ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

# ระบบดาต้ามาร์ท สำหรับ สินค้าคงคลัง
# กรณีศึกษา บริษัทคาโปรแลคตรัมไทย จำกัด (มหาชน)

# DATAMART SYSTEM FOR INVENTORY
# CASE STUDY: CAPROLACTRUM THAI PUBLIC

โดย

**พิศสมร อิงควระ**

**PITSAMORN INGKAVARA**

**อาจารย์ที่ปรึกษา**

**รศ.ดร.วรพจน์ กรีสุระเดช**

วัน เดือน ปี 04 S.A. 2550
เลขทะเบียน H003489
เลขเรียกหนังสือ ฮพ. พ755ร 2549
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."

b11841288x
1111141140

**รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน**
**หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ**
**คณะเทคโนโลยีสารสนเทศ**
**สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง**
**ภาคเรียนที่ 2 ปี การศึกษา 2549**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# DATA MART SYSTEM FOR INVENTORY

# CASE STUDY: CAPROLACTRUM THAI PUBLIC

**PITSAMORN INGKAVARA**

**A SYSTEM DEVELOPMENT PROJECT**

**OF THE REQUIREMENT FOR THE DEGREE OF**

**MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY**

**FACULTY OF INFORMATION TECNOLOGY**

**KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

**2 /2006**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**COPYRIGHT 2007**

**FACULTY OF INFORMATION TECNOLOGY**

**KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | |
|---|---|
| **หัวข้อ** | ระบบ ดาต้ามาร์ท (สำหรับสินค้าคงคลัง) |
| **นักศึกษา** | นางสาวพิศสมร อิงควระ |
| **รหัสนักศึกษา** | 47066438 |
| **ปริญญา** | วิทยาศาสตรมหาบัณฑิต |
| **สาขาวิชา** | เทคโนโลยีสารสนเทศ |
| **แขนงวิชา** | วิทยาการสารสนเทศ |
| **ปีการศึกษา** | 2549 |
| **อาจารย์ที่ปรึกษา** | รศ.ดร.วรพจน์ กรีสุระเดช |

## บทคัดย่อ

การเพิ่มประสิทธิภาพของการมีอยู่ของข้อมูลได้นำเอากระบวนการที่ใช้ในการวิเคราะห์ข้อมูลเชิงธุรกิจ (Business Intelligence) มาใช้เพื่อศึกษาความต้องการในการใช้ข้อมูล ตั้งแต่ระดับล่างสุด เพื่อนำมาจัดการ เรียบเรียง วิเคราะห์ ออกแบบ และ สกัดสารสนเทศจากแหล่งข้อมูลต่าง ๆ มาแปลงเป็นข้อมูลชนิดเดียวกัน (Extract Transform and Load) ให้อยู่ในรูป ด้าต้ามาร์ตสินค้าคงคลัง รวมทั้งวิเคราะห์หาคำตอบเชิงธุรกิจโดยอาศัยเทคโนโลยี OLAP เป็นเครื่องมือในการประมวลผลข้อมูล โดยการสร้างความสัมพันธ์ระหว่างมุมมองต่าง ๆ เพื่อศึกษาเหตุการณ์ปัจจุบัน และ นำข้อมูลมาวิเคราะห์ ค้นหาข้อมูล ตลอดจนพัฒนา ระบบดาต้ามาร์ท เพื่อการวิเคราะห์สินค้าคงคลัง กรณีศึกษาบริษัทดาโปรแลคตรัมไทย จำกัด (มหาชน) ให้ปรากฏเป็นรูปธรรม โดยองค์กรสามารถนำระบบไปใช้งานได้อย่างมีประสิทธิผลต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | |
|---|---|
| **Title** | Data Mart System (Inventory) |
| **Student** | Miss Pitsamorn Ingkavara |
| **Student ID.** | 47066438 |
| **Degree** | Master of Science |
| **Programme** | Information Science |
| **Academic Year** | 2006 |
| **Advisor** | Ph.D. Associate Profressor Worapoj Greesuradat |


## ABSTRACT

The information efficiency addition will take that processing for business intelligence on behalf of finding requirement usage from the lowing level. Which any information is extracted transformed and loaded to data mart for inventory include analysis business information manner by OLAP technology that is a tool for using in multidimensional data processing. As analysis data mart system for inventory, case study Caprolactrum Thai Plc.




เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# กิตติกรรมประกาศ

โครงการพัฒนาระบบงานฉบับนี้สำเร็จได้ ด้วยความกรุณาของรองศาสตราจารย์ ดร.วรพจน์ กรีสุระเดช อาจารย์ที่ปรึกษาโครงการพัฒนาระบบงาน ซึ่งได้ให้คำปรึกษา ข้อชี้แนะ และความช่วยเหลือในหลายสิ่งหลายอย่างจนกระทั่งลุล่วงไปได้ด้วยดีผู้พัฒนาระบบขอกราบขอบพระคุณเป็นอย่างสูงมา ณ ที่นี้

ขอกราบขอบพระคุณ กรรมการสอบโครงการพัฒนาระบบงานทุกท่านที่ให้ความกรุณาให้คำชี้แนะและให้คำแนะนำในการสร้างเครื่องมือในการพัฒนาระบบ

ขอขอบพระคุณพี่ ๆ แผนกระบบสารสนเทศของบริษัท คาโปรแลคตรัมไทย จำกัด (มหาชน) ที่ให้ความกรุณา และ ให้อนุเคราะห์ ในเรื่องของฐานข้อมูลบริษัท การสละเวลาให้คำอธิบายกระบวนการทำงานของฝ่ายงานต่างๆ ของบริษัท คาโปรแลคตรัมไทย จำกัด (มหาชน)

ขอขอบพระคุณผู้บังคับบัญชา Mr. Tariq Rana พี่ ๆ และ เพื่อน แผนก Web Development ของบริษัท อนิลาเยม อินเตอร์เนชันเนล จำกัด ที่ให้ความกรุณา และ ให้การสนับสนุน ในเรื่องการศึกษา

ขอขอบคุณคุณวณิชญา มลคาน และ ครอบครัว สำหรับการให้ความอนุเคราะห์เรื่องอินเทอร์เน็ต และ กำลังใจที่มีให้เสมอ

ขอบคุณและขอบใจ พี่ เพื่อน และน้องภาควิชาวิทยาการสารสนเทศทุกคน ที่คอยถามไถ่ด้วยความห่วงใยว่าเมื่อไหร่จะพัฒนาระบบเสร็จ และขอขอบคุณพิเศษเพื่อนร่วมรุ่น IS18.1 ที่ลุ้นเอาใจช่วยทุกขณะ รวมถึงผู้มีพระคุณทุกท่านที่มิได้เอ่ยนามไว้ ณ ที่นี้

สุดท้ายนี้ กราบขอบพระคุณ คุณพ่อ และ คุณแม่ บุพการีผู้ให้ทุกสิ่งทุกอย่างกับผู้พัฒนาระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

หน้า


บทคัดย่อภาษาไทย..... I

บทคัดย่อภาษาอังกฤษ.....II

กิตติกรรมประกาศ.....III

สารบัญ.....IV

สารบัญตาราง.....VI

สารบัญรูป.....VII

บทที่1 บทนำ.....1

1.1 ความเป็นมาและความสำคัญของปัญหา.....1

1.2 วัตถุประสงค์ของการศึกษาและการพัฒนาระบบ.....2

1.3 สมมุติฐานของการศึกษา.....3

1.4 ขั้นตอนการศึกษาและพัฒนา.....4

1.5 ขอบเขต และ แนวทางในการพัฒนาระบบ.....6

1.6 ประโยชน์ที่คาดว่าจะได้รับ.....7

บทที่2 ทฤษฎีพื้นฐานที่ใช้ในศึกษา และ พัฒนาระบบดาต้ามาร์ทเพื่อการวิเคราะห์สินค้าคงคลัง...9

2.1 กระบวนการที่ใช้ในการวิเคราะห์ข้อมูลทางธุรกิจเพื่อการตัดสินใจ.....9

2.2 แนวคิดเกี่ยวกับคลังข้อมูล และ ดาต้ามาร์ท.....14

2.3 สถาปัตยกรรมคลังข้อมูล และระบบดาต้ามาร์ท.....17

2.4 การเคลื่อนที่ของข้อมูลในคลังข้อมูล.....18

2.5 วิธีการออกแบบฐานข้อมูลสำหรับคลังข้อมูล และ ดาต้ามาร์ท.....18

2.6 การแปลงข้อมูลเข้าสู่ดาต้ามาร์ท.....20

2.7 แนวคิดของเทคโนโลยี OLAP.....21

2.8 คำจำกัดความของเทคโนโลยี OLAP.....23

2.9 OLAP เครื่องมือสำหรับการวิเคราะห์เพื่อสนับสนุสการตัดสินใจ.....25

2.10 การประยุกต์ใช้ OLAP ในระดับองค์กร.....28



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ (ต่อ)

หน้า


บทที่3 การวิเคราะห์ ออกแบบ ระบบดาต้ามาร์ท สำหรับสินค้าคงคลัง กรณีศึกษา บริษัทดาโปรแลคตรัมไทย จำกัด (มหาชน).....33

3.1 ข้อมูลพื้นฐานของกระบวนการทำงานปัจจุบัน การบริหารสินค้าคลัง.....33

3.2 ความต้องการ ของการใช้งาระบบ.....34

3.3 การวิเคราะห์และออกแบบดาต้ามาร์ทในส่วนของ การบริหารสินค้าคงคลัง.....35

3.4 ความต้องการเชิงฟังก์ชัน (Functional Requirements).....39

3.5 ความต้องการเชิงประสิทธิภาพ (Non Functional Requirements).....47

3.6 การวิเคราะห์ และ ออกแบบสิ่งที่เราสนใน ด้วย Class Diagram.....47

3.7 การวิเคราะห์ และ ออกแบบความสัมพันธ์ระหว่างฟังก์ชันการทำงานของระบบโดย Sequence Diagram.....49

บทที่4 การพัฒนาระบบดาต้ามาร์ท สำหรับสินค้าคงคลัง กรณีศึกษา บมจ. ดาโปรแลคตรัมไทย จำกัด (มหาชน).....52

