บทที่ 2

แนวความคิดและเครื่องมือที่ใช้

2.1 ระบบสารนิเทศภูมิศาสตร์ (Geographic Information System:GIS)

ระบบสารนิเทศภูมิศาสตร์ คือ ระบบงานคอมพิวเตอร์ที่สามารถจัดเก็บ แก้ไข ปรับปรุง เปลี่ยนแปลง แสดงข้อสนเทศทางภูมิศาสตร์ที่อ้างอิงจากพื้นผิวโลก (Environmental System Research [ESRI], 1989)

2.1.1 โครงสร้างหลักของซอฟท์แวร์ระบบสารนิเทศภูมิศาสตร์ ประกอบด้วย

1) โปรแกรมสำหรับการนำเข้าข้อมูล (Data entry) เป็นโปรแกรมที่ใช้สำหรับบันทึกข้อมูลเข้าสู่คอมพิวเตอร์ โดยเฉพาะข้อมูลกราฟิกสามารถนำเข้าได้หลายวิธี อาทิ เช่น การแปลงเป็นเชิงตัวเลข (Digitize) การกราดตรวจ (Scan) เป็นต้น รวมถึงการแก้ไขข้อมูลข้อมูลกราฟิกและข้อมูลลักษณะประจำ

2) โปรแกรมการจัดการข้อมูล (Data manipulation) เป็นโปรแกรมที่ใช้สำหรับจัดการข้อมูลในลักษณะต่าง ๆ เช่น การวิเคราะห์ข้อมูล, การวางซ้อนแผนที่ (Overlay), การสร้างแบบจำลองภูมิประเทศ (Terrain modelling) และการสอบถามข้อมูล (Data query)

3) โปรแกรมการแสดงข้อมูล (Data display) เป็นโปรแกรมที่ใช้สำหรับแสดงข้อมูลกราฟิกบนจอคอมพิวเตอร์ รวมถึงการสร้างแผนที่ในรูปของ Plot file

2.1.2 โครงสร้างฐานข้อมูลระบบสารนิเทศภูมิศาสตร์

NCGIA (1990) ได้กล่าวถึงโครงสร้างฐานข้อมูลระบบสารนิเทศเทศภูมิศาสตร์ สามารถแบ่งออกได้เป็น 2 ประเภท คือ ข้อมูลกราฟิก และข้อมูลลักษณะประจำ

1. ข้อมูลกราฟิก (Graphic data) แบ่งออกได้เป็น 2 ประเภทตามโครงสร้างข้อมูลกราฟิกที่จัดเก็บ ดังนี้

ก. โครงสร้างข้อมูลแบบเวคเตอร์ (Vector model) ในโครงสร้างข้อมูลแบบเวคเตอร์ ข้อมูลทางพื้นที่จะถูกจัดเก็บอยู่ใน 4 รูปแบบ ดังนี้

1) จุด (Point) คือ มีค่าพิกัดตำแหน่งในแนวแกน X และแกน Y เพียงหนึ่งค่า

2) เส้น (Line) คือ ชุดของค่าพิกัดตำแหน่งมากกว่าหนึ่งค่า โดยมีจุดเริ่มต้นและจุดปลาย และอาจมีจุดระหว่างจุดทั้งสองในกรณีที่ไม่ใช่เส้นตรง มีความยาวแต่ไม่มีพื้นที่

3) รูปหลายเหลี่ยม (Polygon) คือ มีค่าพิกัดตำแหน่งที่ต่อเนื่องกันโดยมีจุดเริ่มต้นและจุดปลายอยู่ที่ตำแหน่งเดียวกัน มีทั้งความยาว (เส้นรอบรูป) และพื้นที่

4) พื้นผิว (Surface) คือ รูปหลายเหลี่ยมที่มีทั้งความยาว (เส้นรอบรูป) พื้นที่ และค่าระดับความสูงด้วย

ข้อมูลแบบเวคเตอร์ แสดงไว้ดังรูปที่ 2.1(ก)

ข. โครงสร้างข้อมูลแบบแรสเตอร์ (Raster model) โครงสร้างข้อมูลแบบแรสเตอร์ ข้อมูลถูกจัดเก็บอยู่ในตารางกริด ซึ่งมีคุณสมบัติดังนี้

1) ตารางกริด มีจุดเริ่มต้นที่มุมบนซ้าย

2) แต่ละช่องกริด (Cell) บรรจุข้อมูลได้เพียงค่าเดียว

3) ช่องกริดแต่ละช่องจะมีค่าพิกัดตำแหน่ง

4) ตารางกริด 1 ชุด เรียกว่า ชั้นข้อมูล (Layer)

5) ฐานข้อมูลหนึ่งอาจประกอบด้วยชั้นข้อมูลหลาย ๆ ชั้น เช่น ชั้นข้อมูลของชนิดที่ดิน, ชั้นข้อมูลของการใช้ที่ดิน, ชั้นข้อมูลของความลาดเอียง เป็นต้น

ข้อมูลแบบแรสเตอร์แสดงไว้ในรูปที่ 2.1(ข)

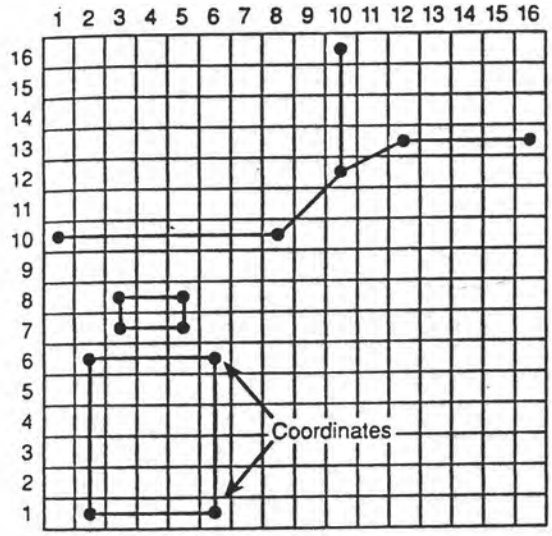
2. ข้อมูลลักษณะประจำ (Attribute data) แบ่งออกได้เป็น 4 ประเภทตามชนิดของแบบจำลองฐานข้อมูล (Model) ได้แก่

ก. แบบจำลองฐานข้อมูลแบบแตกสาขา (Hierarchical model)

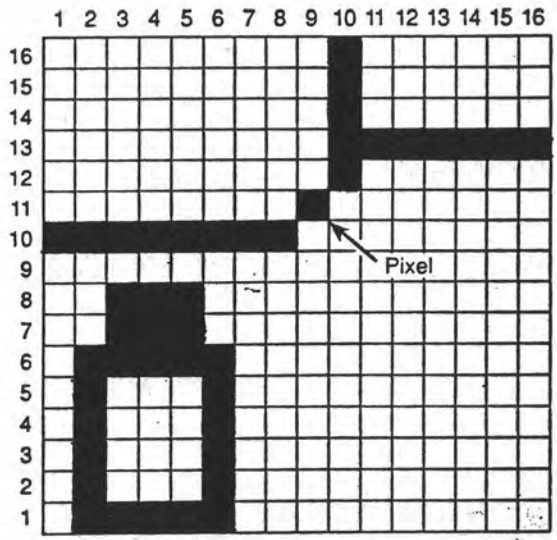
ข. แบบจำลองฐานข้อมูลแบบเครือข่าย (Network model)

ค. แบบจำลองฐานข้อมูลเชิงสัมพันธ์ (Relational model)

ง. แบบจำลองฐานข้อมูลเชิงวัตถุ (Object oriented model)



รูปที่ 2.1(ก) โครงสร้างข้อมูลแบบเวคเตอร์



รูปที่ 2.1(ข) โครงสร้างข้อมูลแบบแรสเตอร์

ระบบสารสนเทศภูมิศาสตร์ส่วนใหญ่ จะมีโครงสร้างฐานข้อมูลลักษณะประจำชนิดแบบจำลองฐานข้อมูลเชิงสัมพันธ์ (Healey, n.d.)

