บทที่ 2
หลักการ ทฤษฎี

2.1 เว็บเซอร์วิส

2.1.1 หลักการของเว็บเซอร์วิส

เว็บเซอร์วิส คือ แอปพลิเคชัน หรือโปรแกรมซึ่งทำงานอย่างใดอย่างหนึ่งในลักษณะให้บริการ โดยจะถูกเรียกใช้งานจากแอปพลิเคชันหรือโปรแกรมอื่นๆผ่านเว็บ การให้บริการของเว็บเซอร์วิสจะมีเอกสารที่อธิบายคุณสมบัติของบริการกำกับไว้ และมีการนำเสนอให้สาธารณชนรับทราบ ผู้ใช้บริการจึงสามารถค้นหา เว็บเซอร์วิส ได้โดยที่ไม่จำเป็นต้องรู้ที่อยู่จริงของแอปพลิเคชันหรือโปรแกรมนั้น

แนวคิดของเว็บเซอร์วิสได้ทำลายกำแพงที่ขวางกั้นระหว่างแพลตฟอร์มลง โดยการเสนอโพรโตรคอลใหม่ที่ชื่อว่า SOAP (Simple Object Access Protocol) ด้วยโพรโตคอลใหม่นี้ เราสามารถเรียกใช้คอมโพเนนต์ใดๆ ก็ได้ ในแพลตฟอร์มใดๆ ก็ได้โดยโพรโตคอลนี้ทำงานอยู่บนโพรโตคอล HTTP ซึ่งเป็นโพรโตคอลสำหรับเว็บ ดังนั้นการเรียกใช้คอมโพเนนต์โดยอาศัยโพรโตคอล SOAP ก็คือการเรียกผ่านเว็บนั้นเอง