4.1 การเตรียมข้อมูลเข้าสู่ระบบ.....52

4.2 การแปลงข้อมูลเข้าสู่ดาต้ามาร์ท.....53

4.3 การกำหนดโครงสร้างของคิวบ์.....59

4.4 วิธีการใช้งานแอพพลิเคชัน OLAP.....59

4.5 ขั้นตอนการใช้งานระบบ ในส่วนของแอพพลิเคชัน.....66

บทที่5 การสรุปผลการศึกษา วิเคราะห์ ออกแบบ และ พัฒนาระบบดาต้ามาร์ตสำหรับสินค้าคงคลัง กรณีศึกษา บริษัทดาโปรแลคตรัมไทย จำกัด (มหาชน).....69

5.1 สรุปผลการพัฒนาระบบ.....69

5.2 อุปสรรค หรือ ปัญหาที่เกิดขึ้น.....70

5.3 ข้อเสนอแนะ.....70

บรรณานุกรม.....71

ประวัติผู้เขียน.....72


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญตาราง


ตารางที่ หน้า

1.1 ตารางแสดงขอบเขต และ แนวทางในการพัฒนาระบบ.....7

2.1 ตารางเปรียบเทียบข้อดี และ ข้อเสียของวิธีการอิมพลีเมนต์ดาต้าแวร์เฮาส์แบบที่ 1.....12

2.2 ตารางเปรียบเทียบข้อดี และ ข้อเสียของวิธีการอิมพลีเมนต์ดาต้าแวร์เฮาส์แบบที่ 2.....13

3.1 ตารางแสดง Meta Data : Item Dimension.....37

3.2 ตารางแสดงคำอธิบายข้อมูล Date Dimension.....37

3.3 ตารางแสดงคำอธิบายข้อมูล Site Dimension.....38

3.4 ตารางแสดงคำอธิบายข้อมูล Location Dimension.....38

3.5 ตารางแสดงคำอธิบายข้อมูล Location Dimension.....39

3.6 ตารางแสดง Use Case Description สำหรับ ฟังก์ชัน Login system.....40

3.7 ตารางแสดง Use Case Description สำหรับ ฟังก์ชัน Connect Datamart.....41

3.8 ตารางแสดง Use Case Description สำหรับ ฟังก์ชัน Implement cube.....42

3.9 ตารางแสดง ฟังกชัน Login System.....43

3.10 ตารางแสดง Use Case connect MSAS.....44

3.11 ตารางแสดง Use Case Description สำหรับฟังก์ชัน Query MDX.....45

3.12 ตารางแสดงฟังก์ชันการทำงาน Retrieve report.....46



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป


| รูปที่ | | หน้า |
|---|---|---|
| 2.1 | ภาพแสดงระดับต่างๆของ Business Intelligence | 10 |
| 2.2 | ภาพแสดงการ Implement BI ด้วยดาต้าแวร์เฮาส์โดยปราศจากดาตามาร์ต | 11 |
| 2.3 | ภาพแผนภาพแสดงการ Implement BI ด้วยดาต้ามาร์ทโดยปราศจากดาต้าแวร์เฮาส์ | 13 |
| 2.4 | ภาพแสดงการ Implement BI โดยมีทั้งดาต้าแวร์เฮาส์และดาต้ามาร์ท | 13 |
| 2.5 | ภาพแสดงสถาปัตยกรรมคลังข้อมูล | 17 |
| 2.6 | ภาพแสดงสถาปัตยกรรมของการนำเอา ข้อมูลจากดาต้ามาร์ทใช้ | 21 |
| 2.7 | ภาพแสดงมุมมองของการคิวรีข้อมูลจากคลังข้อมูลเรียกว่า คิวบ์ | 26 |
| 3.1 | ภาพแสดงแบบจำลองความสัมพันธ์ของข้อมูล บริษัท TCL | 36 |
| 3.2 | ภาพแสดงแบบจำลองความสัมพันธ์ของข้อมูล บริษัท TSL | 36 |
| 3.3 | ภาพแสดงแบบจำลองความสัมพันธ์ของข้อมูล บริษัท UNT | 37 |
| 3.4 | ภาพแสดง Use Case Diagram แสดงกระบวนการทำงานทำงานของระบบดาต้ามาร์ท | 40 |
| 3.5 | Class Diagram แสดงคลาสต่างๆ และความสัมพันธ์ระหว่างคลาส | 48 |
| 3.6 | ภาพแสดงความสัมพันธ์ฟังก์ชัน ETL ของ Item | 49 |
| 3.7 | ภาพแสดงความสัมพันธ์ฟังก์ชัน Connection Data source | 49 |
| 3.8 | ภาพแสดงความสัมพันธ์ฟังก์ชัน Implement Cube | 50 |
| 3.9 | ภาพแสดง Sequence Diagram Logon | 50 |
| 3.10 | ภาพแสดง Sequence Diagram Connection MSAS | 51 |
| 3.11 | ภาพแสดง Sequence Diagram Retrieve Report | 51 |
| 4.1 | ภาพแสดงส่วนติดต่อระหว่างผู้ใช้กับระบบ Microsoft Business Navision Solution | 52 |
| 4.2 | ภาพแสดงโครงร่างข้อมูลแบบดาว (Star Schema) | 54 |
| 4.3 | ภาพแสดงการออกแบบการแปลงข้อมูลจากแหล่งข้อมูลไปยังปลายทาง | 55 |
| 4.4 | ภาพการออกแบบการทำความสะอาดข้อมูลโดยแปลงรูปแบบข้อมูลจากต้นทางไปยังปลายทาง ของ Item Dimension | 55 |
| 4.5 | ภาพแสดงการออกแบบการทำความสะอาด แปลงรูปแบบข้อมูลจากต้นทางไปยังปลายทางของ SiteDimension | 56 |
| 4.6 | ภาพการออกแบบการทำความสะอาด แปลงรูปแบบข้อมูลจากต้นทางไปยังปลายทางของ Location Dimension | 56 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป (ต่อ)


รูปที่ หน้า

4.7 ภาพออกแบบการทำความสะอาด แปลงรูปแบบข้อมูลจากต้นทางไปยังปลายทาง ของ Date Dimension.....57

4.8 ภาพออกแบบการทำความสะอาด แปลงรูปแบบข้อมูลจากต้นทางไปยังปลายทางของ Data Mart Inventory Fact Dimension.....57

4.9 ภาพแสดงกระบวนการในการแปลงข้อมูลกรณีที่มีกระบวนการทำงานถูกต้อง 100 %.....58

4.10 ภาพแสดง SQL Server Business Intelligence Development แสดงการตรวจสอบความถูกต้องขอข้อมูล.....58

4.11 ภาพแสดงการติดต่อ Data Source.....60

4.12 ภาพแสดงการกำหนดสิทธิ์การเข้าใช้บริการเครื่องมือสำหรับการวิเคราะห์ข้อมูล.....60

4.13 ภาพแสดงการตั้งชื่อของ Data Source.....61

4.14 ภาพแสดงเค้าร่างข้อมูลแบบสตาร์ หลังจากติดต่อกับ Data Source สำเร็จ.....61

4.15 ภาพแสดงการเลือกแหล่งข้อมูลที่ต้องการติดต่อ.....62

4.16 ภาพแสดงเค้าร่างข้อมูลแบบสตาร์ หลังจากติดต่อกับ Data Source สำเร็จ.....62

4.17 ภาพแสดงรูปแบบของการสร้างคิวบ์.....63

4.18 ภาพแสดงรูปแบบของตารางเก็บข้อมูลระหว่าง Fact Table และ Dimension.....64

4.19 ภาพแสดงการเลือก Dimension ที่ต้องการใช้งาน.....64

4.20 ภาพแสดง Measure ต่างที่ต้องการใช้งาน.....65

4.21 ภาพแสดงหลังจากการติดตั้งค่าต่าง ๆ ให้กับคิวบ์.....65

4.22 ภาพแสดงหน้าต่างการติดตั้งพื้นที่ สำหรับการคำนวณแบบ Aggregation.....66

4.23 ภาพแสดงหน้าจอสำหรับผู้ใช้ใส่ชื่อ และ รหัสผ่านเพื่อเข้าใช้งานระบบ.....66

4.24 ภาพแสดงหน้าจอสำหรับผู้ใช้ใส่ชื่อ และ รหัสผ่านเพื่อเข้าใช้งานระบบ.....67

4.25 ภาพแสดงหน้าจอเมื่อผู้ใช้เรียกดูรายงานที่แสดงผลเป็นตัวเลข.....68

4.26 ภาพแสดงหน้าจอเมื่อผู้ใช้เรียกดูรายงานที่แสดงผลเป็นกราฟ.....67


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1
# บทนำ

## 1.1 ความเป็นมาและความสำคัญของปัญหา

ในยุคปัจจุบันที่เทคโนโลยีมีการเปลี่ยนแปลงอย่างรวดเร็ว และตลอดเวลา ยังส่งผลให้ธุรกิจหลายประเภท มีการแข่งขันกันค่อนข้างรุนแรง และมากขึ้นตามลำดับ จึงเป็นสิ่งที่เราไม่สามารถหลีกเลี่ยงได้ การที่องค์กรจะอยู่รอดได้นั้น ยังคงต้องการวิเคราะห์ วางแผน และตัดสินใจได้อย่างถูกต้อง รวดเร็ว เพื่อช่วยให้ธุรกิจสามารถดำเนินไปได้ ดังนั้นข้อมูลจึงนับได้ว่าเป็นปัจจัยที่สำคัญยิ่งต่อการดำเนินการนั้นๆ การนำข้อมูลมาเป็นเครื่องมือเพื่อช่วยสนับสนุนการตัดสินใจเพื่อให้ทันต่อยุคสมัยของการเปลี่ยนแปลง หรือได้ตอบปัญหาเชิงธุรกิจได้ทันต่อเหตุการณ์ ให้กับผู้บริหารระดับสูงขององค์กร การที่จะได้มาซึ่งข้อมูลเหล่านั้นจำเป็นต้องมีการแสวงหาหนทางในการเก็บรวบรวมข้อมูลทั้งในอดีตและปัจจุบันให้ได้มาก เพราะข้อมูลเหล่านั้นมิใช่เป็นเพียงข้อมูลภายในองค์กรเท่านั้น อาจเป็นข้อมูลจากภายนอกองค์กร ซึ่งเป็นคู่แข่งหรือเป็นข้อมูลขององค์กรอื่นๆ ที่อยู่ในธุรกิจประเภทเดียวกันกับเราก็เป็นไปได้ หรือแม้แต่การเลือกสรรข้อมูลที่มีคุณค่าจากกองข้อมูลที่มีขนาดมหึมาเพื่อให้แน่ใจว่าระบบข้อมูลสารสนเทศที่พัฒนาขึ้นมานั้นเป็นข้อมูลที่สามารถตอบสนองต่อความต้องการของผู้บริหารระดับสูงขององค์กรได้อย่างแท้จริงกลยุทธ์ที่เรานำมาสร้างกระบวนการวิเคราะห์ เพื่อให้ได้ข้อมูลสรุปที่สามารถแสดงให้เห็นภาพรวมของธุรกิจทั้งหมด หรือสามารถตอบคำถามระบบธุรกิจได้อย่างครอบคลุม จนสามารถนำมาช่วยในการตัดสินใจให้เป็นไปในทิศทางที่ถูกต้องและรวดเร็ว เรียกว่า ระบบธุรกิจอัจฉริยะหรือ Business Intelligence(BI)