2.2 ซอฟท์แวร์ พีซี อาร์ค อินโฟ (PC ARC/INFO)

2.2.1 ชุดคำสั่งหรือโปรแกรมต่าง ๆ ใน พีซี อาร์ค อินโฟ

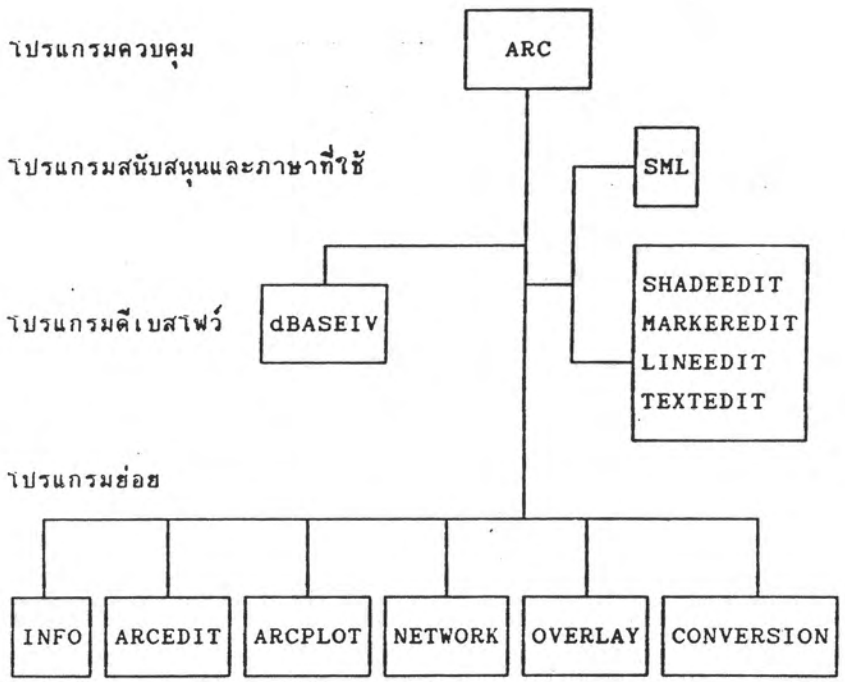
ESRI (1989) ผู้ผลิตซอฟท์แวร์ พีซี อาร์ค อินโฟ ได้อธิบายเกี่ยวกับโปรแกรมดังกล่าวว่า เป็นโปรแกรมประยุกต์ที่ใช้ทำงานด้านระบบสารนิเทศภูมิศาสตร์ สามารถใช้งานได้บนเครื่องไอบีเอ็ม พีซี (IBM PC) หรือบนเครื่อง Compatible 80286 ขึ้นไป โปรแกรมนี้ประกอบขึ้นด้วย โปรแกรมย่อย ๆ หลายโปรแกรม โดยมีโปรแกรมควบคุม (Main program) ควบคุมโปรแกรมย่อย และสามารถใช้งานร่วมกับโปรแกรมดีเบสโฟร์ (dBASE IV) หรือโปรแกรมฟ็อกซ์โปร (Foxpro) ทำให้ผู้ใช้สามารถสร้าง ปรับปรุงเปลี่ยนแปลง และเรียกค้นข้อมูลลักษณะประจำ (Attribute data) ได้ ดังรูปที่ 2.2

ในการวิจัยครั้งนี้ ผู้วิจัยได้ใช้โปรแกรม พีซี อาร์ค อินโฟ ทำการวิจัยเพียงบางส่วนเท่านั้น โปรแกรมที่ใช้ในการวิจัย ได้แก่

1) โปรแกรม พีซี อาร์ค อินโฟ สตาร์ทเตอร์คิท (PC ARC/INFO STARTERKIT) เป็นโปรแกรมควบคุม (Main program) ประกอบด้วยคำสั่งเข้าสู่โปรแกรมย่อย (Subprogram) และฟังก์ชั่นต่าง ๆ ดังนี้

- การนำเข้าข้อมูลกราฟิกด้วยวิธีการต่าง ๆ
- การตรวจสอบข้อผิดพลาดข้อมูลกราฟิก (Error discovery)
- การจัดการแฟ้มข้อมูล (File management) และสร้างพื้นที่การทำงาน (Workspace)
- การทำโปรเจคชั่น (Projection)
- การแปลงค่าพิกัด (Transformation)
- การสร้างตารางข้อมูลลักษณะประจำ (Attribute table)
- การส่งข้อมูลระหว่างเครื่องไมโครคอมพิวเตอร์ กับ เครื่องมินิคอมพิวเตอร์
- การสร้างแผนที่ (Plot file) และการพิมพ์แผนที่

2) โปรแกรม พีซี อาร์คอีดิต (PC ARCEDIT) เป็นโปรแกรมย่อย มีฟังก์ชั่นต่าง ๆ ดังนี้



รูปที่ 2.2 โครงสร้างโปรแกรม พีซี อาร์ค อินโฟ

- การแก้ไขข้อมูลกราฟิก (Graphic data)
- การแก้ไขข้อมูลในตารางข้อมูลลักษณะประจำที่สัมพันธ์กับข้อมูลกราฟิก (Feature attribute table) ที่กำลังแก้ไขอยู่
- การแสดงภาพประกอบซ้อน (Backcoverage)
- การคัดลอกข้อมูลกราฟิกจาก Coverage หนึ่ง ไปยังอีก Coverage

3) โปรแกรม พีซี อาร์คพล็อต (PC ARCPLOT) เป็นโปรแกรมย่อย มีฟังก์ชั่นต่าง ๆ ดังนี้

- การแสดงข้อมูลกราฟิกหรือข้อมูลลักษณะประจำที่ได้สร้างไว้แล้ว ในลักษณะโต้ตอบ (Interactive) กับผู้ใช้
- การกำหนดสัญญลักษณ์ (Symbol), ตำแหน่งภาพ, มาตราส่วน (Scale), หมุนภาพ (Rotate) หรือการกำหนดขนาด สี แบบ ของตัวอักษรหรือส่วนประกอบภาพอื่น ๆ ของข้อมูลเชิงพื้นที่ โดยผลลัพธ์ที่ได้อาจแสดงบนจอภาพ หรือสร้างเป็นแฟ้มข้อมูลสำหรับการส่งพิมพ์ภาพ (Plot file)

4) ภาษา SML (PC ARC/INFO's Simple Macro Language) คือ ชุดคำสั่งซึ่งเป็นภาษาที่ใช้ในการเขียนโปรแกรมอย่างง่ายสำหรับสร้างโปรแกรมต่าง ๆ มีลักษณะเหมือนกับภาษาระดับสูง เช่น การกำหนดตัวแปร การควบคุมการทำงานของโปรแกรม การประเมินผลทางคณิตศาสตร์ การนำเข้าข้อมูลและแสดงผลลัพธ์ ผู้ใช้สามารถเขียนโปรแกรมด้วยภาษา SML เพื่อใช้ในการทำงานต่าง ๆ ดังนี้

- การทำงานต่าง ๆ โดยอัตโนมัติ
- การสร้างคำสั่งสำหรับตัวเองขึ้นมาใช้
- การจัดทำส่วนช่วยเหลือสำหรับผู้ใช้เริ่มต้น หรือการทำงานที่ต้องกำหนดค่าลงไปในในคำสั่ง
- การสร้างเมนูใช้งานตามความต้องการของผู้ใช้

