

บทที่ 2

เอกสารและงานวิจัยที่เกี่ยวข้อง

การวิจัยครั้งนี้ผู้วิจัยได้ศึกษาเอกสารงานวิจัยที่เกี่ยวข้อง และค้นคว้าจากแหล่งข้อมูลบนระบบเครือข่ายอินเทอร์เน็ต โดยมีรายละเอียดที่จะเสนอตามลำดับ ดังนี้

1. แนวคิดเกี่ยวกับทฤษฎีการออกแบบและพัฒนาระบบ
 - 1.1 การออกแบบและพัฒนาระบบสารสนเทศ
 - 1.2 การออกแบบและพัฒนาระบบฐานข้อมูลบนอินเทอร์เน็ต
2. แนวคิดเกี่ยวกับทฤษฎี.NET
3. แนวคิดเกี่ยวกับระบบจัดการฐานข้อมูล
4. หลักการเว็บเบสเทคโนโลยี (Web-based Technology)
5. โครงสร้างการบริหารงานของโรงเรียนจุฬารัตน์ราชวิทยาลัย พิษณุโลก
6. งานวิจัยที่เกี่ยวข้อง

แนวคิดเกี่ยวกับทฤษฎีการออกแบบและพัฒนาระบบ

1. การออกแบบและพัฒนาระบบสารสนเทศ

วงจรการพัฒนาระบบ (System Development Life Cycle: SDLC) วงจรการพัฒนาระบบ (System Development Life Cycle : SDLC) เป็นกระบวนการในการพัฒนาระบบงานที่องค์กรต่างนำกระบวนการดังกล่าวมาใช้ ซึ่งวงจรการพัฒนาระบบประกอบด้วยขั้นตอนต่าง ๆ หลายขั้นตอน ในการดำเนินงานในแต่ละขั้นตอนนี้ กรณีที่ประสบปัญหาหรือข้อผิดพลาดในขั้นตอนใดก็สามารถกลับไปแก้ไขในขั้นตอนที่ผ่านมาได้ ขั้นตอนต่าง ๆ ของวงจรการพัฒนาระบบประกอบด้วยขั้นตอนต่าง ๆ ดังนี้

1.1 การพิจารณาการทำงานจากระบบ เป็นขั้นตอนแรกของการพัฒนาระบบ โดยมองถึงความต้องการในการปรับปรุงวิธีการ หรือการดำเนินงานใด ๆ เพื่อให้ระบบมีความสามารถในการดำเนินงานต่อไปได้อย่างมีประสิทธิภาพ ซึ่งอาจจะสร้างระบบใหม่หรือปรับปรุงระบบเดิม

1.2 การศึกษาปัญหาและการกำหนดทรัพยากร ที่จะต้องมีส่วนในการแก้ไขปัญหาหรือพัฒนาระบบเช่น เทคนิค บุคลากร เวลา และค่าใช้จ่าย ว่าเมื่อดำเนินการพัฒนาระบบแล้วจะคุ้มค่ากับการแก้ไขปัญหาหรือไม่ ทั้งนี้ ถ้าพิจารณาแล้วมีความคุ้มค่าและเป็นไปได้ก็จะดำเนินงานในขั้นตอนต่อไป กรณีที่พิจารณาแล้วไม่คุ้มค่าหรือไม่สามารถดำเนินการได้ก็จะยุติการดำเนินการ

1.3 การศึกษาการทำงานของระบบและกิจกรรมต่าง ๆ เพื่อให้ทราบขั้นตอนการทำงานของระบบเดิม หลังจากนั้นจะทำการวิเคราะห์และกำหนดความต้องการของระบบใหม่

1.4 การออกแบบวิธีการและรายละเอียดต่าง ๆ ซึ่งจะต้องกำหนดลักษณะทั่วไปที่จะนำมาใช้ในระบบใหม่ โดยต้องสอดคล้องกับความต้องการของผู้ใช้ ทั้งนี้จะไม่คำนึงถึงเครื่องคอมพิวเตอร์และรูปแบบของการประมวลผลข้อมูล

1.5 การนำวิธีการและรายละเอียดต่าง ๆ จากขั้นตอนข้างต้น มาทำการออกแบบระบบโดยมองถึงความสามารถทางด้านเทคนิค และวิธีการดำเนินการระบบ เช่น การออกแบบการนำเข้า การประมวลผลและแสดงผล รวมทั้งการออกแบบฐานข้อมูล และเครือข่ายสื่อสาร

1.6 การนำสิ่งที่ได้ออกแบบไว้ มาดำเนินการเขียนโปรแกรม ทดสอบโปรแกรม และติดตั้งระบบให้กับผู้ใช้ ซึ่งในขั้นตอนนี้จะรวมถึงการแนะนำ วิธีการใช้งาน เอกสาร คู่มือการใช้งาน และการฝึกอบรมให้กับผู้ใช้งาน เพื่อให้ผู้ใช้ระบบสามารถใช้งานได้มีประสิทธิภาพ

1.7 การใช้งานระบบงานใหม่อาจจะเกิดปัญหาในการทำงาน ในบางขั้นตอนที่ต้องปรับปรุงแก้ไขดังนั้นจึงต้องมีการบำรุงรักษาหรือแก้ไขระบบงานให้ทำงานได้ดีขึ้น แต่ในบางครั้งการปรับปรุงแก้ไขอาจไม่ใช่วิธีที่ดีที่สุด การเริ่มต้นวงจรการพัฒนาจะเป็นทางเลือกที่จะใช้ในการแก้ไข้ปัญหา

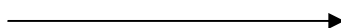
แผนภาพกระแสข้อมูล

แผนภาพกระแสข้อมูล (DFD) เป็นเครื่องมือที่ใช้กันอย่างแพร่หลายในการเขียนแบบระบบใหม่โดยเฉพาะกับระบบที่ หน้าที่ของระบบมีความสำคัญและมีความสลับซับซ้อนมาก โดยส่วนประกอบที่สำคัญของ แผนภาพกระแสข้อมูลมีดังนี้

1) การประมวลผล มีลักษณะเป็นวงกลมและ มีชื่องานที่จะต้องทำอยู่ภายในวงกลมจะมีข้อมูลไหลเข้าและออก การประมวลผลจะเปลี่ยนข้อมูลขาเข้าให้เป็นผลลัพธ์ จึงต้องทราบว่าข้อมูลขาเข้าเป็นอะไร ผลลัพธ์ที่ต้องการเป็นอย่างไร และหน้าที่ทั่วไปของการประมวลผล แต่จะไม่ทราบว่าการทำงานอย่างไร

2) กระแสข้อมูล มีลักษณะเป็นลูกศรและมีชื่อข้อมูลกำกับบนลูกศร หมายถึงข้อมูลที่ไหลไปมาระหว่างการประมวลผล กระแสข้อมูลอาจจะเคลื่อนที่มาจากสิ่งที่อยู่ภายนอกระบบก็ได้ โดยปกติกระแสข้อมูลหนึ่งอาจจะเข้าของการประมวลผลหนึ่ง และอาจจะ เป็นผลลัพธ์ของอีกการประมวลผลหนึ่ง

ชื่อข้อมูล



3) แหล่งเก็บข้อมูล มีลักษณะเป็นเส้นขนาน 2 เส้นและมีชื่อข้อมูลกำกับ หมายถึงที่เก็บข้อมูลต่าง ๆ โดยกระแสข้อมูลที่ไหลเข้าและออกจากแหล่งข้อมูลนี้จะอธิบายได้คือ กรณีที่กระแสข้อมูลหัวลูกศรชี้เข้าแหล่งเก็บข้อมูลแสดงถึงการเขียนหรือปรับปรุงข้อมูล และกระแสข้อมูลหัวลูกศรชี้ออกแสดงถึงการอ่านข้อมูล

ชื่อเพิ่มข้อมูล

4) สิ่งที่อยู่นอกระบบ มีลักษณะเป็นสี่เหลี่ยมผืนผ้าและมีชื่อกำกับอยู่ด้วย หมายถึงบุคคลหรือองค์กรต่าง ๆ ที่อยู่ภายนอกระบบ ซึ่งอาจจะส่งข้อมูลเข้าสู่ระบบหรืออาจจะรับข้อมูลจากระบบก็ได้

สิ่งที่อยู่ภายนอก ระบบ

2. การออกแบบและการพัฒนาระบบฐานข้อมูลบนอินเทอร์เน็ต

การพัฒนาระบบฐานข้อมูลในปัจจุบันไม่จำกัดเฉพาะการใช้ในเครื่องที่มีฐานข้อมูลอยู่เท่านั้น สามารถที่จะพัฒนาระบบฐานข้อมูลสำหรับการทำงานผ่านระบบเครือข่ายอินเทอร์เน็ตได้ ซึ่งในการพัฒนาแอปพลิเคชันในการใช้งานงาน สามารถทำได้หลายวิธี ดังนี้

CGI (Common Gateway Interface) เป็นเทคโนโลยีที่ใช้ในระยะแรก ๆ ในการพัฒนาเว็บร่วมกับฐานข้อมูล ใช้ภาษา Perl เป็นหลัก ส่วนมากใช้กับระบบ UNIX

ASP (Active Server Pages) เป็นเทคโนโลยีของไมโครซอฟต์ที่เพิ่มประสิทธิภาพของ HTML ให้สร้างเว็บเพจแบบไม่ตายตัว (Dynamic) เริ่มแรกมีเฉพาะบนเว็บเซิร์ฟเวอร์ IIS และ PWS บนระบบปฏิบัติการของไมโครซอฟต์ แต่ในปัจจุบันมีเครื่องมือเพื่อพัฒนาบน UNIX และ Windows NT ที่ไม่ได้ใช้ IIS หรือ PWS ได้แก่ Chile! Soft และ Instant Asp ของ Halcyon soft สามารถใช้งานร่วมกับสคริปต์ที่เป็น VBScript หรือ JavaScript ได้เป็นอย่างดี รวมทั้งสามารถใช้ร่วมกับระบบฐานข้อมูลได้หลายรูปแบบ

PHP (Professional Home Page) เป็นเทคโนโลยีใหม่ที่เริ่มเป็นที่นิยม เนื่องจากสนับสนุนเครื่องหลายรูปแบบโดยไม่จำเป็นต้องแก้ไขโค้ดเดิมที่มีอยู่ และที่สำคัญเป็น Open Source ที่สามารถใช้งานได้ฟรี

JSP (Java Server Pages) เป็นเทคโนโลยีที่ใช้ Java เป็นหลัก สามารถใช้ XHTML ได้

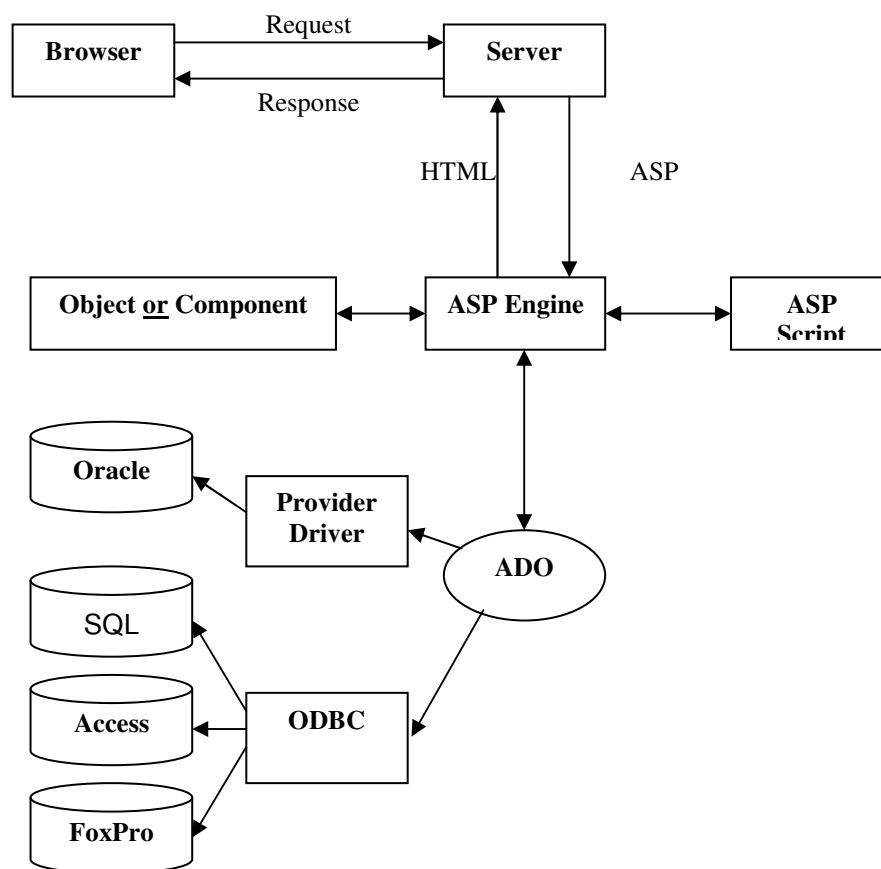
Cold Fusion เป็นเทคโนโลยีที่มีการทำงานคล้ายกับ ASP ต่างกันตรงที่ ASP ใช้ ADO และ VBScript แต่ Cold Fusion จะใช้แทรกพิเศษเพื่อการทำงานที่เกี่ยวข้องกับฐานข้อมูล

