สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

คลังข้อมูลเชิงวัตถุสัมพันธ์
OBJECT-RELATIONAL
INFORMATION WAREHOUSE

นายกานต์ ผู้ภักดี
นายหิรัณย์ อโณวรรณพันธ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2545

เลขหมู่.....
เลขทะเบียน.....49963.....
วัน,เดือน,ปี.....16 เม.ย. 2547
.b.....
.i.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คลังข้อมูลเชิงวัตถุสัมพันธ์

OBJECT-RELATIONAL INFORMATION WAREHOUSE

โดย

นายกานต์ ผู้ภักดี

นายหิรัณย์ อโณวรรณพันธ์

อาจารย์ที่ปรึกษา

รศ.ดร. ศุภมิตร จิตตะยโศธร

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2545

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง คลังข้อมูลเชิงวัตถุสัมพันธ์

OBJECT-RELATIONAL INFORMATION WAREHOUSE

ผู้จัดทำ

1. นาย กานต์ ผู้ภักดี รหัสประจำตัว 42010015
2. นาย หิรัณย์ อโณวรรณพันธ์ รหัสประจำตัว 42010417

_____ อาจารย์ที่ปรึกษา

(รศ.ดร.ศุภมิตร จิตตะยโศธร)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คลังข้อมูลเชิงวัตถุสัมพันธ์

นายกานต์ ผู้ภักดี 42010015
นายหิรัณย์ อโณวรรณพันธ์ 42010417
รศ.ดร. ศุภมิตร จิตตยะโศธร อาจารย์ที่ปรึกษา
ปีการศึกษา 2545

บทคัดย่อ

ปัจจุบันระบบดาต้าแวร์เฮ้าท์ซิ่ง (data warehousing) ที่ใช้ ฐานข้อมูลแบบวัตถุเชิงสัมพันธ์ (object-relational database) ยังมีไม่มากนัก เพราะดาต้าแวร์เฮ้าท์ส่วนใหญ่จะใช้ฐานข้อมูลแบบเชิงสัมพันธ์ (relational database) ซึ่งมีการใช้งานที่ง่าย

ฐานข้อมูลแบบวัตถุเชิงสัมพันธ์นั้นจะสนับสนุนการพัฒนาระบบดาต้าแวร์เฮ้าท์ที่ไม่สามารถทำได้เมื่อใช้ฐานข้อมูลแบบเชิงสัมพันธ์และฐานข้อมูลแบบวัตถุ (object-oriented database) ซึ่งฐานข้อมูลแบบวัตถุเชิงสัมพันธ์ได้มีการรวมเอาข้อดีของฐานข้อมูลแบบเชิงสัมพันธ์และฐานข้อมูลแบบวัตถุเข้าไว้ด้วยกัน

เทคโนโลยีแบบออบเจ็กรีเลชั่นนั้นสามารถที่จะแก้ปัญหาในข้อมูลประเภทโครงสร้าง (slowly changing attributes) ของเอนทีตี้ได้ โดยใช้โครงสร้างของข้อมูลซับซ้อนที่เกี่ยวกับเวลา ซึ่งไม่สามารถแก้ได้ถ้าใช้ฐานข้อมูลแบบเชิงสัมพันธ์ เพราะข้อมูลประเภทโครงสร้างเป็นประเด็นหลักในการพัฒนาดาต้าแวร์เฮาท์ซิ่ง

วิทยานิพนธ์ฉบับนี้ได้พัฒนารูปแบบของระบบดาต้าแวร์เฮ้าท์ ที่ใช้ฐานข้อมูลแบบวัตถุเชิงสัมพันธ์ เพื่อแก้ปัญหาข้อมูลประเภทโครงสร้าง และ ปัญหาความซ้ำซ้อนของข้อมูล ซึ่งไม่สามารถแก้ไขได้โดยฐานข้อมูลแบบเชิงสัมพันธ์ หรือ ฐานข้อมูลแบบวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Object-Relational Information Warehouse

Karn Pupakdee

Hirun Anowannaphan

Assoc. Prof. Dr. Suphamit Chittayasothon Advisor

ABSTRACT

Recently, data warehousing bases on object-relational database is limited work because mostly data warehouse use relational databases which are easy for using.

It has been found out that the object-relational database can support an improved data warehouse that might not be achieved using relational or object-oriented databases. Object-relational database is combining advantages of relational and object-oriented database.

Object-relational technology is capable of capturing and handling the slowly changing attributes of entities by using complex data structures involving time, which would not be possible using a relational database. Because slowly changing attributes remains a major issue in data warehousing.

This thesis develops the data warehouse model that uses object-relational database to solve problem slowly changing attributes and multivalue attributes. These problem can not solve by using relational or object-oriented databases.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้ไม่อาจสำเร็จได้ หากไม่ได้รับความร่วมมือจากหลายๆฝ่ายด้วยกัน บุคคลแรกที่ต้องกล่าวถึงเพราะเป็นส่วนสำคัญที่ทำให้วิทยานิพนธ์นี้เสร็จลงได้ก็คือ อาจารย์ ศุภมิตร จิตตยะโศธร อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ให้ความเอาใจใส่ คำแนะนำ และความช่วยเหลือเสมอมา ซึ่งต้องขอขอบพระคุณเป็นอย่างมาก

ขอขอบพระคุณ คุณธนะพงษ์ จารุเวคิน ตำแหน่ง IT special list บริษัท ไอบีเอ็ม ประเทศไทย ที่ให้ความช่วยเหลือในด้านโปรแกรม DB2 และขอขอบพระคุณบุคคลสำคัญที่สุดที่ทำให้ข้าพเจ้ามีวันนี้ ก็คือ บิดา มารดา อันเป็นที่เคารพรักยิ่ง ซึ่งได้เลี้ยงดูผู้เขียนมาเป็นอย่างดี พร้อมทั้งให้โอกาสในการศึกษาอย่างเต็มที่ และยังให้กำลังใจ เอาใจใส่เสมอมา ในทุกๆด้านอันหาที่เปรียบมิได้ ข้าพเจ้าขอระลึกในพระคุณอันสุดประมาณ และ ขอกราบขอบพระคุณมา ณ ที่นี้

กานต์ ผู้ภักดี

หิรัณย์ อโณวรรณพันธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้าที่
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VI
สารบัญภาพ	VII
บทที่ 1 บทนำ	1
1.1 ความสำคัญและที่มา	1
1.2 วัตถุประสงค์ของงานวิจัย	2
1.3 ขอบเขตของงานวิจัย	2
1.4 วิธีการดำเนินงาน	3
บทที่ 2 ดาต้าแวร์เฮ้าท์ซิ่ง	4
2.1 ดาต้าแวร์เฮ้าท์ซิ่งและประโยชน์ของดาต้าแวร์เฮ้าท์ซิ่ง	4
2.2 รูปแบบของดาต้าแวร์เฮ้าท์	5
บทที่ 3 ฐานข้อมูลเชิงวัตถุสัมพันธ์	7
3.1 ฐานขอ้มูลเชิงสัมพันธ์	7
3.2 ฐานข้อมูลแบบวัตถุ	8
3.3 ฐานข้อมูลแบบวัตถุสัมพันธ์	9
บทที่ 4 โครงสร้างและการออกแบบดาต้าแวร์เฮ้าท์	12
4.1 อีอาร์ไดอะแกรมสำหรับทรานแอคชั่นโพรเซสซิ่งดาต้าเบส	12
4.2 อีอาร์ไดอะแกรมของแวร์เฮ้าท์ดาต้าเบส	17
4.3 การปรับปรุงการออกแบบแวร์เฮ้าท์ดาต้าเบส	22
บทที่ 5 การพัฒนาดาต้าแวร์เฮ้าท์เพื่อนำไปใช้งาน	24
5.1 คำสั่งที่ใช้ในการสร้างออปเจคและตาราง	24
5.1.1 ตารางไดเมนชั่นของเวลา	24
5.1.2 ตารางไดเมนชั่นของสินค้า	26
5.1.3 ตารางไดเมนชั่นของลูกค้า	27
5.1.4 ตารางแสดงข้อมูลการขายสินค้า	33
5.2 คำสั่งที่ใช้ในการเปลี่ยนแปลงข้อมูล	34
5.2.1 ตารางไดเมนชั่นของเวลา	34
5.2.2 ตารางไดเมนชั่นของสินค้า	35
5.2.3 ตารางไดเมนชั่นของลูกค้า	36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.4 ตารางแสดงข้อมูลการขายสินค้า 37
5.3 ตัวอย่างการใช้คำสั่งเอสคิวแอลในการตอบคำถาม 38
บทที่ 6 ผลการทดสอบ 41
บทที่ 7 บทสรุปและคำแนะนำ 44
อ้างอิง ก. ตารางข้อมูล 45
อ้างอิง ข. ตัวอย่างการคิวรี่ข้อมูล 50
บรรณานุกรม 56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 3-1 แสดงคุณสมบัติของฐานข้อมูลทั้ง 3 แบบ ทรานแซคชั่น ดาต้าเบส	11
ตารางที่ 4-1 แสดงถึงแอททริบิวต์ต่างๆของลูกค้าในอีอาร์ไดอะแกรมของ ทรานแซคชั่น ดาต้าเบส	14
ตารางที่ 4-2 แสดงถึงแอททริบิวต์ต่างๆของสินค้าในอีอาร์ไดอะแกรมของ ทรารแซคชั่น ดาต้าเบส	15
ตารางที่ 4-3 แสดงถึงแอททริบิวต์ต่างๆของข้อมูลการขายในอีอาร์ไดอะแกรม ทรานแซคชั่น ดาต้าเบส	16
ตารางที่ 4-4 แสดงถึงแอททริบิวต์ต่างๆของไดเมนชั่นของลูกค้าในอีอาร์ ไดอะแดรม ของแวร์เฮ้าท์ดาต้าเบส	19
ตารางที่ 4-5 แสดงถึงแอททริบิวต์ต่างๆของไดเมนชั่นของสินค้าในอีอาร์ ไดอะแกรม ของแวร์เฮ้าท์ดาต้าเบส	20
ตารางที่ 4-6 แสดงถึงแอททริบิวต์ต่างๆของไดเมนชั่นของเวลาในอีอาร์ ไดอะแกรม ของแวร์เฮ้าท์ดาต้าเบส	21
ตารางที่ 4-7 แสดงถึงแอททริบิวต์ต่างๆของแฟกของการขายในอีอาร์ ไดอะแกรม ของแวร์เฮ้าท์ดาต้าเบส	22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

	หน้า
รูปที่ 2-1 แสดงรูปแบบสตาร์ สกีมา	5
รูปที่ 2-2 แสดงรูปแบบไฮเปอร์คิว	6
รูปที่ 4-1 อีอาร์ ไดอะแกรม ของ ทรานแซคชั่น โพรเซสซิ่ง ดาต้าเบส	12
รูปที่ 4-2 แสดงแอททริบิวต์ของลูกค้า	13
รูปที่ 4-3 แสดงแอททริบิวต์ของสินค้า	15
รูปที่ 4-4 แสดงแอททริบิวต์ของข้อมูลการขาย	16
รูปที่ 4-5 อีอาร์ ไดอะแกรม ของ แวร์เฮ้าท์ดาต้าเบส	17
รูปที่ 4-6 แสดงแอททริบิวต์ของไดเมนชั่นลูกค้า	18
รูปที่ 4-7 แสดงแอททริบิวต์ของไดเมนชั่นสินค้า	20
รูปที่ 4-8 แสดงแอททริบิวต์ของเวลา	21
รูปที่ 4-9 แสดงแอททริบิวต์ของแฟคเทเบิล	22
รูปที่ 5-1 ตารางแสดงไดเมนชั่นของเวลา	26
รูปที่ 5-2 ตารางแสดงไดเมนชั่นของสินค้า	27
รูปที่ 5-3 แสดงการถ่ายทอดคุณสมบัติระหว่าง Person_t กับ Cust_dimension_t	27
รูปที่ 5-4ก ตารางแสดงไดเมนชั่นของลูกค้า	28
รูปที่ 5-4ข ตารางแสดงไดเมนชั่นของลูกค้า(ต่อ)	29
รูปที่ 5-5 แสดงคำสั่งเอสคิวแอลในการสร้างคอมโพสิสไทป์	30
รูปที่ 5-6ก แสดงคำสั่งเอสคิวแอลในการสร้างมัลติแวลูไทป์	31
รูปที่ 5-6ข แสดงคำสั่งเอสคิวแอลในการสร้างมัลติแวลูไทป์(ต่อ)	31
รูปที่ 5-7 ตารางแสดงไดเมนชั่นของข้อมูลการขาย	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มา

แนวความคิดของดาต้าแวร์เฮ้าท์ซิ่งนั้นได้รับการพัฒนามาค่อนข้างนานตั้งแต่ช่วงต้นของปี 80 ในอดีตนั้นฐานข้อมูลหนึ่ง ๆ จะถูกนำมาใช้ทั้งในงาน โอเปอเรชั่นนอล และ การวิเคราะห์ข้อมูล ฐานข้อมูลส่วนใหญ่ก็มักจะถูกออกแบบมาเพื่อใช้กับระบบทรานแซคชั่น โพรเซสซิ่ง (transaction processing system) และการวิเคราะห์ข้อมูลก็จะถูกกระทำโดยมนุษย์ โดยทำการดึงข้อมูลมาจากระบบทรานแซคชั่น โพรเซสซิ่ง และเนื่องจากในอดีตนั้นราคาของซอฟแวร์และฮาร์ดแวร์ที่ใช้ในการบำรุงรักษาฐานข้อมูลนั้นมีราคาค่อนข้างสูง ฐานข้อมูลที่ใช้ในการนำเสนอข้อมูลที่ใช้ในการวิเคราะห์นั้นมีจำนวนเรคอร์ดที่ค่อนข้างจำกัด จึงทำให้การมีฐานข้อมูลเดียวที่รองรับการทำงานทั้งสองประเภทที่กล่าวมาจึงได้รับความนิยมค่อนข้างมาก

แต่ปัจจุบันเมื่อความต้องการทางด้านข้อมูลสูงขึ้น จึงทำให้เกิดข้อจำกัดทางเทคโนโลยีที่จะใช้งานฐานข้อมูลระหว่างทรานแซคชั่น และ การวิเคราะห์ข้อมูลพร้อม ๆ กัน ซึ่งโดยภาพรวมแล้ว ลักษณะของการนำโพรเซสซิ่งทั้งสองกระบวนการมาใช้นั้นมีความแตกต่างกันอย่างสิ้นเชิง

ทรานแซคชั่น โพรเซสซิ่ง ดาต้าเบส จะเกี่ยวข้องกับข้อมูลที่เป็นปัจจุบัน ซึ่งจะมีการเปลี่ยนแปลงของข้อมูลอยู่เรื่อย ๆ ดังนั้นจำนวนของข้อมูลที่มีปริมาณเพิ่มขึ้นจะเป็นตัวลดประสิทธิภาพของ โอเปอเรชั่นนอล ดาต้าเบส ซึ่งข้อมูลที่มีความซ้ำซ้อนจะต้องมีการจัดการที่ดี เพื่อป้องกันปัญหาการเปลี่ยนแปลงข้อมูล ซึ่งได้แก่กระบวนการเพิ่มข้อมูล (insert) อัพเดทข้อมูล(update) และ ลบข้อมูล(delete) ข้อมูลที่ซ้ำกัน

แต่สำหรับฐานข้อมูลที่ใช้ในการวิเคราะห์ข้อมูลเพื่อตอบคำถามนั้น มักจะมีการทำงานในลักษณะที่อ่านได้เพียงอย่างเดียว การเปลี่ยนแปลงข้อมูลนั้นเกิดขึ้นน้อยมาก ความซ้ำซ้อนของข้อมูลนั้นไม่ใช่ปัญหาของดาต้าแวร์เฮ้าท์ และมักจะเป็นฐานข้อมูลที่มีขนาดใหญ่เนื่องจากมีการเก็บข้อมูลเป็นเวลา การมีกระบวนการเก็บข้อมูลที่ดีจะมีผลผลทำให้ประสิทธิภาพของการตอบคำถามดีขึ้น

จากที่กล่าวถึงคุณลักษณะของทรานแซคชั่น โพสเซสซิ่งและดาต้าแวร์เฮ้าท์ เป็นเรื่องยากที่จะออกแบบฐานข้อมูลที่สามารถรองรับถึงคุณลักษณะของทั้งสองได้ ดาต้าแวร์เฮ้าท์ซิ่งนั้นจะรวบรวมข้อมูลทางธุรกิจหลาย ๆ แบบเข้าไว้ด้วยกัน เช่น ไฟล์ ฐานข้อมูลแบบเชิงสัมพันธ์ ,สปีรีดชีท ,แฟลท ดาต้าเบส และ เท็กไฟล์ทั่ว ๆไป และดาต้าแวร์เฮ้าท์ซิ่งจะดึงข้อมูลจากฐานข้อมูลที่มีชนิดต่าง ๆ กัน ไปเก็บในโครงสร้างฐานข้อมูลอันใหม่ ทำให้กลุ่มของผู้ใช้งานสามารถใช้งานทุก ๆ ข้อมูลจากฐานข้อมูลเพียงอันเดียว

เพราะฉะนั้นแวร์เฮ้าท์ดาต้าเบส จึงใช้วิธีการหลาย ๆ อย่างในการจัดการกับข้อมูล โดยดาต้าแวร์เฮ้าท์ดึงข้อมูลมาจากระบบ ทรานแซคชั่น โพรเซสซิ่ง แปลงข้อมูลให้อยู่ในรูปแบบที่เหมาะสมแล้วจึงทำการจับคู่กับสกีมาที่ต่างออกไปจากสกีมาเดิม ที่สามารถทำงานได้ดีกับปริมาณข้อมูลขนาดใหญ่และทำให้การคิวรี่มีประสิทธิภาพดีขึ้น

สำหรับการออกแบบดาต้าแวร์เฮ้าท์ ได้มีการใช้โมเดลหลาย ๆ แบบในการแก้ปัญหาของดาต้าแวร์เฮ้าท์ ซึ่งก็มีโมเดลที่โด่งดังมีการใช้งานอย่างแพร่หลายก็คือ โมเดลรูปดาว หรือที่เรียกกันว่า สตาร์ สกีมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(star schema) ซึ่งใช้หลักการ ดีนอมาไลซ เทเบิ้ล (de-normalized table) และ สโนว์เฟลก สกีมา(Snowflex Schema)ที่แปลงมาจาก สตาร์ สกีมา ใช้การเลือก นอมาไลเซชั่น เฉพาะบาง ไดเมนชั่น เทเบิ้ล (dimension table)

ฐานข้อมูลแบบวัตถุเชิงสัมพันธ์นั้นเป็นการผสมผสานกันระหว่างแนวความคิดในการออกแบบฐานข้อมูลเชิงวัตถุที่มีการใช้งานแบบซับซ้อนและ มีการประกาศโครงสร้างของข้อมูลประเภทใหม่ (user-defined data structure) กับข้อเด่นของฐานข้อมูลเชิงสัมพันธ์ เช่น การนำเสนอข้อมูลอย่างง่าย ๆ ในรูปของตาราง (tabular form) และ สนับสนุนของภาษาเอสคิวแอล(Structure Query Language) สำหรับการประกาศชนิดของข้อมูล(data definition)และการเปลี่ยนแปลงข้อมูล (data manipulation) ทำให้ได้ลักษณะเฉพาะของฐานข้อมูลแบบวัตถุเชิงสัมพันธ์ คือ 1 . การประกาศชนิดของข้อมูลไว้ใช้เอง 2 . สนับสนุนคอมเพล็กออบเจ็ก (complex object) 3 . การสืบทอดของชนิดของข้อมูล

ผลงานในปัจจุบันที่เกี่ยวกับ ดาต้าแวร์เฮ้าท์ซึ่ง ส่วนใหญ่จะกล่าวถึงเทคโนโลยีแบบรีเลชั่นนอล และมีบางอันที่ใช้เทคโนโลยีแบบออบเจ็ก ซึ่งการออกแบบดาต้าแวร์เฮ้าท์ดาต้าเบสบนเทคโนโลยีแบบทั้งสองนั้นจะแสดงในรูปแบบของโมเดลในรูปดาว แต่เนื่องจากโมเดลรูปดาวที่แต่เดิมจะแสดงให้เห็นในรูปแบบของลอจิคคอล ดาต้า โมเดล(Logical Data Model) ไม่สามารถที่จะแสดงให้เห็นถึงความสัมพันธ์ของแอททริบิวต์ได้ จึงได้มีการพัฒนาโมเดลรูปดาว ให้อยู่ในรูปแบบของ อีอาร์ไดอะแกรม ซึ่งเป็นโมเดลที่นิยมนำมาใช้ในการออกแบบโอเปอร์เรชั่นนอล ดาต้าเบสอยู่แล้วโดยเรียกโมเดลนี้ว่า สตาร์อีอาร์ โมเดล (Micheal K. and II-Yeol Song) แต่สำหรับในฐานข้อมูลแบบวัตถุสัมพันธ์ นั้นสตาร์อีอาร์ โมเดลนั้นจะไม่สามารถแสดง คอมโพสิท แอททริบิทต์ มัลติแวลู แอททริบิวต์ และ ไม่สามารถแสดงการสืบทอดคุณสมบัติของแต่ละ เอ็นทิตี้ไทป์ได้ ซึ่งเป็นส่วนสำคัญของฐานข้อมูลแบบวัตถุเชิงสัมพันธ์ ปริญญานิพนธ์นี้ได้พัฒนาโมเดลเพิ่มขึ้นในส่วนแสดงคอมโพสิท แอททริบิทต์ มัลติแวลู แอททริบิทต์ และ การสืบทอดของแต่ละ เอ็นตีตี้ไทป์ และยังได้แสดงให้เห็นถึงการใช้คำสั่งเอสคิวแอลในการเข้าถึงข้อมูลที่ถูกจัดอยู่ในรูปแบบของเทคโนโลยี ออปเจ็ครีเลชั่นนอล ซึ่งจะแสดงในบทที่ 4 และบทที่ 5