การเขียนโปรแกรม SML ทำได้โดยการใช้โปรแกรม Text editor ต่าง ๆ ช่วยในการเขียนโดยให้มีนามสกุลแฟ้มข้อมูลโปรแกรมเป็น .SML โปรแกรม SML ที่ถูกสร้างขึ้นสามารถทำงานได้ในทุก ๆ โปรแกรมย่อยของโปรแกรม พีซี อาร์ค อินโฟ

การเรียกใช้โปรแกรม SML สามารถเรียกใช้ได้ใน 3 วิธี คือ

- พิมพ์คำสั่ง ARC ตามด้วยชื่อโปรแกรม SML
- พิมพ์ &RUN ตามด้วยชื่อโปรแกรม SML

- พิมพ์ @ ตามด้วยชื่อโปรแกรม SML

ข้อจำกัดของภาษา SML มีดังนี้

- ตัวแปร มีได้ไม่เกิน 50 ตัว และมีชื่อตัวแปรเป็นตัวเลข 1 ถึง 50
- ตัวแปรแบบ Character string มีขนาดไม่เกิน 64 ตัวอักษร
- ตัวแปรแบบ Numeric เป็นตัวเลขที่มีจุดทศนิยมคงที่เหมือนกับภาษาฟอร์แทน คือ 17.5 หมายถึงมีจำนวนเต็ม 11 หลัก และมีจุดทศนิยม 5 ตำแหน่ง

2.2.2 แฟ้มข้อมูลกราฟิก

Coverage เป็นชื่อเรียกแผ่นข้อมูลกราฟิกหรือแผนที่ เมื่อผู้ใช้สั่งสร้าง Coverage ซอฟท์แวร์จะสร้างเป็นแฟ้มสารบบภายใต้ชื่อเดียวกับ Coverage ให้โดยอัตโนมัติ แฟ้มสารบบจะประกอบด้วยแฟ้มย่อย ๆ ที่จำเป็นต่อการทำงานของซอฟท์แวร์ ข้อมูลกราฟิกของแต่ละ Coverage จะเก็บเป็นอิสระต่อกัน (ESRI, 1989) ดังแสดงไว้ในรูปที่ 2.3

2.2.3 สาลักษณ์ข้อมูลกราฟิก (Feature)

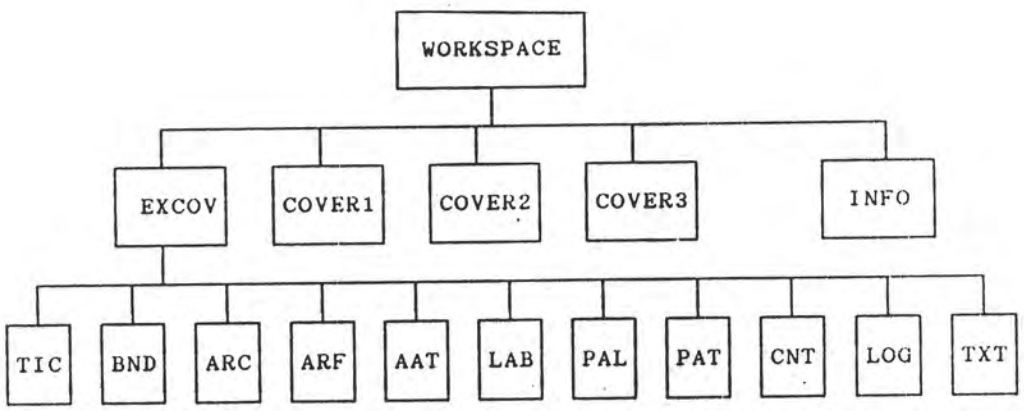
สาลักษณ์ข้อมูลกราฟิก (Feature) ของซอฟท์แวร์ พีซี อาร์ค อินโฟ สามารถจำแนกออกได้เป็น 2 กลุ่ม ตามความสำคัญ (ESRI, 1989) คือ

ก. สาลักษณ์ปฐมภูมิ (Primary features) คือ องค์ประกอบพื้นฐานของภาพกราฟิก ได้แก่ จุด เส้น บัพ และรูปหลายเหลี่ยม แบ่งออกได้เป็น 4 ลักษณะ คือ

1) จุด คือ ข้อมูลที่แสดงด้วยค่าพิกัด 1 ค่า จุดหนึ่งจุดอาจหมายถึง

- จุด (Point) แต่ละจุดจะมีรหัสประจำตัว (User_ID) ที่ผู้ใช้กำหนดขึ้น
- จุดป้าย (Label Point) ของรูปหลายเหลี่ยม ซึ่งกำกับด้วยรหัสประจำตัวที่ผู้ใช้กำหนดขึ้น
- ใช้เป็นตำแหน่งที่วางข้อความกำกับ (Annotation) ของรูปหลายเหลี่ยม ดังนั้น Label Point จึงแทนได้ทั้งจุดและรูปหลายเหลี่ยม การเก็บบันทึกข้อมูลทั้ง 2 ประเภท ควรที่จะแยกกันเก็บคนละ Coverage

2) เส้น (Arcs) ใช้แทนสาลักษณ์ (Lines feature) หรือเส้นขอบเขต



TIC-Tic coordinates and IDs
BND-Coverage minimum and maximum coordinates
ARC-Arc coordinates and topology
ARF-Arc cross-reference file
AAT-Arc Attribute Table
LAB-Label point coordinates and topology
PRF-Polygon/Point cross-reference file
PAT-Polygon/Point Attribute Table
CNT-Polygon centroid table
LOG-Cov./workspace history file
PBX-Polygon index pointer file
PFF-Polygon Filter File
PAL-Polygon topology
TBN-Annotation index file
TBX-Annotation index pointer file
TOL-Coverage processing tolerances
TXT-Coverage annotation features
XBN-Point index file
XBX-Point index pointer file
ABN-Arc Index file
XBX-Point index pointer file
MSK-Edit area masks
PBN-Polygon index file

รูปที่ 2.3 แฟ้มข้อมูลต่าง ๆ ภายใน Coverage

ของรูปหลายเหลี่ยม (Borders of polygon) ในหนึ่งเส้นอาจประกอบด้วย เส้นย่อยหลาย ๆ เส้นที่ต่อเนื่องกัน แต่ละเส้นจะมีรหัสประจำตัวที่ผู้ใช้กำหนดขึ้น (User_ID)

3) บัพ (Node) คือ จุดเริ่มต้นหรือจุดปลายของเส้น

4) รูปหลายเหลี่ยม (Polygon) ประกอบด้วย Arc เส้นเดียวกันหรือหลายเส้นต่อกันเป็นเส้นขอบเขต โดยมี จุดป้าย (Label Point) อยู่ภายใน แต่ละจุดป้าย (Label Point) จะมีรหัสประจำตัวที่ผู้ใช้กำหนดขึ้น (User_ID)

ข. สาลักษณ์ทุติยภูมิ (Secondary Features) แบ่งออกได้เป็น 3 รูปแบบ คือ

1) จุดควบคุม (Control Point) เรียกว่า Tic ใช้เป็นจุดอ้างอิงกลุ่มข้อมูลกราฟิกที่จะทำการบันทึกข้อมูล หรือเป็นตัวควบคุมตำแหน่งพิกัดของกลุ่มข้อมูลกราฟิกซึ่งอาจเป็นระบบพิกัดมาตรฐานทั่วไปก็ได้ แต่ละ Tic จะมีรหัสประจำตัวที่ผู้ใช้กำหนดขึ้น

2) ขอบเขตภาพ (boundary features) เรียกว่า BND ใช้เป็นตัวบอกขอบเขตของกลุ่มข้อมูลกราฟิก โดยการเก็บค่าพิกัดต่ำสุดและสูงสุดของภาพไว้