โดยจะทำงานบน Cold Fusion Server Software ซึ่งเป็นเซิร์ฟเวอร์ที่ทำงานได้ทั้ง Microsoft IIS, Netscape Enterprise Server และ Unix (Apache)

ในที่นี้จะแนะนำการใช้ Active Sever Pages หรือ ASP ซึ่งเป็นเทคโนโลยีที่บริษัท ไมโครซอฟท์พัฒนาขึ้นมา เอกสาร ASP เป็นเท็กซ์ไฟล์ที่ประกอบด้วยภาษาสคริปต์ เช่น VBScript หรือ Jscript (เป็นภาษาสคริปต์ของไมโครซอฟท์ที่คล้ายกับ JavaScript) ร่วมกับแท็กของ HTML แล้วเก็บไว้ที่เว็บเซิร์ฟเวอร์ โดยเอกสารที่เป็น ASP จะมีแท็ก ASP กำกับอยู่ (จะใช้แท็ก <% %>) ถ้าหากใช้กับบราวเซอร์ที่ไม่สนับสนุนการใช้งาน ASP ก็จะไม่แสดงผล เมื่อบราวเซอร์เรียกใช้งาน ก็จะมีตัวแปล (ASP Interpreter) และถูกเอ็กซีคิวต์ที่เว็บเซิร์ฟเวอร์ แล้วส่งผลลัพธ์ในรูปแบบของเอกสาร HTML ไปแสดงผลที่บราวเซอร์นั้น ซึ่งจะทำงานในลักษณะไม่ตายตัว (Dynamic) ทำให้สามารถแสดงข้อมูลที่เป็นข้อมูลล่าสุด ซึ่งต่างจากเว็บเพจแบบเดิมที่เป็นลักษณะตายตัว (Static)

2.1 หลักการทำงานของ ASP

ASP จะทำงานโดยมีตัวแปลและเอ็กซีคิวต์ที่ฝั่งเซิร์ฟเวอร์ เรียกการทำงานว่าเป็น Server Side ส่วนการทำงานของบราวเซอร์เราจะเรียกว่า Client Side การทำงานจะเริ่มจากที่ผู้ใช้ส่งความต้องการผ่านเว็บบราวเซอร์ทาง HTTP (HTTP Request) ซึ่งอาจเป็นการกรอกแบบฟอร์ม หรือใส่ข้อมูลที่ต้องการ ข้อมูลเหล่านั้นจะเป็นเอกสาร ASP (ไฟล์มีนามสกุลเป็น .asp) เมื่อมาถึงเว็บเซิร์ฟเวอร์ก็จะถูกส่งไปให้ ASP เพื่อทำหน้าที่แปลคำสั่งแล้วเอ็กซีคิวต์คำสั่งนั้น ซึ่ง ASP อาจจะใช้โอปเจ็คต์ คอมโพเนนต์ หรือ ADO หลังจากานั้น ASP จะสร้างผลลัพธ์ในรูปแบบเอกสาร HTML ส่งกลับไปให้เว็บเซิร์ฟเวอร์เพื่อส่งต่อไปให้บราวเซอร์แสดงผลทางฝั่งผู้ใช้ต่อไป (HTTP Response) ซึ่งลักษณะการทำงานแบบนี้จะคล้ายกับการทำงานของ CGI (Common Gateway Interface) หรืออาจกล่าวได้ว่า ASP ก็คือโปรแกรม CGI ประเภทหนึ่ง มีการทำงานแสดงได้ดังรูป 2-1



ภาพ 1 แสดงหลักการทำงานของ ASP

2.2 รูปแบบการเชื่อมต่อฐานข้อมูล

2.2.1 การติดต่อฐานข้อมูลโดยใช้ ODBC

การติดต่อแบบนี้จะต้องมีการกำหนดชื่อ DSN (Data Source Name) ให้กับ ODBC โดยจะต้องกำหนดชื่อไฟล์ฐานข้อมูลและชนิดของไดร์เวอร์ว่าเป็นอะไร ข้อดีสำหรับการติดต่อแบบนี้ ก็คือการเปลี่ยนจากฐานข้อมูลแบบหนึ่ง ไปเป็นอีกแบบหนึ่งทำได้ง่าย เช่น ต้องการเปลี่ยนจาก Access ไปเป็น SQL Server หรือ Oracle ก็เพียงแค่เปลี่ยนคุณสมบัติภายในเท่านั้น โดยไม่จำเป็นต้องแก้โปรแกรมเพราะคำสั่งในการติดต่อฐานข้อมูลไม่ว่าจะเป็นแบบไหนก็จะเขียนเหมือนกัน คือ

```
Set ObjDB=Server.CreateObject ("ADODB.Connection")
```

```
ObjDB.Open DSN-name, User-name,Password
```

โดยที่ DSN-name คือชื่อของ DSN ที่สร้างและกำหนดไดร์เวอร์ไว้แล้ว

ข้อเสียของวิธีนี้ก็คือ จะต้องไปเซตอัพ ODBC และสร้าง DSN ที่เซิร์ฟเวอร์นั้น ในการกรณีนี้ การที่จะเข้าไปเซตอัพเซิร์ฟเวอร์ที่ให้บริการฟรีอาจทำได้

2.2.2 การติดต่อกับฐานข้อมูลโดยใช้ DSNLess

การติดต่อแบบนี้ไม่ต้องมีการสร้าง DSN จึงมีข้อดีในการที่สามารถตัดขั้นตอนการเซตอัพ ODBC ออกทั้งหมด ทำให้สะดวกในการใช้งาน โดยไม่จำเป็นต้องเดินทางไปเซตอัพที่เซิร์ฟเวอร์ ส่วนข้อเสียของวิธีนี้ก็คือ ถ้าต้องการเปลี่ยนรูปแบบของฐานข้อมูล เช่น จาก MS Access ไปเป็น SQL Server จะต้องไปแก้ไปที่โปรแกรมในทุกไฟล์ที่มีการติดต่อ

รูปแบบการติดต่อแบบ DSNLess กับฐานข้อมูลที่สำคัญ ๆ มีดังนี้

FoxPro	<pre>Set ObjDB=Server.CreateObject("ADODB.Connection") ObjDB.Open "DRIVER={Microsoft Foxpro Driver (*.dbf)};DBQ=path"</pre> <p>โดยที่การกำหนด path จะทำให้ใช้ไฟล์ .dbf ได้ทุกไฟล์ เนื่องจาก FoxPro จัดเก็บข้อมูลแยกในแต่ละไฟล์</p>
MS Access	<pre>Set ObjDB=Server.CreateObject("ADODB.Connection") ObjDB.Open "DRIVER={Microsoft Access Driver (*.mdb)};DBQ=path and database"</pre> <p>โดยที่ path and database คือพาทและตำแหน่งของไฟล์ฐานข้อมูล เนื่องจากใน 1 ฐานข้อมูลประกอบด้วยออบเจกต์ เช่น ตาราง คิวรี ได้มากมาย ที่ต่างจาก FoxPro ที่เก็บแยกไฟล์ อาจกล่าวได้ว่า 1 ตารางใน Access เท่ากับ 1 ไฟล์ใน FoxPro นั่นเอง</p>
SQL Server	<pre>Set ObjDB=Server.CreateObject("ADODB.Connection") ObjDB.Open "DRIVER={SQL Server};SERVER=server-name; DATABASE=database_name UID=user_Name; PWD=password"</pre> <p>โดยที่ Server-name คือชื่อเซิร์ฟเวอร์ หรือ IP ของเซิร์ฟเวอร์ก็ได้ database-name คือชื่อของฐานข้อมูล user-name คือชื่อของผู้ใช้ password คือรหัสผ่านของผู้ใช้</p>

2.3 การดำเนินการกับฐานข้อมูล

ในการดำเนินการกับฐานข้อมูลที่จะนำเสนอต่อไปนี้จะใช้ MS SQL Server 7.0 เป็น DBMS โดยจะมีการสร้างตารางข้อมูลเพื่อเป็นตัวอย่าง จำนวน 3 ตาราง ดังนี้

ตาราง Books

Column Name	Datatype	Length	Precision	Scale	Allow Nulls	Default Value	Identity	Identity Seed	Id
book_id	char	3	0	0	<input type="checkbox"/>		<input type="checkbox"/>		
title	nvarchar	50	0	0	<input type="checkbox"/>		<input type="checkbox"/>		
publisher	nvarchar	50	0	0	<input type="checkbox"/>		<input type="checkbox"/>		
order_date	datetime	8	0	0	<input type="checkbox"/>		<input type="checkbox"/>		

ตาราง Borrow

Column Name	Datatype	Length	Precision	Scale	Allow Nulls	Default Value	Identity	Identity Seed	Id
book_id	char	3	0	0	<input type="checkbox"/>		<input type="checkbox"/>		
member_id	char	3	0	0	<input type="checkbox"/>		<input type="checkbox"/>		
return_date	datetime	8	0	0	<input type="checkbox"/>		<input type="checkbox"/>		

ตาราง Member

Column Name	Datatype	Length	Precision	Scale	Allow Nulls	Default Value	Identity	Identity Seed	Id
id	char	3	0	0	<input type="checkbox"/>		<input type="checkbox"/>		
name	nvarchar	50	0	0	<input type="checkbox"/>		<input type="checkbox"/>		
address	nvarchar	50	0	0	<input type="checkbox"/>		<input type="checkbox"/>		

2.3.1 การติดต่อกับฐานข้อมูล

ในการติดต่อกับฐานข้อมูลที่สร้างขึ้นจาก MS SQL Server 7.0 โดยใช้คำสั่ง ASP จะกำหนด ดังนี้

ตัวอย่างคำสั่งของ ASP กรณีติดต่อโดยใช้ DSNLess

```
<%
Set con=Server.CreateObject("ADODB.Connection")
con.Open "Driver={SQL Server};SERVER=suphakit; DATABASE=LabSQL; UID=sa;
PWD="
%>
```

ตัวอย่างคำสั่ง ASP กรณีติดต่อโดยใช้ DSN โดยสร้าง DSN ชื่อ LABSQL

```
<%
Set con=Server.CreateObject("ADODB.Connection")
con.Open "LABSQL","sa","
%>
```

2.3.2 การแสดงข้อมูล

ขั้นตอนการอ่านข้อมูลเข้ามาแล้วแสดงผลบนบราวเซอร์ มีดังต่อไปนี้

1) การติดต่อกับฐานข้อมูล โดยใช้วิธีการดังข้อ 12.4.1 ที่กล่าวมาแล้ว

```
Dim con (อาจจะไม่มีหรือไม่มีก็ได้)
Set con=Server.CreateObject("ADODB.Connection")
con.Open "Driver={SQL Server};SERVER=suphakit; DATABASE=LabSQL;
UID=sa; PWD="
```

2) เขียนคำสั่ง SQL เพื่อให้แสดงข้อมูลตามฟิลด์ที่ต้องการ ดังตัวอย่างต้องการแสดงทุกฟิลด์

```
strSQL = "SELECT * FROM Books"
```

3) สั่งให้คำสั่ง SQL ทำงาน

```
Dim rsBooks (อาจจะไม่มีหรือไม่มีก็ได้)
Set rsBooks=Server.CreateObject("ADODB.Recordset")
Set rsBooks=con.Execute (SQL)
```

4) เขียนคำสั่งเพื่อดึงข้อมูลตั้งแต่เรคคอร์ดแรกไปจนถึงเรคคอร์ด

สุดท้าย

```
Do While Not rsBooks.EOF
```



```

Response.Write rsBooks.Fields("Book_ID") & " "
...
Response.Write rsBooks.Fields("Order_Date") & "<BR>"
rsBooks.MoveNext
Loop
5) การปิดการเชื่อมต่อกับฐานข้อมูล
con.Close
Set con=Nothing