1.2 วัตถุประสงค์ของงานวิจัย

1.2.1 ศึกษาและพัฒนาวิธีการออกแบบฐานข้อมูลเชิงวัตถุสัมพันธ์ เพื่อประยุกต์ใช้งานกับดาต้าแวร์เฮ้าท์จริง โดยใช้โปรแกรม IBM DB2 เวอร์ชั่น 8.1

1.2.2 สร้างต้นแบบบฐานข้อมูลแบบวัตถุเชิงสัมพันธ์ เพื่อใช้งานบนดาต้าแวร์เฮ้าท์ โดยใช้วิธีการออกแบบที่ได้พัฒนาขึ้นมา

1.3 ขอบเขตของงานวิจัย

งานวิจัยนี้ได้ศึกษาถึงระบบฐานข้อมูลแบบเชิงวัตถุที่ใช้งานบนดาต้าแวร์เฮ้าท์ ในกรณีศึกษาที่เหมาะสมสำหรับใช้งานระบบฐานข้อมูลแบบวัตถุเชิงสัมพันธ์ โดยรูปแบบของระบบฐานข้อมูลดังกล่าวจะถูกปรับปรุงโดยการนำเอาแบบจำลองอีอาร์ (ER - Diagram) ของปีเตอร์-เชน ซึ่งมักจะถูกนำมาใช้ในการออกแบบฐานข้อมูลเชิงสัมพันธ์มาประยุกต์กับแนวความคิดในการออกแบบดาต้าแวร์เฮ้าท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งมักจะใช้วิธีการของโมเดลรูปดาว เพื่อที่จะแสดงให้เห็นถึงความสัมพันธ์ของแอททริบิวต์ภายในแต่ละไดเมนชั่น และนอกจากนี้ยังได้พัฒนาฐานข้อมูลนี้โดยใช้แนวความคิดแบบฐานข้อมูลเชิงวัตถุสัมพันธ์

1.4 วิธีการดำเนินงาน

งานวิจัยนี้เริ่มด้วยการศึกษาทฤษฎีพื้นฐานต่างๆที่เกี่ยวข้องกับงานวิจัย ซึ่งก็มีเรื่องหลัก ๆ อยู่ 2 เรื่องด้วยกัน คือ ดาต้าแวร์เฮ้าท์ ฐานข้อมูลแบบวัตถุเชิงสัมพันธ์ ซึ่งมีรายละเอียดดังในบทที่ 2 และ 3 จากนั้นก็เอาความรู้ที่ได้ศึกษาทั้งหมดมาพัฒนาออกแบบ ดาต้าแวร์เฮ้าท์ใหม่ที่ใช้ฐานข้อมูลแบบวัตถุเชิงสัมพันธ์ ซึ่งมีรายละเอียดในบทที่ 4 และ การใช้งานจริงบนดาต้าแวร์เฮ้าท์ โดยใช้ IBM DB2 เวอร์ชั่น8.1 ซึ่งมีรายละเอียดในบทที่ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ดาต้า แวร์เฮ้าท์ซิ่ง

จากการศึกษางานวิจัยต่าง ๆ มีงานวิจัยหลาย ๆ งานที่พยายามสร้างดาต้าแวร์เฮ้าท์บนเทคโนโลยีเชิงวัตถุสัมพันธ์ และมีหลายงานวิจัยที่เสนอแบบจำลองฐานข้อมูลเชิงวัตถุสัมพันธ์ สำหรับใช้งานกับดาต้าแวร์เฮ้าท์ซิ่ง ซึ่งทำให้รู้ว่าฐานข้อมูลแบบวัตถุเชิงสัมพันธ์ได้มีการใช้งานร่วมกับดาต้าแวร์เฮ้าท์ทางธุรกิจมาแล้วไม่น้อยกว่า 4 ปี

ในบทนี้จะแบ่งออกเป็น 2 ส่วน โดยส่วนแรกจะกล่าวถึงดาต้าแวร์เฮ้าท์ซิ่งและประโยชน์ของดาต้าแวร์เฮ้าท์ซิ่ง และกล่าวถึงความแตกต่างระหว่าง แวร์เฮ้าท์ ดาต้าเบส (warehouse database) กับ โอเปอร์เรชั่นนอล ดาต้าเบส (operational database) ส่วนที่สองกล่าวถึงชนิด ต่าง ๆ ของแบบจำลองข้อมูล (data model) ที่ใช้งานกับ แวร์เฮ้าท์ดาต้าเบส พร้อมทั้งศึกษาข้อดีและข้อเสียของแต่ละชนิด

2.1 ดาต้าแวร์เฮ้าท์ซิ่งและประโยชน์ของดาต้าแวร์เฮ้าท์ซิ่ง

ดาต้าแวร์เฮ้าท์จะเป็นที่เก็บของข้อมูล ซึ่งรวบรวมมาจากข้อมูลดิบที่มาจากที่ต่าง ๆ เพื่อประโยชน์ในการตอบคำถามและการวิเคราะห์ นิยามของดาต้าแวร์เฮ้าท์ซิ่งได้รับการเสนอครั้งแรกโดย Inmon and Kelly 1993

โอเปอเรชั่นนอลหรือระบบทรานแซคชั่น จะเป็นตัวช่วยในการทำงานขององค์กรโดย แต่ระบบนั้นจะไม่สามารถจัดการข้อมูลจำพวก การวางแผน การคาดคะเนแนวโน้ม และ การวิเคราะห์ข้อมูลทางธุรกิจ โครงสร้างของข้อมูลประเภทโอเปอเรชั่นนอล มีความซับซ้อนค่อนข้างมากเพราะจะต้องมีการนอมาไลซ์ (normalize) ข้อมูลก่อนจัดเก็บ ทำให้ข้อมูลนั้นไม่เหมาะสมกับการนำเสนอกับผู้ใช้ (Chelluri and Kumar 2001)

แต่สำหรับข้อมูลที่อยู่ในดาต้าแวร์เฮ้าท์มีการจัดเก็บข้อมูลในอดีตเพื่อให้นักวิเคราะห์ระบบใช้วิเคราะห์ถึงประสิทธิภาพในอดีตขององค์กรเพื่อทำการวางแผนกิจกรรมในอนาคตขององค์กร ทำให้นักวิเคราะห์ต้องมีการศึกษาข้อมูลที่เก็บไว้ขนาดเทราไบท์ (terabyte) ในฐานข้อมูลจำนวนมาก Hanson 1997 ได้นำเสนอแนวความคิดสองวิธีการที่จะนำเสนอข้อมูลจากหลาย ๆ ฐานข้อมูลที่มีในดาต้าแวร์เฮ้าท์ให้กับกลุ่มผู้ใช้ โดยที่กลุ่มผู้ใช้จะทำการติดต่อเพียงครั้งเดียว วิธีการแรก คือ การรวบรวมแต่ละฐานข้อมูลเข้าไว้ด้วยกันเป็นฐานข้อมูลแบบเชิงสัมพันธ์ หรือ ระบบออบเจ็กออเรนเทิต มัลติดาต้าเบส (object-oriented Multidatabase) ที่บรรจุข้อมูลที่สำคัญที่เข้าถึงฐานข้อมูลถัดไป หรือเข้าถึงดาต้าซอร์ด(data source) วิธีการที่สอง คือ การรวบรวมข้อมูลที่สำคัญมาเป็นฐานข้อมูลใหม่ วิธีที่สองนั้นเป็นถูกนำมาใช้กับงานทางด้านทางธุรกิจภายในรูปแบบของดาต้าแวร์เฮ้าท์ซิ่ง

เนื่องมาจากว่าหาเราสร้างฐานข้อมูลหนึ่งขึ้นมาเพื่อทำงานทั้งทรานแซคชั่นแบบออนไลท์ และวิเคราะห์ข้อมูลสำหรับตัดสินใจ จะเป็นการลดประสิทธิภาพของการทำระบบ โอเปอเรชั่นนอล และ การตอบสนองกับผู้ใช้ (Chelluri and Kumar 2001)

Furlow อธิบายถึง ข้อดี 5 ประการของดาต้าแวร์เฮ้าท์ซิ่ง ข้อแรก คือ ดาต้าแวร์เฮ้าท์ซิ่งนั้นจะแสดงข้อมูลในมิติเดียวโดยจะดึงข้อมูลมารวมกันจากฐานข้อมูลต่าง ๆ จากหลาย ๆ ที่ของภายในองค์กร

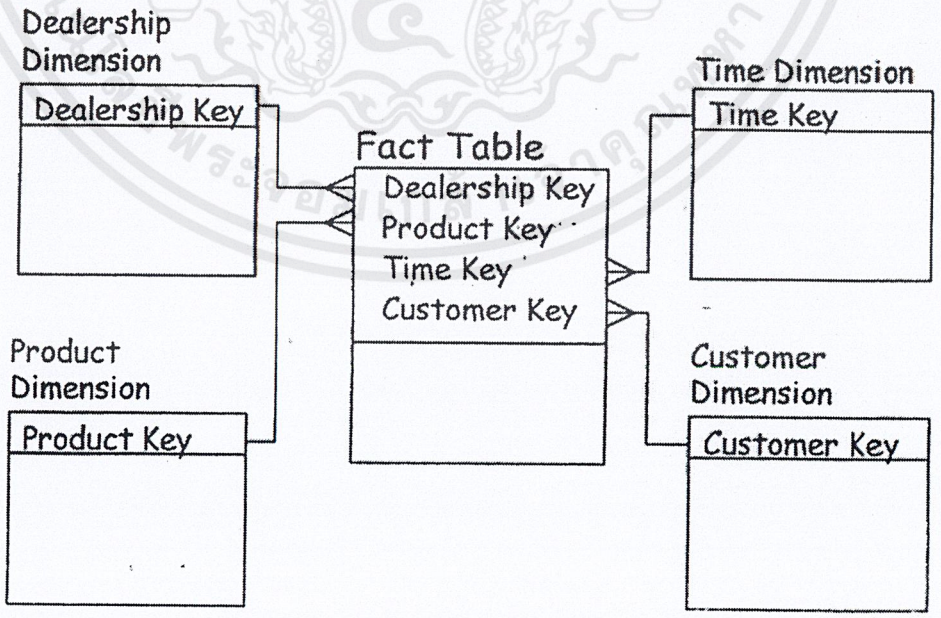
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้การตัดสินใจนั้นง่ายขึ้น ข้อสอง ง่ายต่อการเข้าถึงข้อมูล ซึ่งจะนำไปสู่การเพิ่มปริมาณการผลิต และผลกำไรที่เพิ่มขึ้น ข้อสาม ประสิทธิภาพที่เพิ่มขึ้นของระบบโอเปอเรชั่นนอล ข้อสี่ ความยืดหยุ่นของดาต้าแวร์เฮ้าท์ที่ยอมให้มีการเปลี่ยนแปลงโครงสร้างภายใน ซึ่งจะสามารถทำการเปลี่ยนได้โดยทั้งจากระบบคอมพิวเตอร์ และ โครงสร้างทางธุรกิจ ข้อสุดท้าย ดาต้าแวร์เฮ้าท์ทำให้องค์กรสามารถจัดการกับข้อมูลได้ดีขึ้นทำให้เป็นผลดีในการแข่งขันกันทางธุรกิจ ทำให้เข้าใจความต้องการของลูกค้า และ ทำให้ถึงความต้องการของตลาด

Furlow 2001 ได้กล่าวถึงข้อเสียของดาต้าแวร์เฮ้าท์ คือ ความซับซ้อนตัวดาต้าแวร์เฮ้าท์ และความยุ่งยากในการพัฒนาตัวระบบ การสร้างระบบดาต้าแวร์เฮ้าท์หนึ่ง ๆ จะต้องใช้เงินและเวลาอย่างมาก

2.2 รูปแบบของดาต้าแวร์เฮ้าท์

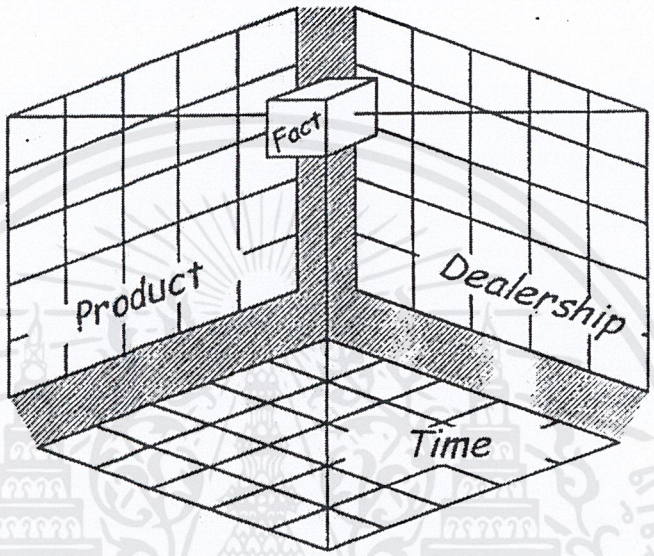
Kimball 1997 อธิบายเกี่ยวกับไดเมนชั่นนอล โมเดลลิ่ง (Dimensional Modeling) ว่าเป็นเทคนิคในการออกแบบดาต้าแวร์เฮ้าท์ซึ่งระดับตรรกะ ไดเมนชั่นนอล โมเดลลิ่ง พยายามนำเสนอข้อมูลในรูปแบบมาตราฐาน ซึ่งมีประสิทธิภาพสูงในการเข้าถึงข้อมูล ที่มีมากับไดเมนชั่นนอลและประกอบกับระเบียบที่ใช้กับ รีเลชั่นนอล โมเดล ทุก ไดเมนชั่นนอล โมเดล จะประกอบไปด้วย ตารางหนึ่งอันที่เก็บมัลติพาท คีย์ (multipart key) ซึ่งในที่นี้จะเรียกว่าแฟค เทเบิ้ล (fact table) และเซตของตารางเล็ก ๆ ซึ่งจะเรียกว่าไดเมนชั่น เทเบิ้ล (Dimension table) โดยแต่ละ ไดเมนชั่น เทเบิ้ล นั้นจะมีไพรมารี คีย์ ที่จะสอดคล้องกับหนึ่งในชุดมัลติพาท คีย์ที่อยู่ในแฟค เทเบิ้ล ลักษณะของโครงสร้างจะมีลักษณะคล้ายกับดาว ดังนั้นไดเมนชั่นนอล ดาต้า โมเดล จึงได้ถูกเรียกว่าโมเดลรูปดาว หรือสตาร์ สกีมา ประโยชน์ของไดเมนชั่นนอล โมเดล นั้นจะอยู่ที่ ความยืดหยุ่น ความสามารถในการปรับตัว ความสามารถในการขยาย และการทำให้อยู่ในรูปมาตราฐาน รูปที่ 2.1 เป็นรูปแสดงการออกแบบดาต้าแวร์เฮ้าท์โดยใช้สตาร์ สกีมา



รูปที่ 2-1 แสดงการใช้โมเดลรูปดาวในการออกแบบฐานข้อมูลแบบมัลติไดเมนชั่นนอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Agrawal et al 1997 ได้อธิบายถึงเทคนิคอีกอย่างในการแสดงภาพลักษณ์ของไดเมนชั่นนอล ดาต้า โมเดล หรือ มัลติไดเมนชั่นนอล ดาต้า โมเดล (multidimensional data model) ซึ่งจะแสดงอยู่ในรูปของทรงลูกบาศก์ โดยข้อมูลจะอยู่ในเซลล์ของทรงลูกบาศก์ แกนต่าง ๆ จะแสดงถึงเอนทิตี้หรือตัวแปรซึ่งจะเรียกว่าไดเมนชั่น ดังรูปที่ 2.2



รูปที่ 2-2 ทรงลูกบาศก์แสดงมัลติไดเมนชั่นนอลดาต้าเบส

จากแบบจำลองข้อมูลที่กล่าวมาไม่ว่าจะเป็น ไดเมนชั่นนอล โมเดลลิ่ง และ มัลติไดเมนชั่นนอล ดาต้า โมเดลลิ่ง จะมีโครงสร้างที่คล้ายกันเพียงแต่ใช้เทคนิคที่ต่างกัน ไดเมนชั่น โมเดลลิ่ง ใช้ฐานข้อมูลแบบเชิงสัมพันธ์ แต่ มัลติไดเมนชั่นนอล ดาต้า โมเดลลิ่ง จะใช้ฐานข้อมูลแบบหลายมิติ Trujillo et al. 2001 เสนอว่า มัลติไดเมนชั่นนอล โมเดลลิ่งนั้น เหมาะกับฐานข้อมูลแบบวัตถุ และ Giovinazzo 2000 นั้นเสนอ รีเลชั่นนอล ไดเมนชั่น โมเดลลิ่ง ในการใช้งานกับฐานข้อมูล เพราะว่าจำนวนเซลล์ของมัลติไดเมนชั่นนอล ดาต้าเบส นั้นจะมีเซลล์ที่ว่างอยู่ทำให้มีการเสียพื้นที่ในการเก็บข้อมูลไปไม่น้อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
ฐานข้อมูลเชิงวัตถุสัมพันธ์

ในโลกธุรกิจของระบบฐานข้อมูลมีผลิตภัณฑ์อยู่ไม่มากที่เป็นที่ยอมรับกัน ฐานข้อมูลที่มีการใช้งานกันอย่างเด่นชัดก็คือ ฐานข้อมูลแบบเชิงสัมพันธ์ และ ฐานข้อมูลแบบวัตถุ

ในช่วงไม่กี่ปีที่ผ่านมานี้บริษัทซอฟแวร์ใหญ่ ๆ อย่าง IBM ,Informix ,Oracle และ Sybase ได้สร้างผลิตภัณฑ์ที่สนับสนุนการทำงานฐานข้อมูลในรูปแบบของ ออบเจ็กรีเลชั่น ของตัวเองขึ้นมา บริษัทเหล่านี้ได้โฆษณาถึงผลิตภัณฑ์ที่ออกมาใหม่ของตนเอง ว่าเป็นเทคโนโลยีถัดมาจากเทคโนโลยีแบบรีเลชั่น ซึ่งก็คือ เทคโนโลยีแบบออบเจ็กรีเลชั่น จุดประสงค์หลักของการสร้างนั้นก็เพื่อลดข้อจำกัดของเทคโนโลยีแบบรีเลชั่น ที่ไม่สามารถทำงานกับแอพพลิเคชั่นใหม่ ๆ ได้ เช่น การเก็บรูปภาพ และ การเก็บมัลติมีเดีย ออบเจ็ก ลงในฐานข้อมูล เพราะ ออบเจ็ก และ โอเปอเรชั่นที่เกี่ยวข้อง จะมีความซับซ้อนมากตามไป ส่วนฐานข้อมูลแบบวัตถุนั้นถูกออกแบบมาเพื่อรองรับกับ ออบเจ็ก และ โอเปอเรชั่นที่เกี่ยวข้องกับ รูปภาพ , ระบบข้อมูลทางภูมิศาสตร์ ,มัลติมีเดียออบเจ็ก ,3-D และ ข้อมูลที่ต้องเกี่ยวข้องกับเวลา แต่เทคโนโลยีทั้งสองแบบก็ไม่สามารถทำงานในส่วนที่ต้องการได้ เป็นเหตุผลสำคัญที่นำไปสู่การพัฒนาเทคโนโลยีแบบใหม่ขึ้นมา นั่นก็คือ เทคโนโลยีแบบออบเจ็กรีเลชั่น ซึ่งเป็นการรวมเอาความสามารถของเทคโนโลยีแบบรีเลชั่น และ เทคโนโลยีแบบออบเจ็ก เข้าไว้ด้วยกัน

ในบทนี้ได้เปรียบเทียบถึงความแตกต่างระหว่างเทคโนโลยีแบบออบเจ็กรีเลชั่นกับเทคโนโลยีแบบรีเลชั่นในส่วนที่ได้พัฒนาเพิ่มเข้ามาและส่วนของประสิทธิภาพของเทคโนโลยีแบบออบเจ็ก แนวความคิดของฐานข้อมูลแบบเชิงสัมพันธ์ กับ ฐานข้อมูลแบบเชิงวัตถุจะอยู่ในส่วน 3.1 และ 3.2 ส่วน 3.3 นั้นจะกล่าวถึงแนวความคิดของฐานข้อมูลเชิงวัตถุสัมพันธ์ และข้อดีของฐานข้อมูลเชิงวัตถุสัมพันธ์ ที่เหนือกว่าฐานข้อมูลแบบเชิงสัมพันธ์ และ ฐานข้อมูลแบบวัตถุ

3.1 ฐานข้อมูลแบบเชิงสัมพันธ์ (RDB)

รีเลชั่นนอล โมเดลได้รับการเสนอโดย Dr. E. F. Codd ในปี ค.ศ.1970 และมีการพัฒนาตั้งแต่นั้นเรื่อยมา โดย IBM Oracle และบริษัทอื่น ๆ ส่วนมาตราฐานของฐานข้อมูลแบบเชิงสัมพันธ์ถูกกำหนดโดย ANSI (the American National Standard Institute) เป็นเอสคิวแอล (ANSI 1986) หรือ SQL1 เรียกว่า SQL-86 และมาตราฐานอันถัดมาเรียกว่า SQL2 หรือ SQL-92

