บทที่ 2

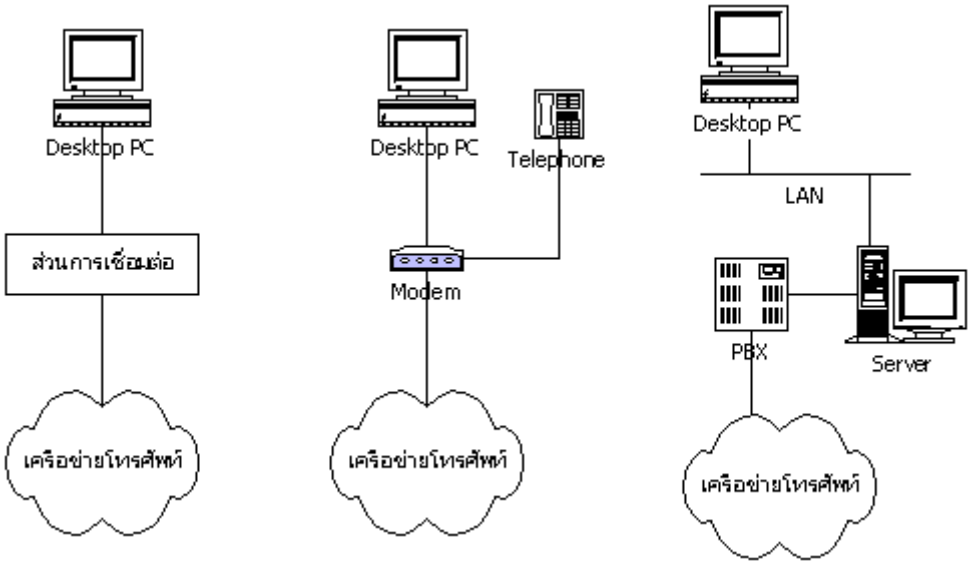
ความรู้เบื้องต้นเกี่ยวกับการติดต่อกับระบบโทรศัพท์

2.1 การเชื่อมต่อคอมพิวเตอร์กับระบบโทรศัพท์

ในปัจจุบันเทคโนโลยีทางด้านไมโครคอมพิวเตอร์ได้มีขีดความสามารถสูงขึ้น จนสามารถเชื่อมโยงการทำงานกันเป็นเครือข่ายเดียวกัน รับส่งข้อมูลถึงกัน จึงเกิดมีแนวคิดที่จะทำให้เครื่องไมโครคอมพิวเตอร์และระบบสื่อสารโทรศัพท์ทำงานร่วมกันได้ ที่เรียกว่า CTI (Computer Telephony Integration) ซึ่งจะทำให้เครื่องไมโครคอมพิวเตอร์ทำหน้าที่เป็นเสมือนโทรศัพท์ และใช้หมายเลขโทรศัพท์ในการติดต่อถึงกัน ข้อดีของการใช้เครื่องไมโครคอมพิวเตอร์คือ การมีหน่วยประมวลผลกลางที่ทำงานได้อย่างมีประสิทธิภาพ [http://web.ku.ac.th/schoolnet, 2542]

2.1.1 แบบจำลองของ CTI

เพื่อเป็นมาตรฐานกลาง มาตรฐานของ CTI จึงเป็นมาตรฐานที่บริษัทชั้นนำต่างๆ พัฒนาขึ้น เพื่อว่าระบบเครือข่าย CTI จะเป็นเครือข่ายที่เชื่อมโยงถึงกันได้ทั่วโลก แบบจำลองอ้างอิงที่ใช้แสดงดังภาพประกอบที่ 2.1 (ก)



(ก) (ข) (ค)

ภาพประกอบที่ 2.1 แบบจำลองของการเชื่อมต่อคอมพิวเตอร์กับเครือข่ายโทรศัพท์ [http://web.ku.ac.th/schoolnet, 2542]