3) ข้อความกำกับ (Annotation) ข้อมูลนี้ใช้ในการแสดงผล (Display) เท่านั้น จะไม่นำมาใช้ในการวิเคราะห์

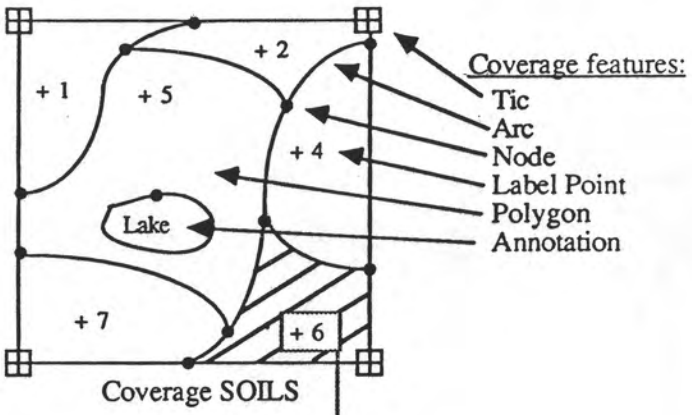
สาลักษณ์ข้อมูลกราฟิกของซอฟท์แวร์ พีซี อาร์ค อินโฟ แสดงไว้ดังรูปที่ 2.4

2.2.4 ลักษณะการจัดเก็บข้อมูลลักษณะประจำของโปรแกรม อาร์ค อินโฟ

ข้อมูลลักษณะประจำที่สัมพันธ์กับข้อมูลกราฟิก จะถูกจัดเก็บอยู่ในรูปของตารางซึ่งโปรแกรมสร้างขึ้น หลังจากผู้ใช้ ใช้คำสั่งสร้างความสัมพันธ์เชิงโทโพโลยีให้กับข้อมูลกราฟิก เรียกว่า ข้อมูลตารางลักษณะประจำของสาลักษณ์ (Features attribute tables) โดยมีรหัสประจำตัวภายในที่โปรแกรมกำหนดให้ (Internal_ID) เป็นตัวเชื่อมโยงกับสาลักษณ์ทางกราฟิกแต่ละตัว สำหรับผู้ใช้งานจะใช้รหัสประจำตัวที่ผู้ใช้กำหนดขึ้น (User_ID) ในระหว่างการบันทึกข้อมูลกราฟิก ตารางข้อมูลลักษณะประจำมีประเภทต่าง ๆ ดังนี้

1) ตารางข้อมูลลักษณะประจำของรูปหลายเหลี่ยม (Polygon attribute table)

2) ตารางข้อมูลลักษณะประจำของจุด (Point attribute table) มีโครงสร้างของตารางเหมือนกับตารางข้อมูลลักษณะประจำของรูปหลายเหลี่ยม แต่ข้อมูลในเขตข้อมูล (Item) AREA และ PERIMETER มีค่าเท่ากับ 0



Feature attribute table SOILS.PAT

RECNO	AREA	PERIMETER	SOILS#	SOILS-ID	SOIL	CLASS	SUITABILITY
1	- 36.0	24.0	1	0	—	—	—
2	3.0	9.0	2	1	A3	113	HIGH
3	2.5	8.5	3	2	C6	95	LOW
4	15.0	15.0	4	3	B7	212	MODERATE
5	4.0	8.5	5	4	B13	201	MODERATE
6	2.0	4.5	6	5	Z22	86	LOW
7	5.5	12.0	7	6	A6	77	HIGH
8	4.0	7.0	8	7	A1	117	LOW

รูปที่ 2.4 สาลักษณ์ข้อมูลกราฟิกของโปรแกรม พีซี อาร์ค อินโฟ

3) ตารางข้อมูลลักษณะประจำของเส้น (Arc attribute table)

4) ตารางข้อมูลลักษณะประจำของพิกัดของจุดควบคุมและเลขประจำตัว (Tic coordinates and IDS)

5) ตารางข้อมูลพิกัดต่ำสุดสูงสุดของคอฟเวอร์เลจ (Coverage minimum - and maximum coordinates)

โครงสร้างตารางข้อมูลลักษณะประจำของสาลักษณ์ข้อมูลกราฟิก แสดงไว้ในรูปที่ 2.5 (ESRI,1989)

2.2.5 ข้อจำกัดของซอฟท์แวร์ พีซี อาร์ค อินโฟ

ข้อจำกัดของโปรแกรม สามารถแบ่งออกได้เป็น 2 ส่วน ดังนี้

1) ข้อจำกัดทางด้านซอฟท์แวร์

- จำนวนสาลักษณ์ข้อมูลกราฟิก ไม่เกิน 32,000 รูป ต่อ 1 Coverage
- จำนวนจุดควบคุมมีได้ไม่เกิน 5,000 จุด ต่อ 1 Coverage
- จำนวนเส้น มีได้ไม่เกิน 5,000 เส้น ต่อ 1 พื้นที่รูปปิด
- จำนวนจุดบนเส้น (Vertices) มีได้ไม่เกิน 500 จุด ต่อ 1 เส้น
- จำนวนเส้นที่เชื่อมโยงกับบัพ (Node) ไม่เกิน 20 เส้น ต่อ 1 บัพ
- จำนวน Coverage ไม่เกิน 250 Coverage ต่อ 1 พื้นที่การทำงาน (Workspace)
- การรวมข้อมูลกราฟิกเข้าด้วยกันทำได้ไม่เกิน 100 Coverage
- ขนาดระเบียนในตารางข้อมูลลักษณะประจำของสาลักษณ์ (Feature - attribute table) 1 เรคคอร์ด ไม่เกิน 128 เขตข้อมูล (Item) หรือไม่เกิน 4,000 ตัวอักษร (Character)
- ชนิดข้อมูลแบบ dBASE MEMO ควรจะเก็บไว้อีก 1 แฟ้มข้อมูล แยกออกจากตารางข้อมูลลักษณะประจำของสาลักษณ์ เมื่อต้องการเชื่อมโยงความสัมพันธ์ จึงใช้คำสั่ง JOIN

2) ข้อจำกัดทางด้านฮาร์ดแวร์

- โปรแกรม พีซี อาร์ค อินโฟ ต้องการหน่วยความจำอย่างน้อยที่สุด 480 Kbytes (ESRI, 1989)

EXCOV.PAT หรือ EXCOV\PAT.DBF

ITEM NAME	ITEM WIDTH	ITEM TYPE	NUMBER DECIMAL	
AREA	13	N	6	- เนื้อที่ของรูปหลายเหลี่ยม
PERIMETER	13	N	6	- ความยาวของเส้นรอบรูป
EXCOV_	11	N	0	- รหัสประจำตัวภายใน
EXCOV_ID	11	N	0	- รหัสประจำตัวโดยผู้ใช้

$RECNO	AREA	PERIMETER	EXCOV_	EXCOV_ID
1	678.2510	456.7806	1	15
2	345.4977	125.8951	3	8
.				.
.				.
.				.
58	190.6700	79.9880	49	13

EXCOV.AAT หรือ EXCOV\AAT.DBF

ITEM NAME	ITEM WIDTH	ITEM TYPE	NUMBERL DECIMAL	
FNODE-	11	N	0	- หมายเลขจุดเริ่มต้นของเส้น
TNODE_	11	N	0	- หมายเลขจุดปลายของเส้น
LPOLY_	11	N	0	- หมายเลขรูปหลายเหลี่ยมด้านซ้าย
RPOLY_	11	N	0	- หมายเลขรูปหลายเหลี่ยมด้านขวา
LENGTH	13	N	6	- ความยาวของเส้น
EXCOV_	11	N	0	- รหัสประจำตัวภายใน
EXCOV_ID	11	N	0	- รหัสประจำตัวโดยผู้ใช้