ฐานข้อมูลแบบเชิงสัมพันธ์จะประกอบด้วยแถวและ คอลัมน์ เป็นความสัมพันธ์ในรูปของตารางข้อมูลที่อยู่ในตารางนั้นเป็นมุมมองแบบลอจิกคัลของฐานข้อมูลเพื่อแสดงให้ผู้ใช้งานและโปรแกรมเมอร์ดู ส่วนข้อมูลที่เก็บอยู่ในระบบดิสคอมพิวเตอร์นั้นจะเรียกว่ามุมมองแบบอินเทอนอล ฟิลด์ในเรคคอร์ดจะเรียกว่าแอททริบิวต์ แต่ละคอลัมน์จะมีชนิดของข้อมูล เช่น ข้อมูลชนิดตัวเลข ข้อมูลชนิดอักขระ ข้อมูลชนิดวันที่ เป็นต้น ซึ่งก่อนที่จะมีการเก็บข้อมูลลงฐานข้อมูลจะต้องมีการตรวจสอบความถูกต้องข้อมูลก่อน เรียกว่า เงื่อนไข มีทั้งเงื่อนไขแบบโดเมน ,เงื่อนไขแบบคีย์ ,เงื่อนไขแบบเอนตีตี้ อินทีกริตี้ (Entity integrity),และ เงื่อนไขแบบเรฟเฟอเรนเชียว อินทีกริตี้ (Referential Integrity) ซึ่งทำให้แน่ใจได้ว่าจะไม่มีข้อมูลที่ผิดปกติอยู่ในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฐานข้อมูลแบบเชิงสัมพันธ์ใช้ภาษาเอสคิวแอล(Structure Query Language : SQL) เป็น ภาษาดาต้า ดิฟฟินิชั่น(Data Definition : DDL) หรือเป็นภาษาดาต้า แมนิพพูเลชั่น(Data Manipulation : DML) ซึ่งภาษาเอสคิวแอลนั้นจะรวมคำสั่งที่ใช้ในการกำหนดชนิดของข้อมูล ,คำสั่งที่ใช้การเปลี่ยนแปลงข้อมูล , คำสั่งคิวรี่ และ คำสั่งที่ใช้กำหนดเงื่อนไข การคิวรี่นั้นจะแบ่งตามลักษณะของการคิวรี่ ถ้าเป็นการคิวรี่ตารางเพียงอันเดียวก็ยังง่ายอยู่ แต่ถ้ามีการคิวรี่ตารางหลาย ๆ ตาราง ก็จะกลายเป็นเรื่องที่ยุ่งยากเพราะจะต้องมีการเชื่อมตาราง,การทำเนสติ้ง,การทำข้อมูลที่เป็นเซต และ อื่น ๆ กระบวนการทั้งหมดจะกระทำบนข้อมูลในฟิลด์ของเรคคอร์ด ตัวอย่างของโปรแกรมพัฒนาฐานข้อมูลแบบเชิงสัมพันธ์ที่มีอยู่ในปัจจุบันได้แก่ ออราเคิลของบริษัทออราเคิล ,DB2 ของบริษัทไอบีเอ็มและ Microsoft SQL Server ของบริษัทไมโครซอฟท์ และด้วยมาตรฐานของเอสคิวแอลทำให้ผู้ใช้งานง่ายต่อการย้ายดาต้าเบส แอพพลิเคชั่นข้ามระบบฐานข้อมูลของแต่ละบริษัทได้ ตัวอย่างเช่น ผู้ใช้สามารถเข้าถึงข้อมูลที่เก็บไว้แต่ละฐานข้อมูลได้โดยไม่ต้องเปลี่ยนแปลงคำสั่งของเอสคิวแอล ข้อดีอีกอย่างคือการเข้าถึงข้อมูลได้เร็วแม้ว่าฐานข้อมูลจะมีขนาดใหญ่ ข้อเสียหลักของฐานข้อมูลแบบเชิงสัมพันธ์ คือ ไม่สามารถเก็บข้อมูลประเภทรูปภาพได้ ไม่สามารถใช้ชนิดข้อมูลแบบพิเศษได้ เช่น จำนวนเชิงซ้อน, อาเรย์ และแอพพลิเคชั่นที่มีการเชื่อมโยงของข้อมูลที่ซับซ้อน (Elmasri และ Navathe 2000)

3.2 ฐานข้อมูลแบบวัตถุ (OODB)

ความมุ่งหวังที่จะนำเสนอโครงสร้างข้อมูลที่ซับซ้อนให้ง่ายนั้นทำให้เกิดแนวความคิดการออกแบบระบบเชิงวัตถุขึ้นมาและนำไปสู่การพัฒนาฐานข้อมูลเชิงวัตถุ แนวความคิดของแอปสแตรค ดาต้า ไท(Abstract Data Type) ได้ถูกพัฒนาขึ้นโดยการซ่อนโครงสร้างของข้อมูลเอาไว้และให้มีโอเปอเรชั่นภายนอกทำการประยุกต์ใช้ผ่านทางออบเจ็กที่กำหนดไว้ ซึ่งกลายเป็นแนวความคิดของเอนแคปซูเลชั่น ลักษณะสำคัญของภาษาประเภทออบเจ็ก คือ

1. เอนแคปซูเลชั่น เหมือนกับชั้นตรวจสอบที่ใช้ในการป้องกันโค้ดและข้อมูลจากโค้ดอื่นที่ประกาศไว้ด้านนอกชั้นตรวจสอบ
2. อินเฮอริแท่น เป็นการรับเอาคุณสมบัติของออบเจ็กต้นแบบมาใช้และสามารถสร้างคุณสมบัติใหม่เอาไว้ได้ด้วย
3. โพลิมอฟิซึม (Bahrami 1999) เป็นอนุญาตให้โอเปอเตอร์ตัวเดียวกันสามารถทำงานคนละแบบได้โดยขึ้นอยู่ชนิดของออบเจ็กที่มีโอเปอเรเตอร์

ฐานข้อมูลแบบวัตถุจะทำงานกับแบบจำลองข้อมูล ที่สนับสนุนคุณลักษณะของออบเจ็กอย่างเช่น แอปสแตรค ดาต้า ไท ฐานข้อมูลแบบวัตถุจะมีการสร้างตัวชี้เฉพาะข้อมูล(Object Identify:OID) มาให้ ดังนั้นแต่ละออบเจ็กจะไม่มีการซ้ำกัน ซึ่งตัวชี้เฉพาะข้อมูลจะคล้ายกับคีย์หลักในแบบรีเลชั่น โมเดลข้อมูลในฐานข้อมูลจะได้รับการจัดการผ่านเซตของความสัมพันธ์สองอย่าง

1. ความสัมพันธ์ที่บรรยายถึงความสัมพันธ์ภายในของดาต้า ไอเท็ม
2. ความสัมพันธ์ที่บรรยายถึงความสัมพันธ์แบบอินเฮอริแท่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบจะใช้ประโยชน์ของความสัมพันธ์ทั้งสองชนิดด้วยการจับคู่ดาต้า ไอเท็ม(Data Item) กับโพรซิเดอรอล เมธอด(Procedural Method) ผลคือจะมีการสร้างความสัมพันธ์ขึ้นโดยตรงระหว่างแอพพลิเคชั่น ดาต้า โมเดล กับ ดาต้าเบส ดาต้า โมเดล ความสัมพันธ์ที่เชื่อมโยงกันระหว่างแอพพลิเคชั่นกับฐานข้อมูลนั้นจะแสดงด้วยโค้ดที่ความกระชับ รัดกุม,โครงสร้างข้อมูลอย่างง่าย ๆ ,การดูแลรักษา และการนำเอาโค้ดกลับมาใช้อีก ภาษาประเภทออบเจ็ก อย่างเช่น C++ , Java นั้นสามารถที่จะทำการลดขนาดของโค้ดให้เล็กลงได้โดยไม่ต้องแปลเป็นภาษาทางดาต้าเบส

อุปสรรคที่สำคัญของฐานข้อมูลแบบวัตถุคือประสิทธิภาพในการเข้าถึงข้อมูลต่ำ ซึ่งไม่เหมือนกับฐานข้อมูลแบบรีเลชั่น เพราะการคิวรี่ของระบบแบบออบเจ็กนั้นจะมีความซับซ้อนอย่างมาก ดังนั้นระบบแบบออบเจ็กจึงไม่เหมาะสมฐานข้อมูลที่มีขนาดใหญ่

3.3 ฐานข้อมูลเชิงวัตถุสัมพันธ์ (ORDB)

จุดประสงค์ในการออกแบบระบบแบบออบเจ็กรีเลชั่น ก็เพื่อรวมเอาข้อดีของระบบแบบรีเลชั่นกับระบบแบบออบเจ็ก อย่างเช่น สกาลาบิลิตี้(Scalability) และการสนับสนุนการสร้างชนิดข้อมูล ฐานข้อมูลเชิงวัตถุสัมพันธ์นั้นจะอาศัยดาต้าโมเดลที่พยายามที่จะรวบรวมเอาคุณลักษณะของออบเจ็กไปไว้ในฐานข้อมูลแบบเชิงสัมพันธ์ ข้อมูลจะถูกเก็บอยู่ในรูปของตาราง แต่สำหรับตารางบางอันข้อมูลที่จัดเก็บในตารางโครงสร้างข้อมูลจะอยู่ในรูปของแอบสแตรค ดาต้า ไท ซึ่งฐานข้อมูลเชิงวัตถุสัมพันธ์จะต้องสนับสนุนรูปแบบของเอสคิวแอลด้วย ดังนั้นจึงได้มีการออกมาตราฐานเอสคิวแอลมาใหม่ในปี 1999 เรียกว่า SQL3 หรือ SQL-99 ฐานข้อมูลเชิงวัตถุสัมพันธ์นั้นจะมีรูปแบบเป็นรีเลชั่นนอล โมเดล เพราะข้อมูลจะถูกเก็บอยู่ในฟอร์มของตารางมีแถวและคอลัมน์ และใช้เอสคิวแอลเป็นภาษาในการคิวรี้ และการแสดงผลของการคิวรี้ก็จะแสดงผลเป็นตาราง หรือเซตของคู่ลำดับ แต่รีเลชั่นนอล โมเดลต้องได้รับการปรับปรุงอย่างมากในส่วนที่สนับสนุนถึงคุณลักษณะของออบเจ็ก โปรแกรมมิ่ง

Stonebraker และ Brown 1999 ได้กล่าวถึงคุณลักษณะที่สำคัญของฐานข้อมูลเชิงวัตถุสัมพันธ์ว่า

1. การสร้างชนิดข้อมูล
2. สนับสนุนออบเจ็กที่ซับซ้อน
3. การอินเฮอริแท่น
4. การปฏิบัติกฎของระบบ

ฐานข้อมูลเชิงวัตถุสัมพันธ์จะยอมให้ผู้ใช้งานสามารถประกาศชนิดของข้อมูล ,ฟังก์ชั่น และโอเปอเรเตอร์ เองได้ เป็นคุณลักษณะของฐานข้อมูลเชิงวัตถุสัมพันธ์เพื่อเพิ่มประสิทธิภาพโดยรวมของระบบ

ตัวอย่างสกีมาของลูกจ้างที่ ฐานข้อมูลเชิงวัตถุสัมพันธ์รองรับ

EMPLOYEE(ID, Name (last, first, middle), Sex, BirthYear, Address (Street, City, State, Zip), Phones (Location, Number), DepartName, DateOfHire, Salary, Picture)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งในตัวอย่างนั้นจะเห็นได้ว่า Name และ Address เป็นคอมโพสิตแอตทริบิว หมายความว่าแอตทริบิวนั้นประกอบขึ้นจากหลาย ๆ แอตทริบิวต์, ส่วนPhones เป็นคอมโพสิตแอตทริบิวและมัลติเวลู มัลติเวลู หมายความ มีหลายข้อมูลอยู่ภายในแอตทริบิวเดียวกัน เพราะคน ๆ เดียวสามารถมีเบอร์โทรศัพท์ได้เบอร์ และ Picture เป็นแอตทริบิตประเภทรูปภาพ มีบางแอตทริบิวจากตัวอย่างที่สนับสนุนด้วยเทคโนโลยีแบบรีเลชั่น

Stonebraker และ Brown 1999 ได้แบ่ง DBMS แอพพลิเคชันเป็น 4 ชนิดด้วยกัน

1. ข้อมูลธรรมดาที่ไม่มีการคิวรี้
2. ข้อมูลธรรมดาที่มีการคิวรี้
3. ข้อมูลซับซ้อนที่ไม่มีการคิวรี้
4. ข้อมูลซับซ้อนที่มีการคิวรี้

จากการแบ่งประเภทของข้อมูลออกเป็น 4 ประเภททำให้สามารถอธิบายลักษณะของข้อมูลในฐานข้อมูลประเภทต่าง ๆ ได้ คือ ฐานข้อมูลประเภทรีเลชั่นนอลนั่นเจะเก็บข้อมูลชนิดข้อมูลธรรมดาที่มีการคิวรี้ ส่วนฐานข้อมูลประเภทออบเจ็กรีเลชั่นก็จะเก็บข้อมูลประเภทข้อมูลซับซ้อนที่มีการคิวรี้

ข้อดีสำคัญของฐานข้อมูลเชิงวัตถุสัมพันธ์คือสกาลาบิลิตี้ขนาดใหญ่ ที่จัดการกับข้อมูลขนาดใหญ่ถึงจะมีข้อดีหลายอย่าง แต่ฐานข้อมูลเชิงวัตถุสัมพันธ์ก็มีข้อเสียเหมือนกันคือสถาปัตยกรรมแบบออบเจ็กรีเลชั่นนอล โมเดลไม่เหมาะสำหรับเวป แอพพลิเคชั่นที่ต้องการความเร็วสูง แต่ข้อดีข้ออื่น เช่น พื้นที่ในการเก็บข้อมูลที่มีขนาดใหญ่ ความเร็วในการเข้าถึงข้อมูล และความสามารถในการจัดข้อมูลของฐานข้อมูลแบบวัตถุ ทำให้ฐานข้อมูลเชิงวัตถุสัมพันธ์น่าจะสามารถยึดครองตลาดฐานข้อมูลได้ จากการสนับสนุนของผู้ขายผลิตภัณฑ์หลัก ๆ ของระบบฐานข้อมูล และ คุณลักษณะของระบบออบเจ็กรีเลชั่นนอลเองที่จะทำให้ออบเจ็กรีเลชั่นเป็นที่นิยม

ฐานข้อมูลเชิงสัมพันธ์	ฐานข้อมูลเชิงวัตถุ	ฐานข้อมูลเชิงวัตถุสัมพันธ์
1. ใช้ SQL-92	1. ภาษาทางด้านออปเจ็คจะช่วยลดขนาดของภาษาทางด้านดาต้าเบส	1. ใช้ SQL - 99
2. เก็บข้อมูลในรูปของตาราง	2. เก็บข้อมูลในรูปของแอปสแตรค ด้าต้าไทป์	2. เก็บข้อมูลในรูปของตาราง
3. ไม่รองรับ Complex Attribute	3. รองรับ Complex Attribute	3. รองรับ Complex Attribute
4. ไม่สามารถสร้างโครงสร้างข้อมูลแบบใหม่ได้	4. สามารถสร้างโครงสร้างข้อมูลประเภทใหม่ได้	4. สามารถสร้างโครงสร้างข้อมูลประเภทใหม่ได้
5. เข้าถึงข้อมูลได้ง่ายและรวดเร็ว	5. การเข้าถึงข้อมูลจะต้องทำผ่านเมทธอดเท่านั้น	5. เข้าถึงข้อมูลได้ง่าย และสามารถเข้าถึงโดยผ่านทางเมท-ธอดหรือไม่ผ่านก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ไม่รองรับความสามารถของออปเจ็ค	6. รองรับความสามารถของออปเจ็ค	6. รองรับความสามารถของออปเจ็ค
7. เหมาะกับฐานข้อมูลประเภทโอเปอร์เรชั่นนอล ดาต้าเบส	7. เหมาะกับฐานข้อมูลขนาดกลาง	8. เหมาะกับฐานข้อมูลขนาดใหญ่

ตารางที่ 3-1 แสดงคุณสมบัติของฐานข้อมูลทั้ง 3 แบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

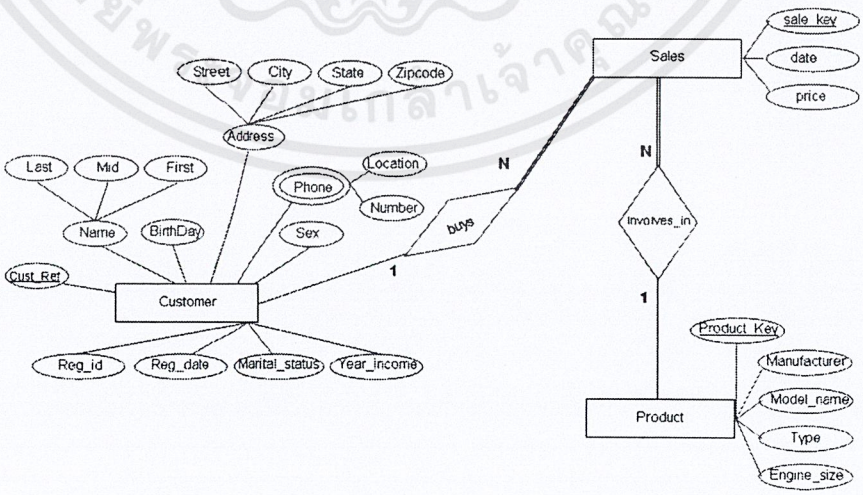
บทที่ 4
โครงสร้างและการออกแบบดาต้าแวร์เฮ้าท์

ข้อได้เปรียบของฐานข้อมูลเชิงวัตถุสัมพันธ์(Object – Relational Database)กับฐานข้อมูลเชิงสัมพันธ์(Relational Database)นั้นได้ถูกกล่าวมาแล้วในบทที่ผ่านมา ซึ่งถ้าหากพิจารณาถึงความต้องการในการใช้งานฐานข้อมูลในปัจจุบันแล้ว การออกแบบดาต้าแวร์เฮ้าท์โดยใช้วิธีการออกแบบฐานข้อมูลเชิงวัตถุสัมพันธ์จึงเป็นทางเลือกที่เหมาะสมที่สุด ในบทนี้จะกล่าวถึงโมเดลการออกแบบซึ่งจะแสดงให้เห็นในรูปแบบของการออกแบบฐานข้อมูลเชิงวัตถุสัมพันธ์ และเพื่อให้ง่ายต่อการเข้าใจ จึงนำวิธีการออกแบบดาต้าแวร์เฮ้าท์ในรูปแบบดาว(Star Schema)มาใช้

ในหัวข้อ 4.1 นั้นเราจะกล่าวถึงการออกแบบฐานข้อมูลโดยใช้โมเดลความสัมพันธ์ของชนิดของเอนทิตี้ (E-R Model) และจะใช้โมเดลความสัมพันธ์ของชนิดของเอนทิตี้ดังกล่าวในการออกแบบดาต้าแวร์เฮ้าท์ในหัวข้อ 4.2 สุดท้ายในหัวข้อ 4.3 นั้นเราจะแนะนำการปรับปรุงวิธีการออกแบบซึ่งได้บรรยายไปแล้วในหัวข้อที่ 4.2 เพื่อนำไปประยุกต์ใช้กับความต้องการในอนาคต

4.1 อีอาร์ไดอะแกรมของทรานแซคชั่นโพรเซสซิ่งดาต้าเบส

ในการศึกษาการประยุกต์นำเอาการออกแบบคลังข้อมูลเชิงวัตถุสัมพันธ์(Object – Relational Information Warehouse)นั้น ในกรณีนี้เราจะนำเอากระบวนการขายรถมาเป็นตัวอย่างการออกแบบ โดยที่คนขายรถนั้นจะเก็บข้อมูลการขายลงในฐานข้อมูลของการขายรถ สำหรับการศึกษานี้เราจะกล่าวถึงเพียงโมเดลออกแบบเพียงเท่านั้น แต่จะไม่กล่าวถึงวิธีการออกแบบโมเดลดังกล่าว สำหรับข้อดีของฐานข้อมูลที่ออกแบบด้วยวิธีการออกแบบเชิงวัตถุสัมพันธ์นั้น คือการรองรับข้อมูลในลักษณะComposite Attributeและลักษณะข้อมูลแบบ มัลติแวลูแอททริบิวต์ได้ ซึ่งทำให้ผิดหลักการของเฟริสนอร์มอลฟอร์ม(First Normal Form) ที่เป็นที่ต้องการในการออกแบบฐานข้อมูลเชิงสัมพันธ์



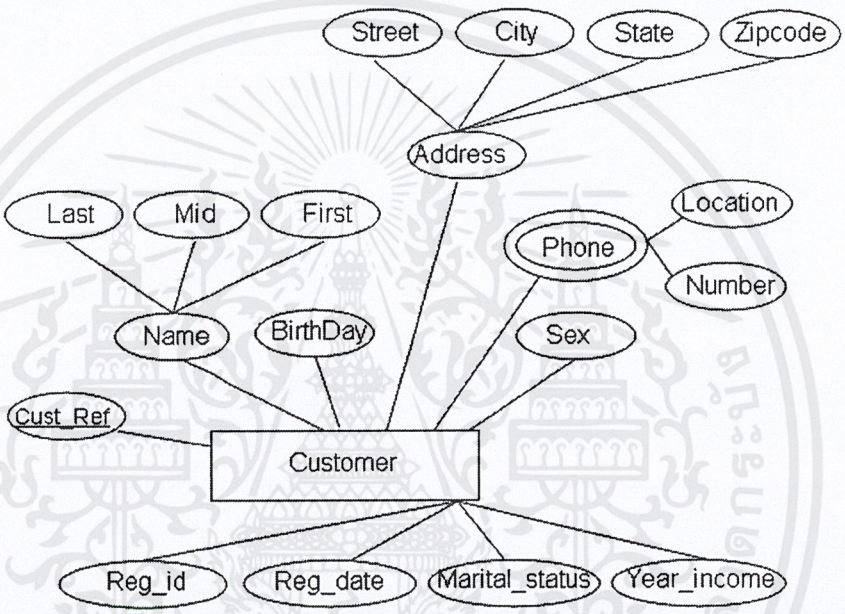
รูปที่ 4-1อีอาร์ไดอะแกรมของทรานแอคชั่นโพรเซสซิ่งดาต้าเบส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4-1 จะแสดงให้เห็นถึงอีอาร์ไดอะแกรม(ER Diagram)ของทรานแอคชั่นดาต้าเบส (Transaction Database) โดยใช้ Peter Chen's classical entity relationship model ซึ่งจะประกอบไปด้วย 3 ชนิดของเอนทิตี้ คือ

Customer

เป็นชนิดของเอนทิตี้ที่ดูแลข้อมูลต่างๆ ของลูกค้า ประกอบด้วยแอททริบิวต์ต่าง ๆ ดังรูป