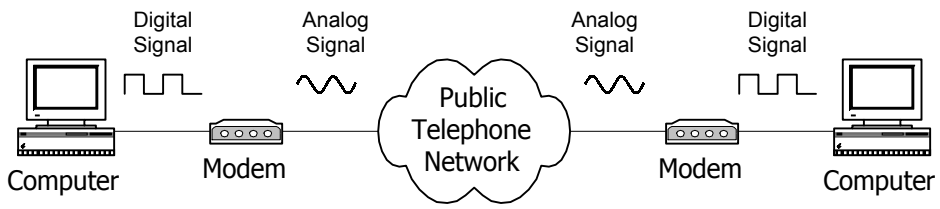
แบบจำลองอ้างอิงนี้แบ่งออกเป็นสามส่วนคือ ที่ปลายทางด้านหนึ่งมีเครื่องไมโครคอมพิวเตอร์เป็นเทอร์มินัล หรือทำหน้าที่เป็นสถานีปลายทางสำหรับรับส่งเสียง (Voice terminal) อีกปลายด้านหนึ่งเป็นเครือข่ายโทรศัพท์หรือสวิตชิง โดยมีส่วนเชื่อมโยงทำหน้าที่เชื่อมโยงเครื่องไมโครคอมพิวเตอร์กับเครือข่ายโทรศัพท์เข้าด้วยกัน เครื่องไมโครคอมพิวเตอร์ทำงานในฟังก์ชันที่เป็นเทอร์มินัลของเสียงประกอบด้วยอินพุตเสียงและเอาต์พุตเสียง อินพุตรับสัญญาณเสียงจากไมโครโฟน และเอาต์พุตเป็นลำโพง ส่วนการเชื่อมต่อนี้มีการเชื่อมโยงได้หลายแบบ เช่น ถ้าเป็นเครื่องไมโครคอมพิวเตอร์ธรรมดาและเชื่อมเข้าเครือข่ายโทรศัพท์ทั่วไป ก็เชื่อมต่อผ่านโมเด็ม และถ้าเครื่องไมโครคอมพิวเตอร์ต่ออยู่กับเครือข่าย LAN (Local Area Network) ส่วนเชื่อมต่อก็เป็น LAN และสามารถเชื่อมต่อเข้ากับ PBX (Private Branch Exchange) คือชุมสายโทรศัพท์ย่อยขององค์กร แสดงดังภาพประกอบที่ 2.1 (ข) และ (ค)

2.1.2 โมเด็ม

2.1.2.1 หลักการทำงาน

โมเด็ม (Modem) มาจากรากศัพท์ของคำว่า Modulator / Demodulator ซึ่งหมายถึงอุปกรณ์ที่ทำหน้าที่แปลงสัญญาณดิจิทัล (Digital) ให้เป็นสัญญาณแอนะล็อก (Analog) และสัญญาณแอนะล็อกให้เป็นสัญญาณดิจิทัล [Stallings, 2000:787,86,5]

เมื่อมีการติดต่อกันระหว่างเครื่องคอมพิวเตอร์ที่อยู่ในระยะไกล จำเป็นต้องใช้โมเด็มทำการแปลงสัญญาณดิจิทัลจากคอมพิวเตอร์ให้เป็นสัญญาณแอนะล็อก เรียกว่าการมอดูเลต (Modulation) โดยการผสมสัญญาณของข้อมูลดิจิทัลเข้าไปกับสัญญาณอีกสัญญาณหนึ่งที่เป็นคลื่นพาหะ (Carrier) ซึ่งสัญญาณนี้มีความถี่ที่เหมาะกับช่องสัญญาณนั้นๆ เพื่อให้ข้อมูลที่ส่งเดินทางได้ไกลมากขึ้น ทำให้ส่งข้อมูลผ่านทางสายโทรศัพท์ได้ และเมื่อถึงด้านรับก็จะแปลงสัญญาณแอนะล็อกที่ได้รับให้กลับมาเป็นสัญญาณดิจิทัลเหมือนเดิม เรียกว่าการดีมอดูเลต (Demodulation) ซึ่งเป็นการแยกเอาสัญญาณข้อมูลออกมานั่นเอง แสดงดังภาพประกอบที่ 2.2



ภาพประกอบที่ 2.2 การเชื่อมต่อโมเด็ม และการแปลงสัญญาณ

2.1.2.2 **โมเด็มเสียง** (Data/Fax/Voice Modem **หรือ** Voice Modem)

โมเด็มทั่วไปที่ใช้งานกันอยู่นั้นเรียกว่า Data/Fax คือ ส่งได้เฉพาะข้อมูล (Data) และแฟกซ์ (Facsimile หรือ Fax) แต่โมเด็มเสียงนอกจากจะส่งข้อมูลหรือแฟกซ์ได้แล้ว ยังเพิ่มคุณสมบัติการประมวลผลเสียงซึ่งเป็นสัญญาณแอนะล็อกได้ด้วย คือสามารถเล่นเสียงและบันทึกเสียงผ่านคู่สายโทรศัพท์ได้ [http://www.ruencom.com/hardware/peri/modemv.htm, 2542]

หลักการทำงาน

โดยหลักการแล้วโมเด็มเสียงจะเพิ่มส่วนของตัวแปลงสัญญาณเสียงพูด ซึ่งเป็นสัญญาณแอนะล็อกให้เป็นดิจิทัลที่ประมวลผลได้ ทำให้ต้องมีภาคแปลงสัญญาณแอนะล็อกเป็นดิจิทัล (Analog to Digital Converter หรือ ADC) ด้วย เช่น แปลงเสียงพูดที่ได้รับจากหูฟัง/ไมโครโฟน หรือแปลงเสียงที่ได้รับจากผู้ที่ติดต่อเข้ามา ซึ่งเป็นเสียงแบบแอนะล็อกให้เป็นดิจิทัลเพื่อเก็บไว้ในรูปของแฟ้มข้อมูลเสียงในคอมพิวเตอร์ และแฟ้มข้อมูลเสียงจะถูกแปลงเป็นแอนะล็อกเมื่อมีการเล่นเสียง (Playback)