ปัจจุบันองค์กรต่างๆ ได้พยามยามพัฒนาระบบ BI ขึ้นมา โดยระบบดังกล่าวได้มีการจัดเก็บข้อมูลไว้เป็นจำนวนมาก แต่ทว่าขาดการจัดเรียงข้อมูลอย่างเป็นระบบการเข้าถึง และการค้นคืนข้อมูลมีความยุ่งยากและการใช้ข้อมูลผ่านซับซ้อนมากขึ้นส่งผลให้ธุรกิจอาจต้องเสียค่าใช้จ่ายจำนวนมหาศาลในการเก็บรักษาข้อมูลเหล่านั้นไว้โดยไม่จำเป็น เพราะไม่ได้รับประโยชน์จากข้อมูลที่มี นอกจากนี้หากมีการนำข้อมูลมาวิเคราะห์อย่างผิดพลาดอาจจะก่อให้เกิดผลเสียหายต่อกระบวนการทำงานภายในองค์กรได้ ซึ่งเป็นการสูญเสียโอกาสทางธุรกิจไป เพราะฉะนั้นในยุคที่ผู้บริหารมีความต้องการใช้ข้อมูลเป็นเครื่องมือ เพื่อการตัดสินใจการจัดระบบระเบียบข้อมูล เพื่อนำเสนอข้อมูลที่มีคุณค่าและผ่านการกลั่นกรองแล้วแก่ผู้บริหารเพื่อใช้ในการตัดสินใจให้ทันต่อเหตุการณ์ จึงเป็นสิ่งที่จำเป็นอย่างยิ่ง แนวความคิดของการพัฒนาระบบ BI จึงเกิดขึ้น องค์กรมีความจำเป็นต้องจัดการวางแผนการใช้งานข้อมูล(Data Planning and Modeling) โดยอาศัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เทคโนโลยีเป็นเครื่องมือ ประกอบด้วย ระบบข้อมูล และโปรแกรมแอพพลิเคชั่นด้านการวิเคราะห์ สามารถแจกแจงได้ดังนี้

1. ดาต้าแวร์เฮ้าส์(Data Warehouse) คือคลังของข้อมูลที่ได้รับการออกแบบมา เพื่อทำการจัดเก็บข้อมูลจำนวนมากๆ ซึ่งเป็นข้อมูลในอดีตซึ่งเป็นข้อมูลที่ถูกอ้างอิงถึงหรือเก็บรวบรวมข้อมูลสำคัญและจำเป็นจากแหล่งต่างๆ ซึ่งเป็นประโยชน์ต่อการตัดสินใจของผู้บริหาร

2. ดาต้ามาร์ท(Date Mart) คือ คลังข้อมูลขนาดเล็กที่ได้รับการออกแบบมาเพื่อวัตถุประสงค์ใดวัตถุประสงค์หนึ่ง โดยพัฒนาสำหรับฝ่ายงานเฉพาะ เพื่อให้ผู้บริหารสามารถเรียกใช้ข้อมูลเฉพาะที่ต้องการได้อย่างรวดเร็วและมีประสิทธิภาพมากขึ้น ข้อมูลเชิงบริหารนี้จะสามารถช่วยลดปัญหาที่เกิดจากการใช้ข้อมูลจากฐานข้อมูลปฏิบัติการ(Operational database) ซึ่งเป็นการเก็บข้อมูลในรูปแบบ Transaction system ได้ ซึ่งโดยทั่วไปปัญหาที่พบเมื่อต้องการข้อมูลที่ช่วยในการตัดสินใจ

3. เครื่องมือที่ใช้ในการวิเคราะห์ข้อมูลหลายมิติ(OLAP) เป็นเครื่องมือใช้ในการวิเคราะห์ข้อมูลในมุมมองหลายมิติ(Multi Dimensional) ผู้ใช้สามารถพิจารณาข้อมูลระดับรายละเอียด(Drill Down) ข้อมูลตามโครงสร้างของปัจจัย(Dimension) และยังสามารถที่จะทำการปรับเปลี่ยนมุมมองได้ตามต้องการ ทั้งนี้ทั้งนั้นการตอบสนองต่อการคิวรีใช้เวลาที่รวดเร็ว การทำงานไม่ขึ้นกับขนาดและ ความซับซ้อนของฐาน

4. ระบบออกรายงาน คือ ระบบที่เก็บรวบรวมข้อมูล ค้นข้อมูลเพื่อนำมาสร้างรายงานที่ต้องการโดยนำเสนอแก่ผู้บริหารในรูปแบบต่างๆ เช่น รูปภาพภาพกราฟฟิก

ด้วยเหตุผลที่ว่าการแข่งขันในธุรกิจทวีความรุนแรงเพิ่มขึ้น องค์กรต่างๆ ต้องการมีผลกำไรมากขึ้น แสวงหาบริหารจัดการที่ดี และมีประสิทธิภาพ ลดปัญหาเรื่องค่าใช้จ่าย เพื่อก่อให้เกิดต้นทุนต่ำ หลายองค์กรมีความพยายามอย่างมากที่จะนำระบบ BI เข้ามาใช้ เพื่อให้ทันต่อธุรกิจที่ผันแปร ดังนั้น ระบบดาต้ามาร์ทสำหรับสินค้าคงคลังกรณีศึกษาบริษัทดาโปรแลคตรัมไทย จำกัด(มหาชน) หรือ Data Mart System จึงได้ศึกษาความเป็นไปได้ในการพัฒนา และนำมาเป็นโครงการพัฒนาระบบงานแขนงวิชาวิทยาการสารสนเทศ โดยการนำเอาวิธีการพัฒนาระบบของธุรกิจอัจฉริยะมาใช้เป็นแนวทางการสร้างระบบงานในส่วนงานของการจัดการสินค้าคงคลัง สำหรับธุรกิจปิโตรเคมี บริษัทดาโปรแลคตรัมไทย จำกัด(มหาชน)

## 1.2 วัตถุประสงค์ของการศึกษาและการพัฒนาระบบ

โครงการพัฒนาระบบงานแขนงวิชา วิทยาการสารสนเทศชิ้นนี้ เป็นโครงงานเชิงประยุกต์ ซึ่งมีวัตถุประสงค์หลักคือ

1. เพื่อศึกษาทิศทางและแนวโน้มของการนำเทคโนโลยี Business Intelligence มาประยุกต์ใช้ กับองค์กรธุรกิจยุคใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เพื่อศึกษาความต้องการของระบบงานเชิงธุรกิจ วิเคราะห์ ออกแบบ ตลอดจนพัฒนาต้นแบบ ตามกระบวนการที่ใช้ในการวิเคราะห์ข้อมูลทางธุรกิจเพื่อการตัดสินใจ

3. เพื่อพัฒนาศักยภาพด้านการวิเคราะห์การค้นหา Solution ที่เหมาะสมโดยอาศัยเทคโนโลยี ถือเป็นเครื่องมือ ประกอบไปด้วย ระบบข้อมูลและโปรแกรมแอพพลิเคชั่นด้านวิเคราะห์

4. เพื่อศึกษาพื้นฐานการทำงานและวัฒนธรรมองค์กรของกรณีศึกษาที่ได้ทำการวิจัย

5. เพื่อพัฒนาระบบ Business Intelligence ให้ปรากฏเป็นรูปธรรม รวมทั้ง สามารถทำงานได้อย่างถูกต้อง และแสดงผลการวิเคราะห์ให้เกิดประโยชน์ในทางธุรกิจมากที่สุด

6. เพื่อสร้างสร้างทักษะในการปฏิบัติงานจริง รวมทั้งแก้ปัญหาเฉพาะหน้าต่างๆ ภายใต้ Business rule ขององค์กรนั้นๆ

## 1.3 สมมุติฐานของการศึกษา

หลังจากเทคโนโลยีสารสนเทศเริ่มมีบทบาทสำคัญในการบริหารงานและจัดการในองค์กรต่างๆ ทั้งในภาครัฐและเอกชน ทุกหน่วยงานได้พัฒนาระบบสารสนเทศในหน่วยงานของตนให้สามารถจัดเก็บและประมวลผลข้อมูล ซึ่งช่วยเพิ่มประสิทธิภาพการทำงานในองค์กรให้ทันกับความต้องการของตลาด ด้วยเหตุนี้ในปัจจุบันฝ่ายต่างๆ ในองค์กรจึงมีการจัดเก็บข้อมูลไว้มากมายอยู่ในรูปแบบเอกสารหรือตารางต่างๆ ในฐานข้อมูล แต่เนื่องจากข้อมูลเหล่านี้อาจจะกระจัดกระจายอยู่ในแต่ละฝ่ายในองค์กร นับวันข้อมูลเหล่านี้ก็จะมีปริมาณเพิ่มขึ้นเรื่อยๆ โดยไม่ได้นำไปใช้ให้เกิดประโยชน์ในฝ่ายงานการบริหารสินค้าคงคลังของบริษัทดาโปรแลคตรัม จำกัด (มหาชน) ก็เช่นกัน มีการจัดเก็บและประมวลผลปริมาณสินค้าคงคลัง ในแต่ละวันหรือแต่ละเดือน เมื่อเวลาผ่านไปข้อมูลนี้ก็จะถูกเก็บไว้ต่างหากโดยไม่ได้นำมาใช้อีกทั้งๆ ที่ข้อมูลนี้อาจจะมีประโยชน์กับผู้บริหารในแง่ของการวางแผน เพื่อให้การดำเนินงานบรรลุวัตถุประสงค์ของการมีสินค้าไว้บริการลูกค้าในปริมาณที่เพียงพอทันต่อความต้องการของลูกค้า และสามารถลดระดับการลงทุนในสินค้าคลังให้ต่ำที่สุด นอกจากนี้อาจจะมีประโยชน์ในการกำหนดนโยบายหรือทิศทางการดำเนินงานของบริษัทอีกด้วย