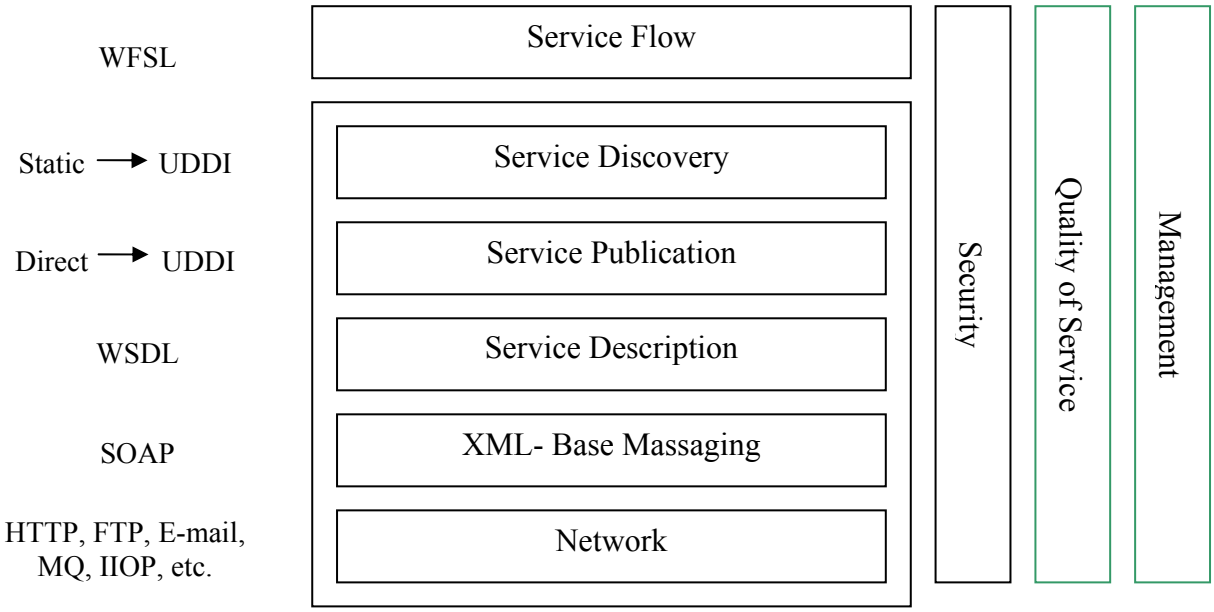
2.1.2 ส่วนประกอบของเว็บเซอร์วิส

- Service Provider ในมุมมองของระบบธุรกิจแล้วส่วนนี้จะเป็นส่วนของเจ้าของการบริการ แต่ถ้าในมุมมองของโครงสร้างแล้วจะหมายถึงส่วนที่รองรับระบบปฏิบัติการที่ Hosts สามารถเข้าถึงบริการได้
- Service Requestor ในมุมมองของระบบธุรกิจแล้วส่วนนี้หมายถึงธุรกิจที่มีความพึงประสงค์ที่จะเรียกใช้บริการ และในแง่ของโครงสร้างหมายถึง Application ที่ทำการค้นหา, เกี่ยวข้อง หรือ เป็นสมาชิก กับบริการ(service)
- Service Registry เป็นส่วนที่เก็บรายการบริการ และคุณลักษณะของบริการนั้นรวมถึงแจ้งกฎกติกาในการเรียกใช้บริการ (Blind) ไว้ สำหรับให้ ผู้ร้องขอ (Service Requestor) สามารถทำการค้นหา (Search) ได้ โดยฝ่ายผู้ให้บริการจะนำรายละเอียดคุณลักษณะของตน (Service Description)ไปเก็บไว้ที่ส่วนของ Service Registry นี้

2.1.3 โครงสร้างเว็บเซอร์วิส

ในการที่จะใช้ ฟังก์ชัน ทั้งสามได้แก่ Publish, Find, Blind นั้นWeb Service จะต้องประกอบไปด้วย Web Service Stack ที่เป็นไปตามมาตรฐานที่กำหนดไว้ในแต่ละชั้น **รูปที่ 1** แสดงถึง Conceptual ของ Web Service Stack Layer ด้านบนจะสร้างขึ้นตามข้อกำหนดเงื่อนไข

ของ Layer ล่าง สำหรับกล่องข้อความในแนวตั้งแสดงถึง Requirement ที่ต้องทำการเพิ่มเข้าไปในทุกๆ level ของ Stack ส่วนด้านซ้ายมือแสดงถึงมาตรฐานที่ใช้ใน Stack แต่ละชั้น



รูปที่2.1 Web Service Conceptual Stack

2.1.4 เทคโนโลยีพื้นฐานของเว็บเซอร์วิส

XML

XML (eXtensible Markup Language) คือ ภาษาของการทำเครื่องหมาย (Markup) อยู่ในรูปแบบ text ที่เป็นมาตรฐานไม่ขึ้นกับระบบ ใช้ในการแสดงและแลกเปลี่ยนข้อมูลบนอินเทอร์เน็ตในปัจจุบัน ผู้ที่ทำหน้าที่รับผิดชอบมาตรฐาน XML คือ World Wild Web Consortium (W3C)

SOAP

SOAP (Simple Object Access Protocol) คือ เป็นโพรโตคอลที่ทำงานได้กับโพรโตคอลเครือข่ายหลายโพรโตคอล เช่น HTTP , SMTP ฯลฯ แต่ในการใช้งานจริงมักพบเฉพาะโพรโตรคอล HTTP

SOAP มีความสามารถให้เราเรียกใช้คอมโพเนนต์หรือเว็บเซอร์วิสข้ามเครื่อง ข้ามแพลตฟอร์ม ข้ามภาษาได้อย่างสบาย โดยอาศัยโพรโตคอลที่มีอยู่เดิมในอินเทอร์เน็ตอย่าง HTTP และรูปแบบข้อความที่

สื่อสารกันด้วยภาษาXMLซึ่งมีลักษณะเป็นข้อความธรรมดาๆปิดล้อมด้วยแท็กทำให้เข้าใจได้ในทุกแพลตฟอร์ม

SOAP ยังมีความสามารถที่เหนือโพรโตคอลแบบเดิม เช่น DCOM , RMI หรือ IIOP โดยSOAP massage สามารถผ่านระบบไฟร์วอลล์ (Firewall) ป้องกันการบุกรุกทั้งนี้เพราะว่า SOAP ทำงานอยู่กับโพรโตคอล HTTP ซึ่งโดยธรรมชาติของไฟร์วอลล์จะเปิดให้การสื่อสารด้วยโพรโตคอล HTTP ผ่านได้อย่างสะดวก