การนำมาใช้งาน

การนำโมเด็มเสียงมาใช้งานคือ สามารถนำมาทำเป็นระบบตอบรับโทรศัพท์อัตโนมัติ โดยเมื่อมีโทรศัพท์เข้ามา แต่ไม่มีผู้รับสาย ก็จะใช้โมเด็มเสียงส่งเสียงตอบอัตโนมัติ ผู้ที่โทรเข้ามาก็อาจจะให้ฝากข้อความไว้ได้ นอกจากนี้ยังใช้งานส่งแฟกซ์กลับได้อีกด้วย แต่เฉพาะโมเด็มเสียงนั้นทำงานด้วยตัวเองไม่ได้ ต้องมีซอฟต์แวร์ควบคุมการทำงานอีกทีหนึ่ง ซึ่งจะทำงานร่วมกับ TAPI

2.1.3 Call Control **และ** Media Processing

หลักการทำงานพื้นฐานสองประการ ที่ระบบคอมพิวเตอร์จะใช้ทำงานร่วมกับเครือข่ายโทรศัพท์ได้ ประกอบด้วยฟังก์ชันการทำงานที่สำคัญ คือ Call Control และ Media Processing [Strathmeyer, 1996]

2.1.3.1 **Call Control** เป็นฟังก์ชันการเรียกเข้าเครือข่ายเพื่อการเชื่อมโยง ทำการควบคุมการติดต่อ และการส่งสัญญาณระหว่างกัน

2.1.3.2 **Media Processing** เป็นเรื่องของการส่งและรับข้อมูลข่าวสารที่วิ่งผ่านไปยังเครือข่าย ในรูปแบบข้อมูลข่าวสารต่างๆ เช่น แฟกซ์, เสียงพูด (Voice), เสียงโทน (Tones) หรือ ข้อมูล (Data)

โดยปกติแล้วงานประยุกต์ทางโทรศัพท์คอมพิวเตอร์ (Computer Telephony Application) จะต้องมีการทำงานทั้งสองอย่างร่วมกัน ซึ่งการทำงานของ Call Control และ Media Processing นี้เลียนแบบมาจากการใช้โทรศัพท์ของมนุษย์นั่นเอง ดังนี้

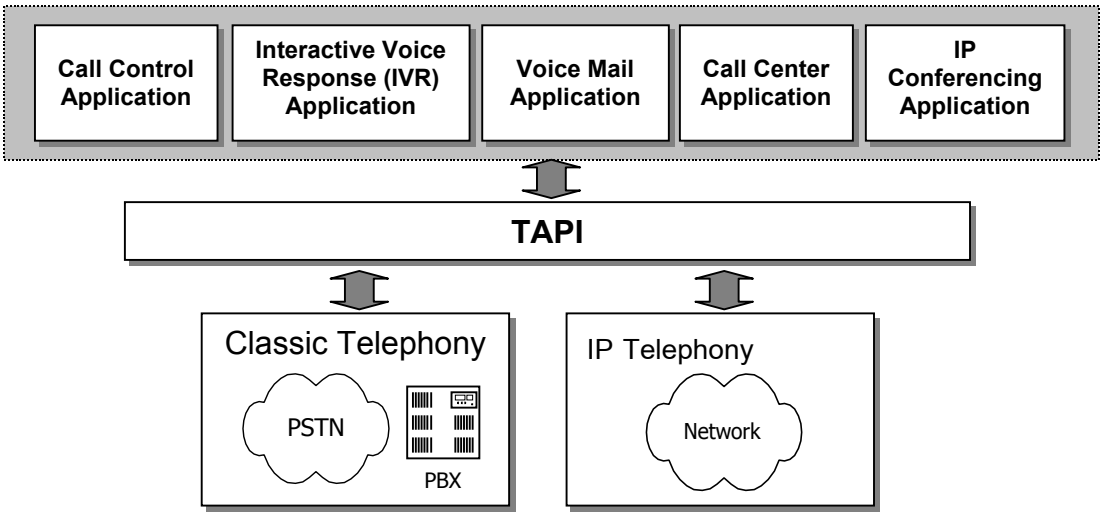
- การยกหูโทรศัพท์ กดหมายเลข แล้วฟังสัญญาณเสียงโทนในการติดต่อ ว่าสำเร็จหรือไม่ จะเปรียบได้กับการทำงานในส่วนของ Call Control
- เมื่อทำการต่อได้สำเร็จแล้ว การพูด และการฟัง กับคู่ร่วมสนทนา จะเปรียบได้กับการทำงานในส่วนของ Media Processing

