบริการแสดงสถานะแบบรวมศูนย์ และทำหน้าที่เป็น Registrar Server นอกจากนี้ยังทำหน้าที่ในการส่งต่อสัญญาณต่าง ๆ ไปยัง Modules ที่เหมาะสมตามที่ได้การตั้งค่าไว้ โดยในระบบเลือกใช้ OpenSIPS ซึ่งองค์ประกอบในส่วนนี้จะไม่ได้มีการแก้ไขเพิ่มเติมจากการทำงานแบบเดิม

5.2.1.3 RLS

RLS เป็น Modules หนึ่งที่ทำงานอยู่เบื้องหลัง SIP Proxy and Registrar โดยมีหลักการทำงานอยู่บนมาตรฐาน RFC 4662 เพื่อใช้ในการลดแบนด์วิดท์ระหว่างเครื่องลูกข่ายและเครื่องแม่ข่าย แต่ในระบบบริการแสดงสถานะแบบรวมศูนย์ที่ออกแบบไว้ในบทที่ 4 ได้เพิ่มกระบวนการส่งข้อมูลบริบท (Relationship และ Level of Trust) จึงทำให้ RLS ของ OpenSIPS ไม่สามารถทำงานตามการออกแบบดังกล่าวได้ ทำให้จำเป็นต้องพัฒนา RLS ขึ้นสำหรับระบบบริการแสดงสถานะแบบรวมศูนย์โดยเฉพาะ ซึ่งการพัฒนา RLS จะอยู่ในรูปแบบของ SIP Servlet โดยการใช้เทคโนโลยี WeSIP ทำหน้าที่ในการติดต่อระหว่าง OpenSIPS กับ SIP Servlet ที่ได้พัฒนาขึ้น โดย SIP Servlet ที่พัฒนาสามารถจัดการ กระบวนการต่าง ๆ ที่เกิดขึ้นดังรูปที่ 5-5



รูปที่ 5-5 ลำดับสัญญาณของ RLS

1. เมื่อ RLS รับสัญญาณ SUBSCRIBE จากผู้ใช้ A ต่อมา RLS จะร้องขอข้อมูล Contact List และ Trust Level จาก Contact List and Trust Server โดยการขอข้อมูลจะเป็นไปตาม รูปแบบของ XCAP ซึ่งเป็นการเก็บข้อมูลในรูปแบบ XML ในระบบการสื่อสารแบบ

รวมศูนย์จะเก็บข้อมูล Contact List และ Trust Level ด้วยรูปแบบของ XCAP โดยใช้ OpenXCAP ทำหน้าที่เป็น XCAP Server

2. OpenXCAP ส่ง Contact List และ Trust Level มาที่ RLS ทำให้ RLS ได้รับบัญชีรายชื่อทั้งหมดของผู้ใช้ A
3. RLS เก็บข้อมูลของผู้ใช้ A ไว้ในฐานข้อมูล ซึ่งในที่นี้ผู้ใช้ A จะมีสถานะเป็นผู้มองสถานะ โดยข้อมูลที่เก็บ เช่น Call-ID, To, From เป็นต้น
4. RLS ส่ง Contact List ไปยัง TP ที่ผู้ใช้ A ใช้งานอยู่เพื่อให้ผู้ใช้ A รู้ว่ามีบัญชีรายชื่อใด
5. RLS จะส่ง Contact List ซึ่งในที่นี้จะเรียกว่า Resource ไปร้องขอข้อมูลแสดงสถานะจาก Presence Server (PS)
6. PS ส่งข้อมูลแสดงสถานะกลับมายัง RLS
7. RLS ตรวจสอบว่าได้รับข้อมูลแสดงสถานะของ Resource ทั้งหมด จากนั้นจัดการรวมข้อมูลแสดงสถานะของแต่ละ Resource ไว้ในข้อมูลแสดงสถานะเดียว แล้วส่งกลับไปยังผู้ใช้ A

5.2.1.4 Contact List and Trust Server

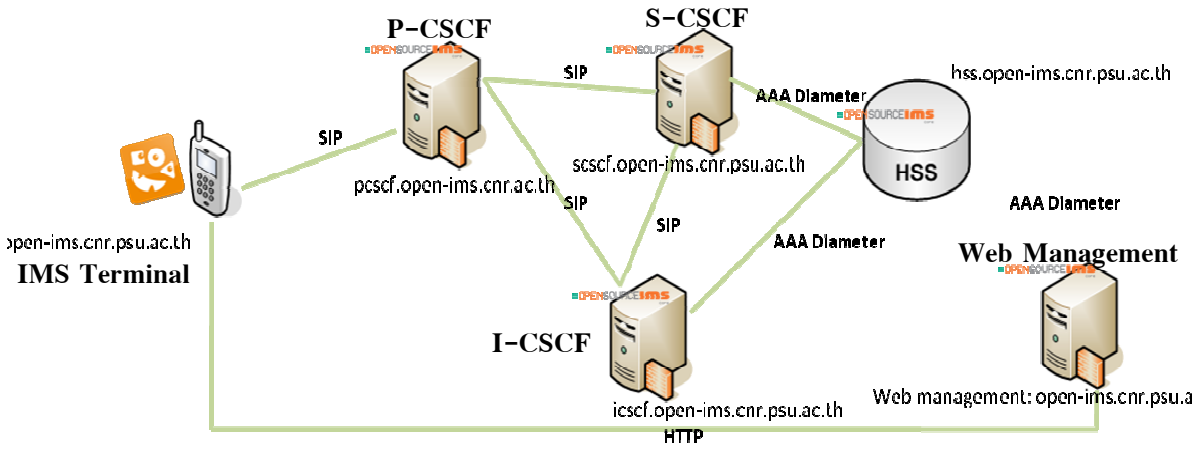
Contact List and Trust Server เป็น OpenXCAP Server [39] ทำหน้าที่ในจัดการเก็บข้อมูลบัญชีรายชื่อและข้อมูลบริบทของแต่ละบุคคล

5.2.1.5 Presence Server

Presence Server เป็น Modules หนึ่งใน OpenSIPS โดยจะทำหน้าที่ในการจัดการสัญญาณ PUBLISH และ SUBSCRIBE ที่เข้ามาในระบบ และแสดงข้อมูลแสดงสถานะไปยังผู้ใช้โดยผ่านสัญญาณ NOTIFY ซึ่งการทำงานปกติของ Presence Server ใน OpenSIPS ยังไม่สามารถทำงานในส่วนของการ Presence Reasoner Module และ Presence Aggregation Module ได้จึงได้พัฒนา Module ทั้งสองขึ้นเป็น SIP Servlet และใช้เทคโนโลยีของ WeSIP ในการเชื่อมต่อ Presence Server ปกติของ OpenSIPS และ SIP Servlet

5.2.2 ส่วนการเชื่อมต่อกับระบบ IMS

ในส่วนนี้ใช้ในการทดสอบการเชื่อมต่อกับระบบ IMS โดยการพัฒนาระบบ IMS จะใช้ระบบ OpenIMSCore เป็นระบบพื้นฐาน และใช้การตั้งค่าให้กับระบบ OpenIMSCore เพื่อใช้ในการสร้างการเชื่อมต่อระหว่างระบบ IMS กับระบบบริการแสดงสถานะแบบรวมศูนย์ โดยองค์ประกอบต่าง ๆ ของระบบ IMS สามารถแสดงได้ดังรูปที่ 5-6



รูปที่ 5-6 ระบบ IMS

จากรูปที่ 5-6 แสดงองค์ประกอบต่าง ๆ ของระบบ IMS โดยองค์ประกอบแต่ละองค์ประกอบมีรายละเอียดดังนี้

1. IMS Terminal ทำหน้าที่เป็นเครื่องลูกข่าย ซึ่งในที่นี้ใช้โปรแกรม Monster [40] สำหรับการทำงานเชื่อมต่อกับ IMS Network เนื่องจากการทำงานรวมกับ IMS Network จำเป็นต้องมีกระบวนการเข้ารหัส และ Header ของสัญญาณที่แตกต่างกับระบบ SIP ทั่วไป ซึ่ง Monster สนับสนุนการทำงานข้างต้น ทำให้ Monster จึงเหมาะสมกับการเป็นเครื่องลูกข่ายของระบบ IMS Network
2. P-CSCF ทำหน้าที่เป็นเครื่องแม่ข่าย ซึ่งเป็นส่วนหนึ่งของ OpenIMSCore โดย P-CSCF จะทำหน้าที่เป็น Outbound/Inbound SIP Proxy Server ของระบบ IMS Network โดยหมายเลข IP ของ P-CSCF สามารถเข้าถึงได้ โดยกระบวนการของ DNS เมื่อ IMS Terminal สอบถามถึง IP Address ของ pcscf.open-ims.cnr.psu.ac.th:4060 ก็จะได้ IP Address 172.30.138.51:4060 ของ P-CSCF เพื่อที่จะส่งสัญญาณไปยัง P-CSCF
3. I-CSCF ทำหน้าที่เป็นเครื่องแม่ข่าย ซึ่งเป็นส่วนหนึ่งของ OpenIMSCore เช่นเดียวกับ P-CSCF โดยจะทำหน้าที่ในการตรวจสอบว่าผู้ใช้ที่ส่งสัญญาณเข้าสู่ระบบสามารถที่จะมี

สิทธิ์ในการใช้งานได้หรือไม่ เมื่อมีสิทธิ์ที่จะใช้งานได้แล้วผู้ใช้ควรจะใช้ S-CSCF ตัวใด โดยการทำงานของกระบวนการข้างต้นจะได้ข้อมูลสิทธิ์การใช้งานจาก HSS โดยใช้ Diameter Protocol และหมายเลข IP ของ I-CSCF ถูกเก็บไว้ใน DNS เช่นเดียวกับ P-CSCF

4. S-CSCF ทำหน้าที่เป็นเครื่องแม่ข่าย ซึ่งเป็นส่วนหนึ่งของ OpenIMSCore โดยจะทำหน้าที่ในการเลือกว่าสัญญาณที่เข้ามาถึง S-CSCF ถ้าไม่ใช้สัญญาณ REGISTER ว่าจะต้องส่งต่อไปยัง Application Server ใด โดยจะกระบวนการ IFC โดยการตั้งค่าดังกล่าวจะอธิบายในหัวข้อถัดไป และการเข้าถึงหมายเลข IP ของ S-CSCF ก็มีลักษณะเดียวกับ P-CSCF และ I-CSCF
5. HSS ทำหน้าที่เป็นฐานข้อมูลที่ใช้ในการเก็บข้อมูลของผู้ใช้ และค่าเริ่มต้น (Network Configuration) ของระบบ IMS Network โดย HSS เป็นส่วนหนึ่งของ OpenIMSCore
6. Web Management ทำหน้าที่เป็นเว็บเซิร์ฟเวอร์ ที่มีวัตถุประสงค์เพื่อความสะดวกในรับค่าติดตั้งต่าง ๆ จากผู้ใช้ให้กับ IMS Network

