บทที่ 3

แนวคิดและทฤษฎี

3.1 แนวคิดเกี่ยวกับสารสนเทศ

- ข้อมูล (Data) หมายถึง ข้อเท็จจริงต่าง ๆ ที่มีอยู่ในธรรมชาติเป็นกลุ่มสัญลักษณ์แทนปริมาณหรือการกระทำต่าง ๆ ที่ยังไม่ผ่านการประมวลผล

- สารสนเทศ (Information) หมายถึงข้อมูลที่ได้ประมวลผลแล้วด้วยวิธีการต่างๆ เป็นความรู้ที่ต้องการสำหรับใช้ทำประโยชน์ สามารถนำไปกระทำกิจกรรมใดกิจกรรมหนึ่งโดยเฉพาะได้

- ความสัมพันธ์ของข้อมูล สารสนเทศ และระบบสารสนเทศ

ระบบสารสนเทศประกอบด้วยส่วนนำเข้า ส่วนกระบวนการหรือส่วนประมวลผลและส่วนผลลัพธ์ ซึ่งตัวข้อมูลเป็นวัตถุดิบในส่วนนำเข้า เพื่อประมวลผลข้อมูลด้วยวิธีการต่าง ๆ และได้สารสนเทศเป็นผลลัพธ์ของระบบสารสนเทศ (จีราภรณ์ รักษาแก้ว, 2535)

คุณสมบัติของสารสนเทศ (Senn, 1989)

3.1.1 ความถูกต้องในที่นี้หมายถึงอัตราส่วนสารสนเทศที่ถูกต้องกับจำนวนสารสนเทศที่ผลิตขึ้นทั้งหมดในช่วงเวลาหนึ่ง ยิ่งสารสนเทศมีความถูกต้องมากเท่าใด ก็ยิ่งมีคุณค่าสำหรับผู้บริหารมากขึ้น

3.1.2 ความทันต่อการใช้งาน สารสนเทศที่ดีนั้นมีความถูกต้องอย่างเดียวยังไม่เพียงพอ แต่ต้องได้รับให้ทันต่อการใช้ประโยชน์ด้วยความหมายของคำว่าทันต่อเหตุการณ์ จะขึ้นอยู่กับความพอใจของผู้ใช้ในแต่ละสถานการณ์

3.1.3 ความสมบูรณ์ สารสนเทศที่สมบูรณ์เกิดจากการรวบรวมข้อเท็จจริงหรือข้อมูลที่มีอยู่อย่างกระจัดกระจายในองค์การได้ในปริมาณที่เพียงพอต่อการผลิตสารสนเทศนั้น

3.1.4 ความกะทัดรัดของสารสนเทศ สารสนเทศที่ดีควรมีความกะทัดรัดได้ใจความที่สมบูรณ์ในตัวเอง สามารถแสดงสาระที่สำคัญตามที่ผู้บริหารต้องการได้ครบถ้วน ซึ่งอาจจะจัดทำ

ได้โดยการสรุปเฉพาะสิ่งที่ผู้บริหารต้องการ

3.1.5 ตรงกับความต้องการ หมายถึงสารสนเทศนั้น คือ สารสนเทศที่ต้องการจะรู้ เป็นสารสนเทศที่สามารถสื่อความหมายให้เกิดการกระทำ ความรู้ และความเข้าใจต่อผู้บริหาร

3.2 ระบบสารสนเทศเพื่อการจัดการ (Management Information System:MIS)

คือระบบสารสนเทศที่ทำหน้าที่ให้สารสนเทศหรือข่าวสาร เพื่อช่วยในการตัดสินใจของผู้บริหาร ในเรื่องของกระบวนการจัดการองค์การ เช่น การวางแผน การจัดองค์การและการควบคุม เพื่อให้องค์การสามารถดำเนินการไปตามวัตถุประสงค์ที่ตั้งไว้ หน้าที่หลักของระบบสารสนเทศเพื่อการจัดการประกอบด้วย

- ให้สารสนเทศเพื่อช่วยในการตัดสินใจของผู้บริหาร
- ให้สารสนเทศแก่ผู้บริหารทุกระดับได้
- ให้สารสนเทศเพื่อช่วยในการแก้ปัญหาทุกรูปแบบของปัญหา
- ให้สารสนเทศที่รวดเร็วและเหมาะกับการใช้งาน

3.2.1 ลักษณะโครงสร้างของระบบสารสนเทศเพื่อการจัดการ (Press, 1990)

ระบบสารสนเทศเพื่อการจัดการ แบ่งระดับการใช้สารสนเทศเป็น 3 ระดับ ดังนี้

3.2.1.1 สารสนเทศเพื่อการวางแผนและตัดสินใจ (Startegic Information) เป็นการใช้สารสนเทศเพื่อกำหนดการวางแผน และกำหนดนโยบายระยะยาวสำหรับผู้บริหารระดับสูงโดยสารสนเทศต้องสามารถตอบสนองความต้องการได้ทุกรูปแบบ(Unstructure)

3.2.1.2 สารสนเทศเพื่อการบริหาร (Managerial Information) เป็นสารสนเทศที่ใช้ในการวางแผน และนโยบายในการบริหาร (Tactical planning and policy implement)

3.2.1.3 สารสนเทศในระดับปฏิบัติการ (Operational Information) ประกอบด้วยสารสนเทศที่ใช้สนับสนุน ควบคุมการปฏิบัติการประมวลผลรายวัน ลักษณะของสารสนเทศมีการกำหนดรูปแบบล่วงหน้า และระยะเวลาที่แน่นอน (Structured)

3.2.2 ลักษณะและองค์ประกอบพื้นฐานของระบบสารสนเทศเพื่อการจัดการ

3.2.2.1 เป็นระบบที่เชื่อมโยงข้อมูลแบบเบ็ดเสร็จ ความหมายของระบบ

เบ็ดเสร็จ (Integrated System) คือมีการรวมข้อมูลเข้าไว้ด้วยกันเพื่อลดปัญหาความซ้ำซ้อน (Redundance) และความไม่สอดคล้อง (Inconsistency) ของข้อมูล และให้ผู้ใช้สามารถเรียกใช้ข้อมูลร่วมกันได้