2.1.4 API (Application Programming Interface)

API เป็นกลุ่มของการทำงานมาตรฐานต่างๆ ที่สามารถถูกเรียกใช้ในงานประยุกต์ต่างๆได้ [Microsoft Corporation, 1997] ซึ่ง API มีความจำเป็นสำหรับการทำงานของ Call Control และ Media Processing โดยผู้พัฒนางานบนระบบวินโดวส์ได้พัฒนา API ที่ทำงานทางด้านโทรศัพท์ขึ้นโดยเฉพาะ เรียกว่า TAPI (Telephony Application Programming Interface) เพื่อรองรับส่วนเชื่อมต่อนี้ ส่วนในกลุ่มของ API ที่ไม่ใช่ทางด้านโทรศัพท์ (Non-Telephony API) พบว่ามีหน้าที่การทำงานหลายอย่าง ที่มีประโยชน์ต่อการใช้งานทาง Computer Telephony โดยเฉพาะการใช้สำหรับควบคุมการทำงานของ Media Processing เช่น การเล่นไฟล์เสียงผ่านลำโพงหลังจากที่มีการสร้างการติดต่อทางโทรศัพท์ขึ้นแล้ว ซึ่งสามารถนำมาใช้ส่งและรับรูปแบบสื่อประสม (Multimedia) ที่เหมือนกัน ผ่านไปบนการติดต่อทางโทรศัพท์ [Strathmeyer, 1996]

2.2 ความรู้เบื้องต้นเกี่ยวกับ TAPI

TAPI เป็นพัฒนาการของ API เพื่อรองรับการการทำงานเชื่อมต่อกันระหว่างเครื่องคอมพิวเตอร์และระบบโทรศัพท์ ซึ่งเป็นมาตรฐานที่พัฒนาขึ้นโดยบริษัทไมโครซอฟต์ [Microsoft Corporation, 1999 (a), 2000] ในปัจจุบัน TAPI เวอร์ชัน 3 ได้สนับสนุนการทำงานทั้งในรูปแบบของเครือข่ายโทรศัพท์ PSTN (Public Switched Telephone Network) และเครือข่ายโทรศัพท์ไอพี (IP telephony) โดย TAPI 3.0 ทำงานอยู่บนระบบปฏิบัติการ Microsoft Windows 2000 ภาพประกอบที่ 2.3 แสดงงานประยุกต์ทางโทรศัพท์บนระบบเครือข่ายโทรศัพท์ PSTN และ IP Telephony โดยมี TAPI เป็นส่วนเชื่อมต่อการทำงาน



ภาพประกอบที่ 2.3 TAPI รองรับการทำงานของงานประยุกต์ ทั้งในระบบเครือข่ายโทรศัพท์ PSTN และ IP Telephony