การเพิ่มประสิทธิภาพของการมีอยู่ของข้อมูลซึ่งได้ค้กล่าวมาแล้วข้างต้นได้นำเอากระบวนการที่ใช้ในการวิเคราะห์ข้อมูลเชิงธุรกิจ(Business Intelligence) มาใช้เพื่อศึกษาความต้องการในการใช้ข้อมูล ตั้งแต่ระดับล่างสุด เพื่อนำมาจัดการ เรียบเรียง วิเคราะห์ ออกแบบและสกัดสารสนเทศจากแหล่งข้อมูลต่างๆ มาแปลงเป็นข้อมูลชนิดเดียวกัน(Extract Transform and Load) ให้อยู่ในรูปค้าต้า-มาร์ท สินค้าคงคลัง รวมทั้งวิเคราะห์หาคำตอบเชิงธุรกิจโดยอาศัยเทคโนโลยี OLAP เป็นเครื่องมือในการประมวลผลข้อมูล โดยการสร้างความสัมพันธ์ระหว่างมุมมองต่างๆ เพื่อศึกษาเหตุการณ์ปัจจุบัน และนำข้อมูลมาวิเคราะห์ทางสถิติ ค้นหาข้อมูล และทำรายงานเพื่อแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Statistical Analysis Querying and Reporting) ตลอดจนพัฒนา ระบบดาต้ามาร์ทสำหรับสินค้าคงคลัง กรณีศึกษาบริษัทดา-โปรแลคตรัมไทย จำกัด(มหาชน) ให้ปรากฏเป็นรูปธรรม โดยองค์กรสามารถนำระบบไปใช้งานได้อย่างมีประสิทธิผลต่อไป

## 1.4 ขั้นตอนการศึกษาและพัฒนา

### 1.4.1 การวิเคราะห์ธุรกิจขององค์กรและการแบ่งขอบเขตของเนื้อหาข้อมูล

1. ศึกษา Output ขององค์กรในส่วนของการจัดการสินค้าคงคลัง โดยการศึกษาจากรายงานข้อมูลที่ได้จากการทำรายการผ่านทางโปรแกรม Microsoft Business Solutions Navision ของบริษัท เพื่อให้ได้ข้อเท็จจริงต่างๆ ที่แสดงผลลัพธ์ข้อมูลตรงตามความต้องการและวัตถุประสงค์ของผู้ใช้งานมากที่สุด

2. ศึกษา Input ขององค์กรในส่วนของการจัดการสินค้าคงคลัง โดยการศึกษาแหล่งที่มาของข้อมูลจากการทำรายการผ่านทางโปรแกรม Microsoft Business Solutions-Navision ของบริษัท เพื่อสำรวจหา Input ของบริษัทว่ามีข้อมูลได้บ้าง

3. ศึกษาหาความสัมพันธ์ระหว่างกระบวนการทำงานของการจัดการสินค้าคงคลัง เพื่อศึกษา Workflow และสิ่งที่อาจจะมีผลต่อระบบ

### 1.4.2 การศึกษาโครงสร้างพื้นฐานในระดับองค์กร

1. ศึกษาโครงสร้างพื้นฐานเชิงเทคนิค (Technical infrastructure) โดยการศึกษาฮาร์ดแวร์ ซอฟท์แวร์ ระบบการจัดการฐานข้อมูล ระบบปฏิบัติการระบบเครือข่ายของบริษัท

2. ศึกษาโครงสร้างพื้นฐานไม่ใช้เทคนิค (Nontechnical infrastructure) โดยการศึกษาข้อมูลพื้นฐาน วิธีการในการทำรายการแบบจำลองข้อมูลเชิงตรรกะ กระบวนการทดสอบระบบ นโยบายขององค์กร

### 1.4.3 การวิเคราะห์ระบบ

1. กำหนดความต้องการของผู้ใช้งาน โดยการสัมภาษณ์ รวมทั้งศึกษาพฤติกรรมการใช้ข้อมูล ของผู้ใช้งาน ซึ่งศึกษาจากการเกิดรายการข้อมูล จากการใช้งานโปรแกรม Microsoft Business Solutions-Navision

2. วิเคราะห์ข้อมูลโดยการศึกษาจากเอกสารศึกษาจากฐานข้อมูลของบริษัท โดยสกัดตารางที่เกี่ยวข้องกับกระบวนการทำงานของแผนกสินค้าคงคลัง โดยผ่านทางโปรแกรม Microsoft SQL Server วิเคราะห์ฟังก์ชันการทำงานของระบบ ในส่วนของการจัดการสินค้าคงคงคลัง โดยศึกษาผ่านทางโปรแกรม Microsoft Business Solutions-Navision เพื่อหาเครื่องมือและเทคโนโลยีที่เหมาะสมมาใช้สำหรับการพัฒนาระบบ

3. วิเคราะห์ข้อมูลที่เกี่ยวข้องกับข้อมูลต่างๆ (Meta data) ซึ่งอยู่ภายในระบบ เพื่อศึกษาความจำกัดความของข้อมูล เพื่อใช้เป็นข้อมูลสำหรับการดำเนินการต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. วิเคราะห์ฟังก์ชันการทำงาน ส่วนติดต่อผู้ใช้ วิธีการเชื่อมต่อระหว่างผู้ใช้กับข้อมูลของระบบคาต้ามาร์ท

**1.4.4 การออกแบบฐานข้อมูลสำหรับคาต้ามาร์ท**

1. กำหนดคาต้าของระบบงานการจัดการสินค้าคงคลังของบริษัท โดยออกแบบเค้าร่างข้อมูลโดยอาศัยแบบจำลองข้อมูล(Data Model) แบบ Star Schema

2. ออกแบบ Fact table ของคาต้ามาร์ทในส่วนงานของการจัดการสินค้าคงคลัง โดยการกำหนดเอนทิตีหลักและกระบวนการที่เกี่ยวกับเอนทิตีนั้นๆ เพื่อให้ทราบถึง dimension table

3. กำหนดแอตทริบิวต์ที่จำเป็นในแต่ละ dimension table เพื่ออธิบายรายละเอียดของ dimension table ร่วมกันในแต่ละ fact table ที่จำเป็นจะต้องมี

4. กำหนดแอตทริบิวต์ที่จำเป็นใน fact table โดยแอตทริบิวต์หลักใน fact table จะมาจาก primary key ในแต่ละ dimension table นอกจากนี้แล้ว รวมทั้งระบุแอตทริบิวต์ที่จำเป็นอื่นๆ ด้วย

5. ออกแบบค่าการคำนวณเบื้องต้นใน fact table โดยการจัดเก็บค่าที่ได้จากการคำนวณให้เป็นแอตทริบิวต์หนึ่งในfact table ถึงแม้ว่าจะสามารถหาค่าได้จากแอตทริบิวต์อื่นๆ ก็ตาม ทั้งนี้เพื่อให้การสอบถามมีประสิทธิภาพมากขึ้น สามารถทำงานด้วยความเร็วที่เพิ่มขึ้น เนื่องจากไม่ต้องคำนวณค่าใหม่ทั้งหมดถึงแม้ว่าจะเกิดความซ้ำซ้อนของข้อมูลในการจัดเก็บบ้างก็ตาม

6. บันทึกคำอธิบายของ dimension table เพื่อให้ผู้ใช้สามารถใช้งานคาต้ามาร์ทส่วนการจัดการสินค้าคงคลัง ให้เกิดประสิทธิภาพ เพราะเกิดความเข้าใจต่อข้อมูลเป็นอย่างดี

7. ออกแบบด้านกายภาพ โดยการกำหนดคิวรีรูปแบบต่างๆ เพื่อให้ผู้ใช้เกิดความสะดวกในการใช้งานและสามารถทำงานกับระบบได้อย่างมีประสิทธิภาพ

8. ออกแบบวิธีการและรูปแบบของการนำข้อมูลจากแหล่งข้อมูลต่างๆ ให้อยู่ในรูปของแพลตฟอร์มของฐานข้อมูลที่ได้ออกแบบไว้ นั่นก็คือการแปลงข้อมูลหรือ Extraction Transformation and Loading (ETL)

9. ออกแบบส่วนติดต่อผู้ใช้งานระบบคาต้ามาร์ท(ส่วน Front - End)

**1.4.5 การสร้างระบบคาต้ามาร์ท**

1. สร้างคาต้ามาร์ทตามโครงสร้างของข้อมูลโดยยึดตามที่ได้ออกแบบไว้ใน Physical Data Model: Star Schema

2. แปลงข้อมูลเข้าสู่คาต้ามาร์ทส่วนสินค้าคงคลัง โดยเริ่มจากวิเคราะห์แหล่งข้อมูล เช่น ปริมาณของข้อมูล จำนวน และชนิดของการเข้าถึงแหล่งข้อมูล แพลตฟอร์มและภาษาโปรแกรมที่ใช้ จากนั้นย้ายข้อมูลที่ต้องการจากระบบเดิมมาไว้ในบริเวณที่ใช้ปรับแต่งข้อมูล หรือเรียกบริเวณนี้ว่า staging area เพื่อนำมาเลือกเฉพาะส่วนที่ต้องการแปลงข้อมูลและตรวจสอบความถูกต้อง หรือการทำความสะอาดข้อมูล โดยกำหนด primary key ของ fact table และ dimension table และกำหนด foreign key ระหว่าง fact table กับ dimension table ทำการย้ายข้อมูลที่ทำความสะอาดแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก staging area ลงสู่เซิร์ฟเวอร์ของดาต้ามาร์ท สร้าง metadata ของแต่ละดาต้ามาร์ท โดยเก็บรายละเอียดของข้อมูลการอัพเดทและส่งออกไปไว้ในดาต้ามาร์ท รวมทั้งตรวจสอบความถูกต้องของข้อมูล