SOAP ก็มีข้อเสีย คือ มีอัตราการรับส่งข้อมูลที่ต่ำ เพราะทำงานอยู่กับโพรโตคอล HTTP

WSDL

WSDL (Web Services Description Language) คือ ภาษาที่ใช้อธิบายคุณลักษณะของ เว็บเซอร์วิส และวิธีการติดต่อกับ เว็บเซอร์วิส โดยใช้ไวยากรณ์ของภาษา XML

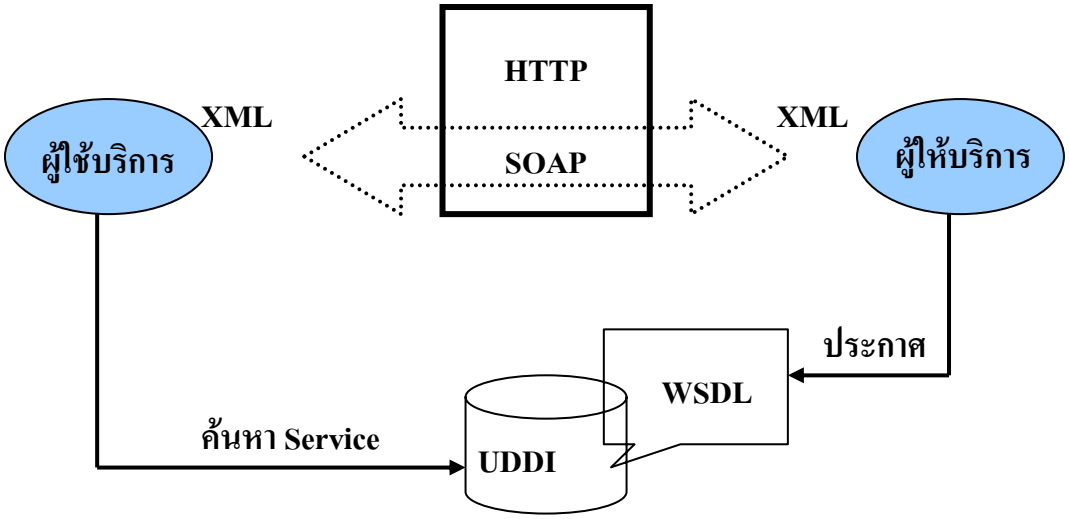
UDDI

UDDI (Universal Description, Discovery and Integration) คือ มาตรฐานในการค้นหาบริการเว็บเซอร์วิส โดย UDDI เปรียบเสมือนฐานข้อมูลขนาดใหญ่ซึ่งมีข้อมูลของเว็บเซอร์วิสที่ให้บริการปัจจุบัน บริษัทผู้ร่วมก่อตั้ง UDDI อย่าง IBM และ ไมโครซอฟท์จะมีเว็บเพจสำหรับทดลองการค้นหาบริการ เว็บเซอร์วิสโดยเว็บไซต์อย่างเป็นทางการของ UDDI ก็มีอยู่เช่นกันคือ http://uddi.org

ส่วน URL ของเว็บเพจสำหรับทดลองบริการค้นหา เว็บเซอร์วิสของ IBM และไมโครซอฟท์อยู่ที่

- http://www-3.ibm.com/services/uddi/testregistry/find
- http://uddi.microsoft.com

2.1.5 การทำงานของเว็บเซอร์วิส



รูปที่2.2 การทำงานของเว็บเซอร์วิส

ขั้นแรกผู้ให้บริการจะต้องประกาศเอกสาร WSDL ไว้ที่ UDDI เพื่อรอให้ผู้ใช้บริการมาค้นหา จากนั้นผู้ใช้บริการจะเข้าไปค้นหาบริการต่างๆที่ผู้ให้บริการทำการประกาศในUDDI ต่อจากนั้นผู้ใช้บริการจะติดต่อกับผู้ให้บริการจะติดต่อกันโดยอาศัยโปรโตคอล SOAP