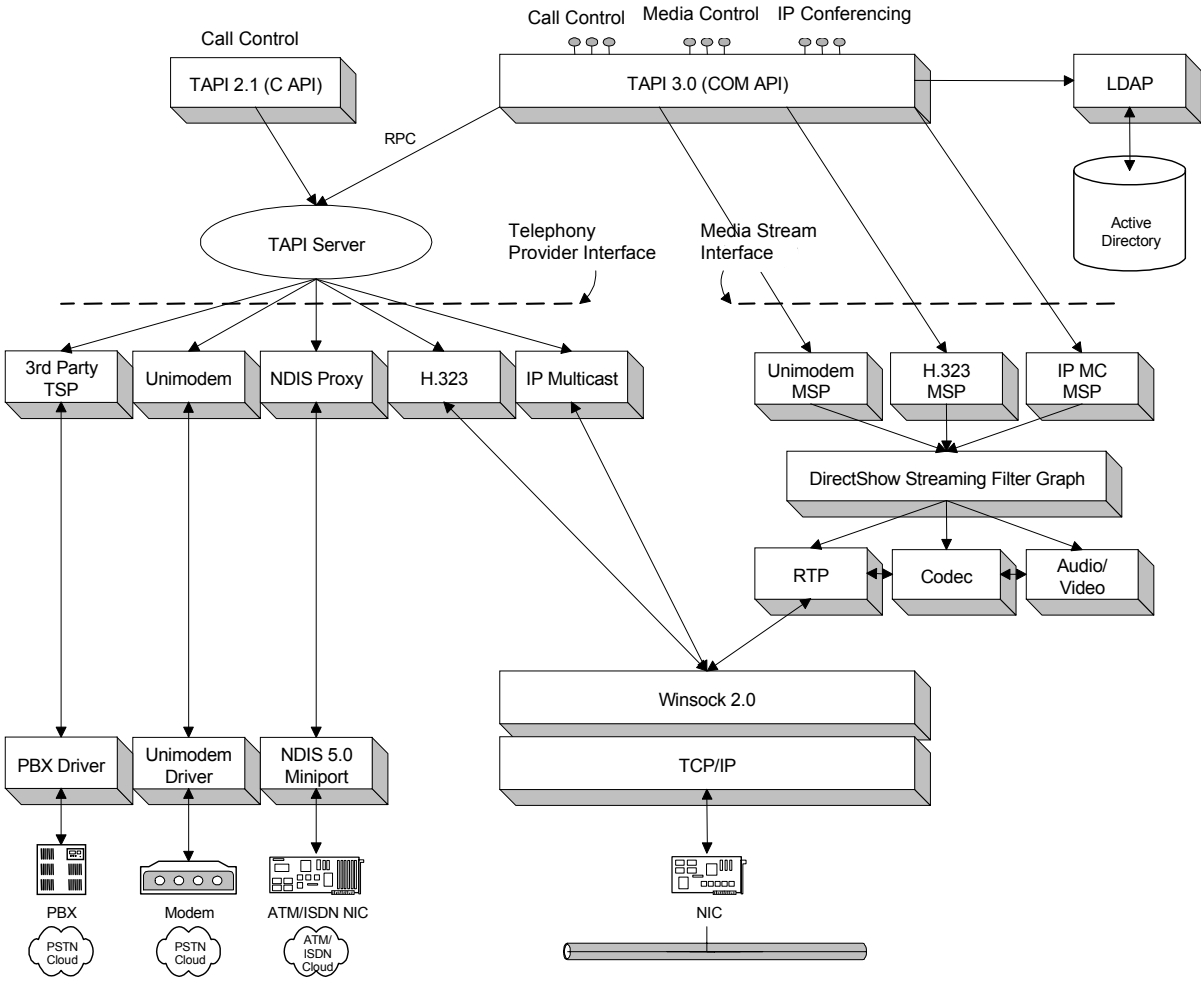
2.2.1 **องค์ประกอบภายใน** TAPI 3.0

TAPI 3.0 ได้รวมส่วนของการทำงานพื้นฐานทางโทรศัพท์ และส่วนการควบคุมกระแสแบบสื่อประสม (Multimedia Stream Control) เอาไว้ด้วยกัน โดย TAPI 3.0 พัฒนามาจาก TAPI 2.1 ซึ่งเป็น API ที่ต้องเขียนโปรแกรมด้วยภาษา C ไปเป็นรูปแบบของคอมโพเนนต์ออบเจ็กต์ (Component Object Model หรือ COM) ที่ให้นักพัฒนางานประยุกต์บน TAPI สามารถเขียนโปรแกรมได้หลายภาษาเช่น C/C++ หรือ Microsoft Visual Basic

นอกจากจะสนับสนุนรูปแบบโทรศัพท์พื้นฐาน (Classic Telephony Providers) แล้ว TAPI 3.0 ยังสนับสนุน Standard H.323 Conferencing และ IP Multicast Conferencing. TAPI 3.0 ใช้ Windows 2000 Active Directory Service เพื่อให้มีการจัดการใช้งานภายในองค์กรที่ง่ายขึ้น และยังสนับสนุนคุณสมบัติของ QoS (Quality-of-Service) ที่เพิ่มคุณภาพของ Conference และความสามารถในการจัดการกับเครือข่าย

สถาปัตยกรรมของ TAPI ดังในภาพประกอบที่ 2.4 แสดงถึงองค์ประกอบของ TAPI 3.0 ซึ่งมีส่วนประกอบหลักอยู่ 4 ส่วน ดังนี้

- TAPI 3.0 COM API
- TAPI Server
- TSP (Telephony Service Provider)
- MSP (Media Stream Provider)



ภาพประกอบที่ 2.4 สถาปัตยกรรมของ TAPI

2.2.1.1 TAPI 3.0 API เป็นชุดของคอมโพเนนต์ออบเจ็กต์ (COM) การพัฒนามาเป็นรูปแบบเชิงวัตถุ (Object-Oriented) ในแบบ COM นี้ ทำให้นักพัฒนาสามารถเขียนโปรแกรมได้หลายภาษา

2.2.1.2 TAPI Server (TAPISRV.EXE) เป็นส่วนประกอบหลักของ TAPI ใช้เป็นส่วนกระบวนการที่ทำการให้บริการ (Service Process) สำหรับ TAPI 3.0 และ TAPI 2.1 โดยฟังก์ชันของ API จะติดต่อกับ TAPI Server เพื่อส่งบริการที่ร้องขอไปยัง Telephony Service Provider