จากรายละเอียดขององค์ประกอบต่าง ๆ ในระบบ IMS ที่อธิบายไว้ข้างต้นระบบ IMS ยังไม่สามารถที่จะทำงานร่วมกับระบบบริการแสดงสถานะแบบรวมศูนย์ได้ ซึ่งจำเป็นต้องพัฒนาการเชื่อมต่อระหว่างระบบบริการแสดงสถานะแบบรวมศูนย์กับระบบ IMS โดยจะอธิบายรายละเอียดในหัวข้อ 5.2.3

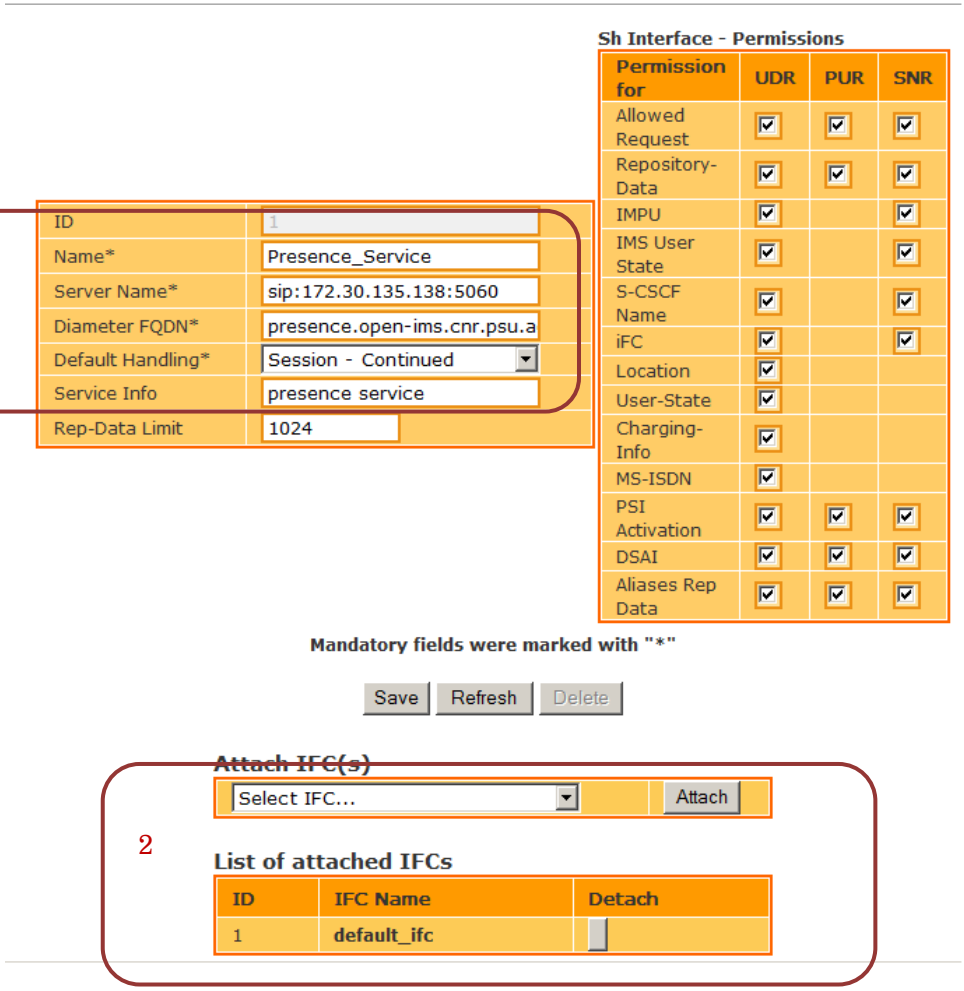
5.2.3 การพัฒนาการเชื่อมต่อระหว่างระบบบริการแสดงสถานะแบบรวมศูนย์กับระบบ IMS

การพัฒนาการเชื่อมต่อระหว่างระบบ IMS และระบบบริการแสดงสถานะแบบรวมศูนย์ เมื่อพิจารณาจากการทำงานของระบบ IMS [21][22] จะพบว่าระบบ IMS จะตัดสินใจในการส่งรับสัญญาณต่าง ๆ ที่ S-CSCF โดยการใช้กระบวนการ IFC (Initial Filter Criteria) ซึ่งระบบ OpenIMSCore ที่ใช้เป็นพื้นฐานในการพัฒนาระบบ IMS ได้ออกแบบโปรแกรมประยุกต์ในรูปแบบเว็บ เพื่อรับข้อมูลการตั้งค่าสำหรับกระบวนการ IFC ไว้แล้ว ในกาพัฒนาการเชื่อมต่อระหว่างระบบบริการแสดงสถานะแบบรวมศูนย์กับระบบ IMS จะให้การปรับตั้งค่าให้ S-CSCF ของ IMS สามารถที่จะรับส่งสัญญาณที่มาจาก SIP Proxy and Registrar Server ของระบบบริการแสดงสถานะแบบรวมศูนย์ได้

5.2.3.1 การตั้งค่ากระบวนการ IFC

โดยการตั้งค่ากระบวนการ IFC จะแบ่งเป็น 2 ขั้นตอน คือ การตั้งค่า Application Service Profile และ การตั้งค่า Trigger Point

การตั้งค่า Application Service Profile เพื่อทำให้ S-CSCF รับทราบว่ามี Application Server ใดบ้างที่ทำงานอยู่ในระบบ โดยการพัฒนาจะมีการตั้งค่าดังรูปที่ 5-7

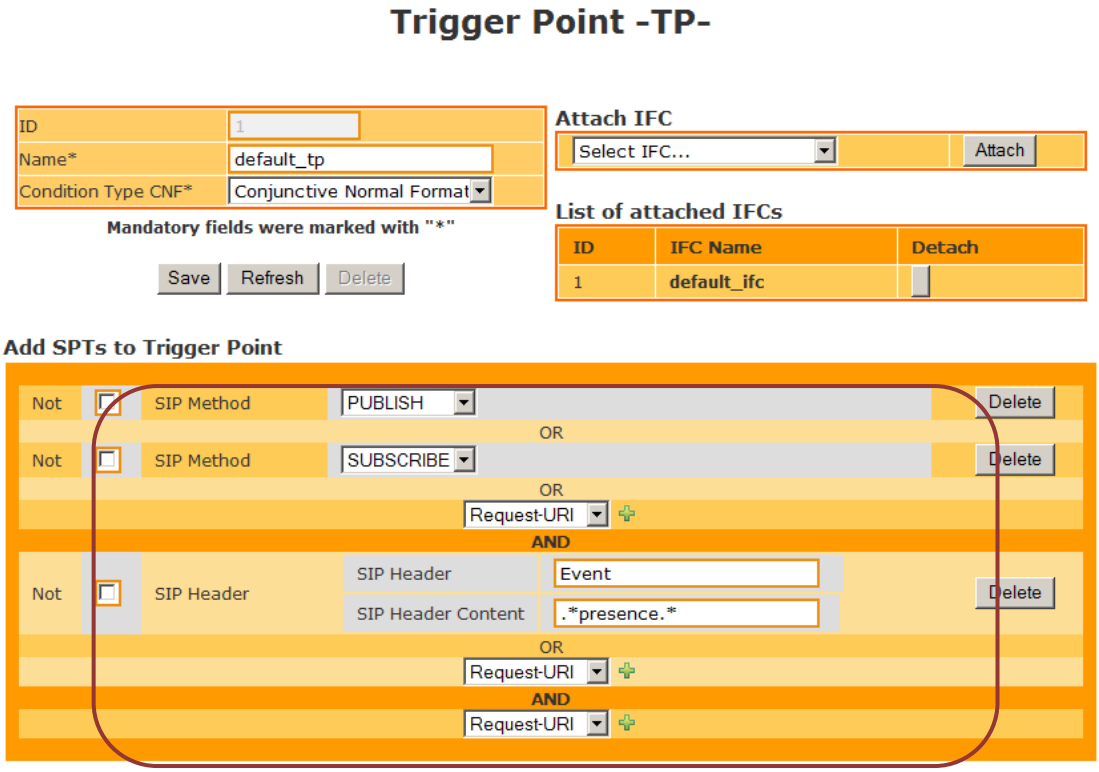


รูปที่ 5-7 การตั้งค่า Application Server Service Profile

จากรูปที่ 5-7 แสดงหน้าเว็บที่ใช้ในการตั้งค่า Application Service Profile โดยในกรอบที่หนึ่งแสดงให้เห็นถึงชื่อ Service โดยในการพัฒนาตั้งค่าเป็น Presence_Service ค่าของ Server Name จะแสดงให้เห็นถึง IP Address และ Port ของ Presence Server ในระบบการสื่อสารแบบรวมศูนย์ โดยตั้งค่าเป็น sip:172.30.135.138:5060 ค่าของ Diameter FQDN จะแสดงถึงชื่อ URI ของ Presence Server โดยจะตั้งค่าเป็น presence.opem-ims.cnr.psu.ac.th และในกรอบ

ที่ 2 แสดงว่า Application Service Profile ของ Presence_Service จะต้องมี IFC ชื่ออะไร โดยในพัฒนาใช้ชื่อ default_ifc

การตั้งค่า Trigger Point เพื่อที่จะทำให้ S-CSCF รู้ว่าจะจัดการสัญญาณที่เข้ามาได้อย่างไร โดยจากรูปที่ 5-8 แสดงให้เห็นถึงการตั้งค่า Trigger Point ที่จะจัดการสัญญาณ PUBLISH และ SUBSCRIBE ที่ใช้ระบบบริการแสดงสถานะระหว่างระบบ IMS และระบบบริการแสดงสถานะแบบรวมศูนย์