3. พัฒนาแอพพลิเคชันในส่วนการวิเคราะห์ข้อมูลเชิงธุรกิจในส่วนสินค้าคงคลังของบริษัท โดยอาศัยเทคโนโลยี Online Analytical Processing System(OLAP) เป็นพื้นฐานของการวิเคราะห์ข้อมูล และใช้เทคโนโลยีเว็บไซด์เป็นส่วนติดต่อผู้ใช้งาน และการแสดงผล

## 1.5 ขอบเขตและแนวทางในการพัฒนาระบบ

ขอบเขตการพัฒนาระบบเริ่มจากการแปลงข้อมูล กำหนดการส่งข้อมูล รวบรวมหรือสร้างข้อมูลภายนอก วางแผนและสร้างรูทีนการแปลงข้อมูล ตรวจสอบความถูกต้องของข้อมูลก่อนนำเข้าสู่คลังข้อมูลเพื่อให้เป็นข้อมูลที่เหมาะสมตลอดจนอาศัยเทคโนโลยี OLAP เป็นแอพพลิเคชันพื้นฐานในการดึงข้อมูลที่เกี่ยวข้องกับสิ่งที่ผู้บริหารต้องการมาแสดงในรูปแบบของรายงาน กราฟ ตาราง และภาพกราฟิก ซึ่งเป็นการวิเคราะห์ข้อมูล ด้านสถิติ การหาความสัมพันธ์ของรายการสินค้าคงคลัง หรือการจำลองสถานการณ์ต่างๆ สามารถแสดงข้อมูลเป็นภาพรวมและเจาะลึกเฉพาะส่วนได้ เพื่อนำไปสู่การค้นหาสาเหตุของปัญหาที่ตรงจุดหาหนทางในการแก้ไขจัดการบริหารสินค้าคงคลังให้ตรงประเด็น และการวางแผนอนาคตได้ถูกทางสอดคล้องกับ KPI ขององค์กร ด้วยเหตุนี้การได้รับข้อมูลที่อัพเดท รวดเร็วและทันการณ์ จะช่วยให้ผู้บริหารสามารถพลิกสถานการณ์ให้เกิดความได้เปรียบในการแข่งขันได้ ทั้งในแง่ความพึงพอใจของลูกค้า ความเข้มแข็งด้านการขาย และระบบงานภายในที่มีประสิทธิภาพเทคโนโลยีที่เลือกใช้ในการพัฒนาระบบดังกล่าวสามารถแบ่งได้ 2 ส่วน คือ ส่วนของ Back-End และส่วนของ Front-End แสดงรายละเอียด ดังนี้คือ

**ตารางที่ 1.1** ตารางแสดงขอบเขตและแนวทางในการพัฒนา

| Service Part | เทนนิคที่ใช้ (technique) | กระบวนการ (Process) | แนวทางสำหรับการพัฒนา(Solution) | User |
|---|---|---|---|---|
| Backend | Architecture database | พัฒนาสถาปัตยกรรมของ Data Warehouse | - กำหนดสถาปัตยกรรมของ database มีโครงสร้างเป็น Relational Database<br>- กำหนด Dimensional Model ขึ้นโดยใช้โครงสร้างลักษณะ Star Schema | Administrator |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 1.1 (ต่อ)

| Service Part | เทนนิคที่ใช้ (technique) | กระบวนการ (Process) | แนวทางสำหรับการพัฒนา(Solution) | User |
|---|---|---|---|---|
| | The data maintenance application | SQL Server Analysis Service( SSAS) | Sql Server Business Intelligence Development | |
| | OLAP Cube | พัฒนา OLAP multidimensional database | - สร้าง Cube ตาม Dimensional Model ของ data warehouse<br>- การเข้าใช้ Relational Database โดยภาษา Structure Query Language (SQL)<br>- Application ในการสร้าง OLAP analysis ใช้ SQL Server Business Intelligence Development | |

## 1.6 ประโยชน์ที่คาดว่าจะได้รับ

สามารถแบ่งได้ 2 ส่วนดังนี้

### 1.6.1 สำหรับองค์กร

1. องค์กรใช้ประโยชน์จากระบบคลังข้อมูล ในแง่สนับสนุนการตัดสินใจของผู้บริหาร โดยผู้บริหารสามารถเรียกใช้ข้อมูลที่ต้องการได้อย่างรวดเร็วและมีประสิทธิภาพมากขึ้น

2. องค์กรสามารถลดปัญหาจากการใช้ข้อมูลฐานปฏิบัติการ (Operational Database) ซึ่งเป็นการเก็บข้อมูลในรูปของ Transaction System

3. ผู้ใช้งานระบบสามารถพิจารณาข้อมูลได้หลายมิติและสามารถหาคำตอบเชิงธุรกิจสนับสนุนการตัดสินใจได้อย่างรวดเร็ว และ เป็นปัจจุบัน

4. องค์กรสามารถค้นหาสาเหตุและแก้ไขปัญหาการจัดการบริหารสินค้าคงคลังได้อย่างตรงประเด็น และสามารถนำผลลัพธ์ที่ได้จากการหาคำตอบไปวางแผนการทำงานในอนาคตได้อย่างถูกทาง และสอดคล้องกับ Key Performance Indicator (KPI) ขององค์กร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**1.6.2 สำหรับนักศึกษา**

**1. ได้รับประสบการณ์ในการนำข้อมูลภายใต้สิ่งแวดล้อมจากการปฏิบัติงานเพิ่มพูนประสบการณ์ และศักยภาพในการพัฒนาระบบงานด้วยวิธีการพัฒนาระบบธุรกิจอัจฉริยะ(Business Intelligence)**

**2. เพิ่มทักษะในการวางแผนโครงการ และ การแก้ปัญหาเฉพาะหน้าได้**

**3. ได้รับความรู้ความเข้าใจเกี่ยวกับระบบงานธุรกิจ Logistic และระบบ Supply Chain เพิ่มมากขึ้น**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 2
# ทฤษฎีพื้นฐานที่ใช้ในศึกษา และ พัฒนาระบบ ดาต้ามาร์ท เพื่อการวิเคราะห์สินค้าคงคลัง

ในบทนี้จะกล่าวถึงทฤษฎีพื้นฐานต่างๆ ที่เกี่ยวข้องในการศึกษา ซึ่งเป็นพื้นฐานสำหรับการพัฒนาระบบ ดาต้ามาร์ท สำหรับสินค้าคงคลังซึ่งเนื้อหาในบทนี้จะกล่าวถึงกระบวนการที่ใช้ในการวิเคราะห์ข้อมูลทางธุรกิจเพื่อการตัดสินใจ เทคโนโลยีที่เกี่ยวข้องกับ BI แนวคิดเกี่ยวกับคลังข้อมูล สถาปัตยกรรมคลังข้อมูล การออกแบบ สร้างดาต้ามาร์ท การอิมพลีเมนต์ BI โดยอาศัยเครื่องมือการวิเคราะห์ข้อมูล อ้างอิงตามแนวคิดของเทคโนโลยี OLAP หลักการออกแบบ และ พัฒนาระบบ OLAPการประยุกต์ใช้ระบบOLAPในระดับองค์กร

## 2.1 กระบวนการที่ใช้ในการวิเคราะห์ข้อมูลทางธุรกิจเพื่อการตัดสินใจ Business Intelligence (BI)

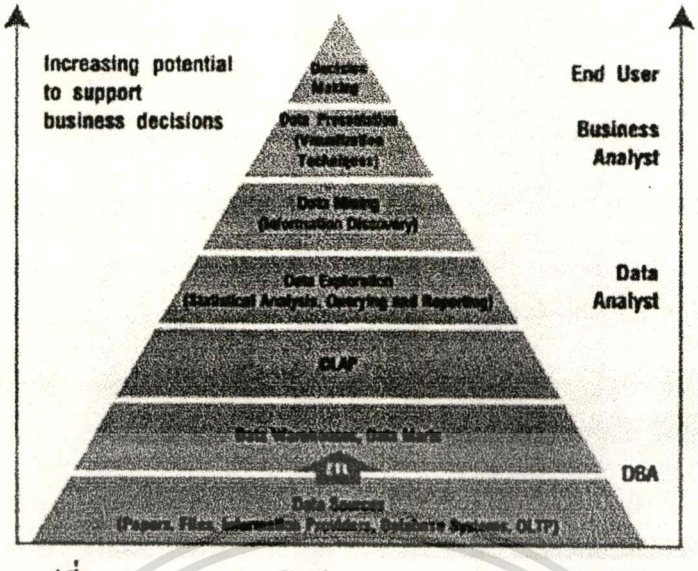
กระบวนการที่ใช้ในการวิเคราะห์ข้อมูลทางธุรกิจเพื่อการตัดสินใจ หรือที่เรียกกันย่อๆว่า BI (Business Intelligence) คือกระบวนการที่นำมาช่วยให้มนุษย์สามารถทำความเข้าใจกับข้อมูลแล้วนำไปใช้ในการตัดสินใจได้ดีขึ้น รวดเร็วมากขึ้น และบรรลุตามวัตถุประสงค์ของธุรกิจมากขึ้น ผลลัพธ์ที่สำคัญ ของระบบนี้ก็คือการเพิ่มประสิทธิภาพและประสิทธิผลของการดำเนินธุรกิจ เทคโนโลยีนี้บางครั้งก็มุ่งที่จะทำให้การไหลของข้อมูลมีความรวดเร็วและสามารถเข้าถึงได้ง่ายขึ้น เช่น การทำรูปแบบ email มาตรฐาน เพื่อลดการสูญเสียเวลา ในการสร้างเอกสารใหม่ การปรับปรุงเอกสารและการใช้เวลาในการกระจายและจัดส่งเอกสารเป็นต้น