3.2.2.2 ใช้คอมพิวเตอร์ช่วยในการประมวลผล (Computer - base system)โดยปกติระบบสารสนเทศเพื่อการจัดการไม่จำเป็นต้องใช้คอมพิวเตอร์ในการประมวลผล แต่ถ้าในหน่วยงานมีการทำงานที่ซับซ้อน มีปริมาณข้อมูลมาก การดำเนินงานโดยบุคลากรอาจได้คำตอบที่ล่าช้าเกินกว่าใช้ให้เกิดประโยชน์ได้เต็มที่ ดังนั้น เพื่อให้การดำเนินการมีประสิทธิภาพจึงจำเป็นต้องนำคอมพิวเตอร์มาช่วยงาน

3.2.2.3 มีการผสมผสานระหว่างฮาร์ดแวร์ และซอฟท์แวร์ เพื่อให้ผู้ใช้มีความสะดวกในการดำเนินงาน (User-machine Interface)

3.2.2.4 จัดเตรียมสารสนเทศ ให้สามารถตอบสนองความต้องการในการบริหารงานทุกระดับ

3.2.2.5 เป็นระบบสนับสนุนการปฏิบัติงาน และช่วยในการวางแผนและตัดสินใจในการบริหารงานขององค์กร

3.3 แนวคิดการพัฒนาระบบสารสนเทศ

เพราะความซับซ้อนของงานวิเคราะห์ระบบในปัจจุบัน ทำให้มีความต้องการมาตรฐานในการพัฒนาระบบงานที่เป็นไปทางเดียวกัน และมีขั้นตอนที่เป็นแนวทางให้ปฏิบัติ โดยมีข้อบกพร่องน้อยที่สุด ดังนั้น วงจรการพัฒนาระบบงานโครงการ (System Development Life Cycle) หรือ SDLC จึงได้ถูกนำมาใช้ตอบสนองความต้องการนี้

วงจรการพัฒนาระบบงาน สามารถแบ่งเป็น 4 ขั้นดังนี้ (Senn, 1989)

- การวิเคราะห์ระบบงาน (System analysis) เป็นขั้นตอนของการศึกษาระบบงานเดิมที่ใช้ในปัจจุบัน (Current system) ปัญหาในระบบงานเดิม ตลอดจนศึกษาถึงความต้องการของธุรกิจ (business needs and requirements) พร้อมกับการประเมินเหตุการณ์ต่าง ๆ เพื่อหาทางเลือกเหมาะสมมาแก้ปัญหา

- การออกแบบและวางระบบงาน (System design) เป็นขั้นตอนวางโครงสร้างของระบบงานทั้งในลักษณะทั่ว ๆ ไป และในลักษณะเฉพาะโดยมีการแจงรายละเอียดที่แน่ชัด

ของแต่ละงาน และระบบงานย่อยของระบบที่ออกแบบขึ้นจะถูกส่งต่อให้ผู้เขียนโปรแกรม เพื่อพัฒนาโปรแกรมให้ระบบปฏิบัติงานจริงได้ในขั้นต่อไป

- การนำระบบงานเข้าสู่ธุรกิจหรือผู้ใช้ (System implementation) เป็นขั้นตอนนำเอาระบบงานมาติดตั้งให้ผู้ใช้หรือธุรกิจ และเพื่อให้แน่ใจว่าระบบงานสามารถทำงานได้ตามจุดประสงค์ ระบบงานต้องถูกตรวจสอบอย่างดี พร้อมให้การอบรมผู้ใช้ระบบ

- การดำเนินการสนับสนุนภายหลังติดตั้งระบบงาน (System support) เป็นการให้คำแนะนำเพื่อช่วยเหลือการใช้ระบบของผู้ใช้ระบบหลังจากติดตั้งระบบไปแล้ว ทั้งยังรวมถึงความต้องการต่าง ๆ ที่อาจเกิดขึ้นหรือเปลี่ยนแปลงไปภายหลังจากระบบถูกติดตั้ง

3.3.1 การวิเคราะห์ระบบงาน ประกอบด้วยขั้นตอนย่อย 4 ขั้นตอนดังนี้

3.3.1.1 การสำรวจเพื่อหาขอบเขตของระบบและความเป็นไปได้ (Survey project scope and feasibility) คือเริ่มศึกษาปัญหาที่ผู้ใช้ประสบ และต้องการที่จะแก้ไขหรือปรับปรุงสภาพให้ดีขึ้น ถ้าผลลัพธ์จากการค้นหาขอบเขต และความเป็นไปได้ของระบบพบว่ามีความเป็นไปได้ที่จะทำการพัฒนา ควรจะทำการวิเคราะห์ต้นทุน-ผลตอบแทน อย่างคร่าว ๆ ด้วย

3.3.1.2 ศึกษาและวิเคราะห์ระบบเดิม เป็นการทำความเข้าใจในระบบงานเดิม ไม่ว่าระบบจะใช้คอมพิวเตอร์อยู่แล้ว หรือปฏิบัติด้วยคนทั้งหมดก็ตาม ผลลัพธ์ที่ได้ คือการแจกแจงปัญหาที่เกิดขึ้น เพื่อใช้เป็นข้อมูลในการพัฒนาระบบต่อไป

3.3.1.3 พิจารณาความต้องการของผู้ใช้ระบบ กำหนดว่าระบบงานข้อมูลใหม่จะพัฒนาจะให้อะไรแก่ผู้ใช้ระบบ รวมถึงค้นหาว่าอะไรบ้างเป็นความต้องการของผู้ใช้จากนั้นจะกำหนดรูปแบบของสิ่งต้องนำเข้าสู่ระบบแฟ้มข้อมูล ขั้นตอนการปฏิบัติต่าง ๆ โดยการกำหนดรูปแบบดังกล่าวจะอยู่ในรูปแบบกว้าง ๆ เรียกว่า (Logical design) ซึ่งไม่สนใจการแจงรายละเอียดด้านเทคนิคมากนัก

3.3.1.4 เลือกแนวทางที่เหมาะสมในการพัฒนาและวางระบบงาน ในการเลือกแนวทางนี้ มีสิ่งที่ใช้เป็นตัวพิจารณา 3 ประการ