รูปที่ 4-2 แสดงแอททริบิวต์ต่าง ๆ ของลูกค้า

แอททริบิวต์	ความหมาย	
แอททริบิวต์ทั่วไป(Simple Attribute)		
1. Reg_id	เป็นคีย์ที่ทำหน้าที่เป็นคีย์แอททริบิวต์ที่บ่งบอกถึงข้อมูลของลูกค้าที่ซื้อรถขณะนั้น ๆ	
2. Sex	เป็น แอททริบิวต์ที่บ่งบอกถึงเพศของลูกค้า เช่น เพศชาย(Male),เพศหญิง(Female) เป็นต้น	
3. BirthDay	เป็น แอททริบิวต์ที่บ่งบอกถึงวันเกิดของลูกค้า	
4.Marital_Status	เป็น แอททริบิวต์ที่บ่งบอกถึงสถานะของลูกค้าว่า แต่งงาน(Married),ยังไม่แต่งงาน (Unmarried),หย่าร้าง(Divored)	
5. Year_income	เป็น แอททริบิวต์ที่บ่งบอกถึงรายได้ต่อปีของลูกค้าขณะซื้อรถ	
คอมโพสิสแอททริบิวต์(Composite Attribute)(1 คนต่อ 1 ค่า)		
1. Name	Last	เป็น แอททริบิวต์ที่บ่งบอกถึงนามสกุลของลูกค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	Mid	เป็น แอททริบิวต์ที่บ่งบอกถึงชื่อกลางของลูกค้า
	First	เป็น แอททริบิวต์ที่บ่งบอกถึงชื่อของลูกค้า
2. Address	Street	เป็น แอททริบิวต์ที่บ่งบอกถึงถนนที่ลูกค้าอาศัยอยู่
	City	เป็น แอททริบิวต์ที่บ่งบอกถึงเมืองที่ลูกค้าอาศัยอยู่
	State	เป็น แอททริบิวต์ที่บ่งบอกถึงรัฐที่ลูกค้าอาศัยอยู่
	Zipcode	เป็น แอททริบิวต์ที่บ่งบอกถึงรหัสไปรษณีย์ของที่อยู่ของลูกค้า
มัลติแวลูแอททริบิวต์(Multivalue Attribute)(1 คนอาจจะมี 1 ค่าหรือมากกว่าก็ได้)		
3. Phone	Location	เป็น แอททริบิวต์ที่บ่งบอกถึงสถานที่ตั้งของหมายเลขที่ติดต่อ เช่น บ้าน(Home),ที่ทำงาน(Office),โทรศัพท์มือถือ(Mobile) เป็นต้น
	Phone_Number	เป็นหมายเลขโทรศัพท์ของลูกค้า
Others แอททริบิวต์with special attention		
1. Reg_date	เป็น แอททริบิวต์ที่บ่งบอกถึงวันที่มีการสั่งซื้อรถ	
2. Cust_Ref	เป็นคีย์แอททริบิวต์ที่บ่งบอกถึงลูกค้าคนไหนที่สั่งซื้อรถ ในกรณีที่มีการสั่งซื้อรถมากกว่า 1 ครั้ง ซึ่ง Reg_id จะเป็นเพียง แอททริบิวต์ที่บ่งบอกถึงลูกค้าที่ซื้อรถในขณะนั้น แต่ไม่สามารถจะบ่งบอกถึงลูกค้าคนเดิมที่ซื้อมากกว่า 1 ครั้งได้ Cust_ref จึงได้ถูกสร้างขึ้นมาเพื่อทำหน้าที่นี้ ซึ่ง Cust_ref นี้จะถูกสร้างขึ้นครั้งแรกที่ลูกค้าคนใหม่ได้ทำการซื้อรถ	

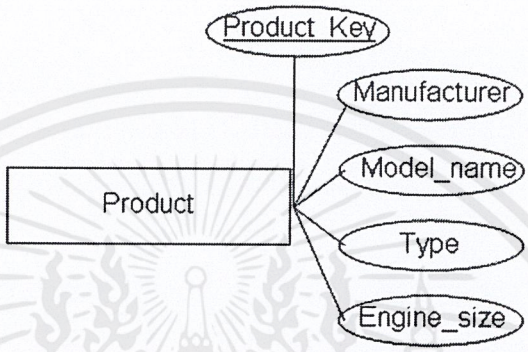
ตารางที่ 4 – 1 แสดงถึงแอททริบิวต์ต่าง ๆ ของลูกค้าในอีอาร์ไดอะแกรม ของทรานแอคชั่นดาต้าเบส

หากพิจารณาถึงการนำไปใช้งานจริง โดยใช้วิธีการของออปเจ็คริเลชั่นแล้ว อาจจะไม่ต้องใช้คีย์แอททริบิวต์(เช่นพวก Customer Key) เลย เพราะแต่ละแถวของตารางนั้นมี OID (Object – Identify) ที่ไม่ซ้ำกันของ ORDBMS ซึ่งทำหน้าที่คล้ายกับคีย์หลัก(Primary Key) แต่อย่างไรก็ตามมันก็ไม่ได้เกิดความเสียหาย หากเราจะมีคีย์หลักอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Product

เป็นชนิดของเอนทิตี้ที่ดูแลข้อมูลต่าง ๆ ของสินค้า ซึ่งในที่นี้ก็คือรถ ประกอบด้วย แอททริบิวต์ต่าง ๆ ดังรูปที่4-3



รูปที่ 4-3 แสดง แอททริบิวต์ต่าง ๆ ของสินค้า

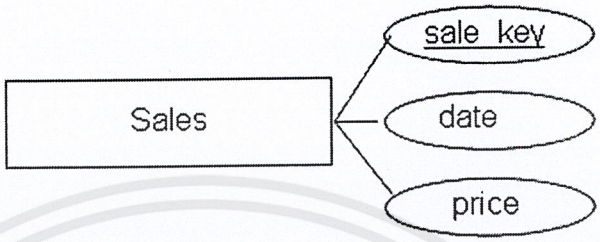
แอททริบิวต์	ความหมาย
แอททริบิวต์ทั่วไป	
1.Product_Key	เป็นคีย์ที่ทำหน้าที่เป็นคีย์แอททริบิวต์ที่บ่งบอกถึงข้อมูลของรถ
2. Manufacturer	เป็น แอททริบิวต์ที่บ่งบอกถึงบริษัทเจ้าของผู้ผลิต
3. Model_name	เป็น แอททริบิวต์ที่บ่งบอกถึงชื่อรุ่นของรถนั้น ๆ
4.Type	เป็น แอททริบิวต์ที่บ่งบอกถึงรายละเอียดต่าง ๆของรถ
5. Engine_size	เป็น แอททริบิวต์ที่บ่งบอกถึงขนาดของเครื่องยนต์

ตารางที่ 4 – 2 แสดงถึงแอททริบิวต์ต่าง ๆ ของสินค้าในอีอาร์ไดอะแกรม ของทรานแอคชั่นดาต้าเบส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sales

เป็นชนิดของเอนทิตี้ที่ดูแลข้อมูลต่าง ๆ ของเกี่ยวข้องกับข้อมูลการขาย ประกอบด้วย แอททริบิวต์ต่าง ๆ ดังรูป



รูปที่ 4-4 แสดง แอททริบิวต์ต่าง ๆ ของข้อมูลการขาย

แอททริบิวต์	ความหมาย
แอททริบิวต์ทั่วไป	
1. Sale_Key	เป็นคีย์ที่ทำหน้าที่เป็นคีย์แอททริบิวต์ที่บ่งบอกถึงข้อมูลการขาย
2. date	เป็น แอททริบิวต์ที่บ่งบอกถึงวันที่มีการขาย
3. Price	เป็น แอททริบิวต์ที่บ่งบอกถึงราคาขายของรถที่ขายไป

ตารางที่ 4 – 3 แสดงถึงแอททริบิวต์ต่าง ๆ ของข้อมูลการขายในอีอาร์ไดอะแกรม ของทรานแอคชั่นดาต้าเบส

นอกจากนี้ชนิดของเอนทิตี้การขายยังมีความสัมพันธ์แบบหนึ่งค่ากับหลาย ๆ ค่า(one – to –many) กับลูกค้าและสินค้าดังนี้ตารางการขายจึงต้องมีคีย์ประกอบ(Foreign Key) ซึ่งได้แก่ Customer_key (อ้างอิงถึง ลูกค้า) และ Product_key (อ้างอิงถึงสินค้า) แต่สำหรับวิธีการทางวัตถุสัมพันธ์นั้นสามารถใช้พอยเตอร์แทน คีย์หลัก/คีย์ประกอบได้ ดังนั้นในตารางการขายจะต้องมีอย่างน้อย 2 พอยเตอร์ที่จะชี้ไปยังตารางลูกค้าและตารางสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

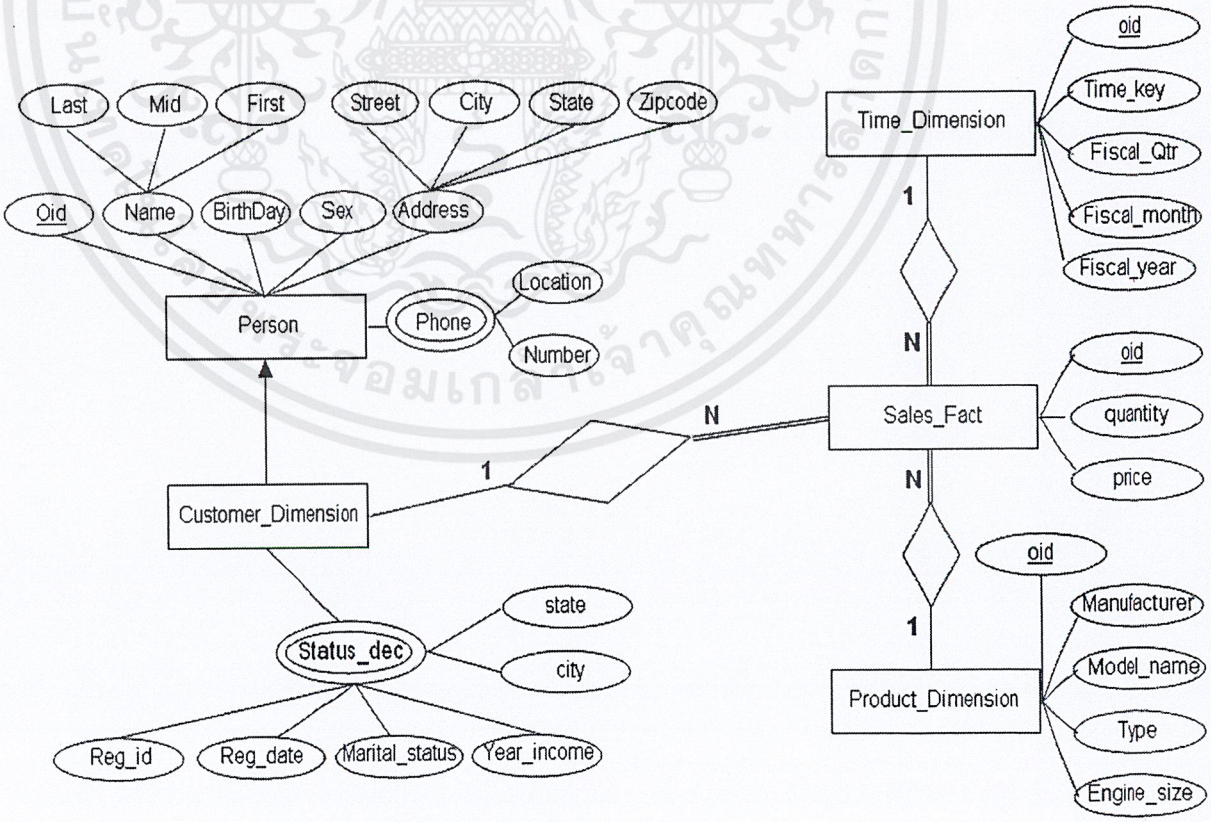
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

4.2 อีอาร์ไดอะแกรมของแวร์เฮ้าท์ดาต้าเบส

จากทรานแอคชั่นดาต้าเบส ที่ได้กล่าวถึงในหัวข้อ 4.1 เราก็จะสร้างอีอาร์ไดอะแกรมของแวร์เฮ้าท์ดาต้าเบสกันในหัวข้อนี้ สำหรับการสร้างอีอาร์ไดอะแกรมของแวร์เฮ้าท์ดาต้าเบสนั้นจะมีพื้นฐานอยู่บนการออกแบบแบบสคาร์สกีมาซึ่งจะประกอบไปด้วยโครงสร้าง 2 อย่างคือ ไดเมนชั่น และ แฟค สำหรับตารางแฟคนั้น จะเป็นตารางที่อยู่ตรงกลางที่แสดงจะเก็บข้อมูลที่เราสนใจ ซึ่งจะแสดงโดยความสัมพันธ์ของตารางไดเมนชั่นที่อยู่โดยรอบ สำหรับตารางไดเมนชั่นนั้นจะอยู่โดยรอบของตารางแฟค และ คีย์หลักของตารางไดเมนชั่นนั้นก็จะไปเกาะเป็น คีย์ประกอบของตารางแฟค

สำหรับการออกแบบสำหรับแวร์เฮ้าท์ดาต้าเบสจะแตกต่างจากทรานแอคชั่นดาต้าเบสตรงที่ทุกตารางในทรานแอคชั่นนั้นมักจะถูกนอร์มอลไลน์(Normalize) เพื่อลดความซ้ำซ้อนของฐานข้อมูล แต่สำหรับตารางแฟคและตารางไดเมนชั่นนั้นจะถูกดีนอร์มอลไลน์(Denormalize) เพื่อเพิ่มประสิทธิภาพและทำให้ง่ายต่อการตอบคำถาม เพราะดาต้าแวร์เฮ้าท์นั้นอยู่บนพื้นฐานของการอ่านข้อมูลจากฐานข้อมูลเพียงอย่างเดียว ดังนั้นการความซ้ำซ้อนของข้อมูลหรือกฎต่าง ๆ จึงไม่เป็นหัวข้อสำคัญของดาต้าแวร์เฮ้าท์

จุดประสงค์ประการสำคัญของดาค้าแวร์เฮ้าท์นั้นอยู่บนการวิเคราะห์ข้อมูลบนพื้นฐานของเวลา ดังนั้นจึงไม่จำเป็นที่ทุก ๆ แอททริบิวต์ในทรานแอคชั่นดาต้าเบสจะต้องมีอยู่ในแวร์เฮ้าท์ดาต้าเบส หากข้อมูลนั้นเป็นข้อมูลที่ไม่ได้ถูกนำมาวิเคราะห์(NonAnalytical Attribute) ก็อาจจะไม่เก็บไว้ใน ตารางไดเมนชั่นก็ได้ แต่ก็มักจะนำไปเก็บในไดเมนชั่นเงา(Shadow Dimension)เพื่อป้องกันมิให้ข้อมูลหายไปหากต้องการใช้



รูปที่ 4-5อีอาร์ไดอะแกรมของแวร์เฮ้าท์ดาต้าเบส

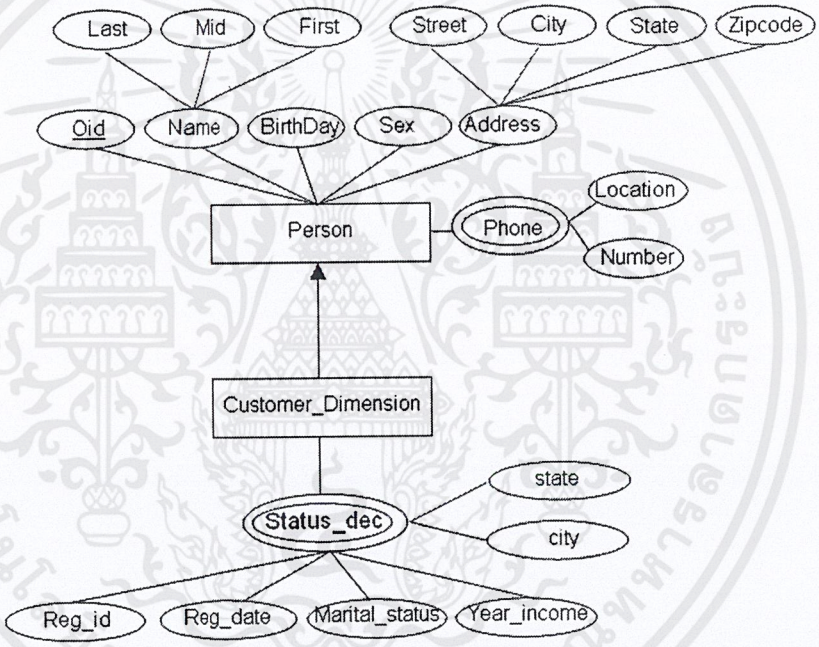
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

49963

จากรูปที่ 4-5 แสดงให้เห็นถึงไดอะแกรมการออกแบบของดาต้าแวร์เฮ้าท์ โดยใช้อีอาร์ไดอะแกรม ซึ่งจะประกอบไปด้วย 3 ไดเมนชั่นและ 1 แฟค

Customer Dimension

เป็นไดเมนชั่นที่ดูแลข้อมูลต่าง ๆ ที่เป็นข้อมูลของลูกค้า สำหรับไดเมนชั่นของลูกค้านั้นมีโครงสร้างหลัก(Root Type)เป็น Person ซึ่ง Customer จะถ่ายทอดคุณสมบัติและความสามารถมาจาก Person



รูปที่ 4-6 แสดง แอททริบิวต์ต่าง ๆ ของไดเมนชั่นของลูกค้า

Attribute	ความหมาย	
แอททริบิวต์อย่างง่ายที่ถูกถ่ายทอดมาจาก Person		
1. Oid	เป็นคีย์ที่ทำหน้าที่เป็นคีย์แอททริบิวต์ที่บ่งบอกถึงข้อมูลของลูกค้าที่ซื้อรถขณะนั้น ๆ	
2. Sex	เป็น แอททริบิวต์ที่บ่งบอกถึงเพศของลูกค้า เช่น เพศชาย(Male),เพศหญิง(Female) เป็นต้น	
3. BirthDay	เป็น แอททริบิวต์ที่บ่งบอกถึงวันเกิดของลูกค้า	
คอมโพสิสแอททริบิวต์(1 คนต่อ 1 ค่า) ที่ถ่ายทอดมาจาก Person		
1. Name	Last	เป็น แอททริบิวต์ที่บ่งบอกถึงนามสกุลของลูกค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

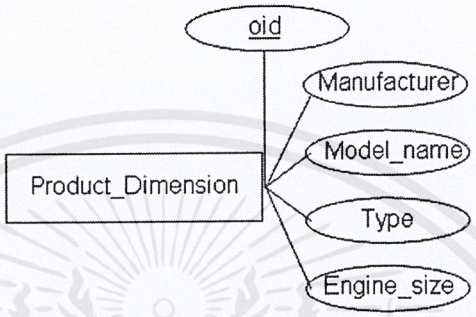
	Mid	เป็น แอททริบิวต์ที่บ่งบอกถึงชื่อกลางของลูกค้า
	First	เป็น แอททริบิวต์ที่บ่งบอกถึงชื่อของลูกค้า
2. Address	Street	เป็น แอททริบิวต์ที่บ่งบอกถึงถนนที่ลูกค้าอาศัยอยู่
	City	เป็น แอททริบิวต์ที่บ่งบอกถึงเมืองที่ลูกค้าอาศัยอยู่
	State	เป็น แอททริบิวต์ที่บ่งบอกถึงรัฐที่ลูกค้าอาศัยอยู่
	Zipcode	เป็น แอททริบิวต์ที่บ่งบอกถึงรหัสไปรษณีย์ของที่อยู่ของลูกค้า
มัลติแวลูแอททริบิวต์(1 คนอาจจะมี 1 ค่าหรือมากกว่าก็ได้) ที่ถ่ายทอดมาจาก Person		
3. Phone	Location	เป็น แอททริบิวต์ที่บ่งบอกถึงสถานที่ตั้งของหมายเลขที่ติดต่อ เช่น บ้าน(Home),ที่ทำงาน(Office), โทรศัพท์มือถือ(Mobile) เป็นต้น
	Phone_Number	เป็นหมายเลขโทรศัพท์ของ
Composite แอททริบิวต์ที่เป็นของ Customer โดยตรง		
1.Status_dec	Reg_id	เป็น แอททริบิวต์ที่ทำหน้าที่เป็นคีย์แอททริบิวต์จะถูกสร้างขึ้นเมื่อมีลูกค้ามาซื้อรถในแต่ละครั้ง
	Reg_date	เป็น แอททริบิวต์ที่บ่งบอกเป็นพอยเตอร์ชี้ไปยัง ไดเมนชั่นของเวลาบ่งบอกถึงวันเวลาที่มีการซื้อรถในแต่ละครั้ง
	Marital_Status	เป็น แอททริบิวต์ที่บ่งบอกถึงสถานะของลูกค้าขณะทำการซื้อรถในแต่ละครั้ง เช่น แต่งงาน(Married),ยังไม่แต่งงาน(Unmarried), หย่าร้าง(Divored) เป็นต้น
	Year_income	เป็น แอททริบิวต์ที่บ่งบอกถึงรายได้ต่อปีของลูกค้าขณะซื้อรถ
	City	เป็น แอททริบิวต์ที่บ่งบอกถึงชื่อเมืองที่ลูกค้าทำการซื้อรถในครั้งนั้น
	State	เป็น แอททริบิวต์ที่บ่งบอกถึงชื่อรัฐที่ลูกค้าทำการซื้อรถในครั้งนั้น

ตารางที่ 4 – 4 แสดงถึงแอททริบิวต์ต่าง ๆ ของไดเมนชั่นของลูกค้าในอีอาร์ไดอะแกรมของแวร์เฮ้าท์ดาต้าเบส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Product Dimension

เป็นไดเมนชั่นที่ดูแลข้อมูลต่าง ๆ ที่เกี่ยวข้องกับสินค้า ซึ่งในที่นี้ก็คือรถยนต์ที่จะทำการขายในแต่ละครั้ง สำหรับไดเมนชั่นของสินค้านั้นก็จะประกอบไปด้วย แอททริบิวต์ต่าง ๆ ดังนี้