$RECNO	FNODE_	TNODE_	LOPY_	RPOLY_	LENGTH	EXCOV_	EXCOV_ID
1	2	3	2	2	5.8254	1	1
2	3	4	2	2	3.3895	2	22
.							.
.							.
.							.
125	8	15	1	2	68.7104	100	78

รูปที่ 2.5 โครงสร้างตารางข้อมูลลักษณะประจำของสาลักษณ์ข้อมูลกราฟิก

EXCOV.TIC หรือ \EXCOV\TIC.DBF

ITEM NAME	ITEM WIDTH	ITEM TYPE	NUMBERL DECIMAL	
IDTIC	11	N	0	- หมายเลขรหัส Tic ที่ผู้ใช้กำหนดให้
XTIC	13	N	6	- ค่าพิกัดทางแกน X
YTIC	13	N	6	- ค่าพิกัดทางแกน Y

$RECNO	IDTIC	XTIC	YTIC
1	1	0.0000	0.0000
2	2	160.0000	0.0000
3	3	0.0000	200.0000
4	4	160.0000	200.0000

EXCOV.BND หรือ \EXCOV\BND.DBF

ITEM NAME	ITEM WIDTH	ITEM TYPE	NUMBERL DECIMAL	
XMIN	13	N	6	- ค่าพิกัดทางแกน X ที่น้อยที่สุด
YMIN	13	N	6	- ค่าพิกัดทางแกน Y ที่น้อยที่สุด
XMAX	13	N	6	- ค่าพิกัดทางแกน X ที่มากที่สุด
YMAX	13	N	6	- ค่าพิกัดทางแกน Y ที่มากที่สุด

$RECNO	XMIN	YMIN	XMAX	YMAX
1	0.0000	0.0000	160.0000	200.000

รูปที่ 2.5(ต่อ) โครงสร้างตารางข้อมูลลักษณะประจำของสาลักษณ์ข้อมูลกราฟิก

2.3 ซอฟท์แวร์ ดีเบสโฟร์ (dBASE IV)

ซอฟท์แวร์ ดีเบสโฟร์ เป็นโปรแกรมประยุกต์ที่ใช้ในการจัดการฐานข้อมูลเชิงสัมพันธ์ ผลิตโดยบริษัท Ashton-Tate โดยนำเอา Query By Example และ Structured Query Lanngauge (SQL) ของบริษัท IBM เข้ามารวมอยู่ด้วย สามารถใช้งานได้ทั้งระบบผู้ใช้เดี่ยว (Single User System) และระบบหลายผู้ใช้ (Multi-User System) (สิทธิชัย ประสานวงศ์, 2533)

2.3.1 การทำงานของซอฟท์แวร์ ดีเบสโฟร์

สิทธิชัย ประสานวงศ์ (2533) ได้กล่าวถึงหน้าที่การทำงานของโปรแกรมดีเบสโฟร์ สามารถแบ่งออกได้เป็น 7 ส่วน หลัก ๆ ดังนี้

1) การจัดการเกี่ยวกับข้อมูลและตารางข้อมูล ได้แก่ การสร้างตารางข้อมูล การแก้ไขโครงสร้างตารางข้อมูล การลบตารางข้อมูล การเพิ่มและลบข้อมูล การแก้ไขข้อมูล การแสดงข้อมูล การเรียงข้อมูล การทำดัชนีข้อมูล การค้นหาข้อมูล เป็นต้น แฟ้มข้อมูลที่ใช้ คือ Database file (.dbf) และ Index file (.ndx)

2) การกำหนดเงื่อนไขด้วย Query By Example (QBE) เป็นการเรียกใช้ข้อมูลตามเงื่อนไขที่กำหนดในตารางสมมติ แฟ้มข้อมูลที่ใช้ คือ Query file (.qbe)

3) การจัดจอภาพ เป็นการจัดรูปแบบของจอภาพตามที่ผู้ใช้กำหนด (Custom screen) เพื่อใช้ในการป้อน แก้ไข หรือการเรียกใช้ข้อมูลจาก Database file แฟ้มข้อมูลที่ใช้ คือ Form file (.scr)

4) การสร้างและพิมพ์แฟ้มรายงาน เป็นส่วนของการจัดการรายงานจาก Database file แฟ้มข้อมูลที่ใช้ คือ Report file (.frm)

5) การสร้างและพิมพ์ป้าย เป็นส่วนของการจัดการป้าย (Label) ในรูปแบบที่ผู้ใช้ต้องการ โดยนำข้อมูลจาก Database file มาใช้ แฟ้มข้อมูลที่ใช้คือ label file (.lbl)

6) การเขียนโปรแกรม เป็นส่วนของการเขียนโปรแกรมเพื่อใช้งานต่าง ๆ แฟ้มข้อมูลที่ใช้คือ Application file (.prg)

7) การใช้ภาษาสอบถาม Structured QueryLangauge (SQL) เป็นส่วนของการใช้ภาษาทางด้านการจัดการฐานข้อมูลแบบเชิงสัมพันธ์ (Relational Database)

2.3.2 โครงสร้างของแฟ้มข้อมูลลักษณะประจำของซอฟท์แวร์ ดีเบสโฟร์

ข้อมูลลักษณะประจำของซอฟท์แวร์ ดีเบสโฟร์ จะจัดเก็บอยู่ในแฟ้มข้อมูลที่มีนามสกุล .dbf ซึ่งแบ่งออกเป็น 6 เขตข้อมูล (Field) ดังตารางที่ 2.1

เขตข้อมูล	ข้อมูล	ขนาดข้อมูล
Character	ใช้เก็บข้อมูลที่เป็นตัวอักษร หรือตัวเลขที่ไม่ได้ใช้ในการคำนวณได้ไม่เกิน 254 ตัว	ไม่เกิน 254 ตัว 254 ไบท์
Numeric	ใช้กับตัวเลขที่มีหลักทศนิยมคงที่ คำนวณได้รวดเร็ว	ไม่เกิน 20 หลัก จำนวนทศนิยมได้ตั้งแต่ 0 - 18 ตำแหน่ง
Float	ใช้กับตัวเลขแบบจุดลอยตัว (floating point) ซึ่งมักอยู่ในรูปของตัวเลขคูณกับ 10 ยกกำลัง ตัวเลขนั้นมีจำนวนมาก	ไม่เกิน 20 หลัก จำนวนทศนิยมได้ตั้งแต่ 0 - 18 ตำแหน่ง
Date	ใช้กับวันที่ เขียนอยู่ในรูป mm/dd/yy	8 ไบท์
Logical	ใช้กับข้อมูลเชิงตรรกศาสตร์ คือ T หรือ Y แทนข้อความจริง F หรือ N แทนข้อความเท็จ	1 ไบท์
Memo	ใช้ในการเก็บข้อความที่มีความยาวมาก โดยแยกเก็บออกเป็นไฟล์ต่าง ๆ จึงมักใช้ในการเก็บส่วนบันทึกข้อความ	เริ่มต้นจะเป็น 10 ตัวอักษร เพิ่มได้ตามข้อความที่พิมพ์เข้าไป