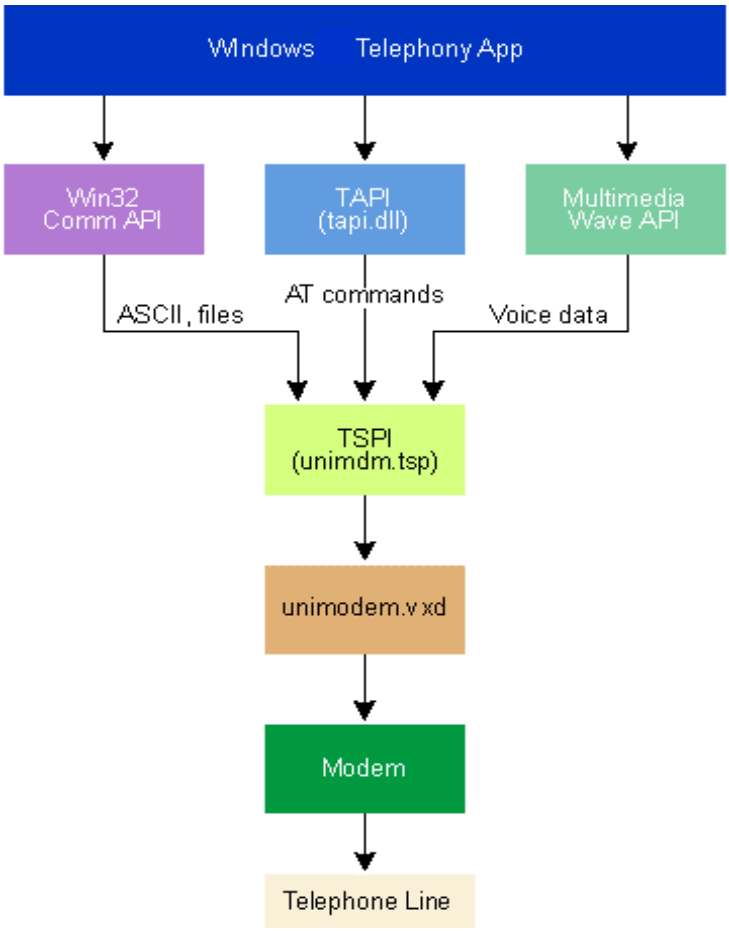
2.2.1.3 TSP เป็นส่วนไดรเวอร์ที่ติดต่อกับอุปกรณ์สื่อสาร โดยมีหน้าที่ในการแปลโปรโตคอลในแบบจำลองการโทร (Call Model) ของ TAPI ไปเป็น โปรโตคอลสำหรับการควบคุมการโทรเรียก (Call Control) โดยเฉพาะของอุปกรณ์ฮาร์ดแวร์ เช่น ตัวแปลงแฟกซ์ (Fax Adapter),

ตัวแปลง ISDN (ISDN Adapter), และ โมเด็ม เพื่อติดต่อควบคุมการทำงานกับอุปกรณ์เหล่านี้ให้สำเร็จ

- **Unimodem TSP** หรือ Universal Modem Driver TSP

เป็นไดรเวอร์ที่สามารถใช้ได้กับโมเด็มมาตรฐานเกือบทั้งหมด สำหรับเวอร์ชันบน Windows 95, Windows 98, และ Windows 2000 สนับสนุนโมเด็มเสียงในลักษณะกระแสแบบทิศทางเดียว (Half-Duplex Streaming) สำหรับ Unimodem บน Windows NT 4.0 หรือ เวอร์ชันที่ต่ำกว่านี้ ไม่สนับสนุนโมเด็มเสียง โดย Unimodem TSP ได้ถูกติดตั้งมาไว้แล้วกับ Windows NT, Windows 95, Windows 98, และ Windows 2000 [Microsoft Corporation, 2000]

Unimodem จัดเป็นส่วนประกอบหนึ่งของ TAPI โดยการทำงานของ TAPI และ Unimodem ทำให้สามารถควบคุมการทำงานของโมเด็มได้ แทนการใช้ชุดคำสั่ง AT สั่งงานโมเด็มโดยตรง ดังแสดงในภาพประกอบที่ 2.5 อธิบายการทำงานดังนี้