2.2 ดอทเน็ตเทคโนโลยี

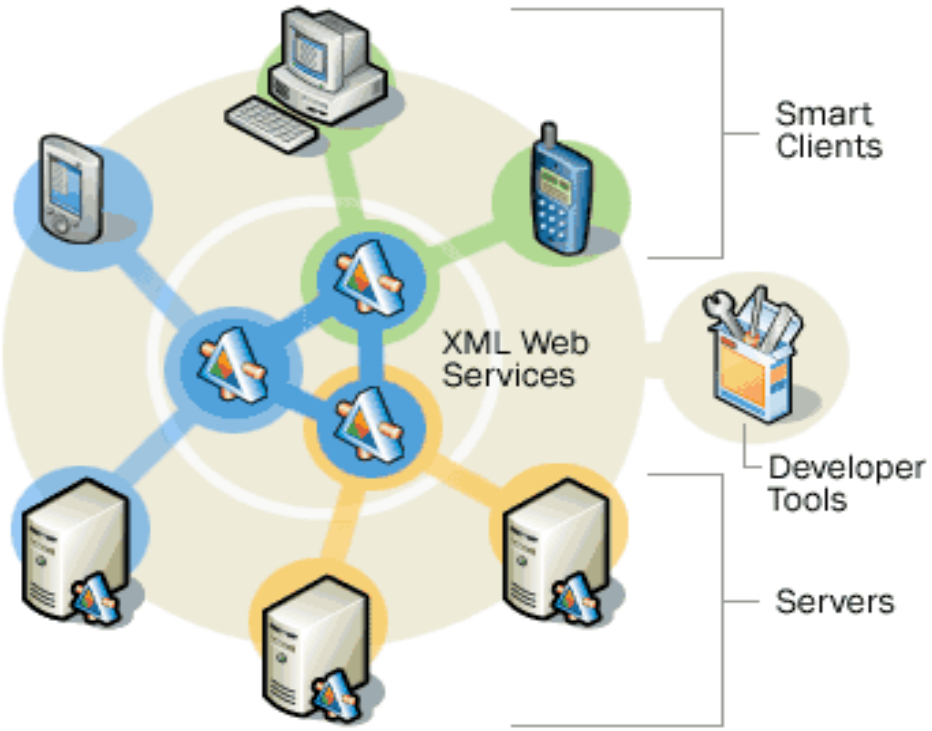
ดอทเน็ตคือแฟรตฟอร์มใหม่ที่ทางไมโครซอฟท์ออกแบบและพัฒนาขึ้นมาเพื่อให้รองรับการทำงานทางด้านอินเตอร์เน็ตเป็นหลักทำให้นักพัฒนาสามารถทำการพัฒนาเว็บแอปพลิเคชั่นที่มีประสิทธิภาพและความสามารถสูงอย่างที่ไม่เคยทำได้ โดยใช้ระยะเวลาในการพัฒนาอันรวดเร็วและง่ายแสนง่าย

ในปัจจุบันอุปกรณ์ไร้สายหลายๆประเภทต่างเริ่มที่จะทำการเชื่อมต่ออินเตอร์เน็ตได้บ้างแล้วซึ่งจะเรียกว่า smart device เช่น โทรศัพท์มือถือ และ พอกเก็ตพีซีแต่เนื่องจากเทคโนโลยีในปัจจุบันมีข้อจำกัดในการแลกเปลี่ยนข้อมูลระหว่างอุปกรณ์ค่อนข้างมาก และต้องเสียค่าใช้จ่ายในการจัดเตรียมและจัดการกับข้อมูลเหล่านั้นโดยไม่จำเป็น เช่น หากองค์กรมีการทำการค้าบนอินเตอร์เน็ตเพื่อการขายสินค้าและบริการผ่านทางเว็ปไซต์อยู่แล้วและมีความต้องการที่จะทำการขายสินคค้าและบริการดังกล่าวผ่านระบบอินเตอร์เน็ตของโทรศัพท์มือถือด้วยในปัจจุบันถือว่าเป็นเรื่องที่ทำได้ยากและต้องเสียเวลาตลอดจนใช้งบประมาณที่สูงมากเนื่องจากข้องมูลและองค์ประกอบของเว็ปไซต์เดิมที่ได้ทำการออกแบบและจัดเตรียม