รูปที่ 4-7 แสดง แอททริบิวต์ต่าง ๆ ของไดเมนชั่นของสินค้า

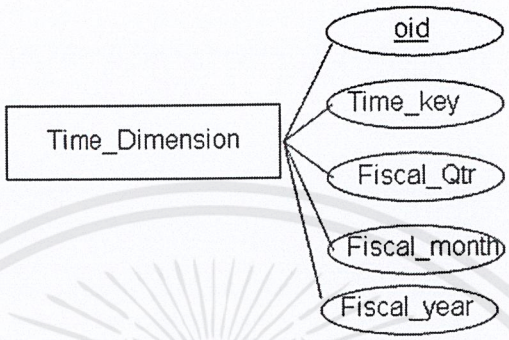
Attribute	ความหมาย
แอททริบิวต์ทั่วไป	
1.oid	เป็นคีย์ที่ทำหน้าที่เป็น คีย์แอททริบิวต์ที่บ่งบอกถึงข้อมูลของรถ
2. Manufacturer	เป็น แอททริบิวต์ที่บ่งบอกถึงบริษัทเจ้าของผู้ผลิต
3. Model_name	เป็น แอททริบิวต์ที่บ่งบอกถึงชื่อรุ่นของรถนั้น ๆ
4.Type	เป็น แอททริบิวต์ที่บ่งบอกถึงรายละเอียดต่าง ๆของรถ
5. Engine_size	เป็น แอททริบิวต์ที่บ่งบอกถึงขนาดของเครื่องยนต์

ตารางที่ 4 – 5 แสดงถึงแอททริบิวต์ต่าง ๆ ของไดเมนชั่นของสินค้าในอีอาร์ไดอะแกรม ของ แวร์เฮ้าท์ดาต้าเบส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Time Dimension

เป็นไดเมนชั่นที่เก็บข้อมูลต่าง ๆ ทางด้านเวลา ซึ่งเป็นส่วนสำคัญของแวร์เฮ้าท์ดาต้าเบสที่จะเก็บข้อมูลอยู่บนพื้นฐานของเวลา สำหรับไดเมนชั่นของเวลานั้นจะถูกสร้างขึ้นโดยมี แอททริบิวต์ต่าง ๆ ดังนี้



รูปที่ 4-8 แสดง แอททริบิวต์ต่าง ๆ ของไดเมนชั่นของเวลา

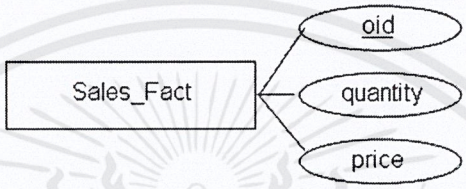
Attribute	ความหมาย
แอททริบิวต์ทั่วไป	
1.oid	เป็นคีย์ที่ทำหน้าที่เป็นคีย์แอททริบิวต์ที่ใช้เป็นตัวชี้ถึงเวลา
2. Time_Key	เป็นคีย์แอททริบิวต์ที่บ่งบอกถึงเวลาเช่นเดียวกับ oid (จะมีหรือไม่ก็ได้)
3. Fiscal_Qtr	เป็นแอททริบิวต์ที่บ่งบอกถึงไตรมาสของบริษัทผู้ขายรถยนต์
4.Fiscal_month	เป็นแอททริบิวต์ที่บ่งบอกถึงเดือนต่าง ๆ ของบริษัทผู้ขายรถยนต์
5. Fiscal_year	เป็นแอททริบิวต์ที่บ่งบอกถึงปีต่าง ๆของบริษัทผู้ขายรถยนต์

ตารางที่ 4 – 6 แสดงถึงแอททริบิวต์ต่าง ๆ ของไดเมนชั่นของเวลาในอีอาร์ไดอะแกรม ของ แวร์เฮ้าท์ดาต้าเบส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sales Fact

เป็นส่วนสำคัญของแวร์เฮ้าท์ดาต้าเบสโดยจะเป็นข้อมูลที่สำคัญที่ถูกนำมาใช้ในการวิเคราะห์ข้อมูล สำหรับตารางแฟคนั้นจะอยู่ตรงกลางของไดอะแกรม โดยจะมีพอยเตอร์ชี้ไปยังตารางไดเมนชั่นที่อยู่โดยรอบ และจะมี แอททริบิวต์ที่ใช้ในการวิเคราะห์ 2 ตัว ดังรูป



รูปที่ 4-9 แสดง แอททริบิวต์ต่าง ๆ ของ Sale Fact

แอททริบิวต์	ความหมาย
แอททริบิวต์ทั่วไป	
1.oid	เป็นคีย์ที่ทำหน้าที่เป็นคีย์แอททริบิวต์ที่ใช้เป็นตัวชี้ถึงแฟคของการขายนั้น ๆ
2. Quantity	เป็น แอททริบิวต์ที่บ่งบอกถึงจำนวนรถที่ขายได้
3. Price	เป็น แอททริบิวต์ที่บ่งบอกถึงราคาของสินค้าที่ขายได้
Pointer to Dimension	
1. cust_dim_ref	เป็นพอยเตอร์ ที่ชี้ไปยังตารางของไดเมนชั่นของลูกค้า
4.time_dim_ref	เป็นพอยเตอร์ที่ชี้ไปยังตารางของไดเมนชั่นของเวลา
5. prod_dim_ref	เป็นพอยเตอร์ที่ชี้ไปยังตารางของไดเมนชั่นของสินค้า

ตารางที่ 4 – 7 แสดงถึงแอททริบิวต์ต่าง ๆ ของแฟคของการขาย ในอีอาร์ไดอะแกรมของแวร์เฮ้าท์ดาต้าเบส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การปรับปรุงการออกแบบแวร์เฮ้าท์ดาต้าเบส

ถึงแม้ว่าการออกแบบแวร์เฮ้าท์ดาต้าเบสที่กล่าวมาในหัวข้อที่ผ่านมานั้น สามารถที่จะทำงาน หรือตอบคำถามได้ แม้จะมีการเก็บข้อมูลในรูปแบบของการเปลี่ยนแปลงบนเวลาที่ต่างกันก็ตาม แต่ถึงอย่างนั้น ก็ยังสามารถที่จะพัฒนาระบบฐานข้อมูลให้ดีขึ้นกว่านี้ได้

จากการออกแบบที่ผ่านมานั้น จะอยู่บนแนวความคิดที่ลูกค้าได้ซื้อรถไปในวันเดียวกับที่ลูกค้าได้กลายเป็นลูกค้าของบริษัท แต่หากสมมติว่าลูกค้าได้เป็นลูกค้าของบริษัทในวันที่ 15 มิถุนายน 1999 และได้ซื้อรถไปในวันที่ 15 สิงหาคม 1999 แล้วข้อมูลที่ถูกเก็บลงในแวร์เฮ้าท์ จะเป็นข้อมูลที่ใหม่ที่เป็นวันที่ลูกค้าทำการซื้อรถ มิใช่วันที่ลูกค้ากลายเป็นลูกค้าของบริษัท ซึ่งทำให้ข้อมูลหายไป ดังนั้นหากจะนำไปพัฒนาต่อก็อาจจะสร้าง “from_date_key” เป็นตัวเก็บวันที่ทำการซื้อรถแทน ซึ่งไม่สามารถทำได้บนทรานแอคชั่นดาต้าเบส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
การพัฒนาดาต้าแวร์เฮ้าท์เพื่อนำไปใช้งาน

จากบทที่ผ่านมานั้น เราได้ศึกษาถึงวิธีการออกแบบดาต้าแวร์เฮ้าท์โดยใช้หลักการของระบบฐานข้อมูลเชิงวัตถุสัมพันธ์ สำหรับในบทนี้เราจะกล่าวถึงการพัฒนาระบบเพื่อนำไปใช้งานจริง

สำหรับเครื่องมือที่ใช้ในการพัฒนาระบบ เราจะใช้ IBM DB2 ซึ่งเป็นเครื่องมือที่ถูกพัฒนาโดยบริษัท ไอบีเอ็ม จำกัด ซึ่งมีอุปกรณ์ที่จำเป็นต่อการพัฒนาระบบฐานข้อมูลเชิงวัตถุสัมพันธ์

สาเหตุประการสำคัญที่ใช้ IBM DB2 ในการพัฒนาดาต้าแวร์เฮ้าท์นั้นคือการที่โปรแกรมดังกล่าวสามารถรองรับการทำงานของระบบฐานข้อมูลเชิงวัตถุสัมพันธ์ และสามารถทำให้โปรเจคนี้สามารถทำได้ นอกจากนี้บริษัทไอบีเอ็มยังมีคู่มือการใช้งานสามารถให้ดาวน์โหลดได้จากเวปไซต์ของบริษัทอีกด้วย

ในบทนี้ เริ่มต้นเราจะกล่าวถึงภาษาโครงสร้างข้อมูล (DDL : Data Definition Language) ในหัวข้อ 5.1 ซึ่งภาษานี้จะรวบรวมคำสั่งที่ใช้ในการสร้างดาต้าแวร์เฮ้าท์ขึ้นมา ในหัวข้อ 5.2 นั้นเราก็จะแสดงภาษาในการจัดการฐานข้อมูล ซึ่งก็จะเกี่ยวข้องกับคำสั่งในการแทรกข้อมูล (Insert) , ลบข้อมูล (Delete) และอัพเดตข้อมูล (Update) และในหัวข้อ 5.3 ซึ่งเป็นหัวข้อสุดท้ายเราจะแสดงให้เห็นถึงการตอบคำถามเพื่อใช้ในกระบวนการทางธุรกิจ สำหรับภาษาที่ใช้นั้นเราจะใช้เป็น เอสคิวแอล 99 (SQL 99) และเพื่อให้ง่ายต่อการอ่าน สำหรับคำสั่งของเอสคิวแอลเราจะใช้เป็นสัญลักษณ์ตัวพิมพ์ใหญ่

5.1 คำสั่งที่ใช้ในการสร้างออปเจคและตาราง

สำหรับการสร้างฐานข้อมูลของดาต้าแวร์เฮ้าท์นั้น เราจะสร้างจากปแผนภาพอีอาร์ในบทที่ผ่านมา และเพื่อให้ง่ายเราจะทำการแปลงชนิดของเอนทิตี้และแอทริบิวต์จากแผนภาพอีอาร์มายังตาราง

สำหรับในหัวข้อนี้เราจะเริ่มจากสร้างตารางไดเมนชั่นง่าย ๆ ได้แก่ ไดเมนชั่นของเวลา(Time Dimension)และ ไดเมนชั่นของสินค้า(Product Dimension)ก่อน หลังนั้นก็จะทำการสร้างตารางไดเมนชั่นที่ซับซ้อนกว่าคือ ไดเมนชั่นของลูกค้า(Customer Dimension) และสุดท้ายเราก็จะสร้างตารางแฟค(Fact Table)

5.1.1 ตารางไดเมนชั่นของเวลา(Time Dimension Table)

สำหรับตารางไดเมนชั่นของเวลา และ ออปเจคของไดเมนชั่นของเวลา(Time Dimension Object) นั้นประกอบด้วย ซิมเปิลแอททริบิวต์(Simple Attribute) ได้แก่ แอทริบิวต์ “oid” เป็นแอทริบิวต์ที่ทำหน้าที่เป็นคีย์แอททริบิวต์ในส่วนนี้ โดยปกติจะไม่แสดงให้เห็นเพราะจะถูกซ่อนอยู่ในตาราง แอทริบิวต์ “Time_Key” ในส่วนนี้จะมีหรือไม่ก็ได้ สำหรับในฐานข้อมูลนี้จะให้มีเพื่อให้ง่ายต่อการอ้างถึงข้อมูลในตาราง ไดเมนชั่นของเวลา แอทริบิวต“Fiscal_year” , “Fiscal_Qtr” และ “Fiscal_month” จะเป็นค่าที่ถูกเก็บลงในตารางไดเมนชั่นของเวลา โดยที่แอทริบิวต์ทั้งสามตัวจะถูกอ้างถึงโดย Time_Key เช่น หาก Time_Key มีค่าเท่ากับ “2002Q3M09” ก็จะอ้างถึงแถวที่มี Fiscal_year เท่ากับ 2002 Fiscal_Qtr เท่ากับ Quarter3 และ มี Fiscal_month เท่ากับ September เป็นต้น

แต่ละแถวใน ตารางไดเมนชั่นของเวลา นั้น สามารถอ้างถึงได้จากตารางอื่น ๆ เช่น หากตาราง Sale_Fact_Table ต้องการอ้างถึงข้อมูลในตารางไดเมนชั่นของเวลา โดยการใช้คำสั่ง REF data-type ใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบฐานข้อมูลเชิงสัมพันธ์ สำหรับระบบฐานข้อมูลเชิงวัตถุสัมพันธ์นี้เราต้องสร้างโครงสร้างของออปเจค (Object Type) ก่อน โดยการใช้คำสั่งเอสคิวแอล 99 ดังนี้

```
CREATE TYPE time_dimension_t AS
(
        time_key        CHAR(10),
        fiscal_year     NUM(4),
        fiscal_qtr      CHAR(10),
        fiscal_month    CHAR(10)
)
MODE DB2SQL
REF USING INTEGER
```

สำหรับประเภท (data-type) ของแอทริบิวต์ของออปเจ็คนั้นจะถูกประกาศในโครงสร้างข้อมูลของออปเจ็ค และเมื่อเราประกาศ Object data-type แล้วเราก็สามารถนำไปประกาศเป็นโครงสร้างข้อมูลใหม่หรือนำไปสร้างเป็นตารางได้แล้ว สำหรับคำสั่ง REF USING INTEGER นั้นเป็นการระบุให้ Oid ที่นำมาใช้ในการอ้างถึงโครงสร้างข้อมูลชนิดนี้ให้เป็นตัวเลข และสำหรับข้อกำหนดต่าง ๆ ของแอทริบิวต์นั้นจะถูกประกาศในขณะทำการสร้างตาราง

สำหรับตารางไดเมนชั่นของเวลาจะใช้ชื่อ time_dim_tab ซึ่งจะถูกประกาศโดยใช้คำสั่ง ดังนี้

```
CREATE TABLE time_dim_tab OF time_dimension_t
(
        REF IS OID USER GENERATED,
        time_key WITH OPTION NOT NULL,
        fiscal_qtr WITH OPTION NOT NULL,
        fiscal_month WITH OPTION NOT NULL,
        fiscal_year WITH OPTION NOT NULL
)
```

แต่ละแถวที่ถูกสร้างขึ้นในตาราง time_dim_tab นั้น จะเป็น time_dimension_t ซึ่งเป็นออปเจ็คที่ถูกสร้างขึ้น จึงเรียกประเภทของตารางนี้ว่า ออปเจ็คเทเบิล (Object Table) นอกจากนี้แต่ละแถวยังสามารถที่จะอ้างถึงได้จากออปเจ็คอื่น ๆ ซึ่งเราจะกล่าวถึงในภายหลัง

แต่ละแถวจะมี ออปเจ็คไอเดนทิไฟล์ (OID) ซึ่งจะใช้อ้างถึงข้อมูลในแต่ละแถว แต่ใน IBM DB2 นั้น ไม่สามารถให้ระบบสร้าง oid ได้ เราจึงต้องสร้าง ลำดับ(Sequence) และ ทริกเกอร์(Trigger)มาเพื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แก้ไข สำหรับการสร้าง oid นั้นเราจะใช้คำสั่ง "REF IS OID USER GENERRATED" สำหรับรูปที่ 5-1 จะแสดงให้เห็นถึงคอลัมน์ต่าง ๆ ของ time_dim_tab

Table time_dim_tab (of time_dimension_t)

Oid	Fiscal_year	Fiscal_Qtr	Fiscal_month
Integer	Number	Char(10)	Char(10)
Object Identifier	NOT NUL	NOT NULL	NOT NULL

รูปที่ 5-1 Object – Relational Representation of the Time Dimension Table

5.1.2 ตารางไดเมนชั่นของสินค้า(Product Dimension Table)

คล้ายกับตารางไดเมนชั่นของเวลา สำหรับตารางไดเมนชั่นของสินค้า นั้นจะใช้ชื่อว่า Prod_dim_tab โดยสำหรับตารางนี้จะประกอบด้วย ซิมเปิลแอททริบิวต์(Simple Attribute) ได้แก่ oid ซึ่งทำหน้าที่เป็นออปเจ็คไอเดนทิไฟล์ ที่ชี้ไปยังแต่ละแถวของตาราง Prod_dim_tab นอกจากนี้ยังมีแอทริบิวต์อื่น ๆ ได้แก่ model_name , manufacturer, type, engine_size และก่อนที่จะสร้าง Prod_dim_tab นั้น เราจำเป็นต้องสร้าง prod_dimension_t ซึ่งเป็นโครงสร้างของข้อมูลขึ้นมาก่อน โดยสำหรับคำสั่งเอสคิวแอลที่ใช้ในการสร้าง Prod_dimention_t คือ

```
CREATE TYPE prod_dimension_t AS
(
        model_name      VARCHAR(30),
        manufacturer    VARCHAR(25),
        type            VARCHAR(50),
        engine_size     CHAR(6)
)
MODE DB2SQL
REF USING INTEGER
```