ตารางที่ 2.1 แสดงชนิดและขนาดของข้อมูลในโปรแกรมดีเบสโฟร์

2.3.3 ข้อจำกัดของซอฟท์แวร์ ดีเบสโฟร์

ข้อจำกัดของโปรแกรม ดีเบสโฟร์ มีดังนี้

1) แฟ้มฐานข้อมูล (Database file)
   - จำนวนเรคคอร์ด 1 พันล้านเรคคอร์ด
   - จำนวนไบท์ 2 พันล้านไบท์
   - ขนาดเรคคอร์ด 4,000 ไบท์
   - จำนวนฟิลด์ 255 ฟิลด์
2) แฟ้มดัชนี (Index file)
   - จำนวนดัชนีใน Multiple index file 47 ดัชนี
   - ขนาดของแฟ้ม 16,384 ไบท์
3) การจัดแฟ้ม
   - เปิดแฟ้มพร้อมกันทุกชนิด 99 แฟ้ม
   - เปิดแฟ้มฐานข้อมูล 10 แฟ้ม
   - เปิดแฟ้มฐานข้อมูลแบบข้อความ ต่อ 1 แฟ้มข้อมูล 1 แฟ้ม
   - เปิดแฟ้มดัชนี ต่อ 1 แฟ้มข้อมูล 10 แฟ้ม
   - เปิดแฟ้มจัดจอภาพ ต่อ 1 แฟ้มข้อมูล 1 แฟ้ม
   - เปิดแฟ้มโปรแกรมการทำงาน 1 แฟ้ม

2.4 ระบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database System)

ระบบการจัดการฐานข้อมูลที่ใช้ในระบบสารสนเทศภูมิศาสตร์ในปัจจุบัน ส่วนใหญ่เป็นระบบฐานข้อมูลเชิงสัมพันธ์ ซึ่งคิดขึ้นโดย คอร์ด (E.F.Codd) ในปี ค.ศ. 1970

ฐานข้อมูลเชิงสัมพันธ์ ได้แก่ ฐานข้อมูลที่ประกอบด้วย รีเลชั่น (ตาราง) ต่าง ๆ ที่มีความสัมพันธ์ซึ่งกันและกัน

รีเลชั่น คือ ตาราง 2 มิติที่

- แต่ละช่องของตารางจะบรรจุข้อมูลเพียงค่าเดียว
- ชื่อหัวข้อในแต่ละคอลัมน์มีความแตกต่างกัน ได้แก่ชื่อของข้อมูลลักษณะประจำ
- ค่าข้อมูลที่อยู่ในแต่ละคอลัมน์ ได้แก่ค่าของข้อมูลลักษณะประจำ ที่ระบุไว้ใน

หัวข้อคอลัมน์นั้น ๆ

- การเรียงลำดับคอลัมน์ถือว่าไม่มีความสำคัญ
- ข้อมูลแต่ละแถวจะต้องแตกต่างกัน
- การเรียงลำดับแถวถือว่าไม่มีความสำคัญ

ตัวอย่างฐานข้อมูลเชิงสัมพันธ์แสดงไว้ในรูปที่ 2.6

2.4.1 แนวความคิดหลักเกี่ยวกับฐานข้อมูลเชิงสัมพันธ์

Date (1986) ได้อธิบายแนวความคิดหลักเกี่ยวกับฐานข้อมูลเชิงสัมพันธ์ ดังนี้

1) คีย์หลัก (Primary key) ประกอบด้วย ค่าของหน่วยข้อมูลย่อย 1 ตัว หรือมากกว่า 1 ตัวก็ได้ ที่สามารถใช้เป็นตัวเจาะจงบอกว่าเรากำลังอ้างอิงถึงข้อมูลแถวไหน (Tuple) และค่าของคีย์หลักจะไม่อนุญาตให้เป็นค่าว่าง (Null values)

2) การเชื่อมความสัมพันธ์ (Relational joins) การเชื่อมความสัมพันธ์ระหว่างตาราง สามารถกระทำได้โดยการกำหนดคีย์นอก (Foreign key) อันได้แก่ หน่วยข้อมูลย่อยของตารางอันหนึ่งที่ซ้ำกับหน่วยข้อมูลย่อย (Attribute) ในอีกตารางหนึ่ง

3) รูปแบบนอร์มัล (Normal forms) รูปแบบนอร์มัล คือ การออกแบบระบบฐานข้อมูลเพื่อให้สามารถเรียกใช้ข้อมูลได้อย่างสะดวกและมีประสิทธิภาพ โดยพยายามให้เกิดความซ้ำซ้อนของการเก็บข้อมูลให้น้อยที่สุด

2.4.2 คุณสมบัติของระบบฐานข้อมูลเชิงสัมพันธ์

1) รูปแบบฐานข้อมูลในระดับเชิงตรรกะ จะแทนเป็นตาราง (Table) ที่สามารถเข้าใจได้ง่าย โดยในตารางจะแบ่งเป็นสดมภ์ (Column) และแถว (Row) จุดตัดของแถวและสดมภ์ เรียกว่า หน่วยข้อมูล (Attribute หรือ Field หรือ Item) และภายในหน่วยข้อมูลเก็บค่า (Value) ผู้ใช้จะต้องสามารถเรียกหรือค้นข้อมูลที่ต้องการได้ทุก ๆ ค่า และหากหน่วยข้อมูลใดเกิดไม่แน่ใจจะมีค่าว่าง (Null) ไว้

2) ระบบจะต้องมีภาษาในระดับสูง เพื่อจัดการข้อมูลที่เก็บอยู่ในตาราง โดยผู้ใช้งานเพียงระบุว่า ต้องการข้อมูลไม่ต้องระบุวิธีการประมวลผลระบบจะจัดการเอง ภาษาในระดับสูงนี้อาจเรียกได้ว่า ภาษาในยุคที่สี่ ภาษาฐานข้อมูลอาจแบ่งประโยคคำสั่งแยกตามลักษณะออกเป็นกลุ่ม ๆ กลุ่มแรกเรียกว่า ภาษานิยามข้อมูล (Data Definition Langauge) จะใช้

เพื่อกำหนดโครงสร้างข้อมูล กำหนดตารางข้อมูลเทียมตามทัศนะของผู้ใช้ กลุ่มที่สอง เรียกว่า ภาษาจัดการข้อมูล (Data Manipulation Langauge) เพื่อการดัดแปลงแก้ไขและการเรียกค้นข้อมูล การควบคุมความถูกต้องของข้อมูลและสิทธิของผู้ใช้ด้านความปลอดภัยของข้อมูล เป็นต้น

3) การเชื่อมต่อระหว่างตารางต่าง ๆ ภายในฐานข้อมูล ผู้ใช้จะไม่มีโอกาสได้รู้เห็นเพราะระบบจะปฏิบัติการโดยอัตโนมัติ โดยระบบเลือกวิธีที่ดีที่สุดและได้ผลดีที่สุด

4) การประมวลผลของฐานข้อมูลเชิงสัมพันธ์ จะเป็นการประมวลผลแบบ Set-Oriented กล่าวคือ เมื่อมีการถามหรือเรียกค้นข้อมูล จะได้คำตอบเป็นชุดหรือเป็นเซตหลาย ๆ แถว (Record) กลับมา ซึ่งต่างกับระบบการจัดการฐานข้อมูลสมัยก่อนที่ให้คำตอบครั้งละหนึ่งแถว แล้วมีการอ่านวนรอบเรียกแถวต่อไป ในฐานข้อมูลเชิงสัมพันธ์จะไม่มีการวนรอบ แต่จะมีการใช้งานผ่านตัวกระทำสัมพันธ์พื้นฐาน เช่น SELECT, PROJECT, JOIN และ DEVIDE กับตัวกระทำการเซต เช่น การผนวก (UNION), การตัดกัน (INTERSECT), การทำผลต่าง (DIFFERENCE) และการทำผลคูณ (PRODUCT)

5) ข้อมูลเกี่ยวกับระบบตั้งแต่โครงสร้างฐานข้อมูล สิทธิการใช้ ฯลฯ ต้องเก็บบันทึกในตารางของระบบ เมื่อฐานข้อมูลมีการเปลี่ยนแปลงโครงสร้าง ตารางของระบบนี้จะมีการเปลี่ยนแปลงแก้ไขโดยอัตโนมัติด้วย