ไว้นั้นไม่สามารถนำไปใช้กับเว็ปไซต์บนมือถือได้โดยทันทีดังนั้นองค์กรของคุณจึงจำเป็นต้องทำการพัฒนาและจัดเตรียมข้อมูลและองค์ประกอบของเว็ปไซต์ขึ้นมาใหม่สำหรับการใช้งานกับเว็ปไซต์บนมือถือโดยเฉพาะและหากมีการเปลี่ยนแปลงข้อมูลของรายการสินค้าหรือบริการใดๆก็ตามคุณก็ต้องทำการเปลี่ยนแปลงข้อมูลบนระบบข้อมูลของทั้งสองรูปแบบควบคู่กันจะเห็นได้ว่าเป็นเรื่องที่ยุ่งยากเสียเวลาเสียงบประมาณข้อมูลมีความซ้ำซ้อน และอาจนำมาซึ่งความไม่น่าเชื่อถือของข้อมูลได้

แต่ไมโครซอฟท์ดอทเน็ทเหตุการณ์ที่กล่าวข้างต้นจะไม่เกิดขึ้นเนื่องจากไมโครซอฟดอทเน็ทจะยอมให้มีการใช้ข้อมูลในทุกๆรูปแบบร่วมกันได้โดยง่าย(โดยหลักการทำงานนี้อยู่ภายใต้การทำงานของ XML Framework ซึ่งเป็นองค์ประกอบหนึ่งของดอทเน็ท)ข้อมูลที่ใช้ทั้งในระบบเว็ปไซต์ปกติและเว็ปไซต์บนมือถือสามารถเป็นข้อมูลในชุดดียวกันและมาจากแหล่งเดียวกันโดยประสิทธิผลที่ได้รับยังคงดีเช่นเดิมซึ่งการทำงานดังกล่างจะใช้ความสามารถของเว็ปเซอร์วิสที่ทำการพัฒนาขึ้นโดยใช้เทคโนโลยีของไมโครซอฟท์นั้นเอง แต่ต้องทำการปรับเปลี่ยนรูปแบบหน้าของเว็ปให้มีความเหมาะสมในการแสดงผลในแต่ละอุปกรณ์เท่านั้น ซึ่งถือได้ว่าเป็นเรื่องที่ง่ายดายและกินเวลาน้อยกว่ามาก

โดยจะเห็นได้ว่าหน้าที่หลักประการหนึ่งของ ดอทเน็ต เฟรมเวิร์คก็คือการทำให้ข้อมูลทุกๆ อย่างที่อยู่ในทุกๆที่ซึ่งส่วนใหญ่มีอยู่บนอินเตอร์เน็ทและทุกรูปแบบสามารถถูกเข้าถึงรวมทั้งทำการแลกเปลี่ยนและใช้งานร่วมกันได้จากทุกๆ Smart Device ดัง**รูปที่2.3** ซึ่งสิ่งเหล่านี้เกิดจากการอาศัยความสามารถของเทคโนโลยี XML และอยู่ภายใต้ความรับผิดชอบของ XML Framework ที่อยู่ใน ดอทเน็ทเฟลมเวร์ค

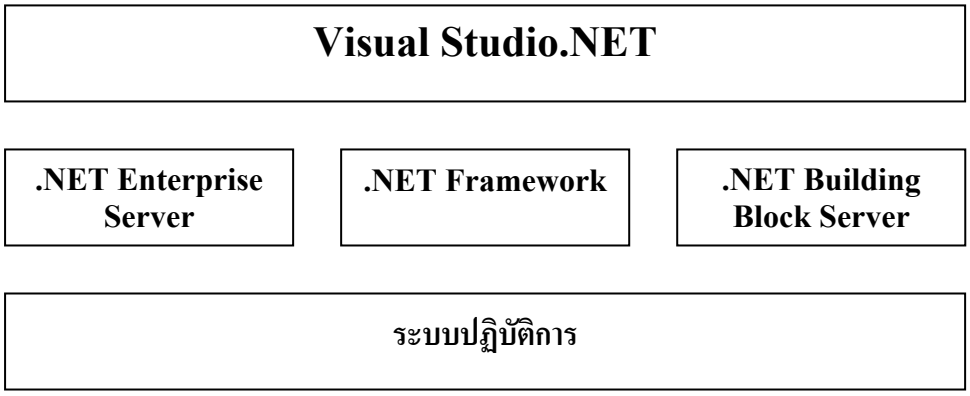


รูปที่2.3 แสดงการใช้งานร่วมกันได้จากทุกๆ Smart Device

2.2.1 แฟรตฟอร์มดอทเน็ต

สิ่งที่ไมโครซอฟท์ประกาศไว้ก็คือต้องการให้ดอทเน็ตเป็นแพลตฟอร์มในการพัฒนาซอฟต์แวร์สำหรับระบบปฏิบัติการวินโดว์รวมไปถึงการพัฒนาเว็บและเว็บเซอร์วิส