องค์ประกอบที่สำคัญของการทำ BI ประกอบด้วยโครงสร้างข้อมูล 2 แบบ คือ ดาต้าแวร์เฮาส์ และ ดาต้ามาร์ท ส่วนประกอบที่เป็นอินเทอร์เฟซเพื่อใช้สนับสนุนการตัดสินใจของผู้ใช้ หรือผู้บริหาร รวมทั้งโพรเซสต่างๆ ที่ใช้ในการรวบรวมข้อมูลเข้าสู่ดาต้าแวร์เฮาส์ หรือ ดาต้ามาร์ท เพื่อใช้ในกระบวนการทางธุรกิจ สิ่งเหล่านี้ถือว่าเป็นลักษณะตามมาตรฐานของ Business Intelligence

### 2.1.1 เทคโนโลยีที่เกี่ยวข้องกับ BI

เทคโนโลยีที่เกี่ยวข้องกับ BI จะมี 3 เทคโนโลยีหลัก ได้แก่ เทคโนโลยีทางด้านดาต้าแวร์เฮาส์ (Data Warehouse) OLAP และดาต้าไมนิ่ง (Data Mining) ซึ่งเทคโนโลยีทั้งหมดที่เกี่ยวข้องกับ BI จะมีหลายระดับสามารถอธิบาย และ แสดงดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 2.1** ภาพแสดงระดับต่างๆของ Business Intelligence

เทคโนโลยีระดับล่างสุดที่เข้ามาสนับสนุน BI คือ แหล่งข้อมูลต่างๆ ได้แก่ ฐานข้อมูล ไฟล์ ฯลฯ ซึ่งให้ข้อมูลพื้นฐานทั่วไป เช่น ข้อมูลเกี่ยวกับลูกค้าแต่ละคนว่าเป็นอย่างไร ซื้อสินค้ามากน้อยแค่ไหนในช่วง 6 เดือนที่ผ่านมา เป็นต้น ระดับที่ถัดมา คือ Data Warehouse ซึ่งเก็บข้อมูลเพื่อใช้ในการวิเคราะห์ เพื่อช่วยให้การค้นหาข้อมูลทำได้รวดเร็วขึ้น สิ่งที่ต้องการสำหรับ Data Warehouse คือ อุปกรณ์จัดเก็บข้อมูลพวกสตอเรจ โดยจะใช้เครื่องมือที่เรียกว่า ETL (Extract, Transform and Load) เช่น Microsoft Data Transformation Services (DTS) เป็นต้น ในการดึงข้อมูลจากแหล่งข้อมูลต่างๆ มาแปลงให้เป็นข้อมูลชนิดเดียวกันทั้งหมด แล้วนำไปเก็บไว้ใน Data Warehouse ผลิตภัณฑ์ทางด้าน Data Warehouse ได้แก่ ผลิตภัณฑ์ทางด้านฐานข้อมูลต่างๆนั่นเอง เช่น Oracle Database ,IBM DB2 UDB ,Microsoft SQL Server เป็นต้น นอกจากนี้ยังมี Data Mart ซึ่งก็คือส่วนย่อยของ Data Warehouse นั่นเอง โดยสามารถแบ่ง Data Warehouse ออกเป็นหลายๆ Data Mart ได้ เช่น ส่วนของการเงิน ส่วนของสินค้าคงคลัง ส่วนของการขาย เป็นต้น ซึ่งทำให้การจัดการทำได้ง่ายขึ้นและการนำเอาข้อมูลไปสร้างความสัมพันธ์และวิเคราะห์ต่อก็ง่ายขึ้นด้วย

**2.1.2 เครื่องมือวิเคราะห์ข้อมูล**

เครื่องมือ (Tool) ที่นำมาใช้ในการวิเคราะห์ข้อมูลทางธุรกิจเพื่อการตัดสินใจหลายประเภท ได้แก่OLAP,Data Exploration และ Data Mining

1. OLAP (On-Line Analytical Processing) เป็นการประมวลผลข้อมูลเชิงวิเคราะห์โดยมีการสร้างความสัมพันธ์ ระหว่างข้อมูลในมิติ (Dimension) หรือมุมมองต่างๆ เช่น ความสัมพันธ์ระหว่างสินค้า สาขา และเวลา เป็นต้น เป็นการวิเคราะห์เหตุการณ์ในปัจจุบัน ผลิตภัณฑ์ทางด้าน OLAP ได้แก่ Oracle OLAP ,IBM DB2 OLAP Server ,Microsoft SQL Server Analysis Services

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

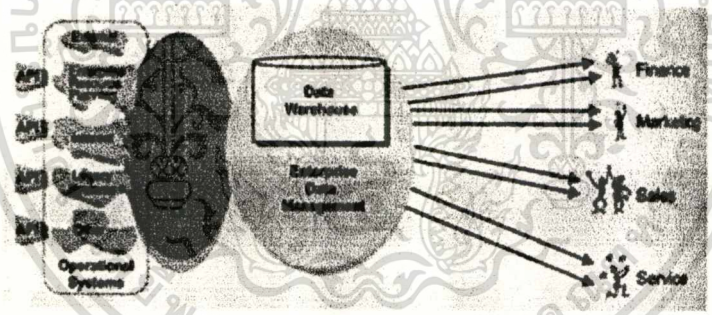
,Hyperion Essbase ,Cognos PowerPlay ,MicroStrategy Intelligent Cubes & MicroStrategy OLAP Services ,SAP Business Information Warehouse

2. Data Exploration เป็นเครื่องมือที่ช่วยนำข้อมูลมาวิเคราะห์ทางสถิติ ค้นหาข้อมูล และทำรายงานเพื่อแสดงผล (Statistical Analysis ,Querying and Reporting) เป็นการสำรวจทำความเข้าใจข้อมูล ผลิตภัณฑ์ทางด้าน Data Exploration ได้แก่ Brio Performance Suite ,Crystal Analysis ,Crystal Reports ,Cognos Impromptu ,MicroStrategy Intelligence Server ,Business Objects ,Excel

3. Data Mining กระบวนการในการดึงสารสนเทศ (Information) ออกจากฐานข้อมูลขนาดใหญ่เพื่อนำไปใช้ประโยชน์ในการตัดสินใจ Data Mining มีหลายโมเดล แต่ละโมเดลจะมีสูตรทางธุรกิจ (Business Formula) เข้ามาเกี่ยวข้องและให้ผลลัพธ์ในรูปแบบที่แตกต่างกัน เช่น เป็นกฎ IF-then หรือ เป็นแผนภูมิการตัดสินใจ (Decision Trees) เป็นต้น ซอฟต์แวร์ในระดับ Data Mining ได้แก่ SPSS Clementine ,SAS Enterprise Miner ,IBM Enterprise Miner ,IBM DB2 Intelligent Miner , Oracle Data Mining เป็นต้น

**2.1.3 การอิมพลีเมนต์ BI** การอิมพลีเมนต์ BI มีอยู่ 3 รูปแบบหลัก ได้แก่

1. การอิมพลีเมนต์เพียงแค่ดาต้าแวร์เฮาส์โดยปราศจากดาต้ามาร์ท การทำงาน ระบบจะไปเก็บข้อมูลจากหลายแหล่งข้อมูลและรวมเข้ากันไว้เป็นข้อมูลศูนย์กลางขององค์กร




**รูปที่ 2.2** ภาพแสดงการ Implement BI ด้วยดาต้าแวร์เฮาส์โดยปราศจากดาตามาร์ต

จากการศึกษา และ ค้นคว้า วิธีการอิมพลีเมนต์เพียงแค่ดาต้าแวร์เฮาส์ สามารถเปรียบเทียบข้อดีและข้อเสียได้ดังนี้

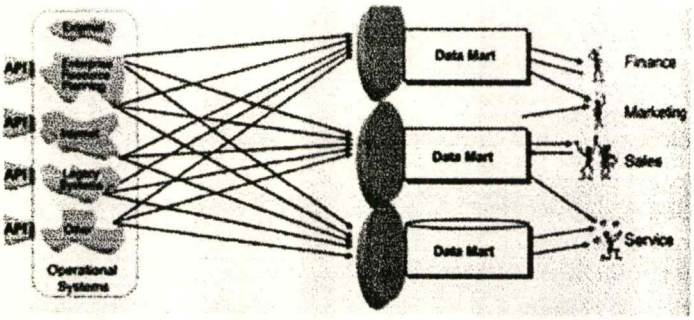
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 2.1** ตารางเปรียบเทียบข้อดี และ ข้อเสียของวิธีการอิมพลีเมนต์ดาต้าแวร์เฮาส์แบบที่ 1

| ข้อดี | ข้อเสีย |
|---|---|
| ▪ ทำให้กระบวนการรวบรวม หรือ Data Acquisition ทำได้ง่ายขึ้น เพราะการดึงข้อมูลจากแต่ละแหล่งข้อมูลนั้น กระทำเพียงแค่ครั้งเดียวเท่านั้น | ▪ เนื่องจากช่วยเรารวบรวมข้อมูลมาไว้ที่ศูนย์กลางเพียงแห่งเดียว ทำให้ข้อมูลที่ทุกคนจะนำไปใช้ เป็นข้อมูลที่มาจากแหล่งเดียวกันซึ่งจะเป็นการสร้างมาตรฐานให้กับตัวข้อมูลที่นำไปใช้ |
| เนื่องจากช่วยเรารวบรวมข้อมูลมาไว้ที่ศูนย์กลางเพียงแห่งเดียว ทำให้ข้อมูลที่ทุกคนจะนำไปใช้ เป็นข้อมูลที่มาจากแหล่งเดียวกัน ซึ่งจะเป็นการสร้างมาตรฐานให้กับตัวข้อมูลที่นำไปใช้ | หากข้อมูลมีปริมาณเพิ่มขึ้นเรื่อยๆ จะทำให้ฐานข้อมูลมีการขยายตัวอย่างรวดเร็ว และมีความซับซ้อนเพิ่มขึ้นเรื่อยๆ ซึ่งจะส่งผลกระทบถึงประสิทธิภาพของระบบและเวลาที่ใช้ตอบสนองต่อผู้ใช้โดยตรง และส่งผลถึงการบำรุงรักษาและการพัฒนาระบบในอนาคตทำได้ยาก เพราะการเปลี่ยนแปลงอะไรก็ตามจะมีผลต่อกระบวนการ Data Acquisition และการเข้าถึงข้อมูลส่วนกลางในดาต้าแวร์เฮาส์ |
| ▪ ช่วยประหยัดค่าใช้จ่ายในส่วนของทรัพยากร เพราะใช้ฐานข้อมูลที่เป็นศูนย์กลางเพียงแค่ที่เดียว ทำให้ไม่ต้องมีหลายฐานข้อมูล ซึ่งจะเป็นการเปลืองทั้งทรัพยากรและค่าใช้จ่าย | ▪ มีผลกระบต่อผู้ใช้ เพราะการเรียกดูข้อมูลที่มาจากกระบวนการ Data Acquisition นั้น การสืบค้นข้อมูลแต่ละครั้งต้องกระทำผ่านทางดาต้าแวร์เฮาส์ ซึ่งค่องข้างมีวิธีการที่ซับซ้อนและยากต่อการใช้งานของผู้ใช้ |