และเมื่อสร้าง Prod_dimension_t ซึ่งเป็นโครงสร้างของข้อมูลขึ้นมาแล้วเราก็จะนำโครงสร้างของข้อมูลดังกล่าวมาสร้างเป็นตาราง ตามคำสั่งเอสคิวแอลดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CREATE TABLE  prod_dim_tab OF prod_dimension_t
(
        REF IS OID USER GENERATED,
        manufacturer WITH OPTION NOT NULL,
        model_name WITH OPTION NOT NULL,
        type WITH OPTION NOT NULL,
        engine_size WITH OPTION NOT NULL
)
```

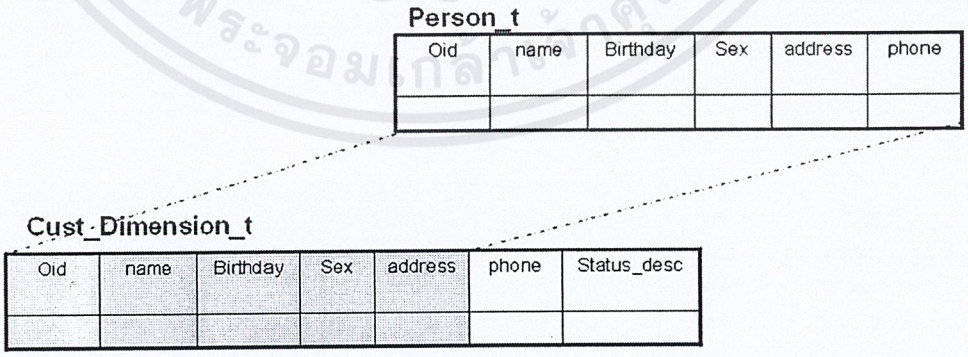
Table Prod_dim_tab (of prod_dimension_t)

Oid	Manufacturer	Model_name	type	Engine_size
Integer	VARCHAR(25)	VARCHAR(30)	VARCHAR(50)	Char(6)
Object Identifier	NOT NUL	NOT NULL	NOT NULL	NOT NULL

รูปที่ 5-2 Object – Relational Representation of the Product Dimension Table

5.1.3 ตารางไดเมนชั่นของลูกค้า

สำหรับตารางไดเมนชั่นของลูกค้าและออปเจ็คไดเมนชั่นของลูกค้านั้นจะแตกต่างจากตารางไดเมนชั่นและออปเจ็คไดเมนชั่นที่ผ่านมา ตรงที่โครงสร้างของไดเมนชั่นของลูกค้านั้นสร้างมาจากโครงสร้างของข้อมูลที่ชื่อ Cust_dimension_t ซึ่ง Cust_dimension_t นั้นจะมีโครงสร้างหลัก(Root Type) เป็น Person_t หมายความว่า Cust_dimension_t นั้นได้รับการถ่ายทอด (Inherit) คุณสมบัติ(Attribute) มาจาก Person_t ดังแสดงให้เห็นได้ในรูปที่ 5-3



รูปที่ 5-3 แสดงการถ่ายทอดคุณสมบัติระหว่าง Person_t กับ Cust_dimension_t

สำหรับไดเมนชั่นนี้มีทั้ง ซิมเปิลแอททริบิวต์(Simple Attribute),คอมโพสิดแอททริบิวต์(Composite Attribute) และมัลติแวลูแอททริบิวต์ (Multivalues Attribute) ดังแสดงให้เห็นในรูปที่ 5-4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

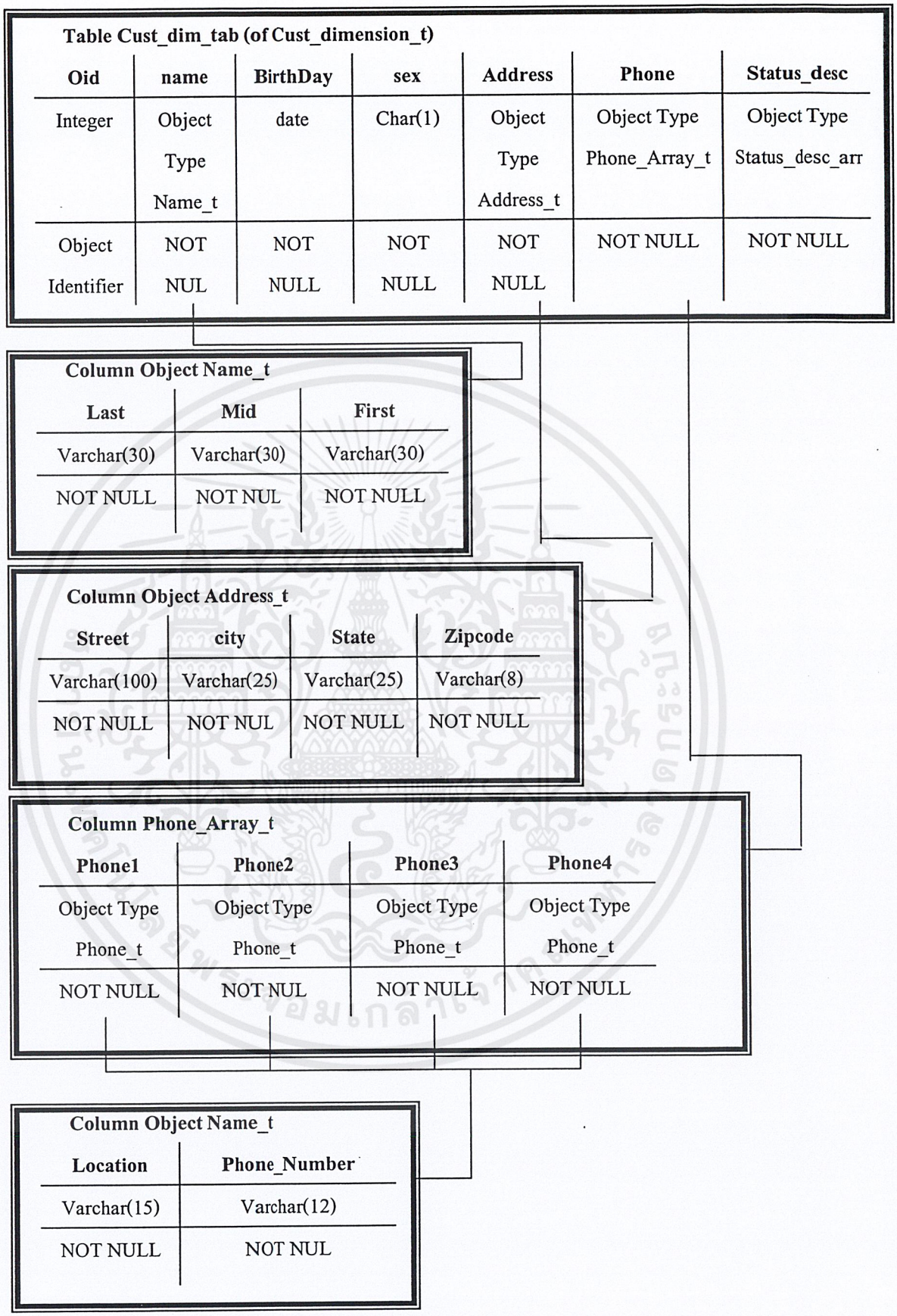


Table Cust_dim_tab (of Cust_dimension_t)

Oid	name	BirthDay	sex	Address	Phone	Status_desc
Integer	Object Type Name_t	date	Char(1)	Object Type Address_t	Object Type Phone_Array_t	Object Type Status_desc_arr
Object Identifier	NOT NUL	NOT NULL	NOT NULL	NOT NULL	NOT NULL	NOT NULL

Column Object Name_t

Last	Mid	First
Varchar(30)	Varchar(30)	Varchar(30)
NOT NULL	NOT NUL	NOT NULL

Column Object Address_t

Street	city	State	Zipcode
Varchar(100)	Varchar(25)	Varchar(25)	Varchar(8)
NOT NULL	NOT NUL	NOT NULL	NOT NULL

Column Phone_Array_t

Phone1	Phone2	Phone3	Phone4
Object Type Phone_t	Object Type Phone_t	Object Type Phone_t	Object Type Phone_t
NOT NULL	NOT NUL	NOT NULL	NOT NULL

Column Object Name_t

Location	Phone_Number
Varchar(15)	Varchar(12)
NOT NULL	NOT NUL

รูปที่ 5-4 ก Object – Relational Representation of the Customer Dimension Table

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

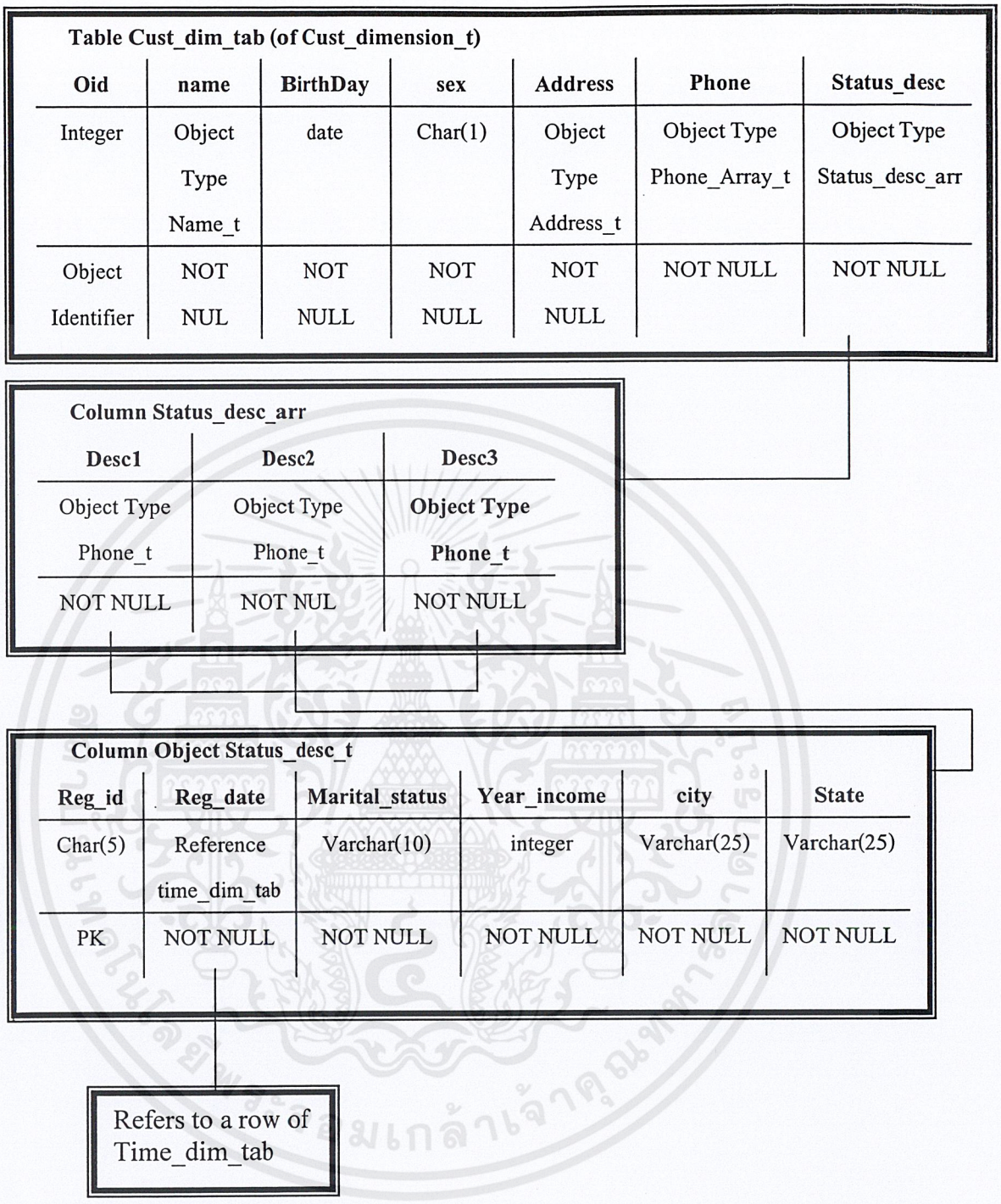


Table Cust_dim_tab (of Cust_dimension_t)

Oid	name	BirthDay	sex	Address	Phone	Status_desc
Integer	Object Type Name_t	date	Char(1)	Object Type Address_t	Object Type Phone_Array_t	Object Type Status_desc_arr
Object Identifier	NOT NUL	NOT NULL	NOT NULL	NOT NULL	NOT NULL	NOT NULL

Column Status_desc_arr

Desc1	Desc2	Desc3
Object Type Phone_t	Object Type Phone_t	**Object Type Phone_t**
NOT NULL	NOT NUL	NOT NULL

Column Object Status_desc_t

Reg_id	Reg_date	Marital_status	Year_income	city	State
Char(5)	Reference time_dim_tab	Varchar(10)	integer	Varchar(25)	Varchar(25)
PK	NOT NULL	NOT NULL	NOT NULL	NOT NULL	NOT NULL

Refers to a row of Time_dim_tab

. ***รูปที่ 5-4 ข Object – Relational Representation of the Customer Dimension Table(Continue)***

สำหรับ IBM DB2 นั้นไม่สามารถที่จะประกาศประเภทของแอทริบิวต์เป็นโครงสร้างซับซ้อนได้ ดังนั้นจึงทำให้เราต้องสร้างโครงสร้างข้อมูลชนิด Array ขึ้นมารองรับความสามารถของมัลติเวลูแอททริบิวต์ ดังรูปข้างต้น ต่อไปนี้เราจะแสดงคำสั่งในการสร้างโครงสร้างข้อมูบ โดยเริ่มจากการประกาศ คอมโพสิสไทป์ก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CREATE TYPE name_t AS
(
        First VARCHAR(20),
        Middle VARCHAR(15),
        Last VARCHAR(20)
)
MODE DB2SQL
CREATE TYPE Address_t AS
(
        No INTEGER,
        Street VARCHAR(100),
        City VARCHAR (25),
        State VARCHAR (25),
        Zipcode VARCHAR (25)
)
        MODE DB2SQL
```

รูปที่ 5-5 แสดงคำสั่งเอสคิวแอลในการสร้างคอมโพสิสไทป์

ต่อมาเราก็จะแสดงคำสั่งในการสร้างมัลติแวลูแอททริบิวต์

```
CREATE TYPE  phone_t AS
(
        location VARCHAR(15),
        phone_num VARCHAR(12)
)
MODE DB2SQL
```

รูปที่ 5-6ก แสดงคำสั่งเอสคิวแอลในการสร้างมัลติแวลูไทป์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CREATE TYPE  phone_array_t AS
(
        phone1 PHONE_T,
        phone2 PHONE_T,
        phone3 PHONE_T,
        phone4 PHONE_T
)
MODE DB2SQL
REF USING INTEGER

CREATE TYPE  status_desc_t AS
(
        Reg_id          CHAR(5),
        reg_date        REF(time_dimension_t),
        marital_status  VARCHAR(10),
        year_income     INTEGER,
        city            VARCHAR(25),
        state           VARCHAR(25)
)
MODE DB2SQL

CREATE TYPE status_desc_arr AS
(
        desc1   status_desc_t,
        desc2   status_desc_t,
        desc3   status_desc_t
)
MODE DB2SQL
REF USING INTEGER
```

รูปที่ 5-6ข แสดงคำสั่งเอสคิวแอลในการสร้างมัลติแวลูไทป์(ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต่อมาเราก็จะสร้างโครงสร้างหลัก ซึ่งก็คือ Person_t

```
CREATE TYPE person_t AS
(
        name            name_t,
        birthday        DATE,
        sex             CHAR(1),
        address         address_t,
        phone           phone_array_t
)
MODE DB2SQL
REF USING INTEGER
NOT FINAL
```

และสุดท้ายเราจะสร้าง Cust_dimension_t ที่ได้รับการถ่ายทอด มาจาก Person_t

```
CREATE TYPE cust_dimension_t UNDER person_t AS
(
        status_desc     status_desc_arr
)
MODE DB2SQL
```

เมื่อสร้างโครงสร้างข้อมูลทั้งหมดเรียบร้อยแล้วขั้นตอนต่อไปคือการสร้างตารางไดเมนชั่นของลูกค้าที่ชื่อ Cust_dimension_tab เราจะใช้คำสั่งเอสคิวแอลในการสร้าง Cust_dimension_tab ดังนี้

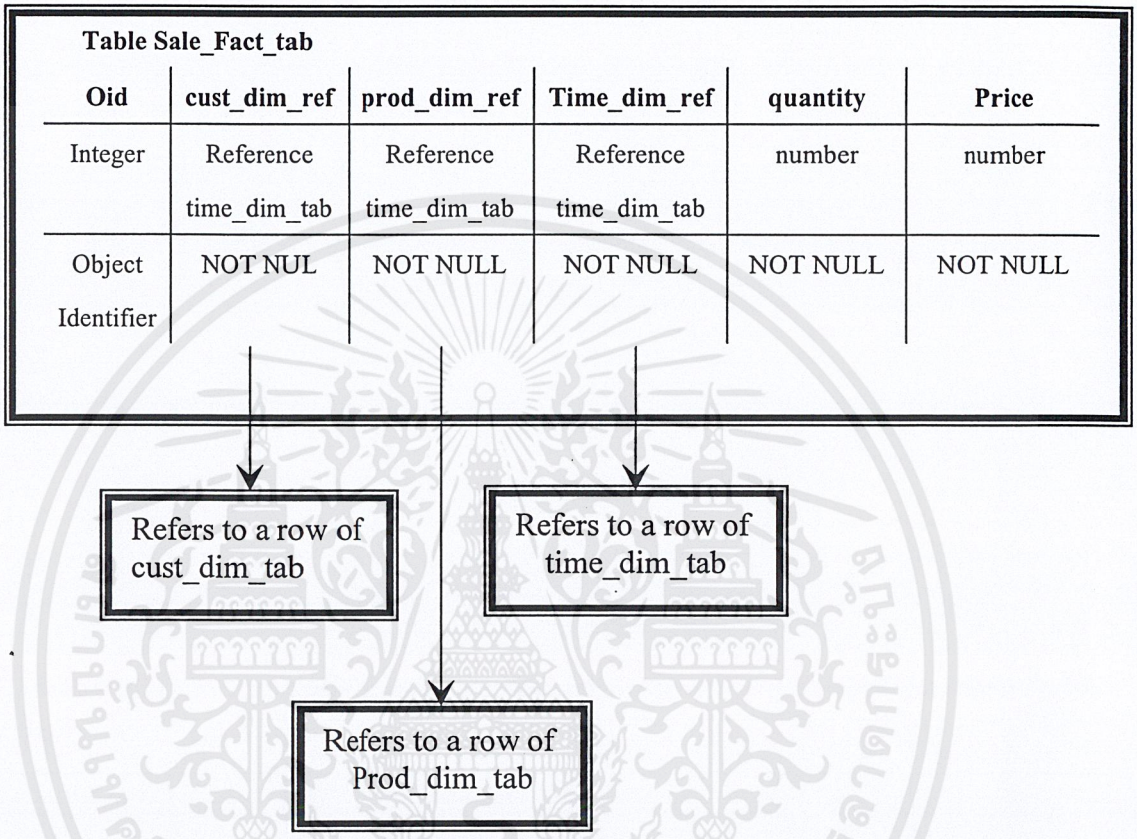
```
CREATE TABLE cust_dim_tab OF cust_dimension_t
(       REF IS OID USER GENERATED,
        name            WITH OPTIONS NOT NULL,
        birthday        WITH OPTIONS NOT NULL,
        sex             WITH OPTIONS NOT NULL,
        address         WITH OPTIONS NOT NULL,
        phone           WITH OPTIONS NOT NULL,
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
status_desc        WITH OPTIONS NOT NULL
)
```

5.1.4 ตารางแสดงข้อมูลการขายสินค้า(Sale Fact Table)

สำหรับตารางแฟคที่อยู่ตรงกลางของไดอะแกรมนั้น จะมีโครงสร้างง่าย ๆ ดังรูปที่ 5-7



รูปที่ 5-7 Object – Relational Representation of the Sale Fact Table

เหมือนกับตารางอื่น ๆ คือจะต้องสร้างโครงสร้างของออปเจ็ค ขึ้นมาก่อน ในที่นี้ก็คือ sale_fact_t ซึ่งในตารางแฟค นั้นจะมีแอททริบิวต์อยู่ 3 ตัวที่ทำหน้าทีเป็นพอยเตอร์ชี้ไปยังตารางอื่น ๆ ซึ่งทั้ง 3 ตัวนี้ จะต้องมีทุกครั้งที่มีการเพิ่มแถวเข้าไปยังตาราง สำหรับคำสั่งที่ใช้ในการสร้าง Sale_Fact_tab นั้นแสดงได้ดังข้างล่างนี้

```
CREATE TYPE sale_fact_t AS
(
cust_dim        REF(cust_dimension_t),
product_dim     REF(prod_dimension_t),
time_dim        REF(time_dimension_t),
quantity        NUM,
price           NUM
)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MODE DB2SQL
REF USING INTEGER

CREATE TABLE sale_fact_tab OF sale_fact_t
(
REF IS OID USER GENERATED,
cust_dim WITH OPTION NOT NULL,
product_dim WITH OPTION NOT NULL,
time_dim WITH OPTION NOT NULL,
quantity WITH OPTION NOT NULL,
price WITH OPTION NOT NULL
)
```

และในตาราง Sale_fact_tab นั้นมีแอททริบิวต์ 3 ตัวที่ทำหน้าที่เป็นพอยเตอร์ดังนั้นจึงต้องทำการ Add Scope เพื่อให้แอททริบิวต์ทั้ง 3 นั้นสามารถชี้ไปได้ถูกที่ สำหรับคำสั่งที่ใช้ในการเพิ่มมุมอง(Add Scope)นั้นจะใช้คำสั่ง ALTER TABLE ดังคำสั่งข้างล่างนี้

```
ALTER TABLE sale_fact_tab ALTER COLUMN time_dim ADD SCOPE time_dim_tab
ALTER TABLE sale_fact_tab ALTER COLUMN product_dim ADD SCOPE prod_dim_tab
ALTER TABLE sale_fact_tab ALTER COLUMN cust_dim ADD SCOPE cust_dim_tab
```

5.2 คำสั่งที่ใช้ในการเปลี่ยนแปลงข้อมูล

โดยปกติแล้ว ดาต้าแวร์เฮ้าท์นั้นจะเก็บข้อมูลชนิดอ่านอย่างเดียว การเก็บข้อมูลลงดาต้าแวร์เฮ้าท์นั้นไม่จำเป็นมากนัก แต่อย่างไรก็ตามหากแอททริบิวต์ในตารางถูกเปลี่ยน ข้อมูลเก่าก็จะถูกแทนที่ด้วยข้อมูลใหม่ที่เข้ามา หรือข้อมูลใหม่อาจจะถูกแทรกลงตารางโดยที่ข้อมูลเก่าไม่ได้ถูกเปลี่ยนแปลง สำหรับการลบข้อมูลนั้นแทบจะไม่จำเป็นในการใช้งานเลย เพราะว่าที่เก็บข้อมูล (Data Storage) มีราคาถูก แต่ถึงอย่างไรก็ตามการอัพเดพ การลบ และการแทรกข้อมูลลงดาต้าแวร์เฮ้าท์ก็ยังคงต้องมีเพราะยังคงมีการเปลี่ยนแปลงข้อมูลในแต่ละไดเมนชั่นอยู่ สำหรับในหัวข้อนี้เราจะกล่าวถึง การแทรก การลบ และการอัพเดตข้อมูล

ในหัวข้อที่ผ่านมานั้นเราแบ่งออกเป็น 4 หัวข้อย่อย แต่ละหัวข้อย่อยก็จะกล่าวถึงแต่ละตาราง สำหรับในหัวข้อนี้เราก็จะพูดถึงภาษาที่ใช้ในการเปลี่ยนแปลงข้อมูลไปทีละตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.1 ตารางไดเมนชั่นของเวลา

สำหรับคำสั่งที่ใช้ในการแทรกข้อมูลลงในตาราง time_dim_tab นั้น จะแสดงให้เห็นดังข้างล่างนี้

```
INSERT INTO time_dim_tab VALUES (
time_dimension_t(0),'1999Q1M01',1999,'Quarter1','January');
```

การอัพเดตข้อมูลสามารถทำได้โดยการใช้คำสั่ง

```
UPDATE time_dim_tab
SET fiscal_year = 1998,
fiscal_qtr='Quarter2',
fiscal_month = 'April'
WHERE time_key = '1999Q1M01'
```

การลบข้อมูลสามารถทำได้โดยการใช้คำสั่ง

```
DELETE FROM time_dim_tab
WHERE time_key = '1999Q1M01'
```

จากการศึกษาเราจะพบว่า คีย์ที่เรานำมาใช้ในการอ้างถึง (time_key : อาจจะมีหรือไม่ก็ได้เพราะยังไงก็มี OID อยู่แล้ว แต่ในที่นี้ให้มีเพราะเพื่อให้ง่ายต่อการอ้างถึง) นั้นเป็นคีย์ที่เป็นลักษณะคีย์ที่มีความหมาย (Meaningful values) ข้อดีของการใช้คีย์ที่มีความหมายคือการที่ไม่ต้องประยุกต์ใช้ในการตอบคำถาม แต่ในทางปฏิบัติแล้วการใช้คีย์ที่มีความหมายเป็นสิ่งที่ควรหลีกเลี่ยงเพราะหากต้องมีการเปลี่ยนแปลงค่าแล้ว จะทำให้คีย์จะต้องถูกเปลี่ยนด้วยซึ่งเป็นสิ่งที่ไม่สมควรทำอย่างยิ่ง

5.2.2 ตารางไดเมนชั่นของสินค้า

สำหรับคำสั่งในการแทรก ลบ หรืออัพเดตข้อมูลใน prod_dim_tab นั้นจะมีลักษณะง่าย ๆ คล้ายกับคำสั่งที่ใช้ในการแทรก ลบ หรืออัพเดตข้อมูลใน time_dim_tab ซึ่งได้กล่าวไปแล้วข้างต้น สำหรับคำสั่งในการแทรกข้อมูลลงใน prod_dim_tab นั้นแสดงได้ดังนี้

```
INSERT INTO  prod_dim_tab VALUES(
        prod_dimension_t(0),
        'Corolla',
        'Toyota',
        'CE 4dr Seden(4cyl,4A)',
        '1800CC')
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลสามารถอัพเดตได้โดยการใช้คำสั่งดังนี้

```
UPDATE prod_dim_tab
SET model_name = 'Civic',
Type  = '2 Door Sedan',
Engine_size = '500CC'
WHERE model_name = 'Corolla'
```

คำสั่งข้างล่างนี้เป็นคำสั่งที่ใช้ในการลบ

```
DELETE FROM prod_dim_tab
WHERE model_name = 'Civic', Type  = '2 Door Sedan'
```