รูปที่ 5-8 การตั้งค่า Trigger Point ของการทำ IFC

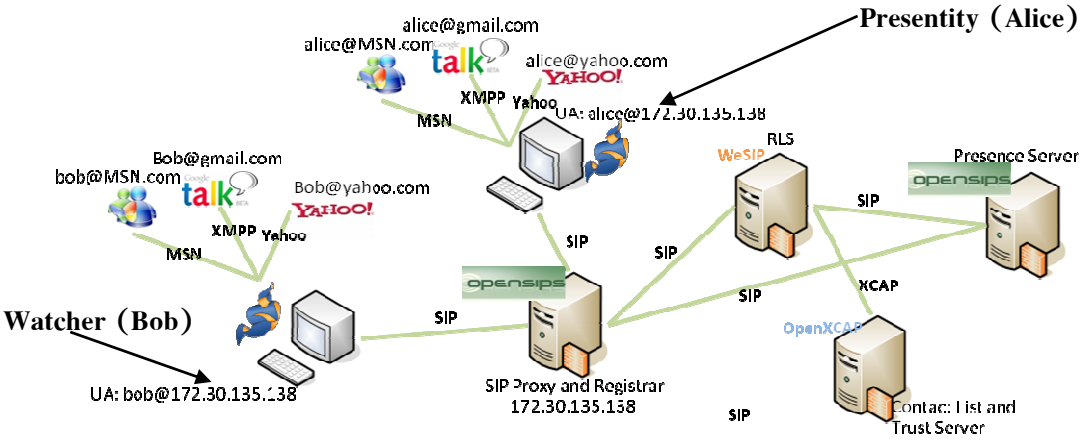
การตั้งค่า Trigger Point จะเป็นการกำหนดเงื่อนไขในการส่งสัญญาณ โดยถ้าเป็นสัญญาณ PUBLISH หรือ SUBSCRIBE และมี Header เป็น Event มีคำว่า “presence” ใน Header อื่น ๆ ก็จะทำการส่งสัญญาณไปยัง Presence Server ในระบบบริการแสดงสถานะแบบรวมศูนย์ที่ได้กำหนดไว้ใน Application Service Profile

5.3 การทดสอบระบบบริการแสดงสถานะแบบรวมศูนย์

ในหัวข้อนี้เป็นการทดสอบระบบบริการแสดงสถานะแบบรวมศูนย์ โดยแบ่งการทดสอบเป็น 3 ส่วน คือ การทดสอบฟังก์ชันรวบรวมข้อมูลแสดงสถานะที่เครื่องลูกข่าย การทดสอบการทำงานของ RLS และการทดสอบการทำงานการเชื่อมต่อระหว่างระบบบริการแสดงสถานะแบบรวมศูนย์กับ IMS โดยแต่ละส่วนมีรายละเอียดดังนี้

ส่วนที่ 1 การทดสอบฟังก์ชันการรวบรวมข้อมูลแสดงสถานะที่เครื่องลูกข่าย

การทดสอบฟังก์ชันการรวบรวมข้อมูลแสดงสถานะที่เครื่องลูกข่าย สามารถแสดงได้ดังรูปที่ 5-9 โดยการทดสอบจะแสดงให้เห็นว่าการทำงานของโปรโตคอลแต่ละโปรโตคอลสามารถทำงานร่วมกันได้



รูปที่ 5-9 การทดสอบส่วนที่ 1 PI Aggregation

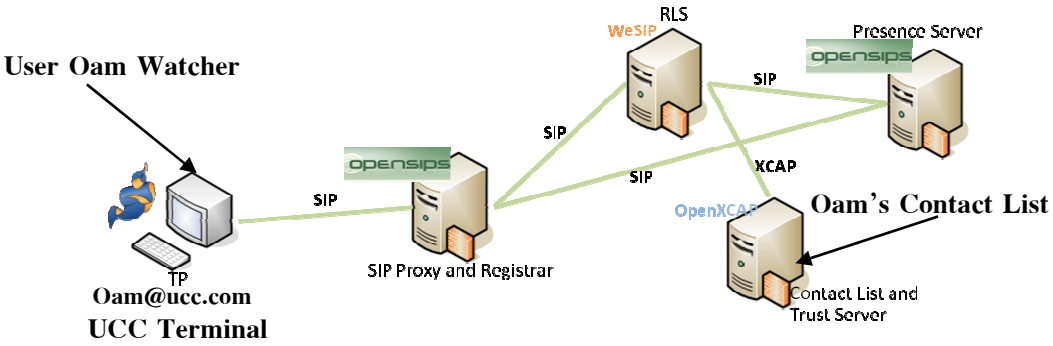
จากรูปที่ 5-9 ในการทดสอบส่วนที่ 1 กำหนดให้มีผู้ใช้สองคนในตัวอย่าง คือ Alice และ Bob เป็นผู้ใช้ระบบการสื่อสารแบบรวมศูนย์ โดยกำหนดให้ Alice เป็นผู้แสดงสถานะ (Presentity) ซึ่ง Alice มีช่องทางการสื่อสาร คือ GTalk, MSN และ Yahoo (Primary Account) และ Bob ได้ถูกกำหนดให้เป็นผู้มองสถานะ (Watcher) ซึ่ง Bob จะทำหน้าที่ในการร้องขอและรับข้อมูลแสดงสถานะของ Alice ผ่านทาง UCC Terminal ที่มีความสามารถในการรวบรวมข้อมูลแสดงสถานะ โดย Alice จะเปลี่ยนแปลงข้อมูลแสดงสถานะในช่องทางการสื่อสารต่างๆ ตามกรณีย่อยดังตารางที่ 5-1 ซึ่ง Bob จะเห็นข้อมูลแสดงสถานะของ Alice เปลี่ยนแปลงไปตามผลที่คาดว่าจะได้รับตามตารางที่ 5-1

ตารางที่ 5-1 ตารางข้อมูลแสดงสถานะของ Alice และผลที่คาดว่า Bob จะได้รับ

	Yahoo	Gtalk	MSN	ผลที่คาดว่า Bob จะได้รับ
กรณีย่อยที่ 1	Available	Busy	Away	Available “Do Not Disturb”
กรณีย่อยที่ 2	Busy	Available	Online	Busy “Do Not Disturb”
กรณีย่อยที่ 3	offline	Busy	Online	Available “Do Not Disturb”

ส่วนที่ 2 การทดสอบการทำงานของ RLS

สำหรับการทดสอบการทำงานของ RLS สามารถแสดงได้ดังรูปที่ 5-10 โดยการทดสอบจะแสดงให้เห็นถึงการทำงานตามกระบวนการของ RLS ที่ได้ออกแบบไว้ได้

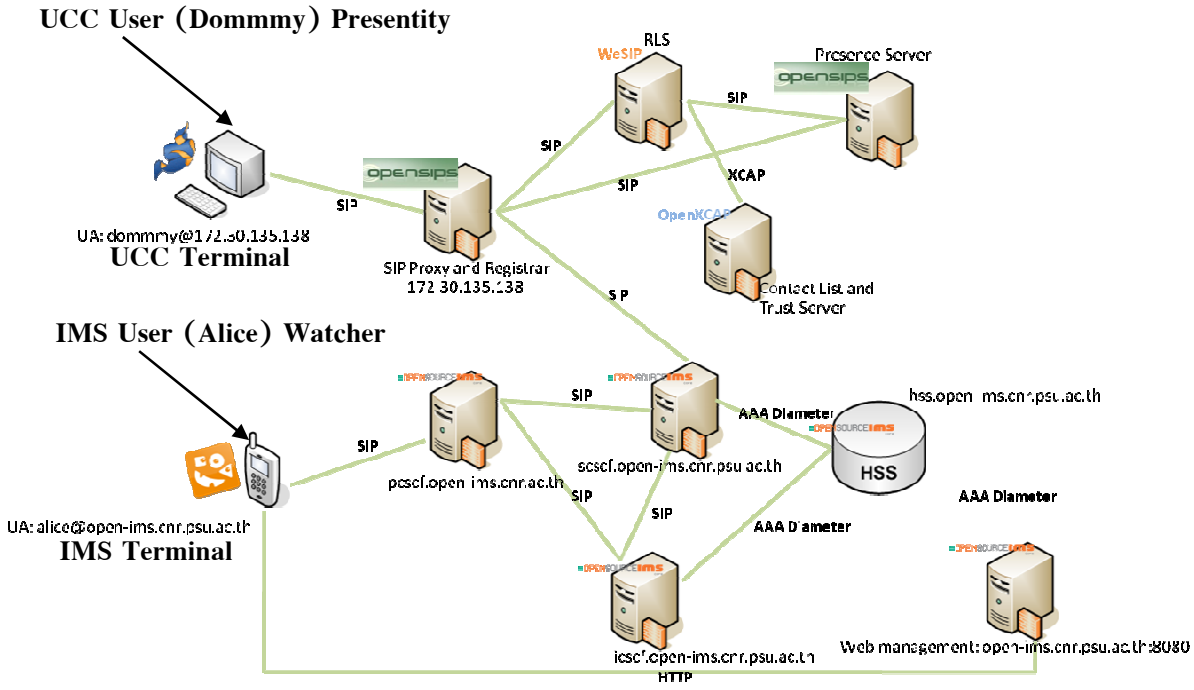


รูปที่ 5-10 การทดสอบส่วนที่ 2 การทดสอบการทำงานของ RLS

จากรูปที่ 5-10 กำหนดให้ Oam เป็นผู้ใช้ในระบบบริการแสดงสถานะแบบรวมศูนย์ ที่ทำงานผ่าน UCC Terminal มี SIP URI เป็น Oam@ucc.com และทำหน้าที่เป็นผู้มองสถานะ โดย Oam จะเริ่มต้นกระบวนการลงทะเบียนเข้าใช้ระบบบริการแสดงสถานะแบบรวมศูนย์ จากนั้น Oam จะร้องขอข้อมูลแสดงสถานะไปยังระบบบริการแสดงสถานะแบบรวมศูนย์ การร้องขอข้อมูลแสดงสถานะจะมีการระบุว่าเป็นการร้องขอข้อมูลแสดงสถานะแบบใช้กระบวนการทำงานของ RLS