2. การอิมพลีเมนต์เพี่ยงแค่ดาต้ามาร์ทโดยปราศจากดาต้าแวร์เฮาส์ เป็นการอิมพลีเมนต์โดยไม่มีฐานข้อมูลส่วนกลางสำหรับจัดเก็บข้อมูล และไม่ได้รับการสนับสนุนจากดาต้าแวร์เฮาส์ แต่ใช้ดาต้ามาร์ทสำหรับจัดเก็บข้อมูล ซึ่งทำให้ได้ข้อมูลที่จำเพาะของแต่ละชนิดข้อมูลหลายๆ ชนิดข้อมูล สามารถแสดงข้อดี และ ข้อเสียได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

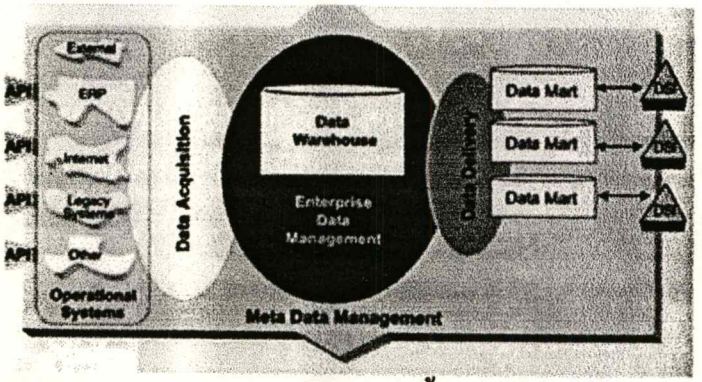



**รูปที่ 2.3** ภาพแผนภาพแสดงการ Implement BI ด้วยดาต้ามาร์ทโดยปราศจากดาต้าแวร์เฮาส์

**ตารางที่ 2.2** ตารางเปรียบเทียบข้อดี และ ข้อเสียของวิธีการอิมพลีเมนต์ดาต้าแวร์เฮาส์แบบที่ 2

| ข้อดี | ข้อเสีย |
|---|---|
| ▪ ดาต้ามาร์ทแต่ละตัวสามารถปรับตัวและรองรับกับกลุ่มผู้ใช้หลายกลุ่มและการสืบค้นข้อมูลในหลายรูปแบบได้ | ▪ เนื่องจากดาต้ามารต์จะถูกสร้างโดยแยกจากกันไม่เกี่ยวข้องกัน ดังนั้นผลลัพธ์ที่ได้จากการพัฒนาก็คือ ข้อมูลในแต่ละดาต้ามาร์ทจะไม่สอดคล้องกันและไม่เป็นมาตรฐานเดียวกัน |
| ▪ มีความรวดเร็วในการตอบสนองกับผู้ใช้ และง่ายต่อการบำรุงรักษาและการปรับเปลี่ยนในอนาคต | |

3. การอิมพลีเมนต์โดยมีทั้งดาต้าแวร์เฮาส์และดาต้ามาร์ท เป็นการอิมพลีเมนต์โดยการนำข้อดีของทั้งสองวิธีข้างต้นมารวมกัน โดยใช้ดาต้าแวร์เฮาส์เพื่อช่วยให้การทำกระบวนการ Data Acquisition ทำได้ง่ายและสะดวกขึ้น และเป็นที่เก็บข้อมูลส่วนกลาง พร้อมกันนี้ก็ใช้ดาต้ามาร์ทในการส่งข้อมูลแต่ละส่วนไปยังผู้ใช้ตามความเหมาะสม สามารถแสดงข้อดี และ ข้อเสียได้ดังนี้




**รูปที่ 2.4** ภาพแสดงการ Implement BI โดยมีทั้งดาต้าแวร์เฮาส์และดาต้ามาร์ท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทั้งนี้การศึกษา ค้นคว้า และ ทำการพัฒนาระบบระบบ ดาต้ามาร์ท เพื่อการวิเคราะห์สินค้าคงคลังของบริษัท คาโปรแลคตรัมไทย จำกัด (มหาชน) ได้เลือกวิธีการอิมพลีเมนต์เพียงดาต้ามาร์ทเท่านั้น ซึ่งสามารถอธิบายแนวคิด และ หลักการต่างๆ ได้ในหัวข้อถัดไป

## 2.2 แนวคิดเกี่ยวกับคลังข้อมูล และ ดาต้ามาร์ท

### 2.2.1 นิยามของคลังข้อมูล

คลังข้อมูล หมายถึง ฐานข้อมูลขนาดใหญ่ขององค์กรหรือหน่วยงานหนึ่งๆ ซึ่งเก็บรวบรวมข้อมูลจากฐานข้อมูลระบบงานประจำวัน หรือเรียกอีกอย่างว่า Operational database และฐานข้อมูลอื่นภายนอกองค์กร หรือเรียกว่า External database โดยข้อมูลที่ถูกจัดเก็บในคลังข้อมูลนั้น มีวัตถุประสงค์ในการนำมาใช้งาน และมีลักษณะของการจัดเก็บแตกต่างไปจากข้อมูลในฐานข้อมูลระบบงานอื่น โดยข้อมูลในคลังข้อมูลจะถูกนำมาใช้เพื่อสนับสนุนการตัดสินใจบริหารงานของผู้บริหาร โดยเฉพาะการเป็นข้อมูลพื้นฐานให้กับระบบงาน เพื่อการบริหารงานอื่น เช่น ระบบ DSS และระบบ CRM เป็นต้น

### 2.2.2 นิยามของดาต้ามาร์ท

ดาต้ามาร์ท คือ ที่รวมของข้อมูลจากข้อมูลปฏิบัติงานและแหล่งข้อมูลอื่น ๆ ที่ได้รับการออกแบบให้รองรับการใช้งานเฉพาะของผู้ใช้ที่มีระดับความรู้ในแนวคิด ข้อมูลที่มีจากฐานข้อมูลของธุรกิจ หรือ ดาต้าแวร์เฮาส์ หรือ เป็นข้อมูลเฉพาะเจาะจง สิ่งที่เกิดขึ้น Data mart ได้รับการใช้โดยความต้องการของกลุ่มผู้ใช้เฉพาะที่มีความรู้ในการวิเคราะห์ เก็บรายละเอียด นำเสนอ และใช้งานได้ง่าย ซึ่งผู้ใช้ดาต้ามาร์ท สามารถคาดหวังว่าข้อมูลที่จะนำเสนอจะอยู่ในรูปที่คุ้นเคย

ในทางปฏิบัติ คำว่าดาต้ามาร์ท และ ดาต้าแวร์เฮาส์ เป็นการแสดงนัยยะของข้อมูลในบางรูปแบบ อย่างไรก็ตามผู้เขียนส่วนใกใช้คำที่ดูเหมือนการออกแบบดาต้ามาร์ท ซึ่งมีแนวโน้มที่จะเริ่มจากการวิเคราะห์ตามความต้องการของผู้ใช้และ ดาต้าแวร์เฮาส์ เริ่มจากการวิเคราะห์ข้อมูลที่มีอยู่จริง และ การรวบรวมข้อมูลในทางที่ข้อมูลสามารถทำไปใช้ต่อดาต้าแวร์เฮาส์เป็นการรวมข้อมูลที่ส่วนกลาง (ซึ่งสามารกระจายทางภายภาพ) ดาต้ามาร์ทเป็นข้อมูลรวมที่ได้จาก ดาต้าแวร์เฮาส์ และทำให้ง่ายในการเข้าถึง และ การประยุกต์สำหรับการออกแบบตามวัตถุประสงค์เฉพาะ โดยทั่วไป ดาต้าแวร์เฮาส์ มีแนวโน้มในการใช้เชิงกลยุทธ์มากกว่า

### 2.2.3 ความสัมพันธ์ระบบคลังข้อมูล ระบบดาต้ามาร์ท กับระบบฐานข้อมูล

ในปัจจุบันมีการใช้ฐานข้อมูลอย่างกว้างขวางในระบบงานทั่วไป จึงมีการวิจัยและพัฒนาวิธีเก็บข้อมูลจำนวนมาก รวมถึงการค้นหาและนำข้อมูลที่ต้องการออกมาจากระบบฐานข้อมูลด้วย แต่เนื่องจากระบบฐานข้อมูลทั่วไป (Operational Database) ที่นิยมใช้อยู่ในปัจจุบันมีหลักในการเก็บข้อมูลที่เน้นในเรื่องการลดความซ้ำซ้อน (redundancy) รักษาความถูกต้อง (integrity) ลดการสูญ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หายของข้อมูล (information lost) และลดความผิดพลาดที่เกิดขึ้นจากการแก้ไขข้อมูล (Update Anomalies)