6) ฐานข้อมูลแบบสัมพันธ์และโครงสร้างสามารถกระทำการเปลี่ยนแปลงได้โดยอาจเปลี่ยแปลงที่เค้าร่างย่อยของส่วนการมองของผู้ใช้ หรือเปลี่ยนค่าพร้อมกันในหลายแถวหรือหลายสคมก์

7) การเปลี่ยนแปลงระบบฐานข้อมูลในระดับต่ำ จะไม่มีผลกระทบต่อส่วนที่อยู่ในระดับสูงหรือชุดคำสั่งที่ใช้งาน ไม่ว่าจะเป็นการเปลี่ยนในระดับภายในหรือในระดับเชิงแนวคิด

8) ระบบฐานข้อมูลต้องเป็นอิสระไม่ขึ้นกับการจัดเก็บตามข้อกำหนด ความถูกต้องของข้อมูล ไม่ขึ้นกับการควบคุมภาษาระดับต่ำเพื่อมาทำลายความถูกต้องของระบบและไม่ขึ้นกับการกระจายของข้อมูลที่อาจแยกกันอยู่หลาย ๆ แห่ง (ดวงแก้ว สวามิภักดิ์, 2533)

2.4.3 ประเภทของความสัมพันธ์ (Types of relationship)

ประเภทของความสัมพันธ์สามารถแบ่งออกได้ เป็น 3 ชนิด ดังนี้

2.4.3.1 ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One-to-one relationship) ตัวอย่างเช่น แต่ละจังหวัดมีศาลากลางจังหวัดเพียงแห่งเดียวเท่านั้น

2.4.3.2 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม (One-to-many relationship) ตัวอย่างเช่น แต่ละจังหวัดสามารถมีโรงแรมได้หลายแห่ง

2.4.3.3 ความสัมพันธ์แบบกลุ่มต่อกลุ่ม (Many-to-many relationship) ตัวอย่างเช่น ภัตตาคารแต่ละแห่งมีรายการอาหารหลายชนิด และอาหารแต่ละชนิดมีอยู่ในหลายภัตตาคาร (Healey, n.d.)

ตัวอย่างชนิดของความสัมพันธ์แสดงไว้ในรูปที่ 2.7

2.4.4 วิธีการออกแบบฐานข้อมูล

จรณิต แก้วกังวาล (2536) ได้จำแนกวิธีการออกแบบฐานข้อมูลเพื่อนำมาใช้งานภายในองค์การออกเป็น 2 วิธีการใหญ่ ๆ คือ

1) วิธีการอุปนัย (bottom-up หรือ inductive approach) เป็นการออกแบบสร้างฐานข้อมูล โดยวิธีการรวบรวมข้อมูลและหรือโปรแกรมที่มีอยู่แล้วจากหลาย ๆ หน่วยงานในองค์กร แนวคิดพื้นฐานการออกแบบประเภทนี้คือ ถือว่าลักษณะงานของแต่ละหน่วยงานมีความซับซ้อน สมบูรณ์แตกต่างกัน ดังนั้น รูปแบบของฐานข้อมูลที่ดีจึงต้องเป็นรูปแบบที่รวบรวมเอาข้อดีของข้อมูลหรือโปรแกรมต่าง ๆ ที่มีอยู่แล้วในหน่วยงานต่าง ๆ มารวบรวมเป็นรูปแบบขนาดใหญ่ทั้งหมด ข้อเสียของวิธีการนี้คือการรวบรวมวิธีการย่อย ๆ เข้าด้วยกันทำได้ไม่ง่ายนัก และเสียเวลาในการที่จะออกแบบและสร้างระบบฐานข้อมูลที่สมบูรณ์

2) วิธีการนิรนัย (top-down หรือ deductive approach) คือ ผู้ที่ออกแบบฐานข้อมูลต้องมีความเข้าใจข้อมูลทั้งระบบ ศึกษาว่าองค์กรมีข้อมูลอะไรบ้างแล้วจึงนำมาออกแบบเป็นโครงสร้างทั้งหมดของฐานข้อมูล ปัญหาของวิธีการนี้ก็คือ จะต้องได้ผู้ที่ศึกษาและเข้าใจระบบทั้งหมดจริง แต่ก็เป็นวิธีการที่เหมาะสมสำหรับการจัดวางระบบฐานข้อมูลที่มีความยุ่งยากมีความหลากหลายของหน่วยงานต่าง ๆ ภายในองค์กร เช่น มีการให้คำจำกัดความของข้อมูลตัวเดียวกันต่าง ๆ กันออกไป เป็นต้น

2.4.5 ขั้นตอนการออกแบบระบบฐานข้อมูล

จรณิต แก้วกังวาล (2536) ได้แบ่งการดำเนินงานออกแบบระบบฐานข้อมูลออกเป็นขั้นตอนตามลำดับ 4 ขั้นตอน ดังนี้

ตารางภัตตาคาร

รหัสภัตตาคาร REST#	ชื่อภัตตาคาร NAME	ที่อยู่ภัตตาคาร ADDRESS	รหัสจังหวัด TOWN#
1	ศรแดง	153 ถ.ตากสิน	36
2	พีพร้อม	23 ถ.สุขุมวิท	46

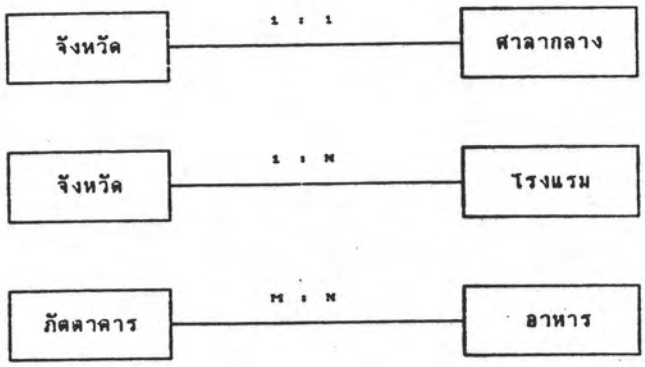
ตารางราคาเครื่องดื่ม

รหัสภัตตาคาร REST#	รหัสเครื่องดื่ม DRINK#	ราคา PRICE#
1	2	30
1	4	25
2	2	20

ตารางเครื่องดื่ม

รหัสเครื่องดื่ม DRINK#	ชื่อเครื่องดื่ม DRINK_NAME
1	กาแฟร้อน
2	ชาร้อน
3	น้ำชิง

รูปที่ 2.6 ตัวอย่างตารางฐานข้อมูลเชิงสัมพันธ์



รูปที่ 2.7 ตัวอย่างชนิดของความสัมพันธ์

1) รวบรวมข้อมูล

2) ทำความเข้าใจกับข้อมูลและการประยุกต์การใช้งาน

3) จัดรูปแบบของข้อมูลเพื่อวางโครงร่างของระบบขั้นต้น (initial conceptual model)

4) วิเคราะห์และลงมือออกแบบระบบที่ต้องการใช้งาน (conceptual model)

ขั้นตอนที่ 1 : การรวบรวมข้อมูล

ในการออกแบบระบบฐานข้อมูล ผู้ออกแบบระบบจะต้องเข้าใจระบบที่ต้องการสร้างทั้งระบบ ต้องรู้ว่ามีข้อมูลอะไรบ้างอยู่ในระบบและต้องการแสดงผลลัพธ์อะไรบ้าง ดังนั้นผู้ออกแบบจึงจำเป็นต้องรวบรวมข้อมูลต่าง ๆ ทั้งหมดที่มีอยู่ในระบบจากผู้ที่เกี่ยวข้องทุกคน หรือถ้าเป็นไปไม่ได้ก็อาจจะต้องเลือกตัวแทนของผู้ที่เกี่ยวข้องแต่ละฝ่ายเข้ามาเป็นผู้ให้ข้อมูลต่าง ๆ