```

2.3.3 การเพิ่มข้อมูล

ในการเพิ่มข้อมูลลงในตารางของฐานข้อมูล จะต้องทำการสร้างฟอร์มเพื่อรับข้อมูลจากผู้ใช้ หลังจากนั้นจะต้องส่งข้อมูลในลักษณะตัวแปรตามฟิลด์ต่าง ๆ แล้วเขียนโปรแกรมลงฐานข้อมูล โดยมีขั้นตอนในการดำเนินการ ดังนี้

- 1) ทำการเชื่อมต่อกับฐานข้อมูลเช่นเดียวกับหัวข้อที่ 2.3.2
- 2) การบันทึกรายข้อมูลลงในตาราง โดยใช้คำสั่ง SQL

```

strSQL="INSERT INTO Books (Book_ID, Title, Publisher, Order_Date) VALUES ("
&_
Request.Form("BookID") & "," &_
....
Request.Form("OrderDate") & ");"

```

```
Set rsBooks = Server.CreateObject("ADODB.Recordset")
```

```
Set rsBooks = con.Execute (strSQL)
```

- 3) ปิดการเชื่อมต่อกับฐานข้อมูล

2.3.4 การแก้ไขข้อมูล

การแก้ไขข้อมูลที่อยู่ในฐานข้อมูล มีวิธีการในการดำเนินการ ดังนี้

- 1) ทำการเชื่อมต่อกับฐานข้อมูลเช่นเดียวกับหัวข้อที่ 1.2.3.2
- 2) ทำการแก้ไขข้อมูล โดยใช้คำสั่ง SQL

```

strSQL="UPDATE Books SET Title=" & Request.Form("Title") &_
" WHERE Book_ID=" & Request.Form("BookID") & ";"

```

```
response.write strSQL & "<BR>"
```

```
Set rsBooks = Server.CreateObject("ADODB.Recordset")
```

Set rsBooks = con.Execute (strSQL)

3) ปิดการเชื่อมต่อกับฐานข้อมูล

2.3.5 การลบข้อมูล

การลบข้อมูลที่อยู่ในฐานข้อมูล มีวิธีการในการดำเนินการ ดังนี้

- 1) ทำการเชื่อมต่อกับฐานข้อมูลเช่นเดียวกับหัวข้อที่ 1.2.3.2
- 2) ทำการแก้ไขข้อมูล โดยใช้คำสั่ง SQL

strSQL="DELETE FROM Books WHERE Book_ID=" & Request.Form("BookID") &

""

response.write strSQL & "
"

Set rsBooks = Server.CreateObject("ADODB.Recordset")

Set rsBooks = con.Execute (strSQL)

3) ปิดการเชื่อมต่อกับฐานข้อมูล

2.3.6 การค้นหาข้อมูล

การค้นหาข้อมูลตามเงื่อนไขแล้วให้แสดง มีวิธีการในการดำเนินการ

ดังนี้

- 1) ทำการเชื่อมต่อกับฐานข้อมูลเช่นเดียวกับหัวข้อที่ 12.4.2
- 2) ทำการค้นหาข้อมูลตามเงื่อนไข โดยใช้คำสั่ง SQL

strSQL="Select * From Books Where Title Like '%M%'"

Set rsBook=Server.CreateObject("ADODB.Recordset")

Set rsBook = con.Execute (strSQL)

3) แสดงข้อมูลที่ได้ออกจากการค้นหา

Do while not rsBook.Eof

Response.Write rsBook.Fields("Book_ID").Value & ", "

...

Response.Write rsBook.Fields("Order_Date").Value & "
"

rsBook.MoveNext

Loop

4) ปิดการเชื่อมต่อกับฐานข้อมูล

แนวคิดเกี่ยวกับทฤษฎี .NET

.NET คือ กรอบการทำงาน (Framework) อันใหม่ของไมโครซอฟท์ ที่ครอบคลุมทั้งส่วนของผู้ใช้ ธุรกิจ และนักพัฒนา โดยมีการขยายกรอบการทำงานจากจุดเดิมคือวินโดวส์ที่เน้นเพียงเครื่องคอมพิวเตอร์(PC ทั้งที่เป็นไคลเอนต์และเซิร์ฟเวอร์) มาเป็นการทำโซลูชัน (Solution) ที่สามารถใช้งานที่ไหนก็ได้ เวลาใดก็ได้ และบนอุปกรณ์ใด ๆ ก็ได้ (Anywhere, Any time and on Any Devices) สิ่งที่สำคัญของ .NET คือ เครือข่ายคอมพิวเตอร์ซึ่งอินเทอร์เน็ตก็เป็นหนึ่งในเครือข่ายหลักที่ใช้ใน .NET และมีการขยายขอบเขตการมองระบบปฏิบัติการ และซอฟต์แวร์ให้กว้างยิ่งขึ้น โดยถือว่าอินเทอร์เน็ตนั้นคือระบบปฏิบัติการ (Operating System) และเว็บไซต์ (ซึ่งให้บริการหนึ่ง ๆ) เป็นซอฟต์แวร์ของระบบปฏิบัติการ (อินเทอร์เน็ต) แต่แทนที่จะเรียกว่าซอฟต์แวร์อย่างเดิม ก็เรียกใหม่ว่าเป็นเซอร์วิส (Services) หรือเว็บเซอร์วิส (Web Services) แทน

ตัวอย่างของกรณีดังกล่าวคือ เราในฐานะผู้พัฒนาเว็บไซต์ ไปดึงเอาเซอร์วิสต่าง ๆ เช่น เคาน์เตอร์ ฐานข้อมูล และห้องสนทนา จากเว็บไซต์อื่น ๆ จากนั้นเพิ่มเติมรายละเอียดอีกนิดหน่อยสำหรับเว็บไซต์เรา เพียงเท่านั้นก็มีบริการใหม่ให้ผู้เข้ามาเยี่ยมชมได้แล้ว

ในอนาคตของ .NET จะมีบริการให้นักพัฒนาเลือกใช้มากมาย นักพัฒนาก็เพียงเรียกใช้และประกอบโซลูชันออกมาให้ดีและเหมาะสมเท่านั้น ฉะนั้นในมุมมองของนักพัฒนา (ระดับทั่ว ๆ ไป) การสร้างโซลูชันก็จะง่ายขึ้น อีกทั้งสามารถให้บริการโซลูชันของตนกับผู้ใช้ โดยผ่านเครือข่ายคอมพิวเตอร์หรืออินเทอร์เน็ตได้ ในขณะที่ไม่ต้องไปกังวลว่าเซอร์วิสต่าง ๆ ที่นำมาประกอบนั้นอยู่ที่ไหนหรือต้องไปนั่งเขียนโปรแกรมเพื่อจัดการเรื่องเน็ตเวิร์กอีก นักพัฒนาเพียงรู้ว่าเซอร์วิสมันทำอะไรให้ได้อ่าง มีข้อจำกัดอย่างไร จากนั้นก็ประกอบและปรับต่างบางอย่างให้เหมาะสมเท่านั้น

สำหรับ .NET ในมุมมองของผู้ใช้ก็เพียงแต่เรียกใช้บริการเท่านั้น บริการต่าง ๆ ก็จะวิ่งเข้ามาให้บริการถึงที่ โดยที่อุปกรณ์ที่ใช้ ๆ นั้นก็ได้จำกัดอยู่เฉพาะคอมพิวเตอร์ แต่จะมีการขยายไปถึงโทรศัพท์มือถือ คอมพิวเตอร์พกพา และอุปกรณ์อื่น ๆ ที่สามารถติดต่อกับเครือข่ายคอมพิวเตอร์ได้ บริการต่าง ๆ ที่นำเสนอมาให้นั้นก็จะนำเสนอในรูปแบบที่เหมาะสมกับอุปกรณ์ของผู้ใช้ ยกตัวอย่างเช่น ต้องการจะซื้อดอกไม้จากร้านค้าบนอินเทอร์เน็ต ถ้าใช้คอมพิวเตอร์ก็อาจมีรูปภาพสีสันสวยงาม แต่ถ้าใช้โทรศัพท์มือถือซึ่งมีหน้าจอเล็กกว่ามาก ก็จะมีขนาดและรูปแบบและรายละเอียดของการนำเสนอที่แตกต่างกันไป (ในขณะที่ทางฝั่งร้านค้านั้นอาจใช้ข้อมูลจากฐานข้อมูลเพียงอย่างเดียว) อีกหน่อยสำหรับผู้ใช้งาน เวลาจะใช้ซอฟต์แวร์ก็อาจไม่จำเป็นต้องซื้อซอฟต์แวร์ทุกตัวมาใช้ อาจมีการเช่าเซอร์วิสแทน โดยเมื่อจะใช้ก็เรียก ระบบก็จะวิ่งเข้าไปดึงบริการมาจากศูนย์ต่าง ๆ เมื่อใช้งานเสร็จก็จบและเก็บเงินจากการใช้นั้น ตัวอย่างหนึ่งที่เรากำลังใช้งานกันประจำ แต่ที่เราไม่รู้สึกรู้สีกอะไรเพราะยังใช้งานฟรีอยู่ (แต่จริง ๆ ไม่ฟรีทีเดียวเพราะเขามีการอัดโฆษณาส่งมาให้เราตลอดด้วย) คือบริการเว็บ

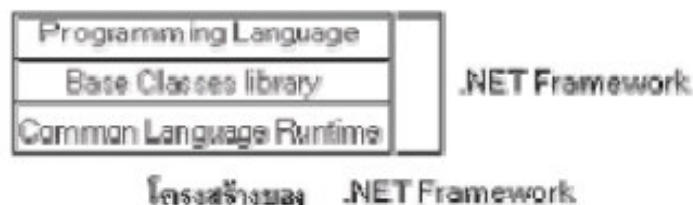
เบสอีเมล เช่น Hotmail เป็นต้น เราเพียงเรียกใช้ เราไม่รู้และไม่จำเป็นต้องรู้เลยว่าระบบพาสปอร์ต (ซึ่งใช้ตรวจสอบผู้ใช้) นั้นอยู่ที่ไหน เขาเก็บข้อมูลของเราอย่างไร ใช้ฐานข้อมูลอะไร อยู่บนระบบปฏิบัติการอะไร เรากำลังใช้บริการจากเครื่องใดในโลก ในฐานะผู้ใช้เราไม่ต้องรับภาระเรื่องการอัปเดตซอฟต์แวร์ ทางศูนย์จัดการให้เราเสร็จสรรพ (นี่เป็นสาเหตุหนึ่งที่ทำให้ Hotmail บางช่วงทำงานได้ไม่เร็วเพราะมีการอัปเดตบริการใหม่ ๆ อยู่เป็นประจำ)

บริการ .NET ที่ทางไมโครซอฟท์มีให้ลองใช้ในช่วงแรกนี้มี 3 ส่วนคือ เซอร์วิสพื้นที่เก็บข้อมูล (Storage) เซอร์วิสการตรวจสอบความเป็นส่วนบุคคล (Authentication) และเซอร์วิสเตือนต่าง ๆ (Notification) ตัวอย่างเช่น ให้พื้นที่เก็บเมลและไฟล์ใน มีบริการตรวจสอบโดยใช้พาสเวิร์ดและมีบริการเตือนต่าง ๆ โดยผ่าน MSN Messenger (ตอนนี้ยังสามารถเตือนผ่านอุปกรณ์ประเภท Mobile ได้อีกด้วย)

อย่างที่ได้อธิบายไปแล้วในตอนต้นว่าไมโครซอฟท์ต้องการที่จะสร้างอะไรที่เป็นมาตรฐานขึ้นมา เพื่อให้ทุกสิ่งทุกอย่าง เพื่อให้ทุกสิ่งทุกอย่างสามารถติดต่อสื่อสารกันได้หมด โดยคิดค้นระบบซึ่งหมายถึงมันนั้นมือว่าจะให้เป็นระบบมาตรฐาน ระบบนี้คือ .NET Framework ซึ่งระบบนี้ไม่ใช่ระบบปฏิบัติการ (OS) แต่เปรียบเสมือนโปรแกรมหนึ่งที่จะสามารถสร้างสภาวะแวดล้อมหนึ่งซึ่งสามารถทำงานในระบบ .NET นี้ได้