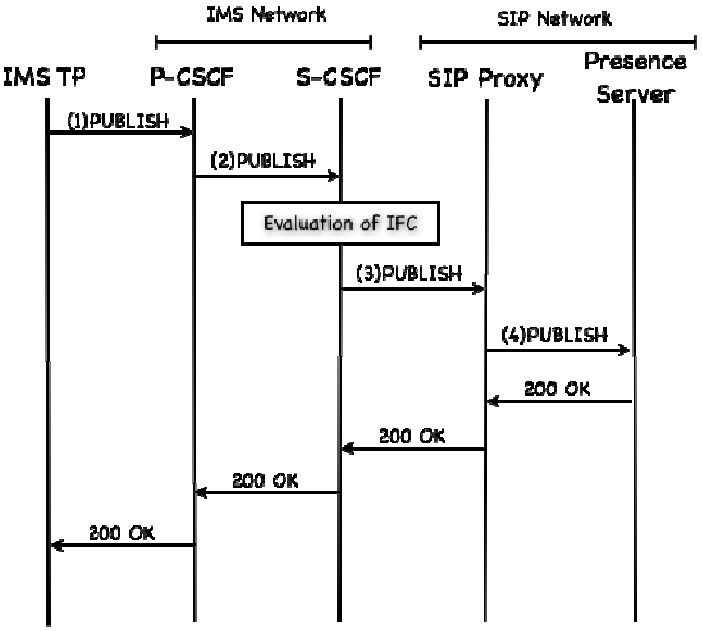
ส่วนที่ 3 การทดสอบการทำงานการเชื่อมต่อระหว่างระบบบริการแสดงสถานะแบบรวมศูนย์กับ IMS

สำหรับการทดสอบการทำงานการเชื่อมต่อระหว่างระบบบริการแสดงสถานะแบบรวมศูนย์กับ IMS สามารถแสดงได้ดังรูปที่ 5-11 โดยการทดสอบจะแสดงให้เห็นถึงการเชื่อมระหว่างระบบบริการแสดงสถานะแบบรวมศูนย์กับระบบ IMS ได้

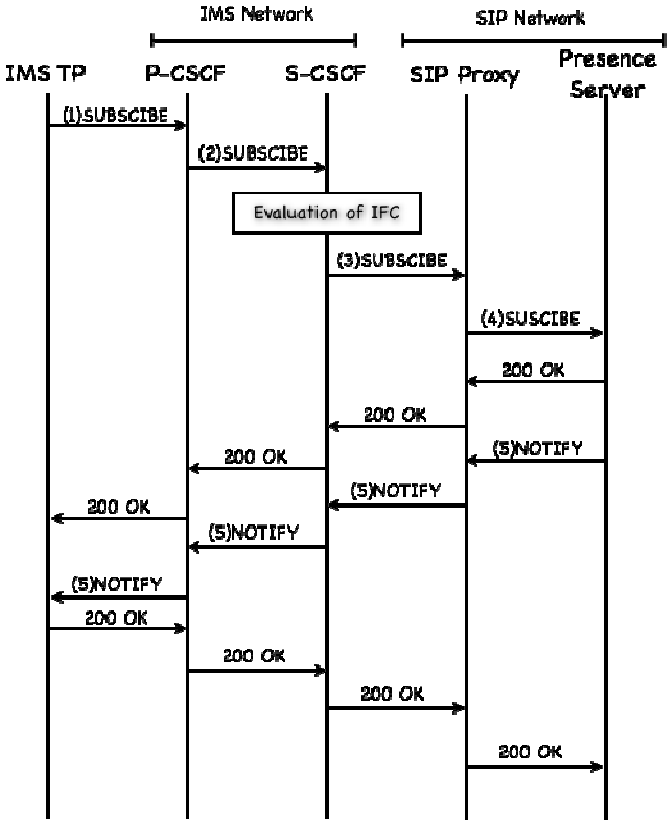


รูปที่ 5-11 การทดสอบส่วนที่ 3 การ Subscription และ Publication ระหว่าง IMS และ SIP Network

จากรูปที่ 5-11 กำหนดให้มีผู้ใช้ 2 คนที่ทำงานอยู่ในระบบบริการแสดงสถานะแบบรวมศูนย์และระบบ IMS โดย Alice ทำงานอยู่บน Monster ซึ่งเป็นเครื่องลูกข่ายที่ทำงานในระบบ IMS และ Dommmy ทำงานอยู่บน SIP Communicator ที่เป็นเครื่องลูกข่ายในระบบบริการแสดงสถานะแบบรวมศูนย์ โดย Alice จะทำการ Publication ข้อมูลแสดงสถานะของตนเอง และ Subscription เพื่อร้องขอข้อมูลแสดงสถานะไปยังระบบบริการแสดงสถานะแบบรวมศูนย์ โดยควรจะมีลำดับของสัญญาณในการ Publication และ Subscription ดังรูปที่ 5-12 และ รูปที่ 5-13 ตามลำดับ



รูปที่ 5-12 IMS Terminal แสดงสถานะของตนเองไปยัง SIP Network



รูปที่ 5-13 การร้องขอข้อมูลแสดงสถานะจาก IMS ไปยัง SIP Network

5.4 ผลการทดสอบ

จากการทดสอบทั้ง 3 ส่วนข้างต้นได้นำมาทดสอบกับระบบที่ได้พัฒนาขึ้น และทำการเก็บรวบรวมผลโดยการจับภาพการแสดงผลจากโปรแกรมเครื่องลูกข่าย และการใช้โปรแกรม Wireshark สำหรับการตรวจสัญญาณ โดยผลของการทดสอบจะแยกแสดงผลเป็นแต่ละกรณีดังนี้

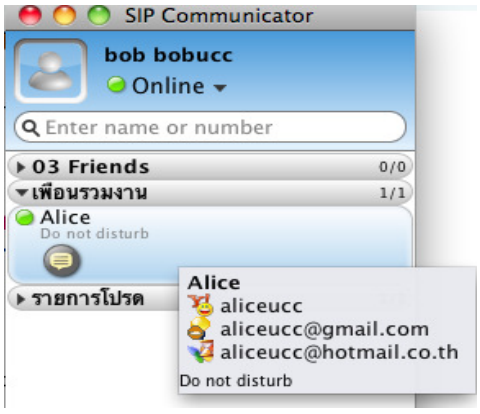
ส่วนที่ 1 การทดสอบฟังก์ชันการรวบรวมข้อมูลแสดงสถานะที่เครื่องลูกข่าย

สำหรับการทดสอบฟังก์ชันการรวบรวมข้อมูลแสดงสถานะที่เครื่องลูกข่ายเป็นการจับภาพการแสดงผลจากเครื่องลูกข่ายของ Bob เมื่อ Alice เปลี่ยนแปลงข้อมูลแสดงสถานะตามกรณีย่อยต่าง ๆ ที่ได้กำหนดไว้ โดยผลการทดสอบแบ่งตามกรณีย่อยได้ดังนี้

จากกรณีย่อยที่ 1 Alice ทำหน้าที่เป็นผู้แสดงสถานะ ซึ่ง Alice มีช่องทางการสื่อสาร 3 ช่องทาง คือ YaHoo, Gtalk และ MSN เมสเซนเจอร์ โดย Alice ได้แสดงข้อมูลแสดงสถานะของแต่ละช่องทางการสื่อสารดังนี้

1. ช่องทางการสื่อสาร YaHoo เมสเซนเจอร์แสดงข้อมูลแสดงสถานะเป็น Available
2. ช่องทางการสื่อสาร Gtalk เมสเซนเจอร์แสดงข้อมูลแสดงสถานะเป็น Busy
3. และช่องทางการสื่อสาร MSN เมสเซนเจอร์แสดงข้อมูลแสดงสถานะเป็น Away

ผลทดสอบการแสดงผลของ Bob ผ่านทาง UCC Terminal ที่มีการรวบรวมข้อมูลแสดงสถานะของ Alice ทั้ง 3 ช่องทางการสื่อสารเป็น Available “Do Not Disturb” ดังรูปที่ 5-14 ซึ่งเป็นผลการทดสอบที่ถูกต้องตามการพัฒนา

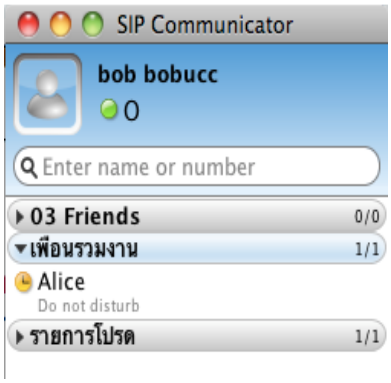


รูปที่ 5-14 ผลการทดสอบจากกรณีย่อยที่ 1

จากกรณีย่อยที่ 2 Alice ได้แสดงข้อมูลแสดงสถานะของแต่ละช่องทางการสื่อสาร ดังนี้

1. ช่องทางการสื่อสาร YaHoo เมสเซนเจอร์แสดงข้อมูลแสดงสถานะเป็น Busy
2. ช่องทางการสื่อสาร Gtalk เมสเซนเจอร์แสดงข้อมูลแสดงสถานะเป็น Available
3. และช่องทางการสื่อสาร MSN เมสเซนเจอร์แสดงข้อมูลแสดงสถานะเป็น Online

ผลการทดสอบการแสดงผลของ Bob ผ่านทาง UCC Terminal ที่มีการรวบรวมข้อมูลแสดงสถานะของ Alice ทั้ง 3 ช่องทางการสื่อสารเป็น Busy “Do Not Disturb” ดังรูปที่ 5-15 ซึ่งเป็นผลการทดสอบที่ถูกต้องตามการพัฒนา

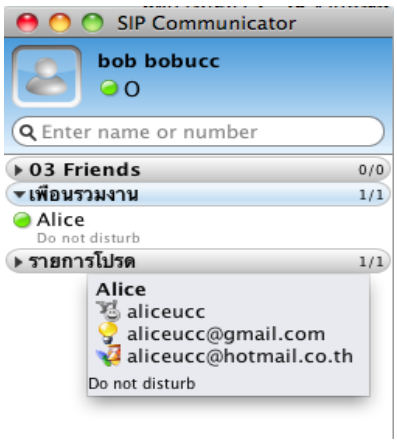


รูปที่ 5-15 ผลการทดสอบจากกรณีย่อยที่ 2

จากกรณีย่อยที่ 3 Alice ได้แสดงข้อมูลแสดงสถานะของแต่ละช่องทางการสื่อสาร ดังนี้

1. ช่องทางการสื่อสาร YaHoo เมสเซนเจอร์แสดงข้อมูลแสดงสถานะเป็น Offline
2. ช่องทางการสื่อสาร Gtalk เมสเซนเจอร์แสดงข้อมูลแสดงสถานะเป็น Busy
3. และช่องทางการสื่อสาร MSN เมสเซนเจอร์แสดงข้อมูลแสดงสถานะเป็น Online