องค์ประกอบของแพลตฟอร์มดอทเน็ตมีลักษณะดังภาพ ซึ่งแบ่งออกได้ 3 ชั้นดัง**รูปที่ 4**



รูปที่ 2.4 องค์ประกอบของแพลตฟอร์มดอทเน็ต

ชั้นล่างสุด คือระบบปฏิบัติการ(Operating System)ในเครื่องเซิร์ฟเวอร์,เครื่องเดสก์ทอป หรืออุปกรณ์ใดๆ ที่โปรแกรมทำงาน

ชั้นที่2 แบ่งเป็นองค์ประกอบ 3 ส่วน คือ .NET Enterprise Server , .NET Framework และ .NET Building Block Services

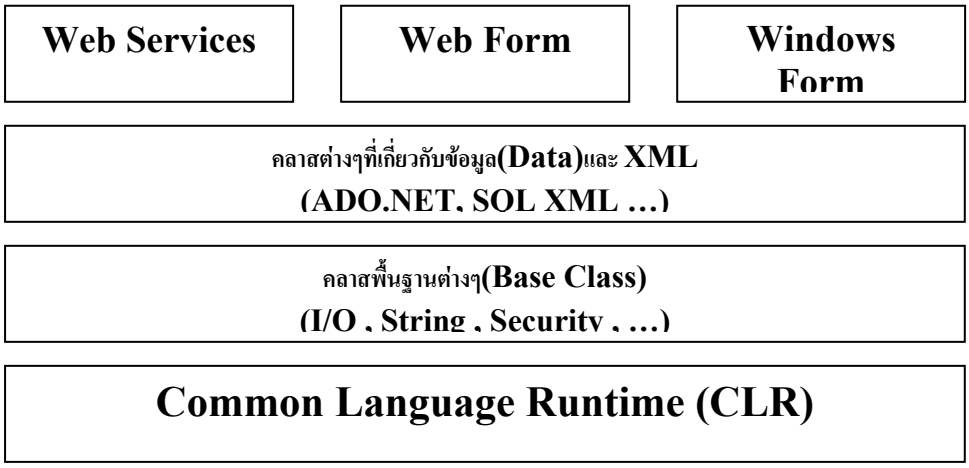
1 .NET Enterprise Server ที่อยู่ในชั้นที่สองก็คือผลิกภัณฑ์ต่างๆ เช่น ฐานข้อมูล SQL Server 2000 , BizTalk Server 2000 , Exchange 2000 เป็นต้น ผลิตภัณฑ์เหล่านี้ถูกออกแบบมาเพื่อรองรับการพัฒนาแอปพลิเคชันในระดับองค์กรไม่เฉพาะในยุค .NET เท่านั้น เพราะก่อนหน้านี้ก็มีผลิตภัณฑ์บางตัวออกมาให้ใช้ก่อนแล้วในแพลตฟอร์มวินโดว์DNA ผลิตภัณฑ์ชุดนี้นับว่าเป็นชุดต่อจาก Microsoft Back Officeโดยจุดหลักที่เปลี่ยนไปจากเดิมคือการรองรับข้อมูลในรูปแบบ XML นั่นเอง ตัวอย่างรายละเอียดของผลิตภัณฑ์ คือMicrosoft SQL Server 2000 คือ ฐานข้อมูลและเครื่องมือวิเคราะห์ที่สมบูรณ์แบบจัดการกับข้อมูลบนเว็บ ด้วยภาษา XML ทำให้มีความสามารถในการแลกเปลี่ยนข้อมูลระหว่างระบบที่มีความแตกต่างกันได้ สามารถสร้าง ใช้งาน และจัดการกับโซลูชันด้าน ระบบการค้าบนอินเตอร์เน็ตด้าน ธุรกิจดาต้าแวร์เฮาส์ได้ย่างรวดเร็ว