ในอนาคตไมโครซอฟท์ก็หวังที่จะนำเอาระบบนี้ไปติดตั้งลงบนอุปกรณ์ทุกชนิด เพื่อให้ทุกอุปกรณ์ทุกอย่างมีระบบ ๆ หนึ่งที่เหมือนกันหมด โดย .NET Framework นั้นมี ส่วนประกอบภายในแบ่งออกเป็น 3 ชั้นใหญ่ๆคือ 1.Programming Language: เป็นรูปแบบของภาษาที่ออกแบบมาเพื่อให้สามารถทำงานในสภาวะที่เป็น .NET ได้โดยที่ทางไมโครซอฟท์ได้เปิดตัวภาษาหลักๆที่จะใช้พัฒนามบน .NET นี้ 3ภาษา ° C# เป็นภาษาใหม่ที่ไม่โครซอฟท์พัฒนามาจาก C++ กับ JAVA เป็นหลัก ° VB.NET เป็นภาษาที่พัฒนามาจาก Visual Basic ในเวอร์ชัน 6.0 ° JScript.net เป็นภาษาที่พัฒนามาจาก JScript ซึ่งเป็น JavaScript ในเวอร์ชันของไมโครซอฟท์ 2.Base Classes Library: Library นั้นเปรียบเสมือนชุดคำสั่งสำเร็จรูปย่อย ๆ ที่เพิ่มเข้ามา ซึ่งส่วนใหญ่จะเป็นชุดคำสั่งที่ต้องใช้งานอยู่เป็นประจำ ดังนั้นจึงมีผู้คิดค้นเพื่ออำนวยความสะดวกในการเขียนโปรแกรม ซึ่ง Library ในภาษาต่างๆ ส่วนใหญ่จะอยู่ในรูปแบบไฟล์ incould แต่ถ้าเป็น ASP สิ่งที่เป็น library ก็คือ คอมโพเนนท์ต่างๆนั่นเอง ซึ่งภายในระบบ .NET จะสร้างสิ่งที่เรียกว่าเป็น Library พื้นฐานขึ้น ทำให้ไม่ว่าจะใช้ภาษาใดในการพัฒนาโปรแกรมก็สามารถที่จะเรียกใช้Libraryที่เป็นตัวเดียวกันได้หมด 3.Common Language Runtime (CLR): นับเป็นสิ่งสำคัญแทบจะที่สุดของระบบ .NET นี้ก็ว่าได้ เพราะ CLR ที่ว่านี้มีหน้าที่ทำให้โปรแกรมที่เขียนขึ้นมาด้วยภาษาต่างๆ กัน กลายเป็นภาษารูปแบบมาตรฐานเดียวกัน ทั้งหมด

เราเรียกภาษาที่ว่าเป็น Intermediate language (IL) ซึ่งเมื่อต้องการที่จะรันโปรแกรมใด CLR ที่ว่านี้จะตรวจสอบเครื่องที่รันว่ามีสภาวะแวดล้อมการทำงานเช่นใดหลังจากนั้นก็คอมไพล์เป็นโปรแกรมที่เหมาะสมต่อการทำงานของเครื่องนั้น ทำให้เราสามารถใช้งานโปรแกรม ต่าง ๆ ได้อย่างมีประสิทธิภาพสูงสุดในแต่ละเครื่อง



ภาพ 2 โครงสร้างของ .Net Framework

ประโยชน์และข้อดีของ .NET Framework นั้นพอจะสรุปออกมาได้เป็นข้อๆดังนี้

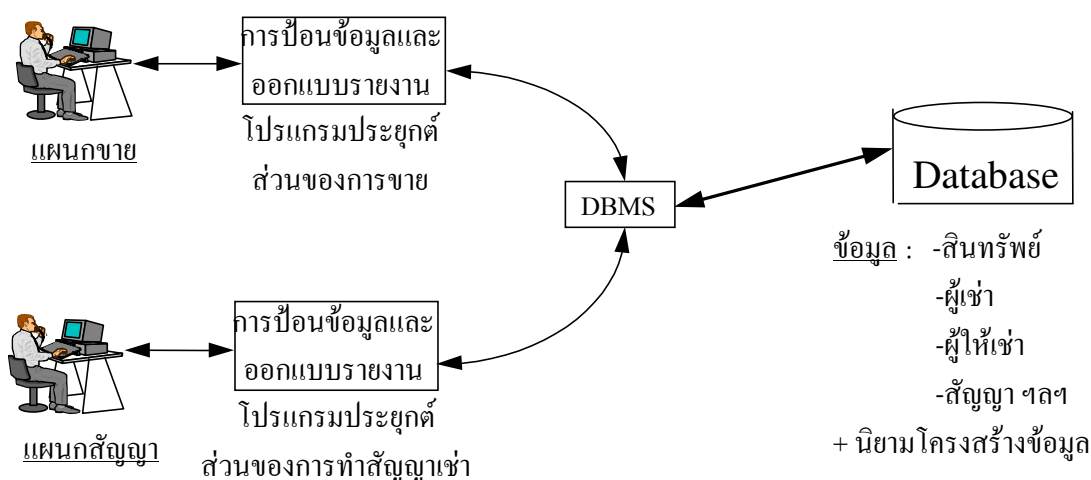
1. เป็นระบบที่มีไลบรารีที่เป็นมาตรฐานเดียวกัน เนื่องจากมีไลบรารีที่เป็นมาตรฐานเดียวกันทั้งหมดทำให้เราไม่ต้องกังวลว่าภาษาที่ใช้เขียนนั้นมีไลบรารีตัวนั้นตัวนี้หรือไม่ รวมทั้งไม่ต้องคอยกังวลว่าถ้าใช้ไลบรารีของภาษาหนึ่งแล้วอีกภาษาหนึ่งจะไม่มีไลบรารีตัวนั้น
2. ไม่ขึ้นกับระบบปฏิบัติการ (OS) เนื่องจากระบบปฏิบัติการ ที่แต่ละบุคคลหรือองค์กรใช้นั้นย่อมไม่เหมือนกัน แต่ภายใน .NET Framework จะไม่มีปัญหาของเพียงแค่มีระบบ .NET Framework ก็จะทำให้สามารถใช้งานโปรแกรมต่าง ๆ ได้ ซึ่งเป็นข้อดีตรงที่เราจะสามารถใช้โปรแกรมต่าง ๆ ได้ทุกระบบปฏิบัติการ
3. ใช้ในการพัฒนาได้ทุกภาษา ทำให้เราไม่ต้องคอยมาศึกษาภาษาใหม่ ๆ เมื่อต้องการสร้างโปรแกรมในแต่ละครั้ง นอกจากนั้นเรายังสามารถเลือก ใช้ภาษาที่เราถนัดที่สุดในการพัฒนาโปรแกรมต่าง ๆ ได้ด้วย
4. มีการควบคุมสิ่งแวดล้อมในการทำงานเป็นอย่างดี เนื่องจากเป็นระบบที่เป็นมาตรฐาน ทำให้การควบคุมจัดสรรระบบต่าง ๆ ทำได้ง่ายขึ้น ไม่ว่าจะเป็นการจัดสรรหน่วยความจำ ด้านการใช้งานเครื่องก็มีความรวดเร็วมากขึ้นลดโอกาสที่เครื่องจะผิดปกติได้เป็นอย่างดี
5. ความปลอดภัยที่มีมากขึ้น .NET Framework สามารถกำหนดสิทธิ์การใช้งาน หรือ permission ของผู้ใช้งานได้มากขึ้นทำให้สามารถกำหนดว่า จะให้โปรแกรมในส่วนตัวใช้งานได้หรือไม่ได้แล้วแต่เฉพาะบุคคล

แนวคิดเกี่ยวกับระบบจัดการฐานข้อมูล (Database Management System - DBMS)

แนวความคิดระบบฐานข้อมูลเริ่มก่อตัวขึ้นประมาณ ค.ศ. 1960 เป็นต้นมา ในช่วงเวลาดังกล่าวมีความพยายามที่จะพัฒนาเทคนิคการจัดเก็บข้อมูลปริมาณจำนวนมาก ๆ เทคนิคแรกจัดเก็บข้อมูลลงแฟ้มข้อมูล (file) แล้วเขียนโปรแกรมดึงข้อมูลมาใช้งานโดยผ่านวิธีการเข้าถึงข้อมูลแบบมาตรฐาน ของระบบปฏิบัติการของเครื่องคอมพิวเตอร์นั้น ๆ ในช่วงเวลาดังกล่าว มีการพัฒนาเทคนิคการจัดการแฟ้มข้อมูลและวิธีการเข้าถึงข้อมูลให้ได้เร็วขึ้น

อย่างไรก็ตาม ข้อมูลที่จัดเก็บจะเพิ่มความซับซ้อนและมีความเกี่ยวโยงกันมากขึ้น ผู้ใช้งานต้องการรวมแฟ้มข้อมูลและโปรแกรมประยุกต์ต่าง ๆ ที่เคยใช้งานแยกเป็นอิสระต่อกันให้ใช้งานร่วมกัน ยิ่งไปกว่านั้นยังต้องการแจกแจงความสัมพันธ์ในระหว่างระเบียบต่าง ๆ นั้นด้วย

ระบบจัดการฐานข้อมูล (DBMS) เป็นกลุ่มโปรแกรมใช้ติดต่อกับข้อมูลในฐานข้อมูล เพื่อเข้าถึงและจัดการกับข้อมูล เช่น เพิ่ม ลบ หรือเปลี่ยนแปลงข้อมูล เป็นต้น คุณลักษณะที่จำเป็นของระบบจัดการฐานข้อมูล ต้องให้ผู้ใช้สามารถ ระบุข้อมูลที่ ต้องการจะได้ในรูปแบบนามธรรม ในขณะที่เดียวกันก็ต้องทำหน้าที่ไปค้นหาข้อมูลที่จัดเก็บในฐานข้อมูลได้ด้วย ดังแสดงในรูป



ภาพ 3 การประมวลผลระบบฐานข้อมูลเช่าที่อยู่อาศัย

เป้าหมายของระบบจัดการฐานข้อมูลแจกแจงได้ดังนี้

1. การเกี่ยวโยงความสัมพันธ์ระหว่างข้อมูล
2. ความสอดคล้องของข้อมูล
3. ความง่ายในการเข้าถึงข้อมูล
4. ความปลอดภัยของข้อมูล

5. การใช้ข้อมูลร่วมกัน
6. ความเป็นอิสระของข้อมูล
7. สมรรถนะของระบบ
8. การควบคุมบริหารงาน

ความเกี่ยวข้องความสัมพันธ์ระหว่างข้อมูล

ระบบจัดการแฟ้มข้อมูลไม่เหมาะสมที่จะใช้จัดการฐานข้อมูลเนื่องจากไม่สามารถกำหนดการเกี่ยวข้องความสัมพันธ์ที่ซับซ้อนระหว่างข้อมูลได้ ความเกี่ยวข้องดังกล่าวสอดคล้องกับความสัมพันธ์ของสิ่งของ (entities) ในระบบงานนั้น ๆ ตัวแบบฐานข้อมูล (Data Model) จะกำหนดโครงสร้างของข้อมูลและวิธีการเชื่อมโยงความสัมพันธ์ระหว่างข้อมูลภายใต้โครงสร้างนั้น ๆ ตัวแบบฐานข้อมูลมีหลายแบบ เช่น ตัวแบบฐานข้อมูลลำดับขั้น (Hierarchical Data Model) ตัวแบบฐานข้อมูลเชื่อมโยง (Network Data Model) หรือตัวแบบฐานข้อมูลเชิงสัมพันธ์ (Relational Data Model) รายละเอียดของตัวแบบต่าง ๆ จะได้อธิบายต่อไป ระบบจัดการฐานข้อมูลจะต้องอิงกับตัวแบบฐานข้อมูลแบบใดแบบหนึ่ง เพื่อใช้กำหนดโครงสร้างข้อมูลและการแทนความสัมพันธ์ด้วยวิธีการของตัวแบบนั้น