ประการที่ 1 ความเป็นไปได้ด้านเทคนิค

ประการที่ 2 ความเป็นไปได้ในทางปฏิบัติ

ประการที่ 3 ความเป็นไปได้ในเชิงธุรกิจ

3.3.1.5 ดำเนินการจัดซื้ออุปกรณ์คอมพิวเตอร์และซอฟต์แวร์

3.3.2 การออกแบบและวางระบบงาน ในส่วนของการออกแบบ และวางระบบงานนี้สามารถจำแนกออกเป็น 2 ขั้นตอน ดังนี้

ขั้นตอนที่ 1 การออกแบบระบบงานใหม่ เพื่อการออกแบบระบบงานให้เห็นเป็นรูปร่าง โดยทำการออกแบบผลลัพธ์ของระบบแฟ้มข้อมูล ข้อมูลส่วนนำเข้าระบบ รวมทั้งวิธีการหรือขั้นตอนการประมวลผลในช่วงต่าง ๆ ของระบบ

ขั้นตอนที่ 2 ดำเนินการสร้างระบบ ก็คือ การดำเนินการเขียนโปรแกรมให้ทำงานตามระบบที่ออกแบบ

3.4 การวิเคราะห์และออกแบบระบบแบบโครงสร้าง

3.4.1 หลักการสำคัญของวิธีการวิเคราะห์และออกแบบระบบแบบโครงสร้าง

- แบ่งงานที่มีขนาดใหญ่ และมีปัญหาที่ซับซ้อนออกเป็นส่วนย่อย ๆ โดยพิจารณาภาพรวมของระบบเป็นหลักก่อน จากนั้นจึงค่อยแตกย่อยออกเป็นส่วน ๆ จนทราบถึงฟังก์ชั่น (Function) หลักของระบบ และกระทั่งทราบถึงกระบวนการ (Procedure) ที่ต้องมีในระบบ

- มีเครื่องมือในการวิเคราะห์และออกแบบระบบที่อยู่ในลักษณะของแผนภาพ (Diagram) ซึ่งผู้วิเคราะห์ระบบสามารถใช้ติดต่อสื่อความเข้าใจกับผู้ใช้ระบบ และผู้บริหารได้สะดวก

- เอกสารประกอบระบบงานจะเกิดขึ้นในขณะทำการวิเคราะห์ และออกแบบระบบ โดยเอกสารนั้นจะอยู่ในรูปแบบเข้าใจได้ง่าย สามารถแก้ไขได้ง่ายเมื่อมีการเปลี่ยนแปลงของระบบ ผู้วิเคราะห์งาน หรือผู้อื่นสามารถใช้ประโยชน์ของเอกสารประกอบระบบงานในการศึกษาระบบ หรือปรับปรุงระบบงานได้สะดวก และถูกต้อง

3.4.2 เครื่องมือใช้ในการวิเคราะห์และออกแบบระบบแบบโครงสร้าง

ประกอบด้วยเครื่องมือหลัก ๆ ดังนี้

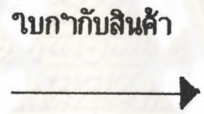
3.4.2.1 แผนภาพกระแสข้อมูล (DATAFLOW DIAGRAM)

เป็นแผนภาพแสดงขั้นตอนดำเนินงานของระบบ ในระดับตรรกะ (LOGICAL LEVEL) แผนภาพจะแสดงถึงแหล่งข้อมูล (TERMINATOR) กระแสข้อมูล (DATA FLOW) การประมวลผล (PROCESS) และหน่วยเก็บข้อมูล (DATA STORE)

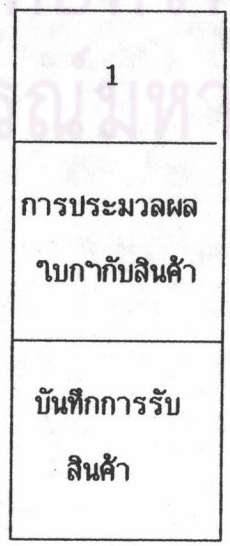
ส่วนประกอบสำคัญของแผนภาพกระแสข้อมูลมีดังนี้

- แหล่งข้อมูล ได้แก่ระบบ หรือหน่วยงานที่เป็นแหล่งกำเนิด หรือสิ้นสุดของข้อมูล สัญลักษณ์ของแหล่งข้อมูลแทนด้วยสี่เหลี่ยมจัตุรัสที่มีชื่อของแหล่งข้อมูลอยู่ภายในเช่น

- กระแสข้อมูล ได้แก่ข้อมูลที่เคลื่อนไหวในระบบ โดยใช้สัญลักษณ์ลูกศรแทนกระแสข้อมูลซึ่งหัวลูกศรแสดงทิศทางการเคลื่อนไหวของข้อมูล และปรากฏชื่อของข้อมูลที่เคลื่อนไหวบนลูกศร เช่น



- การประมวลผลจะใช้สี่เหลี่ยมผืนผ้าแนวตั้งเป็นสัญลักษณ์แทนการประมวลผลในระบบ โดยภายในสี่เหลี่ยมอาจแบ่งเป็น 3 ส่วนคือส่วนบนให้หมายเลขกำกับกระบวนการ ส่วนกลางเป็นหน้าที่ของการประมวลผล และส่วนล่างเป็นคำกริยาที่อธิบายถึงการประมวลผล เช่น



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

- หน่วยเก็บข้อมูล ใช้เส้นคู่ขนานตามแนวนอนที่ปิดหัวข้างหนึ่งเป็นสัญลักษณ์ที่แสดงถึงที่เก็บข้อมูลในระหว่างการประมวลผล เช่น

3.4.2.2 ข้อกำหนดกระบวนการ (PROCESS SPECIFICATION)

ใช้อธิบายรายละเอียด การทำงานของแต่ละการประมวลผลที่มีปรากฏบนแผนภาพกระแสข้อมูลซึ่งรายละเอียดดังกล่าวจะบอกให้ทราบถึงวิธีการแปลงข้อมูลนำเข้าไปเป็นส่วนผลลัพธ์ของการประมวลผล โดยการเขียนข้อกำหนดกระบวนการมักเขียนในรูปของ