ผลการทดสอบการแสดงผลของ Bob ผ่านทาง UCC Terminal ที่มีการรวบรวมข้อมูลแสดงสถานะของ Alice ทั้ง 3 ช่องทางการสื่อสารเป็น Available “Do Not Disturb” ดังรูปที่ 5-16 ซึ่งเป็นผลการทดสอบที่ถูกต้องตามการพัฒนา



รูปที่ 5-16 ผลการทดลองจากกรณีย่อยที่ 3

จากผลการทดสอบทั้งสามกรณีเห็นได้ว่า ได้ผลไปตามที่คาดหวังไว้ แต่เนื่องด้วย SIP Communicator ไม่มีสัญลักษณ์ของสถานะ Busy จึงต้องแสดงผลโดยใช้สัญลักษณ์ Away แทน

ส่วนที่ 2 การทดสอบการทำงานของ RLS

สำหรับการทดสอบการทำงานของ RLS เป็นกระบวนการร้องขอข้อมูลแสดงสถานะที่ระบุการใช้งาน RLS ของ Oam ไปยังระบบบริการแสดงสถานะแบบรวมศูนย์ โดยจะแสดงผลลัพธ์จากการดักจับสัญญาณที่โปรแกรมเครื่องลูกข่ายของ Oam และการจับภาพการแสดงผลของโปรแกรมเครื่องลูกข่ายของ Oam ได้ดังนี้

```
SUBSCRIBE sip:oam@172.30.141.39 SIP/2.0
Call-ID: 2d01f7fc0f5a023d190ca40a553cb9b3@0.0.0.0
CSeq: 1 SUBSCRIBE
From: <sip:oam@172.30.141.39>;tag=959ea5b8
To: <sip:oam@172.30.141.39>
Via: SIP/2.0/UDP 172.30.140.255:5060;branch=z9hG4bK96a477e9dec0c0f0648bf675dc56b
385
Max-Forwards: 69
Event:
Supported: eventlist
Contact: ""oam"" <sip:oam@172.30.140.255:5060;transport=udp;registering_acc=172_
30_141_39>
Accept: application/pidf+xml
Expires: 0
User-Agent: OpenUCC0.1-alphaWindows XP
Content-Length: 0
```

รูปที่ 5-17 สัญญาณ SUBSCRIBE จากเครื่องลูกข่ายของ Oam

จากรูปที่ 5-17 สัญญาณ SUBSCRIBE ที่โปรแกรม Wireshark สามารถดักจับได้จากการที่เครื่องลูกข่ายของ Oam ร้องขอข้อมูลไปยังระบบการสื่อสารแบบรวมศูนย์ โดยสัญญาณ SUBSCRIBE จะมีการระบุ Header Supported เป็น eventlist เป็นการบอกว่าสามารถทำงานร่วมกับ RLS ได้ ซึ่งเป็นผลลัพธ์ที่ถูกต้องตามที่ได้มีการพัฒนาไว้

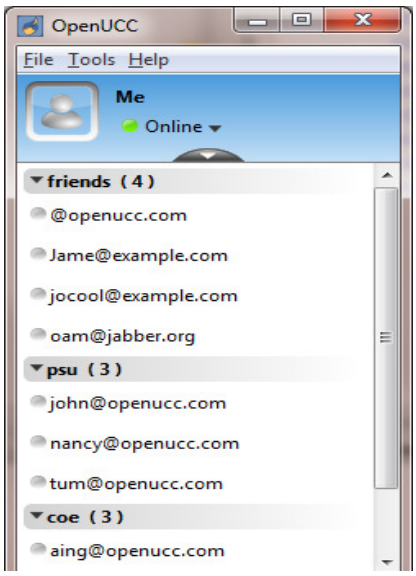
```
NOTIFY sip:oam@172.30.140.255:5060;transport=udp;regist
ering_acc=172_30_141_39 SIP/2.0
Call-ID: 66562da75699fca855951d9948b3729d@0.0.0.0
CSeq: 1 NOTIFY
From: <sip:oam@172.30.141.39>;tag=0419A654435281A6888B557AF9D5C779
To: <sip:oam@172.30.141.39>;tag=88345d12
Max-Forwards: 70
Contact: <sip:172.30.141.39:5060;transport=udp>
Event:  presence
Require: eventlist
SubscriptionState: ative
Expires: 7200
Content-Type: application/pidf+xml
Content-Length: 515
```

รูปที่ 5-18 สัญญาณ NOTIFY สัญญาณแรกจาก RLS ที่ส่งไปยังเครื่องลูกข่ายของ Oam

จากรูปที่ 5-18 แสดงสัญญาณ NOTIFY สัญญาณแรกที่โปรแกรม Wireshark สามารถดักจับได้จากการที่เครื่องลูกข่ายของ Oam รับข้อมูลแสดงสถานะจาก RLS ของระบบการสื่อสารแบบรวมศูนย์ โดยสัญญาณ NOTIFY มีการระบุ Header Require เป็น eventlist เป็นการบอกว่าเป็นสัญญาณที่มาจาก RLS และส่งข้อมูลบัญชีรายชื่อมากับส่วน Body ของสัญญาณดังรูปที่ 5-19 เพื่อให้เครื่องลูกข่ายของ Oam นำเอาข้อมูลบัญชีรายชื่อไปแสดงผลต่อไป

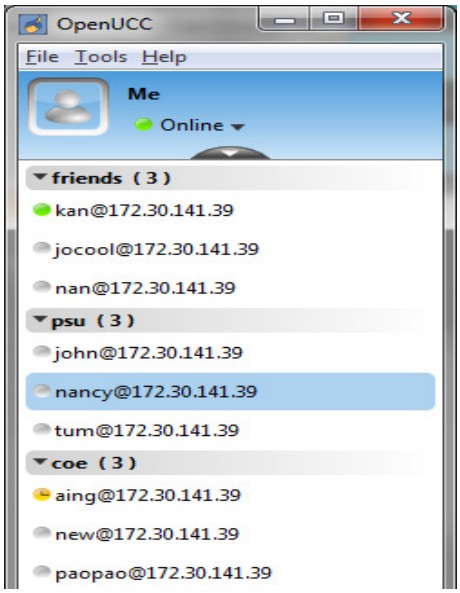
```
<?xml version="1.0" encoding="UTF-8"?>
   <resource-lists xmlns="urn:ietf:params:xml:ns:resource-lists"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <list name="friends">
     <entry uri="sip:oam@jabber.org">
      <display-name>Oam</display-name>
     </entry>
     <entry uri="sip:joe@example.com">
       <display-name>Joe Smith</display-name>
     </entry>
     <entry uri="sip:nancy@example.com">
       <display-name>Nancy Gross</display-name>
     </entry>
    </list>
   </resource-lists>
```

รูปที่ 5-19 บัญชีรายชื่อของ Oam จากส่วน Body ของสัญญาณ NOTIFY สัญญาณแรก



รูปที่ 5-20 SIP Communicator หลังจากรับ NOTIFY สัญญาณแรก

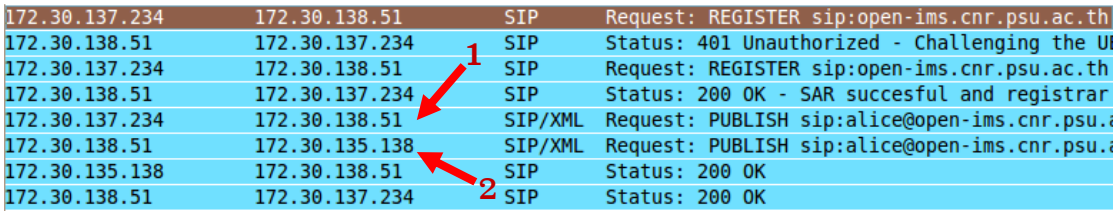
จากรูปที่ 5-20 เป็นการจับภาพที่เครื่องลูกข่ายของ Oam หลังจากรับสัญญาณ NOTIFY สัญญาณแรกจาก RLS ซึ่งเครื่องลูกข่ายของ Oam สามารถแสดงบัญชีรายชื่อได้ ซึ่งเป็นไปตามการพัฒนาของ SIP Communicator จากนั้น RLS ได้ SUBSCRIBE ข้อมูลตามบัญชีรายชื่อที่ส่งมากับสัญญาณ NOTIFY สัญญาณแรก แล้วจะส่งข้อมูลแสดงสถานะกลับมายังเครื่องลูกข่ายของ Oam โดยสามารถจับภาพที่เครื่องลูกข่ายของ Oam ได้ดังรูปที่ 5-21



รูปที่ 5-21 SIP Communicator เมื่อรับ NOTIFY ที่มีข้อมูลการแสดงสถานะ

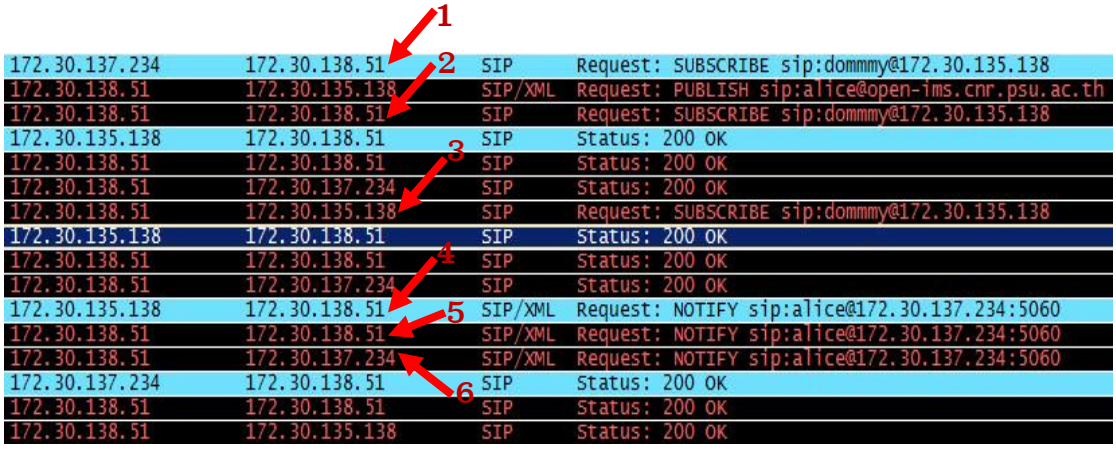
<u>ส่วนที่ 3</u> การทดสอบการทำงานการเชื่อมต่อระหว่างระบบบริการแสดงสถานะแบบรวมศูนย์กับ IMS