ความสอดคล้องของข้อมูล

เมื่อเซตของรายการข้อมูลประกอบไปด้วยสาระความรู้จำนวนมากเก็บไว้ที่ฐานข้อมูล ข้อมูลดังกล่าวต้องสอดคล้องตามสภาพความเป็นจริง ดังนั้นระบบจัดการฐานข้อมูลต้องอนุญาตให้ผู้ใช้งานกำหนดกฎเพื่อคงไว้ซึ่งความสอดคล้องของฐานข้อมูล กฎเหล่านี้เป็นการกำหนดคุณสมบัติหรือเงื่อนไขของข้อมูล ตัวอย่างเช่นในระบบการลงทะเบียนอาจมีข้อบังคับกำหนดให้นักศึกษาลงทะเบียนแต่ละภาคการศึกษาไม่เกิน 21 หน่วยกิต เป็นต้น การรักษาความสอดคล้องของข้อมูลยังรวมไปถึงระบบจำกัดสิทธิการใช้งานเพื่อที่ว่าจัดการกับข้อมูลบางประการจะจำกัดเฉพาะกลุ่มผู้ใช้งานที่รับผิดชอบในเรื่องนั้น ๆ ตัวอย่างเช่น จะให้สิทธิเฉพาะพนักงานการเงินที่จะเข้าไปเปลี่ยนแปลงค่าข้อมูลเงินเดือนได้ เป็นต้น

ความง่ายในการเข้าถึงข้อมูล

ระบบจัดการฐานข้อมูลต้องทำให้ผู้ใช้งานเข้าถึงข้อมูลใด ๆ ในฐานข้อมูลได้โดยง่ายด้วยภาษาเรียกค้นข้อมูล (Query language) ภาษาดังกล่าวควรมีลักษณะเป็นคำอธิบายสิ่งที่ต้องการค้นหามากกว่าจะแสดงขั้นตอนการทำงานเพื่อให้ได้ผลลัพธ์ข้อมูลที่ต้องการ นอกจากนี้การเรียกใช้ข้อมูลอาจทำได้ดังในรูปแบบการโต้ตอบ (interactive) โดยใช้ภาษาเรียกค้นข้อมูลหรืออาจเขียนโปรแกรมประยุกต์ติดต่อเข้าไป ใช้ข้อมูลโดยผ่านระบบจัดการฐานข้อมูล

ความปลอดภัยของข้อมูล

ระบบจัดการฐานข้อมูลต้องสามารถป้องกันข้อมูลจากการล่องละเมิดจากภายนอก เหตุการณ์ดังกล่าวอาจเกิดจากระบบคอมพิวเตอร์เกิดขัดข้อง เช่น ไฟดับ หรือหัวอ่านจานแม่เหล็กไม่ทำงาน โปรแกรมมีข้อผิดพลาด หรือผู้ใช้งานจงใจทำลายข้อมูล เพื่อที่จะป้องกันข้อมูลจากข้อผิดพลาดทั้งที่มาจากระบบโปรแกรมหรือระบบคอมพิวเตอร์ดังกล่าวระบบจัดการฐานข้อมูลต้องกำหนดจุดตรวจสอบ (Check point) เป็นช่วง ๆ เพื่อถ่ายโอนข้อมูลที่เปลี่ยนแปลงถูกต้องครบสมบูรณ์แล้วบันทึกลงจานแม่เหล็กเป็นการถาวร นอกจากนี้ระบบจัดการฐานข้อมูลต้องมีการบันทึกการเปลี่ยนแปลงลงไฟล์ปุม (log file) เก็บไว้เป็นช่วง ๆ เพื่อใช้กู้ฐานข้อมูลย้อนกลับสู่ภาวะปกติล่าสุดเท่าที่จะทำได้

การใช้ข้อมูลร่วมกัน

ดังที่กล่าวมาแล้วเหตุผลหลักอันหนึ่งของแนวความคิดเกี่ยวกับฐานข้อมูล คือการใช้ข้อมูลร่วมกันระหว่างโปรแกรมประยุกต์ โปรแกรมต่างโปรแกรมใช้ส่วนของข้อมูลเดียวกันจะต้องสามารถประมวลผลเสมือนหนึ่งเป็นเจ้าของและใช้ส่วนของข้อมูลนั้นเพียงผู้เดียว ระบบจัดการฐานข้อมูลจะต้องมีวิธีการจัดการข้อมูลที่ใช้ร่วมกันโดยหาวิธีการป้องกันการขัดแย้งของการเข้าใช้ข้อมูลเดียวกันจากหลายโปรแกรมพร้อม ๆ กันและวิธีการแก้ไขปัญหาดังกล่าวด้วย

ความเป็นอิสระของข้อมูล

ความเป็นอิสระของข้อมูลเป็นข้อได้เปรียบสำคัญของการประมวลผลในระบบฐานข้อมูล เปรียบเทียบกับการประมวลผลเพิ่มข้อมูล ในระบบการประมวลผลโดยเพิ่มข้อมูลโปรแกรมจะผูกติดหรือขึ้นตรงกับข้อมูล โปรแกรมต้องรู้โครงสร้างการจัดเก็บและวิธีการเข้าถึงข้อมูลในเพิ่มข้อมูลที่ใช้ หากมีการเปลี่ยนแปลงโครงสร้างหรือวิธีการเข้าถึงข้อมูลก็ต้องเปลี่ยนคำสั่งในโปรแกรมตามโครงสร้างใหม่ด้วย ในทางตรงกันข้ามระบบจัดการฐานข้อมูลทำให้การเขียนโปรแกรมไม่ต้องพะวงกับโครงสร้างการจัดเก็บและวิธีการดึงข้อมูลมาใช้ งาน กล่าวคือระบบจะจัดการกับการเปลี่ยนแปลงโครงสร้างหรือวิธีการเข้าถึง ข้อมูลเมื่อมีความต้องการใหม่เพิ่มขึ้น โดยไม่ต้องแก้ไขคำสั่งในโปรแกรมแต่ประการใด ด้วยการจัดการดังกล่าวทำให้ประหยัดเวลาในการบำรุงรักษา หรือปรับเปลี่ยนระบบงานประยุกต์ให้สอดคล้องกับความต้องการเปลี่ยนแปลงในอนาคต

เราสามารถแยกความเป็นอิสระของข้อมูลได้เป็นสองระดับ คือ ความเป็นอิสระเชิงกายภาพ (Physical independence) และความเป็นอิสระเชิงตรรกะ (logical independence) ดังรายละเอียดจะได้กล่าวภายหลัง

สมรรถนะของระบบ

คุณสมบัติทั้งหลายของระบบจัดการฐานข้อมูลที่ได้กล่าวมาแล้วข้างต้นจะต้องทำให้การดำเนินงานบรรลุผลสำเร็จ กล่าวคือระบบจัดการฐานข้อมูลต้องมี ความสามารถจัดการข้อมูล

จำนวนมากและให้ผลลัพธ์ได้ภายในเวลาเหมาะสม การปรับปรุงสมรรถนะของระบบจะเกี่ยวข้องกับการปรับปรุงเทคนิคการเข้าถึงข้อมูลและการทำให้ดีที่สุด (Optimization) ในทุกสิ่งที่เกี่ยวข้องกับระบบฐานข้อมูล เช่นการออกแบบฐานข้อมูล หรือการเขียนคำสั่งงานที่สั่งงานได้เร็ว เป็นต้น หัวข้อนี้ยังจะต้องมีการศึกษาวิจัยกันต่อไป

การควบคุมบริหารงาน

ผู้จัดการฐานข้อมูล (Database Administrator) เป็นบุคคล หรือทีมงานที่ทำหน้าที่ออกแบบและบำรุงรักษาข้อมูล ทั้งนี้เนื่องจากผู้ใช้งานฐานข้อมูลมีความต้องการที่หลากหลาย ซึ่งบางครั้งอาจขัดแย้งกันได้ประกอบกับลักษณะการใช้งานจะใช้งานร่วมกันในเวลาเดียวกัน ดังนั้นจึงมีความจำเป็นต้องมีผู้รับผิดชอบดูแลฐานข้อมูลเป็นการเฉพาะ รายละเอียดบทบาท / หน้าที่ของบุคคลอื่น ๆ ที่เกี่ยวข้องในระบบฐานข้อมูลจะอธิบายภายหลัง

หลักการเว็บเบสเทคโนโลยี (Web-based Technology)

ในปัจจุบันรูปแบบการนำเสนอและเผยแพร่ข้อมูลข่าวสารทางอินเทอร์เน็ต โดยส่วนมากจะนิยมจะเป็นในรูปแบบของข้อมูลที่ผ่านมาสื่อประเภทเว็บเพจ (Web page) ซึ่งข้อมูลที่น่าสนใจออกไปนั้น สามารถเผยแพร่ ได้ทั้งรูปแบบข้อมูลตัวอักษร ข้อมูลภาพ และข้อมูลเสียง และภาพเคลื่อนไหวต่าง ๆ ที่มีเทคนิคในการนำเสนอที่หลากหลาย การนำเสนอข้อมูลประเภทเว็บเพจ นั้นสามารถที่จะเชื่อมโยงไปยังจุดอื่น ๆ บนหน้าเว็บได้ ตลอดจนสามารถเชื่อมโยงไปยังเว็บอื่น ๆ ในระบบเครือข่ายได้

ข้อดีและลักษณะเด่นของเว็บเพจมีดังนี้

1. การนำเสนอข้อมูลผ่านเว็บ เป็นการนำเสนอด้วยข้อมูลที่สามารถเรียกหรือโยงไปยังจุดอื่น ๆ ในระบบกราฟิก ซึ่งทำให้ข้อมูลนั้น ๆ มีจุดดึงดูดเป็นพิเศษทำให้น่าเรียกดู
2. เป็นระบบ Interactive ซึ่งก็คือการทำงานบนเว็บเป็นการโต้ตอบกับผู้ใช้โดยธรรมชาติ โดยเริ่มตั้งแต่ผู้ใช้เปิดโปรแกรมที่ใช้แสดงผลเว็บ (Browser) พิมพ์ชื่อเรียกเว็บ (URL : Uniform Resource Locator) เมื่อเอกสารแสดงผลผ่านเบราว์เซอร์ ผู้ใช้ก็จะสามารถคลิกเลือกรายการหรือข้อมูลที่สนใจ จึงเป็นการทำงานแบบโต้ตอบไปในตัว
3. ข้อมูลบนเว็บไม่ยึดติดกับระบบปฏิบัติการ (Operating System) เนื่องจากเป็นข้อมูลที่ถูกจัดเก็บเป็น Text File ดังนั้น ไม่ว่าจะถูกเก็บไว้ในคอมพิวเตอร์ที่ใช้ระบบปฏิบัติการที่เป็น Unix หรือ Windows ก็จะสามารถเรียกดูจากคอมพิวเตอร์ที่ใช้ ระบบปฏิบัติการต่างจากคอมพิวเตอร์

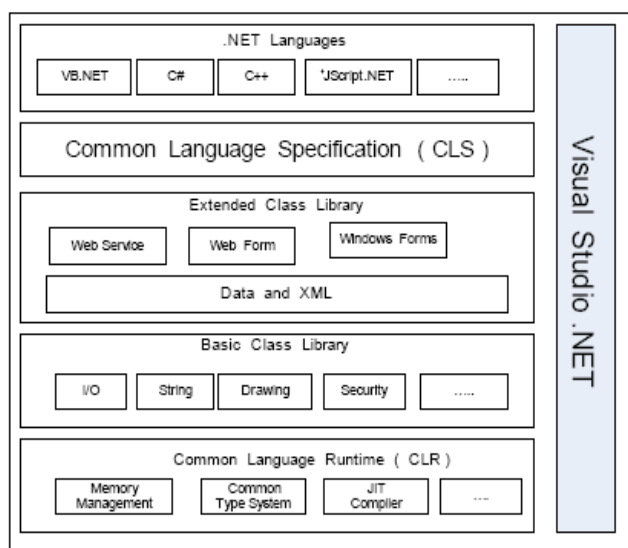
1. ASP .NET ปัจจุบันการใช้งานคอมพิวเตอร์ของแต่ละองค์กรหรือแม้แต่วางหน่วยงานในองค์กรเดียวกันนั้นมีความหลากหลายมาก แตกต่างกันทั้งอุปกรณ์ ระบบปฏิบัติการ

และโปรแกรมที่ใช้งานในขณะที่ความต้องการและความจำเป็นที่จะต้องแลกเปลี่ยนข้อมูลกันหรือทำธุรกิจร่วมกันมีเพิ่มขึ้นความแตกต่างทางด้านโครงสร้างพื้นฐานของระบบคอมพิวเตอร์ที่แต่ละองค์กรใช้จึงกลายเป็นอุปสรรคสำคัญที่ทำให้ความร่วมมือกันทางด้านข้อมูลเป็นไปได้ด้วยความยากลำบากและเสียค่าใช้จ่ายในการดำเนินการสูง