- ภาษาอังกฤษแบบโครงสร้าง (STRUCTURED ENGLISH)
- ต้นไม้การตัดสินใจ (DECISION TREE)
- ตารางการตัดสินใจ (DECISION TABLE)

3.4.2.3 พจนานุกรมข้อมูล (DATA DICTIONARY)

เป็นเอกสารที่ประกอบด้วย ข้อมูลของส่วนย่อยข้อมูล (DATA ELEMENT) ทั้งหมดที่ปรากฏในระบบงาน ซึ่งข้อมูลดังกล่าวจะให้รายละเอียดส่วนย่อยข้อมูลทั้งที่เป็นข้อมูลนำเข้า ผลลัพธ์ของการประมวลผลที่ปรากฏบนแผนภาพกระแสข้อมูลและรายละเอียดของส่วนย่อยข้อมูลที่อยู่บนหน่วยเก็บข้อมูลที่ปรากฏบนแผนภาพกระแสข้อมูล ข้อมูลของส่วนย่อยข้อมูลที่จำเป็นต้องมีในพจนานุกรมข้อมูลมีดังนี้

- ชื่อของส่วนย่อยข้อมูล (NAME)
- ชื่ออื่นๆของส่วนย่อยข้อมูล (ALIASES)
- ความหมายของส่วนย่อยข้อมูล (DESCRIPTION)
- ชนิดและรูปแบบในการจัดเก็บสำหรับส่วนย่อยข้อมูล (FORMAT)
- ตำแหน่งที่มีการบันทึกส่วนย่อยข้อมูล (LOCATION)

3.4.2.4 ผังงานระบบ (SYSTEM FLOWCHART)

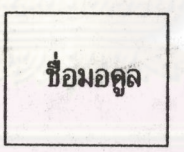
เป็นผังงานที่อธิบายถึงโครงสร้างของระบบทั้งหมด โดยเริ่มตั้ง

แต่ข้อมูลนำเข้าของระบบว่าอยู่ในรูปแบบใดและเข้าสู่ระบบโดยผ่านอุปกรณ์บันทึกข้อมูลชนิดใด การประมวลผลข้อมูลที่มีทั้งการใช้โปรแกรมและไม่ใช้โปรแกรม และผลลัพธ์ที่เกิดขึ้นจากระบบซึ่งอาจเป็นแฟ้มข้อมูล หรือฐานข้อมูลที่บันทึกบนอุปกรณ์ชนิดต่างๆ หรืออาจเป็นรายงานที่ผลิตออกจากระบบ สัญลักษณ์ของผังงานระบบที่ใช้จะปรากฏอยู่บนแผ่นเทมเพลต (TEMPLATE) สำหรับเขียนผังงาน (FLOWCHART)

3.4.2.5 ผังโครงสร้าง (STRUCTURE CHART)

ผังโครงสร้างแสดงความสัมพันธ์ของมอดูล (MODULE) ที่มีในระบบงานในรูปของโครงสร้างแบบต้นไม้ (TREE STRUCTURE) รวมทั้งแสดงถึงข้อมูลนำเข้าที่เข้าสู่มอดูลและผลลัพธ์ที่ได้จากการทำงานของมอดูล สำหรับองค์ประกอบสำคัญของผังโครงสร้างประกอบด้วย

- มอดูล (MODULE) หมายถึงชุดคำสั่งที่กระทำงานใดงานหนึ่ง ใช้สัญลักษณ์แทนด้วยสี่เหลี่ยมภายในมีชื่อของมอดูลอยู่



- การเชื่อมต่อ (CONNECTION) ใช้สัญลักษณ์ลูกศรในการเชื่อมต่อระหว่างมอดูลโดยหางของลูกศรออกจากมอดูลตัวเรียก (CALLING MODULE) ส่วนหัวลูกศรชี้ไปยังมอดูลที่ถูกเรียก (CALLED MODULE)

- คัพพิล (COUPLE) ข้อมูลที่เคลื่อนไหวระหว่างมอดูลโดยอาจเป็นข้อมูลทั่วไปหรือเป็นข้อมูลชนิดตัวบ่งชี้ (FLAG DATA) ใช้สัญลักษณ์

แทนข้อมูลทั่วไปที่มีการเคลื่อนไหว ส่วนข้อมูลชนิดตัวบ่งชี้ที่เคลื่อนไหวใช้สัญลักษณ์

3.4.2.6 ผังไฮโป (HIPO CHART)

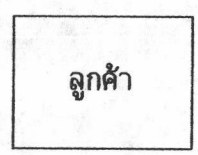
ประกอบขึ้นด้วย 2 ส่วนคือผังลำดับชั้น (HIERARCHY CHART) และผังไอพีโอ (IPO CHART) โดยผังลำดับชั้นแสดงความสัมพันธ์ของมอดูลในลักษณะ จากบนลงล่าง (TOP DOWN) ซึ่งทำให้ทราบว่ามีมอดูลย่อยใดบ้างในระบบงานย่อยหนึ่ง ๆ และแต่ละมอดูลสัมพันธ์กันอย่างไร ส่วนผังงานไอพีโอจะใช้อธิบายถึงส่วนนำเข้าส่วนผลลัพธ์ การประมวลผลของมอดูล รวมทั้งระบุชื่อของมอดูลตัวเรียกของมอดูลปัจจุบันและชื่อของมอดูลที่ถูกเรียกโดยมอดูลปัจจุบัน รูปแบบผังไอพีโอแสดงได้ดังรูปที่ 3.1