สำหรับการทดสอบการทำงานการเชื่อมต่อระหว่างระบบบริการแสดงสถานะแบบรวมศูนย์กับ IMS นี้เป็นการทดสอบการแสดงข้อมูลสถานะของ Alice ที่อยู่ในระบบ IMS ไปยัง Presence Server ที่อยู่ในระบบการสื่อสารแบบรวมศูนย์ และกระบวนการร้องขอข้อมูลแสดงสถานะจาก Alice ที่อยู่ในระบบ IMS ไปยัง Dommmy ที่อยู่ในระบบการสื่อสารแบบรวมศูนย์ โดยผลลัพธ์ที่ได้จากการดักจับด้วยโปรแกรม Wireshark แสดงได้ดังรูปที่ 5-22 และ รูปที่ 5-23 ตามลำดับ นอกจากนี้มีการจับภาพเครื่องลูกข่ายของ Alice ที่สามารถแสดงผลข้อมูลแสดงสถานะของ Dommmy ได้ดังรูปที่ 5-24



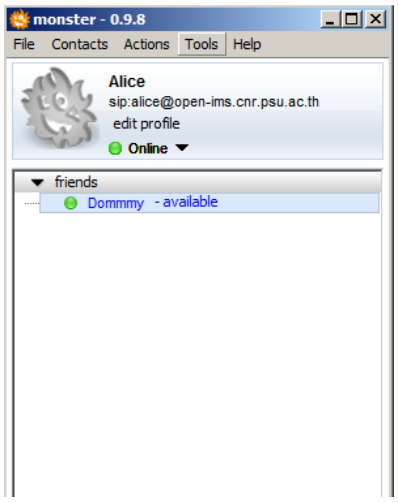
รูปที่ 5-22 การ Publication ในฝั่ง IMS Network

จากรูปที่ 5-22 แสดงให้เห็นกระบวนการ Publication เป็นไปตามรูปแบบของลำดับสัญญาณที่ควรจะได้รับตามการทดสอบส่วนที่ 3 โดยเครื่องลูกข่ายของ Alice เริ่มต้นส่งสัญญาณ PUBLISH ไปยัง P-CSCF ในสัญญาณที่ 1 และ P-CSCF ส่งต่อสัญญาณไปยัง S-CSCF จากนั้น S-CSCF จะพิจารณาสัญญาณด้วยกระบวนการ IFC ซึ่งทำให้ทราบว่า S-CSCF จะต้องส่งสัญญาณต่อไปยัง Presence Server ตามสัญญาณที่ 2



รูปที่ 5-23 การ Subscription และ Notification ในฝั่ง IMS Network

จากรูปที่ 5-23 แสดงให้เห็นกระบวนการ Subscription และ Notification เพื่อร้องขอข้อมูลแสดงสถานะของ Dommmy ซึ่งเป็นไปตามรูปแบบลำดับสัญญาณตามการทดสอบส่วนที่ 3 โดยเริ่มต้นจากเครื่องลูกข่ายของ Alice ส่งสัญญาณ SUBSCRIBE ไปยัง P-CSCF (สัญญาณที่ 1) จากนั้น P-CSCF ส่งสัญญาณต่อไปให้ S-CSCF (สัญญาณที่ 2) เมื่อ S-CSCF ได้รับสัญญาณ SUBSCRIBE ก็จะใช้กระบวนการ IFC พิจารณาว่าควรจะส่งสัญญาณไปยัง Application Server ใด ซึ่งในที่นี้เป็น Presence Server ในระบบการสื่อสารแบบรวมศูนย์ ดังนั้น S-CSCF จึงส่งสัญญาณ SUBSCRIBE ไปให้กับ Presence Server (สัญญาณที่ 3) เมื่อ Presence Server ได้รับสัญญาณ SUBSCRIBE ก็รับทราบได้ว่า Alice ต้องการข้อมูลแสดงสถานะของ Dommmy ก็จะสร้างสัญญาณ NOTIFY ที่ส่งข้อมูลแสดงสถานะของ Dommmy จึงทำการส่งกลับไปยัง S-CSCF (สัญญาณที่ 4) และ S-CSCF จะเห็นว่าเป็น Session เดียวกับการสัญญาณ SUBSCRIBE ก่อนหน้านี้ S-CSCF ก็จะส่งสัญญาณ NOTIFY ต่อไปให้ P-CSCF (สัญญาณที่ 5) จากนั้น P-CSCF ก็จะส่งสัญญาณ NOTIFY ไปยังเครื่องลูกข่ายของ Alice เพื่อแสดงผลข้อมูลแสดงสถานะที่ส่งมาต่อไป (สัญญาณที่ 6) โดยสามารถแสดงผลได้ดังรูปที่ 5-24



รูปที่ 5-24 การแสดงผลของเครื่องลูกข่ายของ Alice (Monster)

5.5 สรุป

ในบทนี้ได้นำเสนอระบบบริการแสดงสถานะแบบรวมศูนย์ที่ได้พัฒนาขึ้น โดยมีการทำงานตามรูปแบบสถาปัตยกรรมที่ได้ออกแบบไว้ในบทที่ 4 ยกเว้น Unified Server ที่จะถือว่าระบบมีได้รับค่าต่าง ๆ เช่น ข้อมูลบริบท บัญชีรายชื่อ และข้อมูล Unified Contact ไว้แล้ว ระบบที่พัฒนาขึ้น จะแบ่งเป็น 2 ส่วน คือ ส่วนระบบบริการแสดงสถานะแบบรวมศูนย์ภายใน และ ส่วนการเชื่อมต่อกับระบบ IMS ในบทนี้ได้แสดง 3 ส่วนที่ใช้ในการทดสอบกับระบบที่ได้พัฒนาขึ้น คือ การทดสอบฟังก์ชันรวบรวมข้อมูลแสดงสถานะที่เครื่องลูกข่าย การทดสอบการทำงานของ RLS และกระบวนการทดสอบการทำงานเชื่อมต่อกับ IMS จากผลการทดสอบของ 3 กรณี ได้แสดงให้เห็นถึงความสามารถที่เพิ่มเติมจากระบบบริการแสดงสถานะแบบเดิม เช่น ระบบสามารถรวบรวมข้อมูลแสดงสถานะที่มีโปรโตคอลแตกต่างกันได้ ซึ่งจะทำให้ความสับสนสำหรับผู้มองสถานะ ระบบสามารถทำงานกับ RLS ที่มีการเพิ่มเติมความสามารถในการส่งข้อมูลบริบท Relationship และ Level of Trust ไปยัง Presence Server เพื่อใช้ในการพิจารณาได้ และระบบสามารถที่จะทำงานร่วมกับระบบ IMS ซึ่งจะทำให้ระบบการสื่อสารแบบรวมศูนย์สามารถที่จะติดต่อกับผู้ใช้ได้ง่ายยิ่งขึ้น

บทที่ 6
สรุปผลการดำเนินงาน

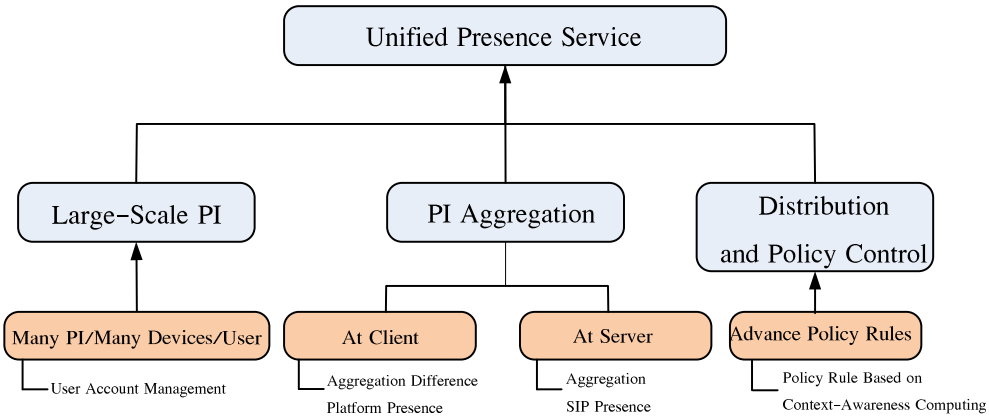
บทนี้กล่าวสรุปผลการวิจัยที่ได้ดำเนินการในวิทยานิพนธ์นี้ รวมทั้งข้อเสนอแนะแนวทางต่าง ๆ ที่จะเป็นประโยชน์ต่อการนำวิทยานิพนธ์นี้ ไปใช้ในการวิจัยเพิ่มเติมในการปรับปรุงระบบบริการแสดงสถานะแบบรวมศูนย์

6.1 สรุปผลการวิจัย

จากปัญหาของ Massive PI ที่เกิดขึ้นกับระบบการสื่อสารแบบรวมศูนย์ดังที่ได้กล่าวไปแล้วในบทที่ 3 ส่งผลให้เกิดการออกแบบระบบการแสดงสถานะแบบรวมศูนย์ทั้งในรูปแบบของการออกแบบกรอบการทำงาน และการออกแบบสถาปัตยกรรม ดังที่กล่าวในบทที่ 4 เพื่อทำให้ระบบการสื่อสารแบบรวมศูนย์มีประสิทธิภาพสูงขึ้น

6.1.1 การออกแบบกรอบการทำงานของระบบการแสดงสถานะแบบรวมศูนย์

กรอบการทำงานของระบบแสดงสถานะแบบรวมศูนย์สามารถสรุปการทำงานเป็น 3 ฟังก์ชันดังแสดงในรูปที่ 6-1