Microsoft (สุรสิทธิ์ คิวประสพศักดิ์ และ นันทน์ แขวงโสภา, 2546, หน้า 14-18) ได้สังเกตเห็นว่าในอนาคตการแลกเปลี่ยนข้อมูลระหว่างกันจะมีมากขึ้นโดยมีปัจจัยสนับสนุนจากเครือข่ายอินเทอร์เน็ต ที่นับวันก็จะมีความเร็วในการเชื่อมต่อเพิ่มขึ้นส่วนทางกับค่าใช้จ่ายที่ต่ำลง เมื่อหลายปีที่แล้ว Microsoft จึงเริ่มต้นโครงการ NGWS ซึ่งย่อมาจาก Next Generation Windows Service หรือ Next Generation Web Services ขึ้น ซึ่ง Microsoft .Net ในปัจจุบันก็คือ ชื่อใหม่ของโครงการนี้นั่นเอง

Microsoft .NET หรือเรียกสั้น ๆ ว่า .NET เป็นเทคโนโลยีใหม่ล่าสุดที่ใช้ในการพัฒนาซอฟต์แวร์สำหรับระบบปฏิบัติการ Windows นำเสนอหลักการพัฒนาซอฟต์แวร์ภาษาอะไรก็ได้ที่เราถนัด ใน .NET มีภาษาโปรแกรมใหม่ ๆ เช่น Visual Basic.NET, C# .NET , C++ .NET , J#.NET หรือแม้กระทั่ง COBOL.NET เป็นต้น

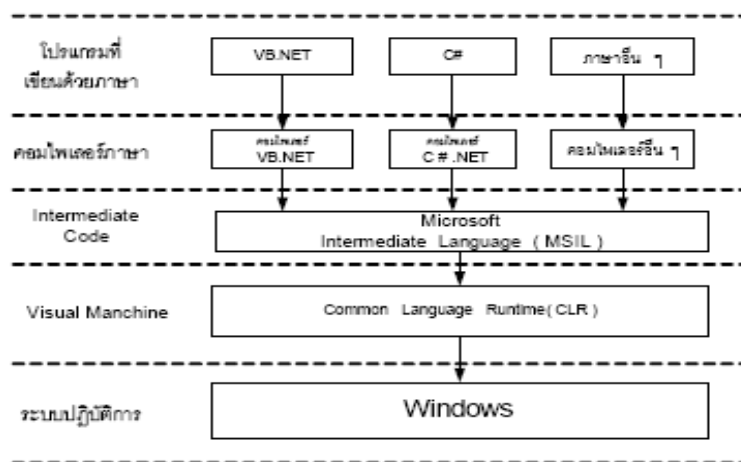
สิ่งสำคัญที่เป็นหัวใจสำคัญของ .NET คือ .NET Framework ซึ่งมีส่วนประกอบต่าง ๆ แสดงในรูปด้านล่าง



ภาพ 4 แสดงสถาปัตยกรรม .NET Framework

.NET ไม่ได้เป็นเพียงภาษาโปรแกรมใหม่เท่านั้น แต่ยังถือเป็นรากฐานในการพัฒนา แอปพลิเคชันยุคใหม่ความสามารถที่โดดเด่นของ .NET ก็คือการพัฒนาโปรแกรม ด้วยภาษาใด ๆ ก็ได้ที่สนับสนุน Common Language Specifications (CLS) ของ .NET ซึ่งช่วยให้นักพัฒนาสามารถเลือกใช้ภาษาใด ก็ได้ (language neutral) โดย .NET Framework จะมีเครื่องมือที่เรียกว่า Microsoft Visual Studio 2005 ซึ่งถือเป็น Integrated Development Environment (IDE) สำหรับการพัฒนาโปรแกรม

โปรแกรมที่เขียนขึ้นมานั้นเมื่อคอมไพล์แล้วจะอยู่ในรูปของ Intermediate language ที่เรียกว่า MSIL (Microsoft Intermediate Language) ซึ่งแนวคิดเดียวกับไบต์โค้ด ของ Java Platform จากนั้นโปรแกรมที่พัฒนาภายใต้ .NET จะสามารถเรียกใช้โปรแกรมที่เขียนด้วยภาษาอื่นได้ ถ้าอยู่ภายใต้มาตรฐาน CLS เหมือนกัน ปัจจุบันมีภาษาโปรแกรมมากกว่า 20 ภาษาที่สนับสนุน CLS เช่น Pascal .NET , Perl .NET หรือ แม้กระทั่ง COBOL.NET เป็นต้น อธิบายขั้นตอนสุดท้ายซึ่งเป็น ส่วนที่สำคัญของ .NET Framework ซึ่งคือ Common Language Runtime (CLR) จะทำหน้าที่ execution engine ในการประมวลผลและจัดการโปรแกรมที่คอมไพล์แล้วให้ทำงานได้บน ระบบปฏิบัติการ Windows โดย CLS จะแปลงโค้ดในรูป MSIL ไปเป็นคำสั่งภาษาเครื่อง โดยใช้เทคโนโลยีในการแปลงแบบ Just-In-Time (JIT) คือแปลงเฉพาะส่วนที่จะนำมาใช้งานเท่านั้น หลังจากนั้นถ้าต้องการนำส่วนอื่น ๆ มาใช้งานอีกก็จะแปลงเพิ่มเฉพาะส่วนนั้น ซึ่งช่วยให้โปรแกรม ทำงานได้เร็วขึ้นนอกจากนี้ CLR ยังทำหน้าที่ติดต่อกับระบบปฏิบัติการ จัดสรรหน่วยความจำให้กับ โปรแกรมต่าง ๆ และ คืนหน่วยความจำที่ไม่ถูกใช้งานแล้วให้กับระบบ (Garbage Collection) มีการ จัดการกับข้อผิดพลาด (exception handling) รวมถึงดูแลเรื่องความปลอดภัย (security management) ด้วยแสดงได้ดังรูปด้านล่าง



สำหรับการพัฒนาโครงการนี้ ทางผู้จัดทำโครงการจะเลือกใช้เครื่องมือภาษา Microsoft Visual Studio 2005 ถือเป็นเวอร์ชันล่าสุดของโปรแกรม Visual Basic ที่ได้รับการพัฒนามาจากภาษา BASIC เป็นภาษาที่เข้าใจง่ายได้รับความนิยมอย่างแพร่หลาย ปัจจุบันเป็นเครื่องมือภาษาหนึ่ง บนชุด Microsoft Visual Studio 2005 แต่ก็ยังมีเวอร์ชัน Express Edition ที่ไม่ได้รวมอยู่ในชุด Microsoft Visual Studio 2005 อีกด้วย

VB.NET ได้รับการพัฒนาขีดความสามารถเพิ่มเติมขึ้นมาอีกมากมาย และมีโครงสร้างภาษาที่เปลี่ยนแปลงไปมาก คำสั่งและความสามารถเดิมบางส่วนใน Visual Basic 6.0 ก็ถูกยกเลิกไป

ขีดความสามารถที่เพิ่มขึ้นที่สำคัญได้แก่

1) เป็นภาษา OOP (Object – Oriented Programming) เต็มตัวในการทำ inheritance ,overloading , และ overriding เป็นต้น เช่นเดียวกับภาษาสมัยใหม่ทั่วไป เช่น C++ , Dephi หรือ Java เป็นต้น

2) รับเอาความสามารถของ .NET ภายใต้ .NET Framework ซึ่งมีมาตรฐาน CLR และมาตรฐานในส่วนของชนิดข้อมูล ทำให้สามารถเขียนโปรแกรมด้วย VB.NET แล้วไปเรียกใช้งานโปรแกรมที่ภาษาอื่น เช่น C#.NET ได้

3) ได้รับการพัฒนาให้สามารถสร้างเว็บแอปพลิเคชันแบบใหม่เรียกว่า Web Form ซึ่งมีวิธีการสร้างแบบ drag-and-drop เหมือนกับการพัฒนาแอปพลิเคชันบน Windows โดยทั่วไป และสามารถสร้าง Web Service ซึ่งอาศัย XML (Extensible Markup Language) เป็นตัวกลางในการแลกเปลี่ยนข้อมูล

4) รองรับการสร้างเว็บแอปพลิเคชันด้วย ASP .NET สามารถผนวกกับ Web Form และ Web Service ในการสร้างเว็บแอปพลิเคชันได้อย่างรวดเร็ว

5) รองรับการสร้างแอปพลิเคชันแบบ Console ซึ่งถือเป็นแอปพลิเคชันแบบใหม่ใน VB.NET เพื่อช่วยให้สามารถทำงานในลักษณะโปรแกรมที่รันบน DOS คือแสดงผลและรับข้อมูลในลักษณะของข้อความได้- รองรับการพัฒนาโปรแกรมระดับ Threading

6) มีโครงสร้างการจัดการข้อผิดพลาดที่ดีขึ้น และมีโครงสร้างเช่นเดียวกับภาษา OOP โดยทั่วไป- รองรับ ADO.NET ซึ่งเป็นเทคโนโลยีการติดต่อฐานข้อมูลแบบใหม่ ที่มาทดแทน ADO และ RDO ใน VB6

7) มีการจัดการหน่วยความจำที่ดีขึ้น โดยอาศัย CLR

8) มีการจัดการ I/O ได้อย่างมีระบบและมีประสิทธิภาพมากขึ้นด้วยการใช้คลาส System.IO มีคอนโทรลเพิ่มขึ้นอีกมากซึ่งจะสามารถช่วยลดเวลาการพัฒนาแอปพลิเคชันไปได้มาก

ASP.NET เป็นเทคโนโลยีการพัฒนาเว็บแอปพลิเคชันแบบไดนามิกใน .Net Framework ที่อยู่ในชุด VB.NET ได้รับการพัฒนาจาก ASP เวอร์ชัน 3.0 เดิม ซึ่ง ASP จะทำงานบนเครื่องที่ใช้โปรแกรม IIS (Internet Information Service) ของไมโครซอฟท์เป็นเว็บเซิร์ฟเวอร์บนระบบปฏิบัติการ Windows โดยเวอร์ชันแรกรันอยู่บน Windows NT 4.0 , Windows NT Workstation 4.0 และได้รับการพัฒนาต่อเนื่องบน Windows จนกระทั่งปัจจุบันบน Windows XP Professional โครงสร้างของ ASP.NET ประกอบด้วยไฟล์ประเภท HTML บวกกับแท็กพิเศษ และไฟล์ที่นามสกุลเป็น .aspx ซึ่งเก็บชุดคำสั่งหรือสคริปต์ ๆ ไว้ การติดต่อกับผู้ใช้งานจะเป็นรูปแบบ HTML ปกติ แต่การประมวลผลจะอยู่ฝั่งเซิร์ฟเวอร์และส่งผลกลับมาอยู่ในรูปแบบ HTML (และอาจรวมถึงสคริปต์ที่ทำงานฝั่งไคลเอนต์) เพื่อแสดงผลแก่ผู้ใช้งาน

ASP.NET และ ASP เวอร์ชันเดิมมีข้อแตกต่างอยู่หลายจุดด้วยกัน ข้อแตกต่างที่เห็นได้ชัดเจนก็คือ ASP จะทำงานในลักษณะ Interpreter คือแปลคำสั่งที่ละบรรทัด การเขียนโปรแกรมจะอาศัยภาษาสคริปต์ เช่น VBScript , JavaScript และ Jscript ในขณะที่ ASP.NET จะประมวลผลในลักษณะ Compiler คือมีการคอมไพล์โปรแกรมไปอยู่ในรูปแบบ Intermediate Language (IL) ซึ่งจะอยู่ในไฟล์ Page Class ทำให้การทำงานรวดเร็วขึ้น เพราะกรณีที่มีการเรียกเพจเดิมซ้ำอีกครั้งสามารถเรียกไฟล์ Page Class ที่ผ่านการคอมไพล์แล้วมาใช้ได้เลย ไม่ต้องคอมไพล์ใหม่ทุกครั้ง ในรายละเอียดการสร้างโปรแกรมก็มีข้อแตกต่างเช่นกัน การสร้างโปรแกรม ASP จะเป็นการรวมเอาคำสั่ง VBScript หรือ Jscript และโครงสร้าง HTML ไว้ในไฟล์เดียวกัน ซึ่งพัฒนาโดยใช้เครื่องมือเช่น FrontPage , Visual InterDev หรือกระทั่งอิดิเตอร์ทั่วไป เช่น Notepad , EditPlus แต่การสร้างโปรแกรม ASP.NET สามารถแยกส่วนแสดงผลและโปรแกรมไว้คนละไฟล์กันได้ เป็นแนวคิดที่เรียกว่า Code Behind การสร้างโปรแกรมด้วย ASP.NET สามารถใช้ภาษาโปรแกรมใด ๆ ก็ได้ที่สนับสนุน .NET เช่น VB.NET , C#.NET เป็นต้น ขั้นตอนการสร้างจะใช้วิธีลากคอนโทรลมาวางบนฟอร์มที่เรียกว่า Web Form แล้วเขียนโปรแกรมกำกับอ็อบเจกต์ของคอนโทรลต่าง ๆ เหล่านั้น ซึ่งมีลักษณะเดียวกันกับการสร้างโปรแกรมที่ทำงานบน Windows นอกจากนี้แล้วข้อแตกต่างที่สำคัญก็คือ ASP เดิมจะทำงานในลักษณะที่เรียกว่า stateless คือไม่มีการเก็บสถานะของผู้ใช้งานไว้ ส่วน ASP.NET จะทำงานในลักษณะ Stateful คือสามารถควบคุมและเก็บสถานะผู้ใช้ไว้ได้