วิธีการที่ใช้ในการรวบรวมข้อมูลมีหลายวิธี เช่น วิธีการใช้แบบสอบถาม สัมภาษณ์ สังเกตการปฏิบัติงานทำงานของผู้ใช้ หรือวิธีการประชุมสัมมนาร่วมกัน เป็นต้น

สิ่งที่ผู้ออกแบบจะต้องศึกษารวบรวมข้อเท็จจริงของระบบประกอบด้วย กฎเกณฑ์การสร้างข้อมูล กฎเกณฑ์การเรียกใช้ข้อมูล วิธีการจัดการข้อมูล กฎเกณฑ์การตรวจสอบความถูกต้องแม่นตรง (validity) ของข้อมูล และกฎเกณฑ์การยกเลิกการใช้ข้อมูลชุดนั้น ๆ

ขั้นตอนที่ 2 : ทำความเข้าใจกับข้อมูล

วิธีการที่นิยมใช้กันมากในการศึกษาเพื่อวางรูปแบบระบบฐานข้อมูล คือ วิธีการแยกแยะความสัมพันธ์ (factoring) ซึ่งอาจแบ่งได้เป็น 4 ลักษณะคือ

2.1 task-data relationship จุดมุ่งหมายของฐานข้อมูลคืออะไร ชิ้นงานผลลัพธ์ที่ต้องการคืออะไร จะต้องใช้ข้อมูลใดบ้างในการออกเอาต์พุตดังกล่าว

2.2 field-field/key-field/key-key relationship การจัดข้อมูลเป็นฟิลด์ต่าง ๆ นั้นมีกี่ฟิลด์ มีฟิลด์ใดที่จะใช้เป็นคีย์หลัก (key) ในการกำหนดรูปแบบหรือลักษณะของระเบียน และถ้ามีมากกว่า 1 ฟิลด์หลัก แต่ละฟิลด์หลักจะมีความสัมพันธ์กันอย่างไร

2.3 field-record relationship ลักษณะการจัดกลุ่มฟิลด์ที่สัมพันธ์กันเป็น

ระเบียนเป็นไปในลักษณะใด ทำอย่างไรจึงจะไม่ซ้ำซ้อน

2.4 file-file relationship ถ้าแยกเก็บข้อมูลไว้ในหลาย ๆ ไฟล์ จะเชื่อมโยงความสัมพันธ์ระหว่างไฟล์ได้อย่างไร ใช้ฟิลด์หลักใดบ้าง

โดยสรุป ผู้ออกแบบจะต้องพยายามทำความเข้าใจกับระบบในประเด็นต่าง ๆ ดังต่อไปนี้

- แหล่งข้อมูล (จะรวบรวมข้อมูลเพื่อจัดเก็บและใช้ฐานข้อมูลได้จากที่ใด)
- จะคงค่าของข้อมูลเหล่านี้ในฐานข้อมูลนานเท่าใด เมื่อใดจะลบทิ้ง
- ข้อมูลแต่ละตัวแต่ละชุดมีความสัมพันธ์กับข้อมูลตัวอื่น ๆ หรือชุดอื่น ๆ อย่างไร โดยใช้วิธีการแยกแยะความสัมพันธ์
- ข้อมูลเหล่านี้จะทำให้เกิดข้อสนเทศอะไรได้บ้างและอย่างไร
- กระบวนการแปรผลข้อมูลเป็นอย่างไร คำจำกัดความของข้อมูลแต่ละตัวคืออะไร
- ข้อมูลเหล่านี้มีโอกาสถูกเปลี่ยนแปลงบ่อยแค่ไหน อย่างไร
- ข้อมูลเหล่านี้จะถูกจัดเก็บอย่างปลอดภัย ไม่ถูกทำลายโดยอุบัติเหตุหรือสาเหตุอื่น ๆ ได้อย่างไร
- มีกฎเกณฑ์ในการตรวจสอบความถูกต้องของข้อมูลได้อย่างไรบ้าง

<u>ขั้นตอนที่ 3 : การกำหนดรูปแบบขั้นต้นของระบบ</u>

ขั้นตอนถัดมาหลังจากที่ผู้ออกแบบตกลงกันเรียบร้อยในเรื่องของฟิลด์แต่ละไฟล์ (table) ที่จะใช้ภายในระบบ ผู้ออกแบบจะต้องกำหนดร่างของระบบคร่าว ๆ ว่า ภายในระบบจะมีการส่งผ่านข้อมูล และแสดงผลลัพธ์เป็นเอาต์พุต (input-output) ในลักษณะใด นั่นคือเป็นการกำหนดรูปแบบขั้นต้น (initial model) ของระบบ

<u>ขั้นตอนที่ 4 : วิเคราะห์รายละเอียดและลงมือสร้างระบบ</u>

หลังจากที่ได้รูปแบบของระบบขั้นต้นแล้ว ผู้ออกแบบจะต้องศึกษาอย่างละเอียดว่าจะต้องมีการจัดการอย่างไรตามลำดับขั้นตอน แล้วจึงวางรูปแบบของระบบในรายละเอียดเป็นขั้นสุดท้าย

หลักพื้นฐานในการออกแบบระบบ (design principle) ดังนี้

- Long-term adaptability สามารถปรับปรุงแก้ไขในระยะยาวได้ง่าย โดยไม่กระทบกระเทือนต่อโครงสร้างใหญ่ทั้งหมดที่สร้างไว้แล้วมากนัก

- Short-term flexibility สะดวกและง่ายในการติดตั้งและใช้งาน

- Completeness มีคำอธิบาย ความหมายและความสัมพันธ์ของข้อมูลชัดเจน ทำให้ผู้เข้าใจระบบได้ดี

- Parsinomy ไม่มีความซ้ำซ้อนในการออกแบบ ไม่มีข้อจำกัดในการสร้างความสัมพันธ์ระหว่างข้อมูล

- History มีคำอธิบายบอกวิธีการตรวจสอบติดตาม (trace) ข้อมูลหรือเรคอร์ดที่สูญหาย หรือบอกความหมายของข้อมูลที่อาจแปรเปลี่ยนไปตามเวลาที่ใช้ระบบฐานข้อมูลนั้น (เป็นเวลานาน)

- Local properties แฟ้มข้อมูลมีความถูกต้อง มั่นคงและสมบูรณ์ เมื่อปรับเปลี่ยน (update) ข้อมูลตัวใดก็ไม่กระทบกระเทือนโครงสร้างทั้งหมด

- Comparability ข้อมูลที่เก็บไว้มีค่าตรงกับในคำจำกัดความของฟิลด์และระเบียนและสามารถเปรียบเทียบกันได้ถูกต้อง

ในอีกแนวทางหนึ่ง ดวงแก้ว สวามิภักดิ์ (2533) ได้แบ่งขั้นตอนการออกแบบฐานข้อมูลออกเป็น 5 ขั้นตอน ดังนี้

1. เปลี่ยนรูปแบบความต้องการให้อยู่ในรูปความสัมพันธ์
2. นอร์มัลไลซ์รีเลชั่น
3. กำหนดฟิลด์ที่จะเป็นคีย์ต่าง ๆ และคุณสมบัติของคีย์แต่ละตัว
4. พิจารณาข้อจำกัด และกฎเกณฑ์อื่น ๆ
5. นำผลที่ได้จากการออกแบบในขั้นตอนแรกมาผนวกกัน