2 .NET Building Block Services ก็คือ บริการเว็บเซอร์วิสที่ไมโครซอฟต์สร้างขึ้นมาเพื่อให้นักพัฒนาใช้ในช่วงระยะแรกๆนี้มีบริการ2 อย่างที่มักได้ยินได้แก่ Microsoft Passport และ HailStorm โดยที่ลักษณะของบริการ Microsoft Passport ก็คือการที่ผู้ใช้กรอกข้อมูลบางอย่างเช่นยูสเซอร์เนม และพาสเวิร์ด เพียงครั้งเดียว ก็สามารถเข้านอกออกในเว็บไซต์ต่างๆ ที่ใช้ระบบ Microsoft Passport ได้อย่างรวดเร็วโดยไม่ต้องเสียเวลาใส่ยูสเซอร์เนม และพาสเวิร์ดหลายครั้ง ในขณะที่ HailStorm เป็นอีกบริการหนึ่งที่ช่วยอำนวยความสะดวกในการท่องเว็บไซต์ให้กับผู้ใช้ในลักษณะที่เรียกว่า personalization คือมีการเก็บข้อมูลต่างๆ ของผู้ใช้แต่ละคนไว้ รวมถึงสิทธิในการ access ข้อมูลต่างๆในอินเทอร์เน็ตด้วย

3 .NET Framework เป็นโครงสร้างที่ควบคุมการทำงานทั้งหมดที่ใช้ในการพัฒนาSoftware

ชั้นสุดท้าย คือ Visual Studio .NET ซึ่งเป็นเครื่องมือที่ช่วยในการพัฒนาโปรแกรมหรือแอปพลิเคชันให้เป็นไปอย่างง่ายดาย

.NET Framework แบ่งออกเป็นระดับต่างๆ ตามลักษณะของการพัฒนาซอฟแวร์ คือตั้งแต่ระดับระบบปฏิบัติการ (Operation System) , ระดับคอมโพเนนต์ที่ช่วยในการพัฒนาซอฟต์แวร์(เรียกว่าคลาสพื้นฐาน (Base Classes)) ไปจนถึงระดับการแสดงผล อย่าง เว็บเซอร์วิส, เว็บฟอร์ม และวินโดว์ฟอร์มนอกจากนี้ .NET Framework ยังต้องอาศัยเทคโนโลยีอย่างภาษา XML และโพรโตคอล SOAP ในการติดต่อ และ เรียกใช้งานโปรแกรมอื่นๆผ่านระบบอินเตอร์เน็ตด้วย



รูปที่ 2.5 องค์ประกอบของ.NET Framework

ชั้นแรก คือส่วนที่เป็นแอปพลิเคชันในแพลตฟอร์มดอทเน็ตจะเห็นได้ว่า มี 3 แบบ คือ เว็บเซอร์วิส, เว็บฟอร์ม , วินโดว์ฟอร์ม

- เว็บเซอร์วิสก็คือการสร้างคอมโพเนนต์หรือโปรแกรมเพื่อให้บริการผ่านโพรโตคอล SOAP/HTTP นั่นเอง
- เว็บฟอร์ม ก็คือการพัฒนาเว็บแบบใหม่ โดยเราสามารถสร้างยูสเซอร์อินเทอร์เฟซออกมาได้อย่างง่ายๆ เพียงแค่ลากเมาส์ในลักษณะ drag & drop เหมือนกับการพัฒนาโปรแกรม Visual Basic ยังไงยังงั้น ทั้งนี้เนื่องจากไมโครซอฟต์พยายามทำให้การพัฒนาเว็บง่ายขึ้น
- วินโดว์ฟอร์ม เป็นการสร้างโปรแกรมที่ทำงานในเครื่องพีซี คล้ายๆ กับฟอร์ม Visual Basic แบบเดิม