โครงสร้างการทำงานใน ADO.NET ผู้ที่เคยใช้ Visual Basic 6.0 พัฒนาโปรแกรมด้านฐานข้อมูลมาก่อนจะทราบว่าจะใช้เทคโนโลยี ADO (ActiveX Data Object) เป็นเครื่องมือในการติดต่อและจัดการข้อมูลแบบ connected database คือจะต้องเปิดการเชื่อมต่อกับฐานข้อมูล

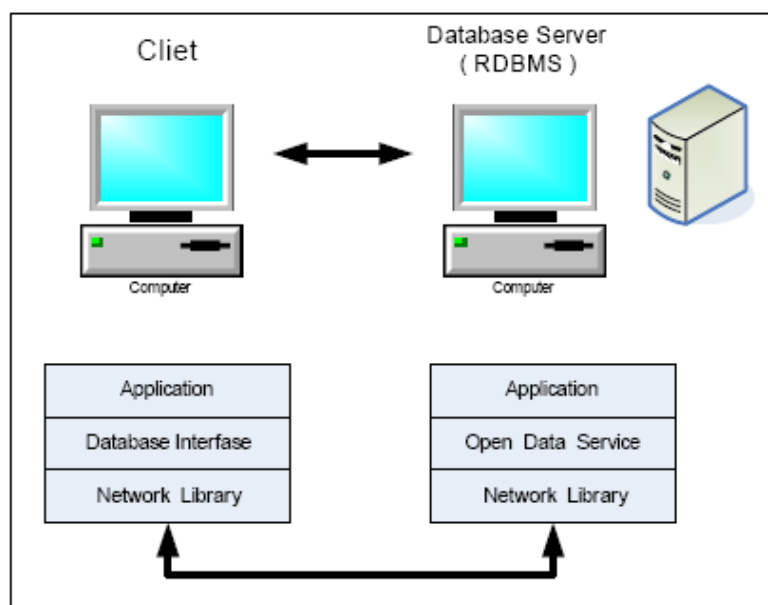
ในตลอดช่วงเวลาใช้งานซึ่งจะทำให้การเพิ่ม ลบ แก้ไขหรือการกระทำใด ๆ ก็ตามที่เกี่ยวข้องกับข้อมูลจะเป็นการกระทำกับข้อมูลจริงในฐานข้อมูลนั้นโดยตรง ซึ่งมีข้อดีคือทำให้ข้อมูลถูกปรับปรุงให้ทันสมัยและถูกต้องอยู่ตลอดเวลา แต่ข้อเสียคือ ถ้ามีผู้ใช้ฐานข้อมูลเป็นจำนวนมากและพร้อม ๆ กันหรือการประมวลผลข้อมูลที่มีปริมาณมากจนทำให้ระยะเวลาที่เชื่อมต่อกับ เซิร์ฟเวอร์นานเกินไป ซึ่งจะทำให้อัตราการสิ้นเปลืองทรัพยากรบนเครื่องเซิร์ฟเวอร์สูง ส่งผลให้ประสิทธิภาพโดยรวมของระบบลดลงได้ ดังนั้นใน Microsoft Visual Studio 2005 จึงได้นำ เทคโนโลยีตัวใหม่ล่าสุดคือ ADO.NET ที่พัฒนาจาก ADO มาใช้จัดการข้อมูลแบบ disconnected database คือสามารถจัดการข้อมูลในฐานข้อมูลโดยไม่ต้องเปิดการเชื่อมต่อไว้ตลอดเวลาเหมือนใน ADO เนื่องจากข้อมูลจะถูกอ่านจากฐานข้อมูลมาเก็บไว้ในหน่วยความจำก่อน จากนั้นจะตัดการเชื่อมต่อกับฐานข้อมูล ทำให้การกระทำใด ๆ หลังจากนั้นจะเป็นการกระทำกับข้อมูลในหน่วยความจำเท่านั้น เมื่อเสร็จสิ้นกระบวนการแล้วจึงจะบันทึกกลับลงฐานข้อมูล ซึ่งจะช่วยประหยัดทรัพยากรของระบบและเพิ่มประสิทธิภาพการทำงานของเซิร์ฟเวอร์ (ที่จริงแล้ว ADO เองก็มีความสามารถในการทำ disconnected database ได้ แต่โปรแกรมเมอร์จะต้องเป็นผู้ควบคุมด้วยตนเองว่าจะให้มีการเชื่อมต่อหรือยกเลิกการเชื่อมต่อเมื่อใด ซึ่งต่างกับใน ADO.NET ที่จะตัดการเชื่อมต่อให้โดยอัตโนมัติ เมื่อไม่มีความจำเป็นต้องใช้ฐานข้อมูลในแบบ on-line และจะเปิดการเชื่อมต่อใหม่เมื่อจำเป็นต้องทำงานร่วมกับฐานข้อมูลจริง)

2. โปรแกรม Microsoft SQL Server

Microsoft SQL Server (อัมไพ สลินลิขิตกุล, 2544, หน้า 15 - 21) เป็นระบบการจัดการฐานข้อมูลแบบเชิงสัมพันธ์ (Relational Database Management System มีชื่อย่อว่า RDBMS) สามารถติดตั้งและทำงานได้กับระบบปฏิบัติการ Windows 95/98 , Windows NT 4.0 , Windows XP และ Windows 2000 โดย SQL Server นี้ ออกแบบมาให้ทำงานในลักษณะที่เป็น Client – Server Database จึงสามารถรองรับการทำงานจากเครื่องไคลเอนต์ได้เป็นจำนวนมากที่ต่อผ่านทางระบบเครือข่ายหรือเน็ตเวิร์กเข้ามา นอกจากนี้ Microsoft SQL Server ยังออกแบบมาเพื่อใช้ในงานที่เป็นแบบ Stand-Alone Database ได้ด้วย โดยติดตั้งลงบนเครื่องที่ใช้ระบบปฏิบัติการ Window 95/98

นอกเหนือจากความสามารถในด้านการจัดการข้อมูลแล้ว Microsoft SQL Server ยังมีภาษาที่ใช้เพื่อดูแล บำรุงรักษา และลบข้อมูลจากฐานข้อมูลเรียกว่า Transact-SQL โดยยึดตามมาตรฐานทั้ง SQL-89 และ SQL-92 นอกจากนี้ Transact-SQL ยังมีคำสั่งช่วยในการทำงานเพิ่มเติมจากส่วนที่เป็นมาตรฐานเพื่อช่วยการทำงานคล่องตัวมากขึ้น

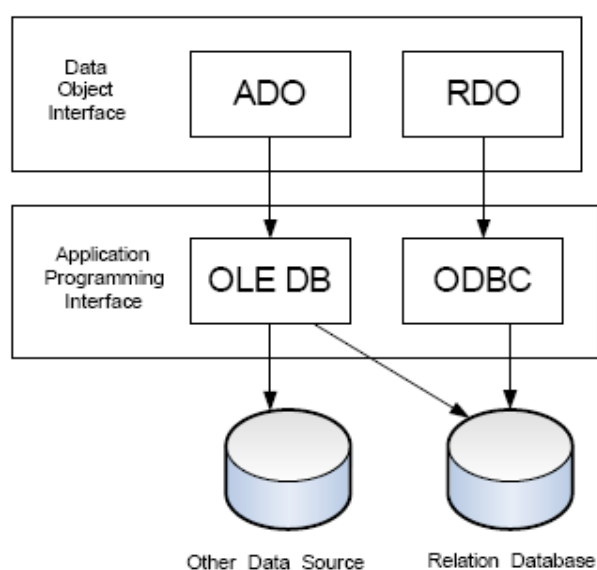
สถาปัตยกรรมของ SQL Server โดยที่ SQL Server จะมีส่วนที่ทำหน้าที่ติดต่อสื่อสารกับทางฝั่งไคลเอนต์อยู่ด้วย แต่ก็แยกส่วนที่จัดการเน็ตเวิร์กและโปรโตคอลออกจากส่วนที่เป็นแอปพลิเคชัน ทำให้แอปพลิเคชันนี้สามารถทำงานอยู่บนเน็ตเวิร์กแบบใดก็ได้



อธิบายความหมายจากรูปได้ดังนี้

- 1) Application หมายถึงโปรแกรมที่พัฒนาขึ้นเพื่อใช้ข้อมูลจากฐานข้อมูลโดยผ่านอินเตอร์เฟซของโปรแกรมที่เรียกว่า API (Application Programming Interface เป็นวิธีการเขียนโปรแกรมให้สามารถติดต่อกับฐานข้อมูล)
- 2) Database Interface หมายถึง อินเตอร์เฟซที่ใช้แอปพลิเคชันเพื่อติดต่อไปยัง SQL Server เช่น ODBC (Open Database Connectivity) หรือ OLE DB (Object Linking and Embedding Database)
- 3) Network Library หรือเรียกว่า Net-Library หมายถึงซอฟต์แวร์ที่ทำหน้าที่จัดส่งข้อมูลผ่านไปทางเน็ตเวิร์ก โปรแกรมนี้จะต้องติดตั้งทั้งด้านของเซิร์ฟเวอร์และไคลเอนต์ สำหรับ SQL Server มี Net-Library ที่จะทำงานกับโปรโตคอลหลายประเภท เช่น TCP/IP ,Named-Pipes, Novell IPX/SPX, Banyan VINES/IP และ AppleTalk ADSP

- 4) Open Data Service เป็นอินเทอร์เฟซระหว่าง Net-Library และแอปพลิเคชันทางฝั่งเซิร์ฟเวอร์ เช่น SQL Server เอง และ extended stored procedure ต่างก็เป็นแอปพลิเคชันฝั่งเซิร์ฟเวอร์
- 5) Open Data Service นี้จะทำหน้าที่ในการเชื่อมต่อ และรับ request จากทางไคลเอนต์แล้วส่งต่อไปให้กับ SQL Server ทำการประมวลผล รวมทั้งรับผลที่ได้นั้นแล้วส่งกลับไปที่ไคลเอนต์



การทำงานร่วมกับแอปพลิเคชัน ของ SQL Server

การพัฒนาโปรแกรมเพื่อให้ติดต่อกับฐานข้อมูลทำได้โดยใช้ชุดคำสั่งหรือฟังก์ชันที่ SQLServer จัดเตรียมไว้ให้ที่เรียกว่า API ซึ่งวิธีการที่จะต่อกับฐานข้อมูลมีดังนี้คือ

- 1) OLE DB (Object Linking and Embedding Database) เป็น อินเทอร์เฟซชนิด Component Object Model (COM based) ที่สนับสนุนการทำงานของแอปพลิเคชันที่ใช้ OLE DB การทำงานของ OLE DB ได้ถูกออกแบบมาเพื่อใช้กับ Relational database และได้มีการพัฒนาให้สามารถติดต่อกับระหว่างโปรแกรมฐานข้อมูลหลาย ๆ แบบ เช่น Access, Oracle เป็นต้น
- 2) ODBC (Open Database Connectivity) เป็นอินเทอร์เฟซที่ช่วยในการติดต่อกับฐานข้อมูลหรือ แอปพลิเคชัน ทั้งหลายที่ใช้ ODBC
- 3) Data Object Interfaces มี 2 ส่วน คือ