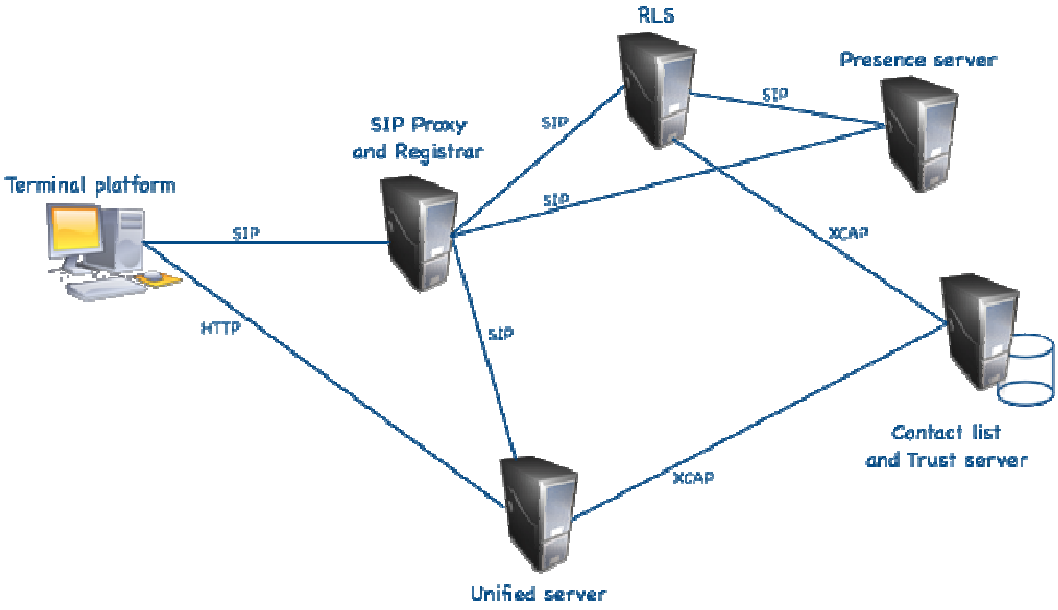
รูปที่ 6-1แสดงถึงฟังก์ชันภายในของ Unified Presence Service เพื่อที่จะใช้จัดการกับปัญหา Massive PI

1. Large-Scale PI เป็นฟังก์ชันในการจัดการข้อมูลการแสดงสถานะอย่างชาญฉลาด เมื่อผู้ใช้แต่ละคนมีจำนวนของข้อมูลแสดงสถานะและบัญชีรายชื่อเป็นจำนวนมาก

2. PI Aggregation เป็นฟังก์ชันในการรวมข้อมูลแสดงสถานะให้เป็นแบบรวมศูนย์อย่างชาญฉลาด
3. Distribution and Policy Control เป็นฟังก์ชันที่ใช้สำหรับการกำหนดสิทธิการเข้าถึงข้อมูลแสดงสถานะอย่าชาญฉลาด

6.1.2 การออกแบบสถาปัตยกรรมของระบบการแสดงสถานะแบบรวมศูนย์

สถาปัตยกรรมของระบบการแสดงสถานะแบบรวมศูนย์ถูกออกแบบอยู่บนพื้นฐานของโปรโตคอล SIP ดังแสดงรูปที่ 6-2 โดยระบบที่ได้ออกแบบไว้สามารถรองรับฟังก์ชันการทำงานของระบบแสดงสถานะแบบรวมศูนย์ตามกรอบการทำงานที่ได้ออกแบบไว้อย่างครอบคลุม

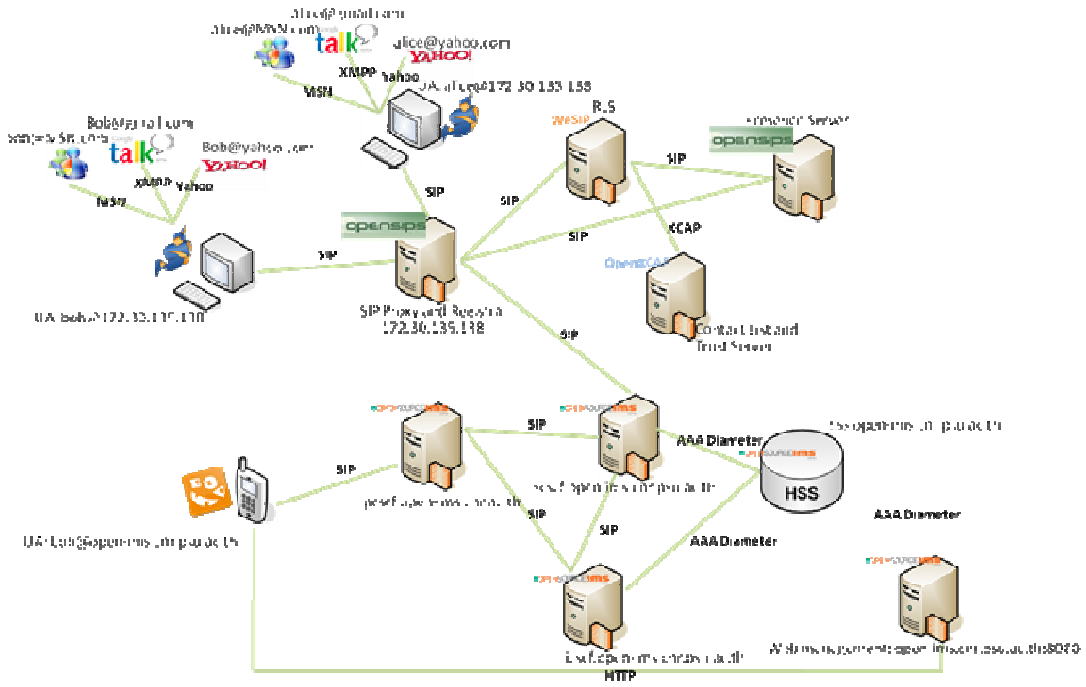


รูปที่ 6-2 ภาพรวมของสถาปัตยกรรมของระบบแสดงสถานะแบบรวมศูนย์

1. กระบวนการ Large-Scale PI ถูกจัดการที่ Unified Server ซึ่งทำหน้าที่จัดเก็บ ข้อมูลบัญชีรายชื่อของผู้ใช้ที่มีในระบบ และเป็นส่วนที่คอยรับข้อมูลบริบท (Relationship และ Level of Trust) เมื่อผู้ใช้ต้องการรายชื่อช่องทางของ ผู้ที่ต้องการติดต่อ โดย Unified Server ช่วยตรวจสอบความถูกต้องของข้อมูลที่ผู้ใช้ใส่เข้ามาเพื่อป้องกันความสับสนของข้อมูล

2. กระบวนการ PI Aggregation ถูกออกแบบให้การทำงานเป็นสองส่วน คือ ที่ Terminal Platform และ Presence Server โดยการทำ PI Aggregation ที่ Terminal Platform เป็นการรวบรวมข้อมูลแสดงสถานะของโปรแกรมอื่น ๆ ที่ไม่ได้ใช้โปรโตคอล SIP และการทำ PI Aggregation ที่ Presence server จะเป็นการรวบรวมข้อมูลแสดงสถานะที่ได้มาจากโปรโตคอล SIP ทั้งหมด
3. กระบวนการ Smart Distribution and Policy Control ที่เป็นการจำกัดสิทธิการเข้าถึงของข้อมูลแสดงสถานะ เป็นการทำงานร่วมกันระหว่าง Presence Server และ Contact List and Trust Server ซึ่งใช้ระบบล่วงรู้บริบทมาช่วยในการตัดสินใจ โดยมี RLS เป็นตัวเชื่อมในการส่งสัญญาณระหว่างเครื่องแม่ข่ายทั้งสอง โดยข้อมูลบริบทได้ถูกออกแบบอยู่ในรูปของ Amoeba model เพื่อสร้างความสัมพันธ์ของผู้ใช้ได้ ในหลายมิติ

นอกจากการออกแบบสถาปัตยกรรมของระบบที่ได้ออกแบบตามกรอบการทำงานแล้ว สถาปัตยกรรมของระบบยังครอบคลุมไปถึงการเชื่อมต่อกับระบบโทรศัพท์เคลื่อนที่ยุคที่สาม (3G) อีกด้วยส่งผลให้ภาพรวมสุดท้ายของสถาปัตยกรรมเป็นดังแสดงในรูปที่ 6-3



รูปที่ 6-3 ภาพรวมของสถาปัตยกรรมแบบรวมศูนย์เมื่อรวมกับระบบโทรศัพท์เคลื่อนที่ยุคที่สาม

6.1.3 การพัฒนาระบบบริการแสดงสถานะแบบรวมศูนย์เพื่อทดสอบการทำงาน

การทดสอบสถาปัตยกรรมของระบบแสดงสถานะแบบรวมศูนย์เพื่อต้องการตรวจสอบว่าระบบที่ถูกออกแบบมีความสามารถที่จะรองรับกรอบการทำงานทั้งหมดหรือไม่ สำหรับการทดสอบระบบที่สามารถแสดงได้ดังรูปที่ 6-3 โดยการทดสอบแบ่งออกเป็น 3 ส่วน ด้วยกัน คือ การทดสอบฟังก์ชันการรวบรวมข้อมูลแสดงสถานะที่เครื่องลูกข่าย การทดสอบการทำงาน RLS และการทดสอบการทำงานการเชื่อมต่อกับระบบ IMS โดยผลการทดสอบทั้ง 3 ส่วน สามารถทำงานได้อย่างถูกต้อง

6.2 ข้อเสนอแนะ

จากวิทยานิพนธ์นี้ยังพบจุดด้อยในบางประการ เช่น การออกแบบข้อมูลบริบท Presence Ontology และการออกแบบสถาปัตยกรรมให้ระบบบริการแสดงสถานะแบบรวมศูนย์อยู่บนพื้นฐานของระบบ IMS โดยข้อเสนอแนะในการปรับปรุงการทำงานข้างต้นมีรายละเอียดดังนี้

6.2.1 การออกแบบข้อมูลบริบทและ Presence Ontology Model

จากวิทยานิพนธ์นี้ไม่ได้เน้นการออกแบบข้อมูลบริบทและ Presence Ontology Model ทำให้การออกแบบความสัมพันธ์ของผู้ใช้ในระบบ เป็นเพียง Relationship และ Level of Trust ซึ่งในแต่ละบุคคล อาจจะมีความสัมพันธ์ที่มากกว่าหนึ่งความสัมพันธ์ในช่วงเวลาหนึ่งๆ และการให้ระดับความเชื่อใจเป็นเปอร์เซ็นต์อาจจะไม่เหมาะสม เพราะผู้ใช้ก็จะให้เปอร์เซ็นต์ของระดับความเข้าใจที่แตกต่างกันได้ โดยที่ความหมายอาจจะเหมือนกัน ทำให้การสร้างความสัมพันธ์แบบ Amoeba อาจไม่ครอบคลุมในทุกกรณี ซึ่งถ้ามีการออกแบบความสัมพันธ์ดังกล่าวให้มีความซับซ้อนและถูกต้องมากยิ่งขึ้นก็อาจจะทำให้ระบบการสื่อสารแบบรวมศูนย์ นั้นมีความฉลาดมากยิ่งขึ้น และอาจจะสามารถสร้างการทำงานแบบใหม่ๆ ได้มากยิ่งขึ้น