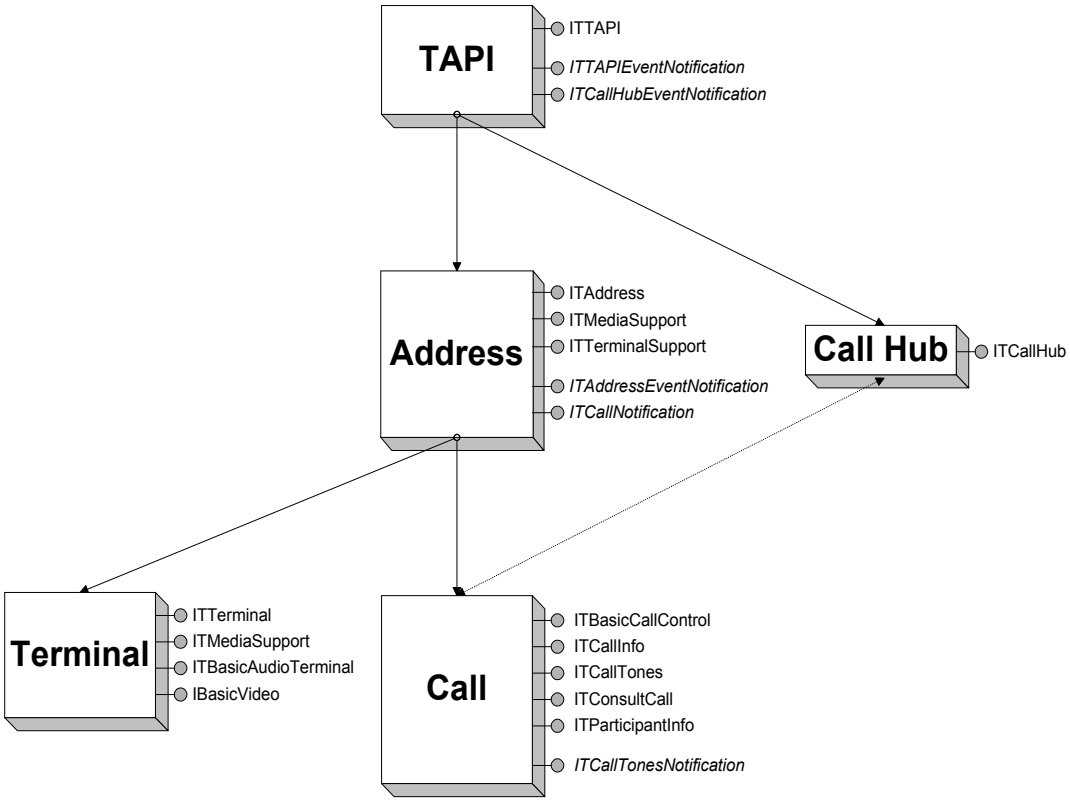
ภาพประกอบที่ 2.5 รูปแบบสถาปัตยกรรมอย่างง่ายของ Unimodem

TAPI มีหน้าที่จัดการเกี่ยวกับการโทรติดต่อกัน โดยจะทำการส่งชุดคำสั่ง AT หรือ AT+V ที่ใช้ควบคุมการทำงานของโมเด็ม (AT Commands สำหรับ Data Modem, AT+V Commands สำหรับ Voice Modem) ไปยังโมเด็ม โดย TAPI ไม่ได้มีหน้าที่ส่งข้อมูล เมื่อต้องการส่งข้อมูลผ่านสายโทรศัพท์ จะต้องใช้ API อื่น เช่น ใช้ Win32 comm API สำหรับการส่ง ASCII หรือ ข้อมูลแบบไบนารี หรือ ใช้ Multimedia Wave API สำหรับการส่งข้อมูลเสียง เป็นต้น

TAPI จะอยู่ในแฟ้มข้อมูลชื่อ Tapi.dll เมื่อมีการทำงานจะไปเรียก TSP API ซึ่งหน้าที่การทำงานต่างๆ ของโมเด็มจะเก็บอยู่ใน TSP และถูกใช้งานอยู่ในลักษณะของ DLL (Dynamic Link Library) ที่มีชื่อเรียกว่า unimdm.tsp โดย Unimodem จะทำหน้าที่แปลคำสั่งของ TAPI ไปเป็นชุดคำสั่ง AT หรือ AT+V แล้วส่งชุดคำสั่งนี้ไปยัง unimodem.vxd ซึ่งเป็นไดรเวอร์ที่จะติดต่อกับโมเด็มอย่างแท้จริง

2.2.1.4 MSP ทำหน้าที่สำหรับการใช้กระแสแบบสื่อประสม (Media Stream) ในการโทร โดยจะสนับสนุน API ชื่อ DirectShow เป็นหลักในการควบคุมกระแสแบบสื่อประสม

2.2.2 Call Control Model



ภาพประกอบที่ 2.6 ความสัมพันธ์ของออบเจ็กต์ใน TAPI 3.0

ออบเจ็กต์ (Object) หลักใน TAPI 3.0 มีอยู่ 5 ออบเจ็กต์ ดังแสดงในภาพประกอบที่ 2.6 คือ

- TAPI
- Address
- Terminal
- Call
- CallHub

อธิบายรายละเอียดของออบเจ็กต์ได้ดังนี้

2.2.2.1 TAPI Object เป็นจุดเริ่มต้นของการทำงานของงานประยุกต์สำหรับเข้าสู่ TAPI 3.0 ออบเจ็กต์นี้เป็นตัวแทนของทรัพยากรทางโทรศัพท์ทั้งหมดที่เครื่องคอมพิวเตอร์จะมีการใช้งานได้ โดยออบเจ็กต์นี้จะให้งานประยุกต์ระบุแอดเดรสของทรัพยากรที่ต้องการจะใช้งาน