3) ADO (Active X Data Objects) เป็นออบเจ็คที่ช่วยให้พัฒนาแอปพลิเคชันด้วย OLEDB ทำได้ง่ายขึ้น (เพราะ OLE DB เป็น API ระดับ Low Level) มักนำ ADO มาใช้งาน

ในหลายๆ ภาษา เช่น Microsoft Visual Basic, Microsoft Visual C++, Active Server Pages (ASP)

4) RDO (Remote Data Objects) เป็นออบเจ็คที่ช่วยพัฒนาแอปพลิเคชันด้วย OLE DB ทำได้ง่ายขึ้น เช่นกัน แต่มักใช้ในกรณีที่ใช้ข้อมูลข้ามเครื่อง

5) SQL Server Component เป็นส่วนประกอบต่างๆ ของ Microsoft SQL Server มีรายละเอียดของคอมโพเนนต์ที่สำคัญดังนี้

6) SQL Server Services เป็นเซอวิสที่จะช่วยในการควบคุมการทำงานของ SQL Server ในกรณีที่ SQL Server ติดตั้งอยู่บนเครื่องที่เป็น Windows NT นั้นเซอวิสต่างๆ จะทำงานอยู่บน Windows NT

7) MS SQL Server Service เป็นเซอวิสหลักของ SQL Server โดยควบคุมการทำงานของเซอวิสอื่นๆ ทั้งหมด จัดการเกี่ยวกับทรานแซกชัน ดูแลและจัดการไฟล์ของฐานข้อมูล ควบคุมการใช้ทรัพยากรของระบบ ดูแลการทำงานแบบภาวะพร้อมกัน (Concurrency Control) การจัดการเกี่ยวกับความปลอดภัย ความถูกต้องและความสอดคล้องกันของข้อมูล

8) SQL Server Agent Service เป็นเซอวิสที่ทำงานร่วมกับ MS SQL Server Service โดยควบคุมการทำงานของ Multi Server Job ได้แก่งานที่ติดต่อแลกเปลี่ยนข้อมูลระหว่าง Server (Replication) การแจ้งเตือน (Alert) การจัดการด้าน Operator

9) Microsoft Distributed Transaction Coordinator (MS DTC) Service เป็นเซอวิสที่ควบคุมการทำงานของชุดคำสั่ง (Transaction) ที่ต้องการติดต่อหรือจัดการข้อมูลระหว่างเซิร์ฟเวอร์ซึ่งมีมากกว่า 1 เครื่องให้ข้อมูลบนเครื่องเหล่านั้นมีความถูกต้องและสอดคล้องกัน ควบคุมการคอมมิท ของข้อมูลโดยใช้หลัก Two-Phase Commit (ซึ่งจะช่วยให้มั่นใจได้ว่าการจัดการกับข้อมูลในแบบต่างๆ บนเซิร์ฟเวอร์ซึ่งมีมากกว่า 1 เครื่องนั้นจะถูกต้องครบถ้วนและสมบูรณ์เสมอ)

10) Microsoft Search Service เป็นเซอวิสที่ช่วยในการหาข้อมูลในลักษณะ Text Searching โดยเซอวิสนี้ เป็นทางเลือกหนึ่งที่จะติดตั้งหรือไม่ติดตั้งก็ได้ ในขณะที่ติดตั้ง SQL Server การจัดการเซอวิสต่างๆ เหล่านี้สามารถควบคุมโดยอาศัย Service Manager โดยจะปรากฏอยู่ที่ ทาสก์บาร์ที่มุมขวาล่างของหน้าจอ ให้ดับเบิลคลิกที่ไอคอนนั้น หรืออาจควบคุมโดยผ่านทาง Windows NT

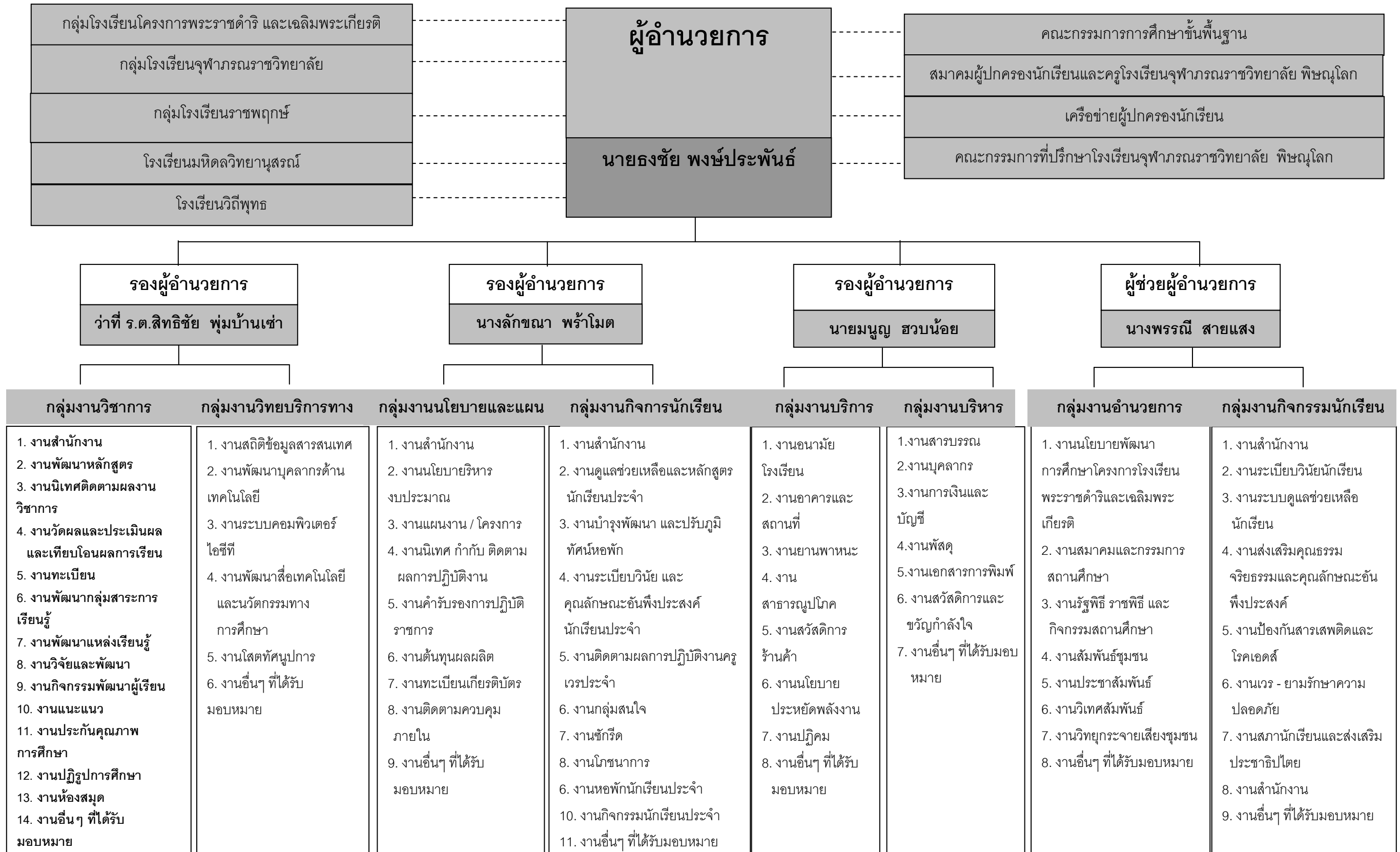
11) SQL Server Enterprise Manager มีลักษณะเป็นกราฟิก ช่วยให้ทำงานกับระบบจัดการฐานข้อมูลของ SQL Server ได้ง่ายขึ้น โดยเลือกหรือคลิกตำแหน่งที่ต้องการ ก็จะแสดงผลลัพธ์หรือเมนูถัดที่แสดงตัวเลือกได้ การทำงานโดยใช้ Enterprise Manager ได้แก่

- 12) ใช้ติดต่อกับเซิร์ฟเวอร์ต่างๆ โดยการรีจิสเตอร์ไปยังเซิร์ฟเวอร์นั้น
- 13) จัดการเกี่ยวกับดาต้าเบส เทเบิล อินเด็กซ์ วิว Stored Procedure
- 14) ทำการสำรองและติดตั้งกลับในส่วนฐานข้อมูลของ SQL Server
- 15) ตรวจสอบการทำงานของผู้ใช้แต่ละคน และการทำงานของระบบ
- 16) ควบคุมการแจ้งเตือน (Alert) และการสร้างล็อกไฟล์เพื่อบันทึกการทำงานของข้อผิดพลาดของ SQL Server

17) จัดการด้าน Replicate Management ซึ่งหมายถึงกระบวนการในการคัดลอกข้อมูลหรือ ทรานแซกชันจาก SQL Server เครื่องหนึ่งไปยังอีกเครื่องหนึ่ง

โครงสร้างการบริหารงานของโรงเรียนจุฬาภรณราชวิทยาลัย พิชณโลก

โครงสร้างการบริหารงานโรงเรียนจุฬาภรณราชวิทยาลัย



งานวิจัยที่เกี่ยวข้อง

นุชรัตน์ (2545) ทำการพัฒนาระบบการจัดการเอกสารอิเล็กทรอนิกส์สำหรับองค์กรรัฐกรณีสึกษาคณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ เป็นระบบที่พัฒนาขึ้นเพื่อดำเนินการรับ-ส่ง ค้นหา และจัดเก็บเอกสารในองค์กร พัฒนาด้วยโปรแกรม Visual Basic 6 ส่วนของการเก็บข้อมูลใช้ SQL Server 7 และทำการเชื่อมต่อกับฐานข้อมูลผ่าน ODBC โปรแกรมที่พัฒนาขึ้นทำงานในลักษณะ Client/Server ใช้คอมพิวเตอร์เครื่องหนึ่งสำหรับเก็บข้อมูล มีระบบการล็อกคอินเพื่อเข้าใช้ระบบ โปรแกรมที่พัฒนาขึ้นสามารถรับ-ส่งเอกสารระหว่างผู้ใช้งานในระบบ ติดตามได้ว่าเอกสารมีการเปิดอ่านและตอบกลับหรือยัง สามารถจัดเก็บเอกสารที่ผ่านการอนุมัติและรับทราบได้

อังษณา (2545) ทำการพัฒนาระบบคลังเอกสารบนเครือข่ายภายในองค์กร เป็นระบบที่พัฒนาขึ้นมาเพื่อใช้ในการควบคุม ดูแล และจัดการเอกสารต่าง ๆ โดยทำการพัฒนาระบบเครือข่ายในองค์กร ใช้ระบบจัดการฐานข้อมูล MySQL และใช้โปรแกรมภาษา ASP เป็นเครื่องมือทางด้านซอฟต์แวร์ การทำงานของระบบแบ่งเป็น 5 ส่วน คือ ส่วนแรกเป็นการนำเอกสารเข้ามาเก็บในระบบ เอกสารที่จัดเก็บมี 2 แบบ คือ เอกสารที่เป็นข้อความและเอกสารที่เกิดจากการสแกน โดยมีการเก็บรายละเอียดเกี่ยวกับเอกสารไว้ในฐานข้อมูล ส่วนแฟ้มเอกสารจะจัดเก็บไว้ในสารบบของระบบแฟ้มข้อมูล มีการกำหนดความสัมพันธ์ของเอกสาร การกำหนดคำค้นประจำเอกสาร และการกำหนดสิทธิ์การใช้งานเอกสาร ส่วนที่สองเป็นการนำเอกสารที่แก้ไขแล้วกลับเข้ามาจัดเก็บในระบบโดยอาจจะเดิมหรือจัดเก็บเป็นชุดใหม่ ส่วนที่สี่เป็นการลบเอกสารออกจากระบบ ส่วนสุดท้ายเป็นการสืบค้นเอกสาร ปัญหาที่พบในการพัฒนาคือ การอัปโหลดไฟล์ขนาดใหญ่ทำให้ระบบทำงานช้า ดังนั้น ควรจะเปลี่ยนรูปแบบของไฟล์ให้เป็นประเภทที่กินเนื้อที่หน่วยความจำน้อยก่อนการอัปโหลดไฟล์ แนวทางในการพัฒนานั้น ควรพัฒนาระบบการรับ-ส่งเอกสาร เพื่อให้เป็นระบบงานเอกสารที่สมบูรณ์แบบมากยิ่งขึ้น