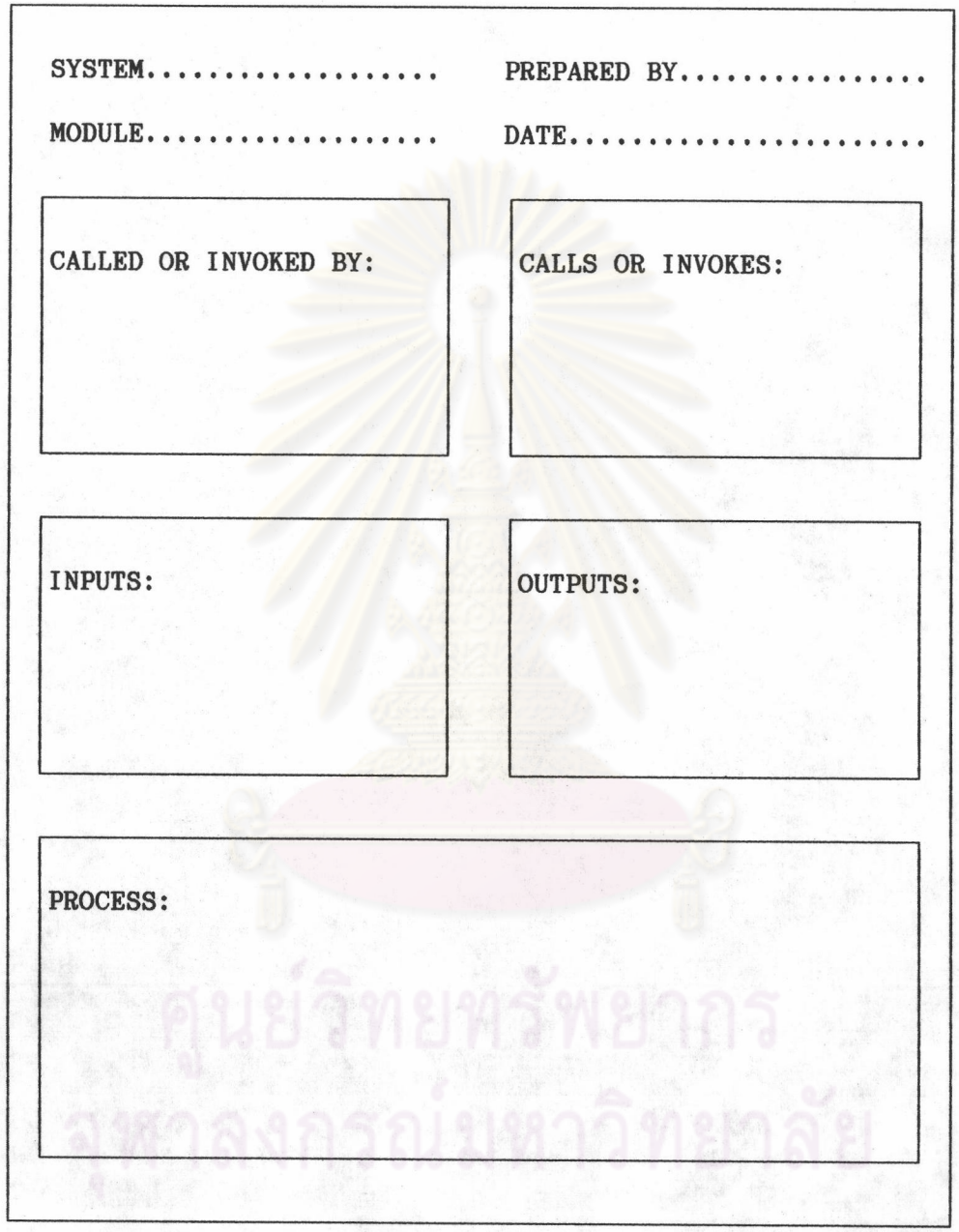
3.4.2.7 แผนภาพความสัมพันธ์เอนทิตี (ENTITY - RELATIONSHIP DIAGRAM)

หน่วยเก็บข้อมูล (DATA STORE) ที่ปรากฏบนแผนภาพกระแสข้อมูลจะให้รายละเอียดเพียงเล็กน้อยเท่านั้นสำหรับข้อมูลที่ถูกจัดเก็บ แผนภาพความสัมพันธ์เอนทิตีเป็นแผนภาพที่แสดงให้ทราบถึงรายละเอียดของหน่วยเก็บข้อมูลบนแผนภาพกระแสข้อมูล ว่ามีข้อมูลใดบ้างที่อยู่ในหน่วยเก็บข้อมูลหนึ่ง ๆ นอกจากนี้ยังบอกให้ทราบถึงความสัมพันธ์ (RELATIONSHIPS) ที่เกิดขึ้นระหว่างหน่วยเก็บข้อมูลต่างๆ

สำหรับส่วนประกอบหลักของแผนภาพความสัมพันธ์เอนทิตีมีดังนี้

- เอนทิตี (ENTITIES) อาจหมายถึง คน สถานที่ สิ่งของ เหตุการณ์ที่เกี่ยวข้องกับระบบที่จะพัฒนา ใช้รูปสี่เหลี่ยมโดยมีชื่อของเอนทิตีอยู่ภายในเป็นสัญลักษณ์แทนเอนทิตี เช่น เอนทิตีลูกค้า แทนด้วย





รูปที่ 3.1 แสดงรูปแบบผังไอพีโอ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

- ความสัมพันธ์ (RELATIONSHIPS) ใช้รูปสี่เหลี่ยมขนมเปียกปูนที่มีชื่อความสัมพันธ์อยู่ภายในแทนความสัมพันธ์ที่เกิดขึ้นระหว่างเอนทิตี โดยจะมีเส้นตรงลากจากเอนทิตีที่มีความสัมพันธ์กันมาบรรจบกันที่สัญลักษณ์ความสัมพันธ์ เช่น เอนทิตี MAN มีความสัมพันธ์ MARRY กับเอนทิตี WOMAN แทนด้วย

- คุณสมบัติของเอนทิตีและความสัมพันธ์ (DESCRIPTION OF ENTITIES & RELATIONSHIPS) ในแต่ละเอนทิตี และความสัมพันธ์ประกอบด้วยแอตตริบิวชุดหนึ่ง ๆ โดยอาจมีแอตตริบิวเพียงตัวเดียวหรือมากกว่าก็ได้แอตตริบิวแต่ละตัวมีชื่อของตัวเอง แอตตริบิวแต่ละตัวจะมีค่าอยู่ในขอบเขตของโดเมน (DOMAINS) ของแอตตริบิวนั้น