2.2.2.2 Address Object เป็นตัวแทนของจุดตั้งต้นหรือปลายทางสำหรับการโทรเรียก โดยงานประยุกต์สามารถรอรับสายโทรเรียกบน Address Object นี้ หรืออาจจะสร้าง Call Object ที่ทำการโทรออกได้จาก Address Object นี้

2.2.2.3 Terminal Object เป็นตัวแทนของอุปกรณ์รับหรือส่งในจุดตั้งต้นหรือปลายทางสำหรับการติดต่อ ที่ใช้ในการโต้ตอบสื่อสารกันสำหรับมนุษย์ เช่น ไมโครโฟน และลำโพง แต่ก็สามารถที่จะเป็นแฟ้มข้อมูล (File) หรืออุปกรณ์อื่นๆ ที่สามารถทำการรับหรือส่งได้

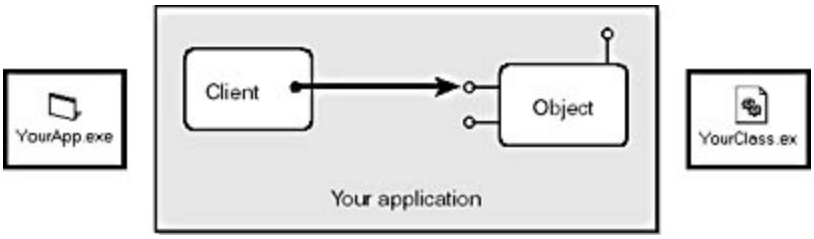
2.2.2.4 Call Object เป็นตัวแทนของการโทรติดต่อถึงกันของแอดเดรส ระหว่างแอดเดรสต่างๆ (การติดต่อกันนี้สามารถกระทำได้โดยตรง หรือผ่านทาง CallHub) โดย Call Control ทั้งหมดจะถูกกระทำผ่าน Call Object และจะมี Call Object สำหรับแต่ละสมาชิกใน CallHub ด้วย

2.2.2.5 CallHub Object เป็นตัวแทนของกลุ่มสำหรับการโทรติดต่อที่มีความเกี่ยวข้องกัน โดยงานประยุกต์จะไม่สามารถสร้าง CallHub Object ได้โดยตรง แต่มันจะถูกสร้างโดยทางอ้อม เมื่อสายโทรเข้าถูกรับผ่าน TAPI 3.0 ในการใช้ CallHub Object นี้ สามารถใช้ระบุผู้เข้าร่วมในการโทรติดต่อคนอื่นๆได้ หรือทำการประชุมสนทนาหลายคน

2.3 ความรู้เบื้องต้นเกี่ยวกับ COM (Component Object Model)

COM เป็นข้อกำหนดสำหรับการเขียนซอฟต์แวร์ในรูปแบบเชิงชิ้นส่วน ที่สามารถนำกลับมาใช้งานใหม่ได้ (Reusable Component) โดยจะอยู่ในรูปของไบนารี (Binary Form) ไม่ขึ้นอยู่กับโปรแกรมต้นฉบับ ทำให้ตัดปัญหาในการแปล (Compile) ออกไป และทำให้สามารถสร้างงานประยุกต์โดยใช้โปรแกรมภาษาได้หลายภาษา [Pattison, 1998]

COM อยู่บนพื้นฐานของการโปรแกรมเชิงวัตถุ (Object Oriented Programming หรือ OOP) ดังนั้น COM จึงเกี่ยวข้องกับไคลเอนต์ (Client) ที่มีการติดต่อกับออบเจ็กต์ แต่ COM เป็นระดับที่สูงกว่า OOP ในด้านการนำกลับมาใช้งานใหม่ ซึ่งไคลเอนต์ และ คลาส (Class) โดยปกติแล้วจะอาศัยอยู่แยกกันในคนละแฟ้มข้อมูลที่เป็นแบบไบนารี (Binary File) ดังแสดงในภาพประกอบที่ 2.7 โดย COM จะให้ไคลเอนต์ติดต่อกับออบเจ็กต์ที่ขณะเวลาทำงาน (Run Time)



ภาพประกอบที่ 2.7 ไคลเอนต์เรียกใช้คลาสที่อยู่แยกกันในคนละแฟ้มข้อมูลแบบไบนารี

COM ยังอยู่บนแนวคิดของการโปรแกรมแบบอินเตอร์เฟส (Interface-based Programming) ซึ่งจะมีลักษณะที่เป็นเหมือนคลาส ที่กำหนดกลุ่มของเมธอด (Method) ต่างๆ ไว้ โดยแต่ละอินเตอร์เฟสจะกำหนดโปรโตคอลสำหรับการติดต่อที่เฉพาะเจาะจงระหว่างไคลเอนต์และออบเจ็กต์

COM จะถูกใช้งานอยู่ในรูปแบบของแฟ้มข้อมูลที่เป็น .DLL หรือ .EXE