ในส่วนของ Presence Ontology Model ถ้าอนุญาตให้ผู้ใช้สามารถที่จะออกแบบรูปแบบ Presence Ontology Model ของตนเองได้ ก็จะทำให้ระบบการสื่อสารแบบรวมศูนย์นั้นมีความยืดหยุ่นสำหรับผู้ใช้มากยิ่งขึ้น เพราะผู้ใช้สามารถที่จะสร้าง Presence Ontology Model ได้เหมาะสมและแสดงตัวตนของการใช้งานระบบการสื่อสารต่างๆ ได้ถูกต้อง

6.2.2 การออกแบบสถาปัตยกรรมให้ระบบบริการแสดงสถานะแบบรวมศูนย์อยู่บนพื้นฐานของระบบ IMS

เนื่องด้วยระบบ IMS ในขณะนี้รองรับกับการทำงานไม่ใช่เฉพาะโทรศัพท์เคลื่อนที่เพียงอย่างเดียว แต่ยังรองรับการเชื่อมต่อในรูปแบบต่าง ๆ ของการสื่อสาร ไม่ว่าจะเป็น PSTN, GSM Network หรือระบบ Internet ADSL เป็นต้น ซึ่งทำให้ระบบ IMS สามารถทำงานครอบคลุมการสื่อสารที่มีรูปแบบการเชื่อมต่อที่แตกต่างกันได้ นอกจากนี้ระบบ IMS ออกแบบขึ้นมาเป็น Common Platform พื้นฐานที่ใช้ในการสร้างการบริการแบบใหม่ ๆ โดยระบบจะอนุญาตให้ผู้พัฒนาสามารถสร้างการบริการใหม่ ๆ ภายในส่วน Application Server ของสถาปัตยกรรม IMS ถ้าระบบบริการแสดงสถานะแบบรวมศูนย์ถูกพัฒนาขึ้นบนสถาปัตยกรรม IMS จะทำให้ระบบทั้งสองไม่จำเป็นต้องสร้างการเชื่อมต่อกัน และระบบจึงไม่จำเป็นจะต้องสนใจในเรื่อง QoS, Firewall หรือ NAT ที่แตกต่างกันระหว่าง Internet Domain และ IMS Domain

เอกสารอ้างอิง

[1] Nomura Research Institute, Ltd, "IT Roadmap 2009," Toyo Keiza INC, Tokyo 2009.

[2] B. Pleasant, "What UC Is and Isn't," http://searchunifiedcommunication.com Last Accessed: April 10, 2010

[3] H. Schulzrinne, "SIMPLE Presence and Event Architecture", Communication System Software and Middleware 2006, pp. 1-9.

[4] R. Shacham, W. Kellerer, H. Schulzrinne, and S. Thakolsri, "Composition for Enhanced SIP Presence," IEEE ISCC, 2007.

[5] S.H. Maes, Universal IM and presence aggregation on technology-specific client, Google Patents, 2009.

[6] C. Ankur, User Presence Aggregation at Server, Google Patents, 2007.

[7] N. Banerjee, D. Chakraborty, K. Dasgupta, S. Mittal, and S. Nagar, "RU-In?-Exploiting Rich Presence and Converged Communications for Next-generation Activity-Oriented Social Networking," Proceedings of the 2009 Tenth International Conference on Mobile Data Management: Systems, Services and Middleware-Volume 00, IEEE Computer Society, 2009, pp. 222-231.

[8] A. Roach, "Session Initiation Protocol (SIP)-specification event notification", RFC 3265, IETF, June 2002.

[9] J. Rosenberg, "Presence authorization rules," RFC 5025, IETF, December 2007.

[10] H. Schulzrinne, H. Tschofenig, J. Morris, J. Cuellar, J. Polk, and J. Rosenberg, "RFC 4745 Common Policy: A Document Format for Expressing Privacy Preferences," IETF. February, 2007.

[11] 3GPP, "Service requirements for the Internet Protocol (IP) multimedia core network subsystem (IMS) Stage 1", TS 22.228, 3rd Generation Partnership Project, March 2008.

[12] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. R. Johnston, J. Peterson, R. Sparks, M. Handley and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, Internet Engineering Task Force, June 2002.

[13] V. Kumar, M. Korpi and S. Sengoodan, "IP Telephony with H.323: Architectures for Unified Networks and Integrated Services", USA: John Wiley & Sons, 2001.

[14] M.Day, J Rosenberg, and H. Sugano, "A model of presence and instant messaging", RFC 2778, Internet Engineering Task Force, Feb. 2000.

[15] P. Saont-Andre, "Extensible messaging and presence protocol (XMPP): core", RFC 3920, IETF, Oct. 2004.

[16] H. Sugano, S. Fujimoto, G. Klyne, A. Bateman, W. Carr, J. Peterson, "Presence Information Data Format", RFC 3863, IETF, Aug. 2004.

[17] J. Rosenberg, "A Data for Presence", RFC 4479, IETF, July 2006.

[18] J. Rosenberg, "The extensible markup language (XML) configuration access protocol (XCAP)", RFC 4827, IETF, July 2006.

[19] S.Ludwig, J.Beda, P. Saint-Andre, R. McQueen, S.Egan, J Hildebrand, "XMPP Extention Protocol-0166: Jingle", XEP-0166 version 1.1, IETF, Dec. 2009.

[20] 3GPP, "Network Architecture", TS 23.002, 3rd Generation Partnership Project, Dec. 2007.

[21] G. Camarillo, M.A.Garcia-Matin, "The 3G IP Multimedia Subsystem: Merging The Internet and The Cellular Worlds", John Wiley and Sons Ltd, 3rd Edition, 2008.

[22] 3GPP, "Presence Service; Architecture and Functional Description; Stage 2", TS 23.141, 3rd Generation Partnership Project, March 2007.

[23] A.B.Roach, B.Campbell, J.Rosenberg, "A Session Initiation Protocol (SIP) Event Notification Extension for Resource Lists", RFC 4662, IETF, Aug. 2006.

[24] M. Mintz-Habib, A. Rawat, H. Schulzrinne, and X. Wu, "A VoIP emergency services architecture and prototype," Computer Communications and Networks, 2005. ICCCN 2005. Proceedings. 14th International Conference on, 2005, pp. 523-528.

[25] NEC, "Unified Communications," March 2008.

[26] A. Mar n-Lopez, F.A.M. DanielD az-Sanchez, F.R. Garc a, J.L. Rizaldos, A. CuberoVega, I. Ormaetxea, J.L. Lopez, S. Fernandez, F. Ortigosa, and E.M. IvanKotuliak, "Interoperability of IP Multimedia Subsystems: The NetLab Approach," Wireless and Mobile Networking: Second Ifip Wg 6.8 Joint Conference, Wmnc 2009, Gdansk, Poland, September 9-11, 2009, Proceedings, Springer-Verlag New York Inc, 2009, p. 20.

[27] S. Schumann, E. Mikoczy, S. Massner, and M. Maruschke, "Presence service integration using interconnected IP multimedia core networks (IM-CN)," 2009.

[28] R. Vargic, M. Krhla, S. Schumann, and I. Kotuliak, "IMS Interworking Using IBCF," Convergence and Hybrid Information Technology, 2008. ICCIT'08. Third International Conference on, 2008.

[29] 3GPP, "Interworking between the IM CN subsystem and IP networks" TS 29.162, 3rd Generation Partnership Project, Dec. 2009.

[30] ETSI-TISPAN, "IP Multimedia Call Control Protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP)" Stage 3, ETSI ES 283 003, Version 1.8.0, September 2007.

[31] ETSI-TISPAN, "Universal Mobile Telecommunications System (UMTS)-3G Security-Access Security for IP-based Services," ETSI ES 133 203, Version 7.7.0, October 2007.

[32] ETSI-TISPAN, "Resource and Admission Control: DIAMETER Protocol for Session based Policy Set-up Information Exchange between the Application Function (AF) abd the Service Policy Decision Function (SPDF)," ETSI ES 183 017, Version 1.4.0, August 2007.

[33] Parlay X Web Service Specification, Version 3.0,

[34] J. Rosenberg, "The extensible markup language (XML) configuration access protocol (XCAP)", RFC 4827, IETF, July 2006.

[35] [1] M. Eisenstadt and M. Dzbor, "Buddyspace: Enhanced presence management for collaborative learning, working, gaming and beyond," JabberConf Europe, 2002.

[36] OpenSIPS, Available from: http://www.opensips.org/ , Last Accessed: April 10, 2009.

[37] WeSIP, Available from: http://www.wesip.com/mediawiki/index.php/Main_Page , Last Accessed: April 10, 2009.

[38] OpenIMSCore, Available from: http://www.openimscore.org: May 10, 2009.

[39] OpenXCAP Project, Available from: http://www.openxcap.org: May 10, 2009.

[40] Monster IMS Client, Available from: http://www.**monster**-the-**client**.org: May 10, 2009.

[41] SIP Communicator, Available from: http://www.sip-communicator.org: May 10, 2009.

ประวัติผู้เขียน

ชื่อ สกุล นายพีระพงษ์ บุญเลิศ

รหัสประจำตัวนักศึกษา 4812103

วุฒิการศึกษา

วุฒิ	ชื่อสถาบัน	ปีที่สำเร็จการศึกษา
วิศวกรรมศาสตรบัณฑิต (วิศวกรรมคอมพิวเตอร์)	มหาวิทยาลัยสงขลานครินทร์	2548

ทุนการศึกษา (ที่ได้รับในระหว่างการศึกษา)

ทุนศิษย์ก้นกุฏิ คณะวิศวกรรมศาสตร์ ปีการศึกษา 2548

การตีพิมพ์เผยแพร่ผลงาน

[1] P. Boonlert, S. Sae-Wong, T. Kamolphiwong, S. Kamolphiwong, “Design Framework and Architecture of Unified Presence Service”, Proceedings of 2010 International Conference on Communications and Mobile Computing (CMC2010), Shenzhen, April 2010.