3.5 ระบบฐานข้อมูล

ฐานข้อมูล (Database) คือ โครงสร้างสารสนเทศ (Information) ที่ประกอบด้วย เอนติตี (Entity) หลาย ๆ ตัวซึ่งเอนติตีเหล่านี้ต้องมีความสัมพันธ์กันสำหรับประเภทของฐานข้อมูลในปัจจุบันมี 3 ประเภทได้แก่ (Date, 1981)

ประเภท 1 ฐานข้อมูลแบบรีเลชันนัล (Relational database)

ข้อมูลถูกเก็บในรูปแบบของตาราง(Table) ซึ่งตารางนี้ก็คือ Relation

ประเภท 2 ฐานข้อมูบแบบลำดับชั้น (Hierachical database)

ข้อมูลถูกเก็บโดยมีโครงสร้างแบบต้นไม้ (Tree) สัมพันธ์เป็นลำดับชั้นแบบแม่ (Parent) และลูก (Child) ข้อมูลในระดับบนสุดเรียกว่ารากฐาน (Root)

ประเภท 3 ฐานข้อมูลแบบเนทเวอร์ค (Network database)

โครงสร้างข้อมูลแบบนี้มีความซับซ้อน การเก็บข้อมูลเป็นลักษณะลิสท์เชื่อมโยง (Link List) หรือตัวชี้ (Pointer) จากเรคคอร์ดแม่ (Parent record) เชื่อมโยงไปหาเรคคอร์ดลูก (Children Record)

3.5.1 ประโยชน์ของการประมวลผลโดยใช้ฐานข้อมูล (ดวงแก้ว สวามิภักดิ์, 2533)

- ลดความซ้ำซ้อนของข้อมูล
- สามารถหลีกเลี่ยงความขัดแย้งของข้อมูลได้ในระดับหนึ่ง
- สามารถใช้ข้อมูลร่วมกันได้
- สามารถควบคุมความเป็นมาตราฐานได้
- สามารถจัดหาระบบความปลอดภัยที่รัดกุมได้
- สามารถควบคุมความคงสภาพของข้อมูลได้
- สามารถสร้างสมดุลในความขัดแย้งของความต้องการได้
- เกิดความเป็นอิสระของข้อมูล

3.6 สถาปัตยกรรมระบบฐานข้อมูล (Database architecture) (Date, 1981)

ระบบฐานข้อมูลได้แบ่งระดับข้อมูลออกเป็น 3 ระดับคือ

- ระดับภายใน (Internal หรือ Physical level) เป็นระดับที่แสดงถึงการจัดเก็บข้อมูลจริง ๆ ว่าถูกแทนที่อย่างไรในระดับกายภาพ (Physical)

- ระดับหลักการ (Conceptual level) เป็นระดับถัดขึ้นมาอันได้แก่การมองระดับเอนติตี และความสัมพันธ์ระหว่างเอนติตี และความสัมพันธ์ระหว่างเอนติตีทั้งหมดรวมทั้งกฎเกณฑ์ต่าง ๆ เกี่ยวกับข้อมูล

- ระดับภายนอก (External หรือ View level)เป็นระดับสูงสุดอันเป็นระดับข้อมูลที่มองเห็นจากการใช้งานของผู้ใช้แต่ละคน

ในขณะที่เรียกใช้และแก้ไขข้อมูล ย่อมทำให้ค่าของข้อมูลในระบบมีการเปลี่ยนแปลงอยู่ตลอดเวลา เรียกเหล่าข้อมูลในฐานข้อมูลในขณะใดขณะหนึ่งว่า อินสแตนท์ (Instance) ของฐานข้อมูล ส่วนเค้าร่างที่ได้จากการออกแบบฐานข้อมูลโดยส่วนรวม คือ เค้าร่างฐานข้อมูล (Database schema) ซึ่งตามปกติแล้วไม่ควรมีการเปลี่ยนแปลง หรือมีก็ไม่ควรบ่อยนัก

เค้าร่าง (Schema) ในฐานข้อมูลแบ่งได้หลายระดับระดับต่ำที่สุด ได้แก่ระดับเค้าร่างภายใน (Internal schema) ในระดับกลาง ได้แก่ ระดับเค้าร่างหลักการ (Conceptual schema) และระดับขั้นนอกสุด ได้แก่ ระดับเค้าร่างภายนอก (External schema)

3.7 ระบบจัดการฐานข้อมูล (Database Management system)

หมายถึงซอฟต์แวร์ที่มีความสามารถในการจัดการเกี่ยวกับฐานข้อมูล ซึ่งได้แก่การมีตัวการจัดเก็บ การแก้ไข และควบคุมดูแลการเรียกใช้ฐานข้อมูล โดยมีระบบควบคุมความปลอดภัยของฐานข้อมูล ซึ่งผู้ใช้สามารถเรียกใช้งานได้สะดวก เหมือนสื่อกลางระหว่างผู้ใช้และโปรแกรมต่าง ๆ ที่เกี่ยวกับการใช้ฐานข้อมูล

3.7.1 หน้าที่ของระบบจัดการฐานข้อมูล (Fleming, 1989)

- ทำหน้าที่ติดต่อกับตัวจัดการระบบแฟ้มข้อมูล
- ควบคุมความคงสภาพ โดยทำการควบคุมค่าของข้อมูลในระบบให้อยู่ในกรอบที่ถูกต้อง
- ควบคุมระบบความปลอดภัย ป้องกันการใช้ข้อมูลจากผู้ที่ไม่ได้รับอนุญาต
- การสร้างระบบสำรองและการฟื้นสภาพ จัดการทำข้อมูลสำรองและถ้าเกิดปัญหาต่าง ๆ จะสามารถใช้ระบบข้อมูลสำรองในการฟื้นสภาพได้
- ควบคุมการใช้ข้อมูลในสภาพที่มีผู้ใช้พร้อมกันหลายคน

3.8 <u>การออกแบบฐานข้อมูล (Database design)</u> (ดวงแก้ว สวามิภักดิ์, 2533)