5.2.3ตารางไดเมนชั่นของลูกค้า

สำหรับไดเมนชั่นนี้จะแตกต่างจากไดเมนชั่นอื่น เนื่องจากไดเมนชั่นของลูกค้านั้นจะประกอบไปด้วยข้อมูลประเภทโครงสร้างซึ่งได้แก่ข้อมูลประเภท คอมโพสิสแอททริบิวต์ , มัลติแวลูแอททริบิวต์ ซึ่งภาษาที่ใช้ในการเปลี่ยนแปลงข้อมูลในดาต้าแวร์เฮ้าท์ก็จะมีลักษณะที่แตกต่างออกไป ดังตัวอย่างข้างล่างนี้ก็จะเป็นการแทรกข้อมูลลงในตารางไดเมนชั่นของลูกค้า ที่ชื่อ cust_dim_tab

```
INSERT INTO cust_dim_tab VALUES (
cust_dimension_t(0),
name_t()..First('Jonathan')..Middle('-')..Last('Abbott'),
'12/11/1999',
'M',
Address_t()..Street('142 S Sheridan Road')..City('Tulsa')..State('Oklahoma')..Zipcode('74112')
,phone_array_t()
..phone1(phone_t()..Location('Home')..phone_num('918-835-3161'))
..phone2(phone_t()..Location('Mobile')..Phone_num('338-331-4463'))
..phone3(phone_t()..Location('')..phone_num(''))
..phone4(phone_t()..Location('')..phone_num(''))
,status_desc_arr()
..desc1(status_desc_t()
  ..Reg_id('R01')
  ..reg_date((select oid from time_dim_tab where time_key = '1999Q3M09'))
  ..marital_status('Unmarried')
  ..year_income(60000)
  ..city('Tulsa')
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  ..state('Oklahoma'))
)
```

สำหรับคำสั่งที่ใช้ในการอัพเดตข้อมูลนั้นเนื่องจากว่า IBM DB2 ยังไม่สนับสนุนการทำงานในรูปแบบข้อมูลประเภทโครงสร้างมากนัก ทำให้การอัพเดตข้อมูลในตารางที่มีข้อมูลประเภทโครงสร้างอยู่นั้นจำเป็นต้องเอาข้อมูลเก่าเก็บลงไปด้วย เพื่อไม่ให้ข้อมูลเก่าหายดังแสดงได้ในตัวอย่างข้างล่างนี้

```
UPDATE cust_dim_tab
SET status_desc = (status_desc_arr()
..desc1(status_desc_t()
  ..Reg_id('R01')
  ..reg_date((select oid from time_dim_tab where time_key = '1999Q3M09'))
  ..marital_status('Unmarried')
  ..year_income(60000)
  ..city('Tulsa')
  ..state('Oklahoma'))
..desc2(status_desc_t()
  ..Reg_id('R02')
  ..Reg_date((select oid from time_dim_tab where time_key = '2001Q1M01'))
  ..marital_status('Married')
  ..Year_income(66000)
  ..city('Skokie')
  ..state('Illinolis'))
..desc3(status_desc_t()
  ..Reg_id('R03')
  ..Reg_date((select oid from time_dim_tab where time_key = '2002Q1M03'))
  ..marital_status('Married')
  ..Year_income(70000)
  ..city('Tulsa')
  ..state('Oklahoma'))
)
WHERE name..first = 'Jonathan'
```

คำสั่งสุดท้ายเป็นคำสั่งที่เกี่ยวข้องกับการลบข้อมูลในตาราง แสดงให้เห็นได้ดังนี้

```
DELETE FROM cust_dim_tab where name.first = 'Jonathan'
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.4 ตารางแสดงข้อมูลการขายสินค้า

แอททริบิวต์ของตารางแฟคนั้นจะเป็นแอททริบิวต์ง่าย ๆ ไม่ซับซ้อน สำหรับการแทรกข้อมูลลงตารางนั้นเราจะใช้คำสั่งดังต่อไปนี้

```
INSERT INTO sale_fact_tab VALUES(
sale_fact_t(0),
(select oid from cust_dim_tab where name..first = 'Jonathan'),
(select oid from prod_dim_tab where model_name = 'CR-V'),
(select oid from time_dim_tab where time_key = '1999Q3M09'),
1,21550)
```

และสามารถอัพเดตข้อมูลได้โดยใช้คำสั่ง

```
UPDATE sale_fact_tab
SET time_dim = (select oid from time_dim_tab where time_key = '1999Q1M01')
WHERE time_dim->time_key  = '2000Q1M02'
```

5.3 ตัวอย่างการใช้คำสั่งเอสคิวแอลในการตอบคำถาม

จุดประสงค์หลักสำคัญของหัวข้อนี้เพื่อที่จะศึกษาการตอบคำถามข้อมูลในแต่ละแบบ ในขณะที่มีการคำนวณหาผลลัพธ์ เราจะสามารถอ้างถึงข้อมูลประเภทที่มีโครงสร้างซับซ้อนได้อย่างไร

ในหัวข้อนี้เราก็จะแสดงถึงตัวอย่างของการตอบคำถามทางด้านธุรกิจ เพื่อให้ง่ายต่อการเข้าใจเราจะแสดงตัวอย่างของการตอบคำถามประเภทง่าย ๆ ก่อน แล้วค่อย ๆ ยากขึ้นไป

ตัวอย่างแรกนั้นเราจะหาคำตอบของจำนวนรถที่ถูกขายไปในแต่ละปี เช่นปี 2000 ตัวอย่างนี้เกี่ยวข้องกับตารางแฟคและ ตารางไดเมนชั่นของเวลา ซึ่งอ้างถึงโดยใช้ time_dim_ref ในตารางตารางแฟค ซึ่งจะต่างกับการตอบคำถามในฐานข้อมูลเชิงสัมพันธ์ตรงที่เราไม่จำเป็นต้องใช้การเชื่อมตาราง(Joint Table) ดังแสดงในตัวอย่าง

```
SELECT count(*)
FROM sale_fact_tab
WHERE time_dim->fiscal_year = 2000
```

คำถามต่อไปเราจะตอบคำถามที่ว่าที่ว่ามีรถกี่คันที่ถูกผลิตในบริษัท Jeep และถูกขายได้ในปี 2000 ซึ่งจะเกี่ยวข้องกับไดเมนชั่น 2 ไดเมนชั่นคือตารางไดเมนชั่นของเวลาและตารางไดเมนชั่นของสินค้าในตาราง Fact Table

```
SELECT count(*)
FROM sale_fact_tab
WHERE time_dim->fiscal_year = 2000 AND product_dim->manufacturer = 'Jeep'
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างต่อไปจะตอบคำถามที่ว่าจงหาชื่อบริษัทผู้ผลิตรถยนต์ที่มียอดจำหน่ายรถสูงที่สุดในปี 2000

```
SELECT sf1.product_dim->manufacturer
FROM sale_fact_tab sf1
WHERE sf1.time_dim->fiscal_year = 2000
GROUP BY sf1.product_dim->manufacturer
HAVING sum (sf1.price) >= all (select sum(sf2.price)
        FROM sale_fact_tab sf2
        WHERE sf2.time_dim->fiscal_year = 2000
        GROUP BY sf2.product_dim->manufacturer)
```

ถึงแม้ว่าการเชื่อมตารางจะไม่จำเป็นต่อการตอบคำถามสักเท่าไรนัก แต่สำหรับการทำซัปคิวรี่ก็ยังคงเป็นสิ่งที่หลีกเลี่ยงไม่ได้

ตัวอย่างต่อไปเป็นการตอบคำถามที่ว่า มีรถกี่คันที่ผลิตโดยบริษัท Jeep และขายให้กับลูกค้าที่เป็นเพศชาย ในปี 2000 จากตัวอย่างนี้เราจะเห็นได้ว่าจะเกี่ยวข้องกับ 3 ไดเมนชั่นของตารางแฟคดังตัวอย่างข้างล่างนี้

```
SELECT count(*)
FROM sale_fact_tab
WHERE time_dim->fiscal_year = 2000 AND product_dim->manufacturer = 'Jeep'
        AND cust_dim->sex = 'M'
```

คำถามต่อไปเป็นคำถามที่ถามว่า มีลูกค้าที่เป็นเพศชายกี่คนที่ซื้อรถในระหว่างปี 2000

```
SELECT count(cust_dim->name..First)
FROM sale_fact_tab
WHERE time_dim->fiscal_year = 2000
AND (cust_dim->status_desc..desc1..marital_status = 'Married'
    OR cust_dim->status_desc..desc2..marital_status = 'Married'
    OR cust_dim->status_desc..desc3..marital_status = 'Married')
AND cust_dim->sex = 'M'
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตัวอย่างข้างต้นจะเห็นว่าเนื่องจากว่าสถานะการสมรสของลูกค้านั้นอยู่ในข้อมูลประเภทโครงสร้างซับซ้อน ดังนั้นการเข้าถึงจึงต้องตรวจสอบจากสถานะทั้งหมดที่มีของลูกค้า

คำถามต่อมาเราก็จะศึกษาการเข้าถึงการตอบคำถามจากข้อมูลประเภทโครงสร้างซับซ้อนทั้งที่เป็นมัลติแวลูแอททริบิวต์และที่ไม่เป็นมัลติแวลูแอททริบิวต์ เป็นคำถามที่ถามว่า มีลูกค้าที่เป็นหญิง และมีสถานะเป็นหม้าย มีที่อยู่อยู่ที่แมนเชสเตอร์ ซื้อรถที่ผลิตโดยบริษัท JEEP ระหว่างปี 2000 จำนวนกี่คน

```
SELECT count(cust_dim->name..first)
FROM sale_fact_tab
WHERE time_dim->fiscal_year = 2000
AND cust_dim->sex = 'F'
AND (cust_dim->status_desc..desc1..marital_status = 'Divorced'
        OR cust_dim->status_desc..desc2..marital_status = 'Divorced'
        OR cust_dim->status_desc..desc3..marital_status = 'Divorced')
AND cust_dim->address..state = 'Messachusetts'
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
ผลการทดสอบ

วัตถุประสงค์ของงานวิจัยชิ้นนี้ก็เพื่อที่จะศึกษาถึงผลการนำเอาเทคโนโลยีแบบออบเจ็กรีเลชั่นมาใช้กับดาต้าแวร์เฮ้าท์ซิ่ง และเพื่อเป็นการตรวจสอบว่าเทคโนโลยีแบบออบเจ็กรีเลชั่นนั้นสามารถเพิ่มประสิทธิภาพการทำงานของดาต้าแวร์เฮ้าท์ได้ ซึ่งแวร์เฮ้าท์ส่วนใหญ่จะใช้ฐานข้อมูลแบบเชิงวัตถุสัมพันธ์ และเพื่อเป็นการพิสูจน์ ว่าฐานข้อมูลแบบวัตถุเชิงสัมพันธ์นั้นจะสามารถแก้ปัญหาด้านข้อมูลโครงสร้างได้ดีกว่าการใช้ฐานข้อมูลแบบเชิงสัมพันธ์

ผลงานวิจัยจะแบ่งออกเป็น 2 ส่วน ในส่วนแรกนั้นจะอยู่ในบทที่ 4 ในเรื่องการพัฒนาฐานข้อมูลบนโอเปอเรชั่นนอล ดาต้าเบส เพื่อให้แวร์เฮ้าท์มีความสมบูรณ์ตามที่ต้องการ แวร์เฮ้าท์ ดาต้าเบสนั้นยอมรับได้ถ้าเกิดมีปัญหาขึ้นระหว่างที่ทำการแปลงคิวรี่ ทำให้ต้องมีการปรับปรุงฐานข้อมูลใหม่ ส่วนที่สองเป็นการใช้งานจริงกับฐานข้อมูลซึ่งแสดงในบทที่ 5 ในระหว่างการพัฒนาด้วยเทคโนโลยีแบบออบเจ็กรีเลชั่นนอลนั้น ได้มี ทำให้รู้ถึงประโยชน์สองอย่างของการพัฒนาด้วยเทคโนโลยีแบบออบเจ็กรีเลชั่นก็คือ การแก้ปัญหาเรื่องข้อมูลประเภทโครงสร้าง และการแปลงคิวรี่ซึ่งจะกล่าวใน 6.1 และ 6.2 ตามลำดับ และปัญหายาก ๆ ด้านเทคนิคของฐานข้อมูลแบบวัตถุเชิงสัมพันธ์จะกล่าวใน 6.3

6.1 การทำงานกับข้อมูลประเภทโครงสร้าง

ดาต้าแวร์เฮ้าท์นั้นจะเป็นน่อนวอลาไทลและจะเก็บเพียงข้อมูลที่เอาไว้เพื่อใช้อ่านเท่านั้น ทูเปิลใหม่ที่ใส่เข้ามาส่วนมากจะใส่ไว้ที่แฟกเทเบิ้ล แต่มีบางอันที่จะใส่ไว้ที่ไดเมนชั่น เทเบิ้ล ตัวอย่างที่น่าศึกษา คือ สมมุติว่ามีลูกค้าใหม่ต้องการที่จะซื้อรถ โดยทฤษฎีนั้นจะไม่มีการอัพเดทข้อมูลที่อยู่ในไดเมนชั่นอย่างแน่นอน ซึ่งไม่ทางที่จะเป็นจริงในโลกของจริง ในกรณีนี้สมมุติว่ารายได้ของลูกค้าเกิดมีการเปลี่ยนแปลงขึ้นในการซื้อรถครั้งถัดมา ถ้าข้อมูลประเภทโครงสร้างมีการแก้ปัญหาไม่ดี จะทำให้เกิดปัญหาในการนำข้อมูลไปใช้ในการวิเคราะห์ได้

มีหลายแนวทางที่ใช้แก้ปัญหาข้อมูลประเภทโครงสร้าง โดยวิธีเหล่านี้จะตั้งอยู่บนเทคโนโลยีแบบรีเลชั่น ซึ่งได้รวบรวมไว้โดย Giovinazzo 2000

- Record Overwriting

  เป็นการเขียนข้อมูลใหม่ทับบนข้อมูลเก่าที่มีอยู่ วิธีนี้จะทำให้ข้อมูลเก่าที่มีอยู่หายไปอย่างถาวร เป็นการละเมิดอินทิกริตี้ของดาต้าแวร์เฮ้าท์ และจะทำให้การวิเคราะห์ข้อมูลเกิดความผิดพลาดได้

- New Record Creation

  จะมีการสร้างเรคคอร์ดใหม่ที่มีข้อมูลอัพเดทขึ้นในดาต้าแวร์เฮ้าท์ โดยไม่มีการเปลี่ยนแปลงข้อมูลของเก่า วิธีนี้จะไม่สามารถบอกถึงความสัมพันธ์กันระหว่างเรคคอร์ดใหม่และเรคคอร์ดเก่าได้ ซึ่งวิธีนี้จะทำให้สูญเสียข้อมูลเกี่ยวกับพฤติกรรมของลูกค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Old/Current Fields

ใช้วิธีการเปลี่ยนแปลงโครงสร้างไดเมนชั่นของเรคคอร์ด ให้ข้อมูลทั้งเก่าและใหม่มีการปรับเข้าหากัน วิธีการนี้ค่อนข้างยุ่งยากตรงที่ต้องคาดเดาจำนวนครั้งที่ข้อมูลจะมีเกิดเปลี่ยนแปลง

- Record Versioning

จะมีการสร้างเรคคอร์ดขึ้นมาเพื่อใช้เก็บเวอร์ชั่นของข้อมูล เรคคอร์ดนี้จะเป็นคีย์หลักในไดเมนชั่น เรคคอร์ด แม้ว่าวิธีจะสามารถแก้ปัญหาได้ แต่จะมีความยุ่งยากในเรื่องความสัมพันธ์ระหว่างแฟกเทเบิ้ล และ ไดเมนชั่น เทเบิ้ล และทำให้การคิวรี่ข้อมูลยากมากขึ้นตาม

- Record Linking

เมื่อไดเมนชั่น เทเบิ้ลมีการเปลี่ยนแปลง มีการสร้างเรคคอร์ดใหม่ขึ้น พร้อมคีย์หลัก และจะมีไคลแอน ไอดี อยู่ที่ฝั่งโอเปอเรชั่นนอล ซิสเต็ม ใช้ลิงค์กับเวอร์ชั่นของข้อมูลที่อยู่ในไดเมนชั่นเรคคอร์ด วิธีนี้เป็นวิธีที่ดีที่สุดในการใช้กับฐานข้อมูลแบบเชิงสัมพันธ์

งานวิจัยนี้ใช้เทคโนโลยีแบบออบเจ็กรีเลชั่นในการแก้ปัญหาด้านข้อมูลประเภทโครงสร้างในไดเมนชั่นต่าง ๆ มี 2 เทคนิคที่ในการแก้ปัญหา อันแรก การให้ข้อมูลประเภทโครงสร้างเก็บเป็นข้อมูลประเภท row type เพื่อให้สามารถเก็บข้อมูลแบบคอมโพสิตและแบบมัลติแวลูได้ อันที่สอง การเก็บข้อมูลเกี่ยวเวลาแบบอ้างอิงคล้ายกับเทมโพรอล ดาต้าเบส เพื่อใช้แสดงช่วงเวลาที่ข้อมูลได้ถูกกระทำ ทั้งสองวิธีเป็นความสามารถเฉพาะในการแก้ปัญหาข้อมูลประเภทโครงสร้าง ซึ่งข้อดีที่ทำให้ทั้งสองวิธีดีกว่า 5 เทคนิคที่ใช้กับฐานข้อมูลแบบเชิงสัมพันธ์นั้นก็คือ

- ข้อมูลในแต่ละเอนติตี้จะมีการจัดเก็บเพียงครั้งเดียวทำให้ใช้พื้นที่ในการจัดเก็บ
- การทำดาต้า แมนิพูเลชั่นสามารถทำได้ง่ายและเร็วกว่า
- ง่ายต่อการเข้าถึงข้อมูลผ่านทางแฟก เทเบิ้ล

6.2 การเปลี่ยนแปลงของคำสั่งคิวรี้

ดาต้าเดฟฟินิชั่น และ ดาต้าแมนิพูเลชั่น ของ ออบเจ็กรีเลชั่นนอล แวร์เฮ้าท์ ดาต้าเบส ได้กล่าวใน 5.1 และ 5.2 ของ บทที่ 5 แล้วซึ่งถ้าเปรียบเทียบกับฐานข้อมูลแบบเชิงสัมพันธ์ คำสั่งที่ใช้จะมีความซับซ้อนมากกว่าและจะซับซ้อนมากขึ้นเมื่อมีการใช้คำสั่งกับข้อมูลประเภทมัลติแวลู และประเภท คอมโพสิต

ในการสร้างไทด์ข้อมูล สร้างตาราง สร้างวิว จะมีการทำขึ้นเพียงครั้งเดียวเท่านั้น และไม่ต้องทำการยุ่งเกี่ยวอีกถ้าโครงสร้างฐานข้อมูลไม่มีการเปลี่ยนแปลง เกิดขึ้นได้ยากมากในกรณีที่มีการออกแบบฐานข้อมูลอย่างดี การทำดาต้าแมนิพูเลชั่นนั้นจะทำจาก ทรานแซคชั่น โพรเซสซิ่ง ดาต้าเบส ไปยัง แวร์เฮ้าท์ดาต้าเบส ต้องมีการทำอย่างเป็นประจำโดยซอฟแวร์อัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 ปัญหาของฐานข้อมูลแบบวัตถุเชิงสัมพันธ์

แม้ว่าฐานข้อมูลแบบวัตถุเชิงสัมพันธ์จะมีข้อดีกว่าฐานข้อมูลแบบเชิงสัมพันธ์ในการสร้างดาต้า แวร์เฮ้าท์ก็ตาม แต่ก็มีข้อเสียอยู่อย่างหนึ่ง คือ ขาดมาตราฐาน ถึงแม้ว่าฐานข้อมูลแบบวัตถุเชิงสัมพันธ์จะมีให้ใช้ได้หลายปีแล้ว แต่ภาษาที่ใช้ในการติดต่อกับฐานข้อมูล SQL3 นั่นยังไม่มีความแน่นอน ผู้ผลิตซอฟแวร์แต่ละรายใช้มาตราฐานของภาษาของแต่ละราย ซึ่งแต่ละรายมีระเอียดของภาษาที่ไม่เหมือนกัน แม้ว่าผู้ผลิตแต่ละจะมีแนวความคิดเดียวกัน

ปัญหาที่สองนั่นคือการไม่มาตราฐานของคุณลักษณะของออบเจ็ก ผู้ผลิตแต่ละรายแก้ปัญหามัลติแวลูไม่เหมือนกัน ซึ่งเป็นการยากที่จะใช้ฐานข้อมูลข้ามผลิตภัณฑ์

ปัญหาที่กล่าวมาจะได้รับการแก้ไขอีกไม่นาน เมื่อมีการออกมาตราฐานของฐานข้อมูลแบบวัตถุเชิงสัมพันธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7
บทสรุปและคำแนะนำ

จากข้อดีทั้งระบบฐานข้อมูลเชิงสัมพันธ์นั้น การผสมผสานความสามารถของฐานข้อมูลเชิงสัมพันธ์กับฐานข้อมูลเชิงวัตถุ อันก่อให้เกิดฐานข้อมูลเชิงวัตถุสัมพันธ์ขึ้นนั้น จากการดำเนินงานที่ผ่านมาทำให้สรุปได้แล้วว่าการนำเทคนิคการออกแบบฐานข้อมูลแบบออปเจครีเลชั่นนอลนั้นช่วยเพิ่มประสิทธิภาพของดาต้าแวร์เฮ้าท์ได้มากกว่าการนำเทคนิครีเลชั่นนอลดาต้าเบสมาออกแบบ 3 ประการคือ