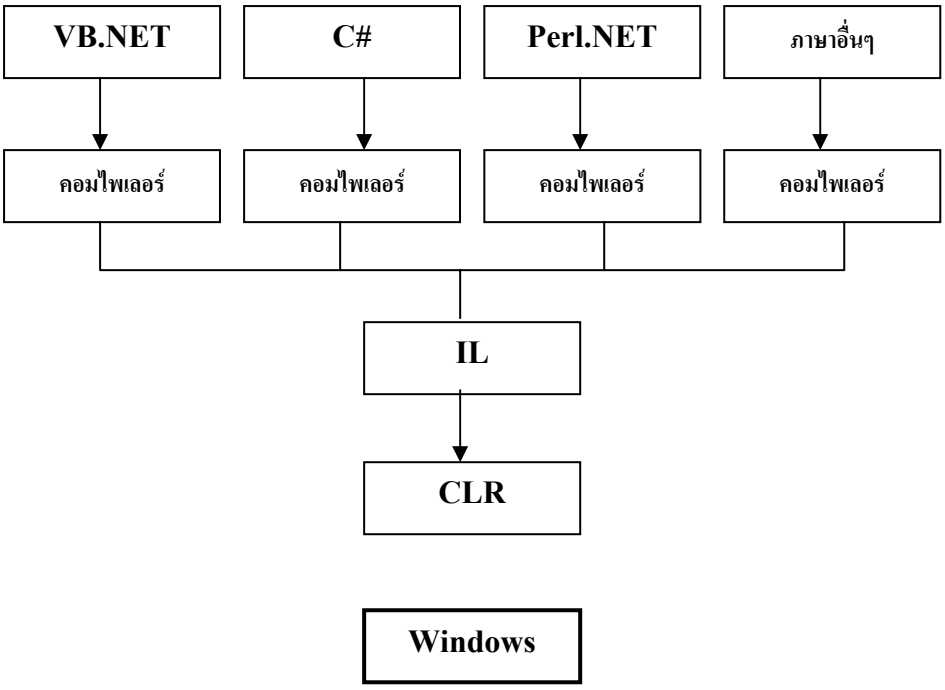
สรุป เว็บฟอร์มเป็นการสร้างเว็บเพจส่วนวินโดว์ฟอร์มคือการสร้างโปรแกรมสำหรับทำงานในเครื่องพีซีนั่นเอง

ชั้นที่2 คือ ชั้นข้อมูล และ XML เป็นกลุ่มคลาสที่ใช้ในการจัดกาและเรียกใช้ข้อมูลต่างๆ เช่นจากฐานข้อมูลต่างๆเช่นจากฐานข้อมูล(Database)หรือข้อมูลในรูปXMLตัวอย่างเช่นคลาสSQL,ADO.NET, XML ฯลฯ

ชั้นที่3 คือ ชั้น คลาสพื้นฐานเป็นกลุ่มของคลาสที่ใช้ในงานทั่วๆไป เช่นคลาสสตริง, I/O, security เป็นต้น หากจะเปรียบเทียบง่ายๆ เพื่อให้เห็นภาพชัดเจน ก็คล้ายๆ กับคลาสในภาษาจาวา นั่นเอง

ชั้นที่4 คือ ชั้น CLR(Common Language Runtime) จัดเป็นสิ่งที่สำคัญมาในการพัฒนาโปรแกรมสำหรับแพลตฟอร์ม .NET หน้าที่ของ CLR คือเป็นสภาวะแวดล้อมแบบรันไทม์(runtime) ในการจัดการโค้ดที่คอมไพล์แล้ว ไม่ว่าจะพัฒนาด้วยภาษาอะไรมาก็ตาม

การพัฒนาโปรแกรมในแพลตฟอร์มดอทเน็ตจะเลือกใช้ภาษาอะไรก็ได้เช่นกันแต่ภาษานั้นจะต้องมีคอมไพเลอร์เฉพาะที่จะคอมไพล์โปรแกรมจากแต่ละภาษามาอยู่ในโปรแกรมอีกรูปหนึ่งที่เรียกว่า IL (Intermediate Language) ซึ่งเป็นรูปแบบเฉพาะที่ทำงานใน CLR บนแพลตฟอร์มวินโดว์นอกจากนี้ทางไมโครซอฟต์ได้มีแผนงานในอนาคตว่า จะพยายามพัฒนาให้ CLR สามารถใช้งานในแพลตฟอร์มอื่นๆได้



รูปที่ 2.6 แสดงการพัฒนาโปรแกรมในแพลตฟอร์มดอทเน็ตในภาษาต่างๆ