กระบวนการทั้งหมดเริ่มจากการรวบรวมความต้องการของผู้ใช้ รวมทั้งกฎเกณฑ์และข้อบังคับต่างๆเพื่อนำมาวิเคราะห์ออกแบบเป็นระบบฐานข้อมูลในระดับสารสนเทศ โดยไม่คำนึงถึงระบบจัดการฐานข้อมูลที่จะใช้ หรือว่าจะทำงานบนเครื่องคอมพิวเตอร์ชนิดใด หลังจากผ่านการออกแบบระดับสารสนเทศแล้ว การออกแบบขั้นต่อไป คือการออกแบบระดับกายภาพ โดยจุดนี้ต้องอาศัยข้อมูลอื่นๆที่เป็นข้อจำกัดทางกายภาพมาร่วมใช้ในการออกแบบด้วย เช่น ข้อมูลเกี่ยวกับการควบคุมความปลอดภัยของข้อมูล เป็นต้น ซึ่งข้อมูลดังกล่าวจะนำมารวมกับระบบฐานข้อมูลที่ออกแบบไว้แล้ว เพื่อผ่านการออกแบบระดับกายภาพได้เป็นโครงสร้างระบบฐานข้อมูลที่สมบูรณ์ ในการออกแบบระดับนี้ ผู้ออกแบบต้องคำนึงถึงความสามารถของระบบจัดการฐานข้อมูลที่ใช้ และประสิทธิภาพของระบบเป็นหลัก

สามารถสรุปขั้นตอนการออกแบบระดับสารสนเทศได้ดังนี้

3.8.1 เปลี่ยนรูปแบบความต้องการให้อยู่ในรูปลักษณะของความสัมพันธ์(RELATION)

3.8.2 นอร์มัลไลซ์ความสัมพันธ์ (NORMALIZE RELATION)

3.8.3 กำหนดเขตข้อมูล (FIELD) ที่จะเป็นคีย์ต่างๆ และคุณสมบัติของคีย์แต่ละตัว

3.8.4 พิจารณาข้อกำหนดและกฎเกณฑ์อื่น ๆ

3.8.5 นำผลที่ได้จากการออกแบบไปผนวกกับผลการออกแบบที่สร้างขึ้นสำหรับผู้ใช้คนอื่นๆ

ผลลัพธ์ของการออกแบบระบบฐานข้อมูลในระดับสารสนเทศ หรือมโนภาพของแบบแผนฐานข้อมูล (CONCEPTUAL SCHEMA) ส่วนใหญ่จะถูกแสดงในรูปของความสัมพันธ์ที่ถูกนอร์มัลไลซ์ (NORMALIZED RELATION) ให้อย่างน้อยอยู่ในรูปแบบบรรทัดฐานระดับที่ 3 (THIRD NORMAL FORM)

สาเหตุที่ต้องมีการนอร์มัลไลซ์ความสัมพันธ์ ก็เพื่อต้องการปรับปรุงคุณสมบัติของความสัมพันธ์ขึ้นเป็นขั้นตอนอย่างมีระบบ เพื่อลดโอกาสที่จะเกิดปัญหาความซ้ำซ้อนของข้อมูล และปัญหาที่จะเกิดขึ้นเมื่อมีการแก้ไขข้อมูลในระบบ (UPDATE ANOMALY) อันเนื่องมาจากการที่ข้อมูลในส่วนต่างๆของระบบมีการขึ้นต่อกันที่ไม่เหมาะสม

อาจจะแบ่งลักษณะของปัญหา ที่อาจเกิดขึ้นเมื่อมีการแก้ไขข้อมูลในระบบได้ 4 ประเภท

- การแก้ไขข้อมูล (UPDATE)
- ความขัดแย้งของข้อมูล (INCONSISTENT DATA)
- การเพิ่มเติมข้อมูล (ADDITIONS)
- การลบข้อมูล (DELETIONS)

3.9 <u>ทฤษฎีการนอร์มัลไลซ์ (Normalization theory)</u> (Fleming, 1989)

ในการออกแบบความสัมพันธ์ ที่จะทำให้การเรียกใช้ข้อมูลในระบบทำได้สะดวก และมีประสิทธิภาพ โดยให้มีความซ้ำซ้อนในการจัดเก็บข้อมูลน้อยที่สุดทำได้ โดยอาศัยทฤษฎีของการนอร์มัลไลซ์โดยเป้าหมายของทฤษฎี คือ การทำให้ความสัมพันธ์ที่ออกแบบอยู่ในรูปแบบบรรทัดฐาน (NORMAL FORM)

3.9.1 รูปแบบบรรทัดฐานระดับที่ 1(FIRST NORMAL FORM : 1NF)

ความสัมพันธ์ที่ถือว่าอยู่ในรูปแบบบรรทัดฐานระดับที่ 1 ก็ต่อเมื่อความสัมพันธ์นั้นไม่มีกลุ่มที่ซ้ำกัน(REPEATING GROUPS)

การปรับความสัมพันธ์ ที่อยู่ในลักษณะของความสัมพันธ์ที่ยังไม่นอร์มัลไลซ์ให้เป็นความสัมพันธ์ที่อยู่ในรูปแบบบรรทัดฐานระดับที่ 1 ทำได้โดยการขจัดกลุ่มที่ซ้ำกันออกจากความ

สัมพันธ์ ตัวอย่างความสัมพันธ์การสั่งที่ยังไม่นอร์มัลไลซ์โดยมีรหัสการสั่งเป็นคีย์หลัก

การสั่ง

รหัสการสั่ง	วันที่สั่ง	รหัสสินค้า	จำนวนสินค้า
12489	020931	ax12	11
12491	020931	bt04	12
		bz06	1
12494	040931	bx01	2

จากความสัมพันธ์พบว่าในการสั่งสินค้าครั้งหนึ่งอาจสั่งสินค้ามากกว่า 1 อย่าง ดังนั้นจึงเกิดกลุ่มที่ซ้ำกันในส่วนของรหัสสินค้า และจำนวนสินค้า

สามารถทำให้อยู่ในรูปแบบบรรทัดฐานระดับที่ 1 โดยการแยกข้อมูลของรหัสการสั่ง 12491 ออกเป็น 2 แถว ดังนี้

การสั่ง

รหัสการสั่ง	วันที่สั่ง	รหัสสินค้า	จำนวนสินค้า
12489	020931	ax12	11
12491	020931	bt04	12
12491	020931	bz06	1
12494	040931	bx01	2