1. ฐานข้อมูลที่มีลักษณะเป็นฐานข้อมูลเชิงวัตถุสัมพันธ์นั้นรองรับความสามารถในการทำมัลติแวลูแอททริบิวต์ ซึ่งมันจะช่วยให้ฐานข้อมูลสามารถรองรับโครงสร้างข้อมูลที่มีความสลับซับซ้อน มีการเปลี่ยนแปลงแบบค่อยเป็นค่อยไป
2. โครงสร้างคำสั่งที่ใช้ในการตอบคำถามนั้นง่าย และมีโค๊ดเพียงไม่กี่บรรทัด ถ้าหากฐานข้อมูลเชิงวัตถุสัมพันธ์ถูกนำมาใช้
3. ฐานข้อมูลรองรับการทำงานแบบถ่ายทอดคุณสมบัติมาจากคลาสต้นแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
ตารางข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table A.1. Test Data of the Time_dim_tab Table

time_key (CHAR(10))	fiscal_year (NUMBER(4))	fiscal_quarter (CHAR(10))	Fiscal_month (CHAR(10))
1999Q1M01	1999	Quarter1	January
1999Q1M02	1999	Quarter1	February
1999Q1M03	1999	Quarter1	March
1999Q1M04	1999	Quarter2	April
1999Q1M05	1999	Quarter2	May
1999Q1M06	1999	Quarter2	June
1999Q1M07	1999	Quarter3	July
1999Q1M08	1999	Quarter3	August
1999Q1M09	1999	Quarter3	September
1999Q1M10	1999	Quarter4	October
1999Q1M11	1999	Quarter4	November
1999Q1M12	1999	Quarter4	November
2000Q1M01	2000	Quarter1	January
2000Q1M02	2000	Quarter1	February
2000Q1M03	2000	Quarter1	March
2000Q1M04	2000	Quarter2	April
2000Q1M05	2000	Quarter2	May
2000Q1M06	2000	Quarter2	June
2000Q1M07	2000	Quarter3	July
2000Q1M08	2000	Quarter3	August
2000Q1M09	2000	Quarter3	September
2000Q1M10	2000	Quarter4	October
2000Q1M11	2000	Quarter4	November
2000Q1M12	2000	Quarter4	November
2001Q1M01	2001	Quarter1	January
2001Q1M02	2001	Quarter1	February
2001Q1M03	2001	Quarter1	March
2001Q1M04	2001	Quarter2	April
2001Q1M05	2001	Quarter2	May
2001Q1M06	2001	Quarter2	June
2001Q1M07	2001	Quarter3	July
2001Q1M08	2001	Quarter3	August
2001Q1M09	2001	Quarter3	September
2001Q1M10	2001	Quarter4	October
2001Q1M11	2001	Quarter4	November
2001Q1M12	2001	Quarter4	November
2002Q1M01	2002	Quarter1	January
2002Q1M02	2002	Quarter1	February
2002Q1M03	2002	Quarter1	March
2002Q1M04	2002	Quarter2	April
2002Q1M05	2002	Quarter2	May
2002Q1M06	2002	Quarter2	June

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table A.2 Test Data of prod_dim_tab Table.

oid (intger)	model_name (VARCHAR(30))	manufacturer (VARCHAR(25))	type (VARCHAR(50))	engine_size (CHAR(6))
1	Corolla	Toyota	CE 4dr Sedan (4cyl, 4A)	1800CC
2	RAV4	Toyota	4WD 4dr SUV (4cyl, 4A)	2000CC
3	Accord	Honda	EX 4dr Sedan (4cyl, 4A)	2300CC
4	CR-V	Honda	EX AWD 4dr Sedan (4cyl, 4A)	2400CC
5	Golf	Volkswagen	GLS TDI 4dr SUV (4cyl, 4A)	1900CC
6	BMW 5 Series	BMW	540i 4dr Sedan (8cyl, 5A)	4400CC
7	A4 Series	Audi	Quattro AWD 4dr Sedan (8cyl, 4A)	3000CC
8	Crown Victoria	Ford	LX 4dr Sedan (8cyl, 4A)	4600CC
9	Grand Cherokee	Jeep	Laredo 4WD 4dr SUV (8cyl, 5A)	4700CC
10	GS 300	Lexus	4dr Sedan (6cyl, 5A)	3000CC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Test Data of the Cust_Dim_Tab

oid	name			sex (CHAR(6))	birthday (DATE)	address				phone_var		status_desc					
	last (VARCHAR(30))	first (VARCHAR(30))	middle (VARCHAR(15))			street (VARCHAR(100))	city (VARCHAR(25))	state (VARCHAR(25))	zipcode (VARCHAR(25))	location (VARCHAR(15))	phone_number (VARCHAR(12))	reg_number (CHAR(3))	reg_date_ref (REF Datatype)	marital_status (VARCHAR(15))	Yearly_income (NUMBER)	city (VARCHAR(25))	state (VARCHAR(25))
1	Abbott	Jonathan		Male	22-12-1964	1421 S Sheridan Road	Tusa	Oklahoma	74112	Home	918-835-3161	R01	1999Q3M09	Unmarried	6000	Tulsa	Oklahoma
										Mobile	338-331-4463	R02	2001Q1M01	Married	66000	Skokie	Illinois
												R03	2002Q1M03		70000	Tulsa	Oklahoma
2	Alexander	Stephen	P	Male	03-04-1973	1700 West Loop South	Houston	Texas	77027	Home	713-621-9720	R01	1999Q4M11	Unmarried	100000	Nashua	New Hampshire
										Office	713-565-7865	R02	2000Q2M06	Unmarried	100000	Houston	Texas
										Mobile	834-736-6935						
3	Alien	Norene		Female	14-06-1956	15 Southwest Park	Westwood	Massachusetts	02090	Home	781-329-3350	R01	1999Q4M12	Married	90000	Westwood	Massachusetts
										Office	781-329-9875	R02	2000Q3M09	Divorced	120000	Westwood	Massachusetts
										Mobile	456-453-9899	R03	2001Q4M12	Married	150000	Westwood	Massachusetts
4	Soffredo	Robert		Male	21-05-1967	9922 Roosevelt Boulveard	Edwardsville	Pennsylvania	18704	Home	570-283-1860	R01	1999Q4M12	Married	50000	Reading	Pennsylvania
										Office	215-676-0989	R02	2001Q4M12	Married	60000	Hawthorne	New Jersey
										Mobile	610-921-0719	R03	2002Q2M06	Married	70000	Edwardsville	Pennsylvania
5	Shefali	Mehta		Female	06-02-1956	3424 Peachtree Road	Boston	Massachusetts	02116	Home	617-572-2022	R01	2000Q1M03	Married	100000	Chicago	Illinois
										Office	782-453-8963	R02	2002Q1M03	Married	100000	Boston	Massachusetts
6	Chimenya	Esther		Female	14-04-1972	588 North Gulph Road	New York	New York	11377	Home	712-525-3636	R01	2000Q1M03	Unmarried	45000	King of Prussia	Philadelphia
										Office	712-527-4242	R02	2001Q2M06	Married	20000	Ottawa	Illinois
										Mobile	263-783-9898	R03	2002Q2M05	Divorced	60000	New York	New York
7	Chessbrough	John		Male	08-09-1964	128 Intervale Road	Burlington	Tesas	50013	Home	972-661-8960	R01	2000Q2M06	Married	90000	Burlington	Texas
										Office	972-381-0817						
										Mobile	445-677-6555						
8	Holmquist	Jim	Alberto	Male	12-10-1953	11711 Berryessa Road	San Jose	California	23434	Home	408-487-3138	R01	2000Q3M08	Married	100000	Redwood City	California
										Office	408-288-4081	R02	2002Q1M01	Married	120000	San Jose	California
										Mobile	737-784-7474						
9	Claydon	Graham		Female	15-09-1971	11220 Allisonville Road	Fishers	Indiana	64038	Home	317-842-1040	R01	2000Q1M03	Unmarried	65000	Elizabethtown	Kentucky
										Office	381-784-8432	R02	2002Q2M05	Married	70000	Fishers	Indiana
										Mobile	633-734-7843						
10	Conway	Warwick		male	24-08-1969	5801 E. 76th Avenue	Commerce City	Colordo	80022	Home	303-286-0406	R01	2000Q3M08	Unmarried	75000	Las Vegas	Nevada
										Office	303-637-9192	R02	2000Q4M11	Married	95000	Salt Lake City	Utah
										Mobile	722-373-9643	R03	2001Q4M12	Divorced	110000	Commerce City	Colorado

Test Data of the sales_fact_tab Table.

oid (integer)	Cust_dim (REF Datatype)	Prod_dim (REF Datatype)	Time_dim (REF Datatype)	Quantity (NUMBER)	Price (NUMBER)
1	1	4	9	1	21,550
2	1	6	25	1	48,650
3	1	1	39	1	15,550
4	2	2	11	1	18,750
5	2	8	18	1	16,500
6	3	1	12	1	14,330
7	3	5	21	1	17,725
8	3	10	26	1	37,650
9	4	3	12	1	22,440
10	4	7	36	1	31,340
11	4	4	42	1	22,705
12	5	9	15	1	28,670
13	5	8	39	1	27,780
14	6	6	15	1	50,885
15	6	5	30	1	18,560
16	6	7	41	1	31,800
17	7	4	18	1	23,655
18	8	3	20	1	22,760
19	8	1	37	1	14,655
20	9	6	15	1	51,845
21	9	8	41	1	27,785
22	10	5	20	1	18,650
23	10	9	23	1	29,605
24	10	10	36	1	39,180

ภาคผนวก ข
ตัวอย่างการอย่างตอบคำถาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. จำนวนรถทั้งหมดที่ขายได้ในช่วงปี 2000

```
SELECT    count(*)
FROM      sale_fact_tab
WHERE     time_dim->fiscal_year = 2000
```

ผลลัพธ์ count

9

2. จำนวนรถที่ผลิตโดย Jeep และขายในช่วงปี 2000

```
SELECT    count(*)
FROM      sale_fact_tab
WHERE     time_dim->fiscal_year = 2000
          and product_dim->manufacturer = 'Jeep'
```

ผลลัพธ์ count

2

3. ผู้ผลิตรถยนต์รายใดมียอดขายสูงสุดในช่วงปี 2000

```
SELECT    sf1.product_dim->manufacturer
FROM      sale_fact_tab sf1
WHERE     sf1.time_dim->fiscal_year = 2000
GROUP BY  sf1.product_dim->manufacturer
HAVING    sum (sf1.price) >= all (SELECT    sum(sf2.price)
                                  FROM      sale_fact_tab sf2
                                  WHERE sf2.time_dim->fiscal_year = 2000
                                  GROUP BY sf2.product_dim->manufacturer)
```

ผลลัพธ์ manufacturer

BMW

4. จำนวนรถที่ผลิตโดย Jeep และขายให้กับลูกค้าที่เป็นผู้ชายในช่วงปี 2000

```
SELECT    count(*)
FROM      sale_fact_tab
WHERE     time_dim->fiscal_year = 2000
          and product_dim->manufacturer = 'Jeep'
          and cust_dim->sex = 'M'
```

ผลลัพธ์ count

1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. จำนวนลูกค้าผู้ชายที่แต่งงานและซื้อรถในช่วงปี 2000

```
SELECT    count(cust_dim->name..First)
FROM      sale_fact_tab
WHERE     time_dim->fiscal_year = 2000
          and (cust_dim->status_desc..desc1..marital_status = 'Married'
          or cust_dim->status_desc..desc2..marital_status = 'Married' or
          cust_dim->status_desc..desc3..marital_status = 'Married')
          and cust_dim->sex = 'M'
```

ผลลัพธ์ count

4

6. จำนวนของลูกค้าผู้หญิงที่หย่าร้างจาก Massachusetts ซื้อรถที่ผลิตโดย Jeep ในช่วงปี 2000

```
SELECT    count(cust_dim->name..first)
FROM      sale_fact_tab
WHERE     time_dim->fiscal_year = 2000
          and cust_dim->sex = 'F'
          and (cust_dim->status_desc..desc1..marital_status = 'Divorced'
          or cust_dim->status_desc..desc2..marital_status = 'Divorced' or
          cust_dim->status_desc..desc3..marital_status = 'Divorced')
          and cust_dim->address..state = 'Messachusetts'
```

ผลลัพธ์ count

1

7. ราคารวมและจำนวนของรถที่มีเครื่องยนต์ ตั้งแต่ 4000 ซีซี หรือขายให้กับลูกค้าที่ยังไม่แต่งงานที่มีเงินเดือนน้อยกว่า 75000 ในช่วงควอเตอร์แรกในปี 2000

```
SELECT    sum(price),sum(quantity)
FROM      sale_fact_tab
WHERE     time_dim->fiscal_year = 2000
          and time_dim->fiscal_qtr = 'Quarter1'
          and product_dim->engine_size >= '4000CC'
          and (cust_dim->status_desc..desc1..marital_status = 'Unmarried'
          or cust_dim->status_desc..desc2..marital_status = 'Unmarried'
          or cust_dim->status_desc..desc3..marital_status = 'Unmarried')
          and (cust_dim->status_desc..desc1..year_income <= 75000  or
          cust_dim->status_desc..desc2..year_income <= 75000 or
          cust_dim->status_desc..desc3..year_income <= 75000)
```

ผลลัพธ์ sum(price)	sum(quantity)
102730	2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. จำนวนของรถที่ขายให้กับลูกค้าที่ซื้อรถมากที่สุดจากช่วงควอเตอร์ที่ 3 ปี 1999 ถึง ควอเตอร์ที่ 2 ปี 2000

```
SELECT    sum(quantity)
FROM      sale_fact_tab sf1
WHERE     (sf1.time_dim->fiscal_year = 1999
          and sf1.time_dim->fiscal_qtr ='Quarter3')
          or (sf1.time_dim->fiscal_year = 2000
          and sf1.time_dim->fiscal_qtr ='Quarter1')
          or (sf1.time_dim->fiscal_year = 1999
          and sf1.time_dim->fiscal_qtr = 'Quarter4')
          or (sf1.time_dim->fiscal_year = 2000
          and sf1.time_dim->fiscal_qtr ='Quarter2')
GROUP BY  sf1.cust_dim->name..first
HAVING    sum(sf1.quantity) >= all (SELECT sum(sf2.quantity)
          FROM      sale_fact_tab sf2
          WHERE     (sf2.time_dim->fiscal_year = 1999
                    and sf2.time_dim->fiscal_qtr ='Quarter3')
                    or (sf2.time_dim->fiscal_year = 2000 and
                    sf2.time_dim->fiscal_qtr =    'Quarter1')
                    or (sf2.time_dim->fiscal_year = 1999 and
                    sf2.time_dim->fiscal_qtr =    'Quarter4')
                    or (sf2.time_dim->fiscal_year = 2000 and
                    sf2.time_dim->fiscal_qtr =    'Quarter2')
          GROUP BY  sf2.cust_dim->name..first)
```

ผลลัพธ์ sum

2

9. ลูกค้าคนใดบ้างที่ซื้อรถ BMW ช่วงปี 2000

```
SELECT    cust_dim->name..first,
          cust_dim->name..last,
          cust_dim->name..middle,
          cust_dim->address..no,
          cust_dim->address..street,
          cust_dim->address..city,
          cust_dim->address..state,
          cust_dim->address..zipcode,
          cust_dim->phone..phone1..location,
          cust_dim->phone..phone1..phone_num,
          cust_dim->phone..phone2..location,
          cust_dim->phone..phone2..phone_num,
          cust_dim->phone..phone3..location,
          cust_dim->phone..phone3..phone_num,
          cust_dim->phone..phone4..location,
          cust_dim->phone..phone4..phone_num
FROM      sale_fact_tab
WHERE     time_dim->fiscal_year = 2000
          and product_dim->manufacturer =    'BMW'
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลลัพธ์ Name('Chimenya','Esther',null)
Address('588 North Gulph Road','New York','New York','11373')
Phone('Office','712-527-4242','Mobile', '263-783-9898')
Name('Claydon','Graham','null)
Address('11220 Allisoville Road','Fishers','Indiana','64038')
Phone('Home','317-842-1040',
'Office','381-784-8432',
'Mobile','633-734-7843')

10. ควอเตอร์ไหนของปีที่มีการขายรถมากที่สุด

```
SELECT      sf1.time_dim->fiscal_qtr
FROM        sale_fact_tab sf1
GROUP BY    sf1.time_dim->fiscal_qtr
HAVING      sum(sf1.quantity) >= all (SELECT sum(sf2.quantity)
            FROM        sale_fact_tab sf2
            GROUP BY    sf2.time_dim->fiscal_qtr)
```

ผลลัพธ์ fiscal_qtr
quarter1
quarter 4

11. รถยี่ห้อใดที่ขายได้ในปี 2000 และมีขนาดเครื่องยนต์ตั้งแต่ 3000CC เป็นต้นไป

```
SELECT      product_dim->manufacturer
FROM        sale_fact_tab
WHERE       product_dim->engine_size >= '3000CC'
            and time_dim->fiscal_year = 2000
GROUP BY    product_dim->manufacturer
```

ผลลัพธ์ manufacturer
BMW
Ford
Jeep

12. จำนวนลูกค้าที่แต่งงาน และซื้อรถในควอเตอร์ที่ 4 ของแต่ละปี

```
SELECT      time_dim->fiscal_year,count(*)
FROM        sale_fact_tab
WHERE       time_dim->fiscal_qtr = 'Quarter4' and
            (cust_dim->status_desc..desc1..marital_status = 'Married'
            or cust_dim->status_desc..desc2..marital_status = 'Married'
            or cust_dim->status_desc..desc3..marital_status = 'Married')
GROUP BY    time_dim->fiscal_year
```

ผลลัพธ์ Year count
1999 2
2001 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

13. รถยี่ห้อใดบ้างที่ขายได้ในปี 2000 และแต่ละยี่ห้อมีจำนวนเท่าไร

```
SELECT      product_dim->manufacturer,count(*)
FROM        sale_fact_tab
WHERE       time_dim->fiscal_year=2000
GROUP BY    product_dim->manufacturer
```

ผลลัพธ์

manufacturer	count
BMW	2
Ford	1
Honda	2
Jeep	2
Volkswagen	2

14. ลูกค้าในรัฐใด มีการซื้อรถมากที่สุด

```
SELECT      cust_dim->address..state
FROM        sale_fact_tab sf1
GROUP BY    cust_dim->address..state
HAVING      count(*) >= all(SELECT count(*)
FROM        sale_fact_tab sf2
GROUP BY    cust_dim->address..state)
```

ผลลัพธ์

state
Messachusetts

15. ยอดขายโดยรวมของรถแต่ละยี่ห้อในช่วง 4 ปี

```
SELECT      product_dim->manufacturer,sum(price)
FROM        sale_fact_tab
WHERE       time_dim->fiscal_year=1999 or time_dim->fiscal_year=2000
            or time_dim->fiscal_year=2001 or time_dim->fiscal_year=2002
GROUP BY    product_dim->manufacturer
```

ผลลัพธ์

manufacturer	sum(price)
Audi	63140
BMW	200030
Ford	72065
Honda	134660
Jeep	58275
Lexus	76830
Toyota	78835
Volkswagen	54935

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. Agrawal, Rakesh, Ashish Gupta, and Sunita Sarawagi. (1997) "Modeling Multidimensional Databases" Proceedings of the 13th International Conference on Data Engineering – 1997, Institute of Electrical and Electronics Engineers, Birmingham

2. Armstrong, Rob. (December, 2001) "Seven Steps to Optimizing Data Warehouse Performance." IEEE Computer 34, no. 12: 76-79.

3. Anahory, S. and D. Murray. (1997) Data Warehousing in the Real World. England:Addison-Wesley.

4. Bahrami, A. (1998) Object-Oriented Systems Development. Illinious: Irwin/McGraw-Hill.

5. Chaudhuri, S. and U. Dayal. (1997) "An Overview of Data Warehousing and OLTP Technology." ACM Sigmod Record 26, no. 1

6. Chelluri Kiran and Vijay Kumar. (2001) "Data Classification and Management in Very Large Data Warehouses" Proceeding of the Third International Workshop on Advances Issues of E-Commerce and Web-Based Information System(WECWIS'01) Institute of Electrical and Electronics Engineers, San Juan, California, June 21-22

7. Chen, Wei-Chou, Tzung-Pei Hong, and Wei-Yang Lin. (1999) "A Composite Data Model in Object-Oriented Data Warehousing," Proceedings of the 31st Internationa Conference on Technology of Object-Oriented Language and Systems, Institute of Electrical and Electronics Engineers, Nanjing, China, September 22-25

8. Elmasri, R. A. and S. B. Navathe. (2000) Fundamentals of Database System. California:Addison-Wesley.

9. Furlow, Gerri. (July/August, 2001) "The Case for Building a Data Warehouse." IEEE IT Professional 3, no. 4:31-34.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. Giovinazzo, W. A. (2000) "Object-Oriented Data Warehouse Design: A star Schema." New Jersey: Prentice Hall PTR

11. Inmon, W. H. (2002) "Building the Data Warehouse. Wellesley": John Wiley & Sons

12. Inmon, W. H. and C. Kelley. Rdb/VMS: (1993) Developing the Data Warehouse.Boston: QED Publishing Group.

13. Kimball, R. (1997) "A Dimensional Modeling Manifesto." DBMS Online 10, no.9

14. Kimball, (1996) R. The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses. New York: J. Wiley

15. Michael Krippendorf and Il-Yeol Song, "The Translation of Star Schema into Entity-Relationship Diagrams", College of Information Science and Technology Drexel University, Philadelphia, PA 19104

16. Sarda, N. L. (1999) "Temporal Issues in Data Warehouse Systems," Proceeding of the International Symposium on Database Applications in Non-Traditional Environments (DANTE'99), Institute of Electrical and Electronics Engineers, Kyoto, Japan, November 28-30

17. Stonebraker, M. and P Brown. (1999) Object-Relational DBMS: Tracking the Next Great Wave. San Francisco: Morgan Kaufmann.

18. Trujillo, Juan, Manuel Palomar, Jaime Gomez, and Il-Yeol Song. (December, 2001) "Designing Data Warehouses with OO Conceptual Models." IEEE Computer 34, no. 12 :66-75.

เวปไซต์อ้างอิง

1. www.ibm.com
2. http://www7b.boulder.ibm.com/dmdd/library/techarticle/0206brown/0206brown1.html
3. http://www7b.boulder.ibm.com/dmdd/library/techarticle/0206brown/0206brown2.html

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้