จากผลการทำดังกล่าว ทำให้ต้องเพิ่มแอตตริบิวของคีย์หลัก โดยตามทฤษฎีกำหนดว่า คีย์หลักใหม่จะประกอบด้วยคีย์หลักเดิมผนวกกับแอตตริบิวที่เป็นคีย์หลักของกลุ่มที่ซ้ำกัน อันได้แก่ รหัสสินค้า ดังนั้นคีย์หลักใหม่จึงประกอบด้วย รหัสการสั่งและรหัสสินค้า

3.9.2 รูปแบบบรรทัดฐานระดับที่ 2(SECOND NORMAL FORM : 2NF)

นิยาม : แอตตริบิวที่ไม่ใช่ส่วนหนึ่งของคีย์หลัก จะเรียกว่าแอตตริบิวที่ไม่ใช่คีย์ (NONKEY ATTRIBUTE)

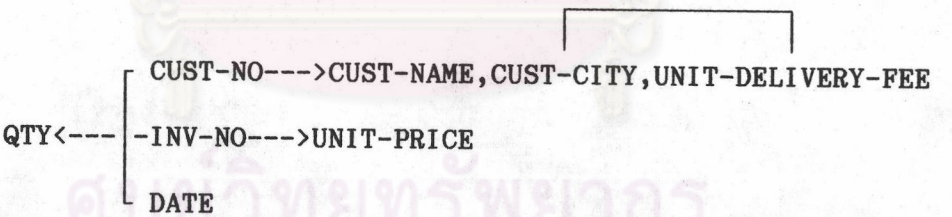
- ความสัมพันธ์ที่ถือว่าอยู่ในรูปแบบบรรทัดฐานระดับที่ 2 ก็ต่อเมื่อความสัมพันธ์นั้นอยู่ในรูปแบบบรรทัดฐานระดับที่ 1 และไม่มีแอตตริบิวที่ไม่ใช่คีย์ตัวใดไปขึ้นอยู่กับเฉพาะส่วนใดส่วนหนึ่งของคีย์หลัก การปรับความสัมพันธ์ที่อยู่ในรูปแบบบรรทัดฐานระดับที่ 1 ให้อยู่ในรูปแบบบรรทัดฐานระดับที่ 2 ทำได้โดยการสร้างความสัมพันธ์ขึ้นมาใหม่สำหรับการขึ้นต่อกันที่เป็นปัญหา

ตัวอย่าง ความสัมพันธ์ CUSTOMER-ORDER ประกอบด้วยแอตตริบิว

CUST-NO,CUST-NAME,INV-NO,DATE,CUST-CITY,UNIT-PRICE

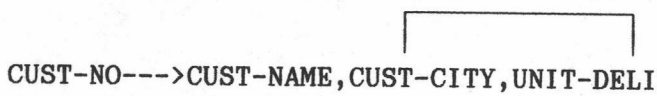
QTY,UNIT-DELIVERY-FEE

คีย์หลักคือ (CUST-NO,INV-NO,DATE) มีการขึ้นต่อกันดังนี้



ปรับให้อยู่ในรูปแบบบรรทัดฐานระดับที่ 2 โดยการแยกความสัมพันธ์เดิมออกเป็นความสัมพันธ์ย่อย จนกระทั่งแอตตริบิวที่ไม่ใช่คีย์ขึ้นอยู่กับเฉพาะคีย์หลักของแอตตริบิวที่ไม่ใช่คีย์นั้น ได้เป็น

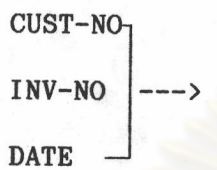
ความสัมพันธ์ CUSTOMER



ความสัมพันธ์ INVENTORY

INV-NO--->UNIT-PRICE

ความสัมพันธ์ ORDER



3.9.3 รูปแบบบรรทัดฐานระดับที่ 3(THIRD NORMAL FORM : 3NF)

นิยาม : แอตตริบิว (หรือกลุ่มของแอตตริบิว) ใดก็ตามที่สามารถเลือกแอตตริบิวตัวอื่นได้เรียกว่าตัวเลือก (DETERMINANT)

- ความสัมพันธ์ที่ถือว่าอยู่ในรูปแบบบรรทัดฐานระดับที่ 3 ก็ต่อเมื่อความสัมพันธ์นั้นอยู่ในรูปแบบบรรทัดฐานระดับที่ 2 และตัวเลือกใด ๆในความสัมพันธ์ที่มีแอตตริบิวที่ไม่ใช่คีย์ไปขึ้นอยู่กับมัน ตัวเลือกดังกล่าวต้องเป็นคีย์คู่แข่ง

การปรับความสัมพันธ์ในรูปแบบบรรทัดฐานระดับที่ 2 ให้อยู่ในรูปแบบบรรทัดฐานระดับที่ 3 ทำได้โดยดึงแอตตริบิวที่ไม่ใช่คีย์เหล่านี้ออกมาสร้างเป็นความสัมพันธ์ใหม่โดยให้แอตตริบิว (หรือกลุ่มแอตตริบิว)ที่มันขึ้นอยู่เป็นคีย์หลักของความสัมพันธ์ใหม่ที่สร้างขึ้นมา

ตัวอย่าง : จากความสัมพันธ์ CUSTOMER-ORDER ที่ปรับให้อยู่ในรูปแบบบรรทัดฐานระดับที่ 2 แล้วแต่พบว่าความสัมพันธ์ CUSTOMER มีแอตตริบิวที่ไม่ใช่คีย์ คือ UNIT-DELIVERY-FEE ขึ้นอยู่กับแอตตริบิว CUST-CITY ซึ่งเป็นแอตตริบิวที่ไม่ใช่คีย์

การปรับให้ความสัมพันธ์ CUSTOMER อยู่ในรูปแบบบรรทัดฐานระดับที่ 3 ทำได้โดยแยก CUST-CITY , UNIT-DELIVERY-FEE ออกมาสร้างเป็นความสัมพันธ์ใหม่ได้ดังนี้

ความสัมพันธ์ CUST

CUST-NO--->CUST-NAME,CUST-CITY

ความสัมพันธ์ DELIVERY-FEE

CUST-CITY--->UNIT-DELIVERY-FEE

I17002904