เนื่องจากฐานข้อมูลทั่วไป (Operational Database) มีลักษณะดังได้กล่าวมาแล้วจึงมีความสามารถเพียงแค่การเรียกใช้ข้อมูลที่มีอยู่ แต่ไม่สามารถจะนำมาช่วยในการสนับสนุนการตัดสินใจได้ เพราะเมื่อมีการเรียกใช้ข้อมูลจะต้องเรียกใช้ข้อมูลจากฐานข้อมูลขนาดใหญ่ ซึ่งมีข้อมูลจำนวนมหาศาลและมีการแตกตารางที่นอร์มัลไลซ์ (normalized table) แล้วออกเป็นหลายตาราง จึงไม่รองรับคำถามที่ต้องการจะนำมาใช้ช่วยในการสนับสนุนการตัดสินใจ (decision support queries) มีการรวม (join) กันของตารางต่างๆที่ซับซ้อน ซึ่งจะทำให้มีประสิทธิภาพของการค้นหาข้อมูลจากฐานข้อมูลน้อยลง และทำงานช้าลง ไม่สามารถเรียกใช้ข้อมูลที่ต้องการได้ทั้งหมดเพราะมีรูทีนอัตโนมัติ (Automate Routine) จึงมีความสามารถในการค้นหาข้อมูลแบบที่ไม่ซ้ำซ้อนเท่านั้น นอกจากนี้การเก็บข้อมูลในระบบฐานข้อมูลทั่วไป (Operational Database) ยังไม่มีการเก็บข้อมูลย้อนหลัง (historical data) เพื่อใช้ช่วยในการคาดคะเนแนวโน้มที่คาดว่าจะเป็นไปได้ในอนาคต โดยระบบฐานข้อมูลคลังข้อมูลจะแยกกลุ่มข้อมูลสารสนเทศที่ใช้ในการวิเคราะห์ทางธุรกิจออกจากฐานข้อมูลที่ใช้ประจำวัน (Operational Database) มาเก็บอยู่ในระบบจัดการฐานข้อมูล (Relational Database Management Systems) ประสิทธิภาพสูงสุด และทำให้การเรียกใช้ข้อมูลชุดนี้ทำได้อย่างยืดหยุ่น จากเครื่องมือที่อยู่บนเครื่องคอมพิวเตอร์เดสก์ทอปทั่วไป โดยลด off-loading เพิ่มกลไกการช่วยตัดสินใจ ปรับปรุงเวลาที่ตอบสนอง (response time) รวดเร็วขึ้นอย่างมากและผู้บริหารสามารถเรียกข้อมูลรายละเอียดที่จำเป็นที่ถูกเก็บมาก่อนหน้านี้ (historical data) มาใช้ช่วยในการตัดสินใจทางธุรกิจแม่นยำขึ้น

ความแตกต่างอีกประการหนึ่งก็คือผู้ใช้คลังข้อมูลมักจะต้องการจัดกลุ่มข้อมูลด้วยตนเองมากกว่าผู้ใช้ในระบบฐานข้อมูลธรรมดา ยกตัวอย่างผู้ใช้อาจต้องการวิเคราะห์ผลกระทบของการทำการตลาดแบบต่างๆ อาจต้องการจัดกลุ่มการขายสินค้าแยกตามผลิตภัณฑ์ หรือรูปแบบของการจัดผลิตภัณฑ์ เช่น การห่อรวมสินค้าไว้ในบรรจุภัณฑ์สีต่างๆ หรือการรวมผลิตภัณฑ์ต่างรูปแบบไว้ด้วยกัน ในกรณีต่างๆ เหล่านี้ผู้ใช้ต้องการที่จะเลือกจัดกลุ่มข้อมูลได้ตามใจชอบ นอกจากการนำข้อมูลเข้ามารวมกันแล้วผู้ใช้ยังอาจต้องการที่จะแยกแยะข้อมูลในแบบที่ตนเองต้องการได้ ยกตัวอย่างในการจัดทำคลังข้อมูลเกี่ยวกับนักวิจัยและผลงานวิจัยของประเทศ หน่วยงานอาจจัดเก็บข้อมูลเอาไว้เป็นกลุ่มก้อนโดยไม่ได้แยกสาขา แต่ต่อมาผู้ใช้อาจต้องการนำข้อมูลนักวิจัยมาวิเคราะห์แยกแยะว่าทั้งประเทศมีนักวิจัยสาขาต่างๆ เป็นจำนวนเท่าใด ทำงานวิจัยด้านใดบ้าง ใช้เงินด้านวิจัยไปเท่าใด เป็นต้น โดยปกติแล้วการจัดทำฐานข้อมูลให้สามารถวิเคราะห์แยกแยะข้อมูลในแบบนี้ได้นั้นเป็นเรื่องไม่ยาก แต่ในการออกแบบคลังข้อมูลนั้นจำเป็นต้องเผื่อให้ผู้ใช้หลายคนสามารถแยกแยะข้อมูลตามความต้องการที่แตกต่างกันได้ด้วย ผู้ใช้จำนวนมากในปัจจุบันนี้อาจใช้ซอฟต์แวร์หลากหลายประเภทสำหรับเครื่องคอมพิวเตอร์ส่วนบุคคล ผู้ใช้บางคนอาจจะใช้โปรแกรมสเปรดชีต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการวิเคราะห์ข้อมูล และผู้ใช้บางคนอาจต้องการใช้โปรแกรมวิเคราะห์สถิติอื่นๆ ดังนั้นผู้ใช้เหล่านี้อาจจะมีความต้องการในการนำเข้าข้อมูลจากคลังข้อมูลมาไว้ในแฟ้มข้อมูลที่มีรูปแบบตรงกับโปรแกรมที่ตนต้องการใช้ ความต้องการด้านนี้นับว่าสำคัญมากที่สุดในการจัดทำคลังข้อมูล

งานอย่างหนึ่งที่นิยมใช้ฐานข้อมูลกันมากก็คืองานบันทึกข้อมูลธุรกรรมเอาไว้เพื่อประมวลผล ข้อมูลธุรกรรมเหล่านี้ได้แก่ ข้อมูลการสั่งซื้อสินค้าของลูกค้า ข้อมูลการซื้อบัตรโดยสารเครื่องบิน ข้อมูลการฝากหรือถอนเงินของลูกค้าธนาคาร แต่เดิมนั้นการบันทึกข้อมูลธุรกรรมเริ่มต้นด้วยการใช้กระดาษแบบฟอร์มสำหรับให้ลูกค้ากรอกข้อมูล จากนั้นจึงนำแบบฟอร์มมาบันทึกข้อมูลลงในฐานข้อมูลของระบบคอมพิวเตอร์ในแบบแบตช์ (batch) ปัจจุบันนี้การบันทึกข้อมูลธุรกรรมได้เปลี่ยนไปเป็นระบบออนไลน์ (online) เป็นส่วนใหญ่ ในระบบแบบนี้กระบวนการบันทึกข้อมูลมีลักษณะอัตโนมัติมากขึ้นและใช้อุปกรณ์บันทึกข้อมูลที่สามารถเก็บข้อมูลลงในฐานข้อมูลของระบบคอมพิวเตอร์ได้ทันที เช่น การใช้อุปกรณ์ฝากถอนเงินโดยอัตโนมัติ (ATM) ทำให้สามารถประมวลผลการฝากถอนเงินและบันทึกข้อมูลที่เกิดขึ้นได้ทันที หรือในห้างสรรพสินค้าก็มีการใช้เครื่องบริการ ณ จุดขาย (Point of Sale; POS) สำหรับอ่านรหัสแท่ง แสดงราคาสินค้า แล้วบันทึกข้อมูลการขายไปเก็บไว้ในฐานข้อมูลได้ทันที การดำเนินการในลักษณะนี้เรียกกันว่า การประมวลผลธุรกรรมออนไลน์ (On-Line Transaction Processing; OLTP)

**2.2.4 คุณลักษณะของระบบคลังข้อมูล - ระบบค้าต้ามาร์ต**

จากนิยามของคลังข้อมูลที่บอกถึงความแตกต่างกันระหว่างคลังข้อมูลกับฐานข้อมูลปฏิบัติการ ซึ่งสามารถสรุปคุณลักษณะของคลังข้อมูล ได้ดังนี้

1. การแบ่งโครงสร้างตามเนื้อหา (Subject Oriented) หมายถึง คลังข้อมูลถูกออกแบบมาเพื่อมุ่งเน้นไปในแต่ละเนื้อหาที่สนใจโดยที่ ข้อมูลจะต้องถูกสร้างขึ้นจากหัวข้อ (subject) ธุรกิจที่สนใจ ไม่ได้เน้นไปที่การทำงานหรือกระบวนการแต่ละอย่างโดยเฉพาะเหมือนอย่างฐานข้อมูลปฏิบัติการในส่วนของรายละเอียดข้อมูลที่จะจัดเก็บในระบบทั้งสองแบบก็จะแตกต่างกันไปตามความต้องการใช้งานด้วยเช่นกันคลังข้อมูลจะไม่จัดเก็บข้อมูลที่ไม่มีส่วนเกี่ยวข้องกับการประมวลผลเพื่อสนับสนุนการตัดสินใจ ในขณะที่ข้อมูลนั้นจะถูกเก็บไว้ในฐานข้อมูลปฏิบัติการหากมีส่วนเกี่ยวข้องกับกระบวนการทำงาน Integration หรือการรวมเป็นหนึ่งเดียว ซึ่งถือได้ว่าเป็นคุณลักษณะที่สำคัญที่สุดของคลังข้อมูล คือ การรวบรวมข้อมูลจากหลายฐานข้อมูลปฏิบัติการเข้าด้วยกัน และทำให้ข้อมูลมีมาตรฐานเดียวกัน เช่นกำหนดให้มีค่าตัวแปรของข้อมูลในเนื้อหาเดียวกันให้เป็นแบบเดียวกันทั้งหมด โดยที่ ค่าของตัวแปรตัวเดียวในแต่ละฐานข้อมูลอาจต่างกัน ฐานข้อมูลหนึ่งอาจใช้ 0 และ 1 อีกฐานข้อมูลหนึ่งอาจใช้ T และ F ดังนั้น ฐานข้อมูลที่สร้างใหม่จะต้องได้รับการกำหนดค่าตัวแปรให้เหมือนกันเป็นหนึ่งเดียว

2. ความสัมพันธ์กับเวลา (Time Variancy) หมายถึง ข้อมูลในคลังข้อมูล จะต้องจัดเก็บโดยกำหนดช่วงเวลาเอาไว้ โดยจะสัมพันธ์กับการดำเนินธุรกิจของหน่วยธุรกิจนั้น เพราะในการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

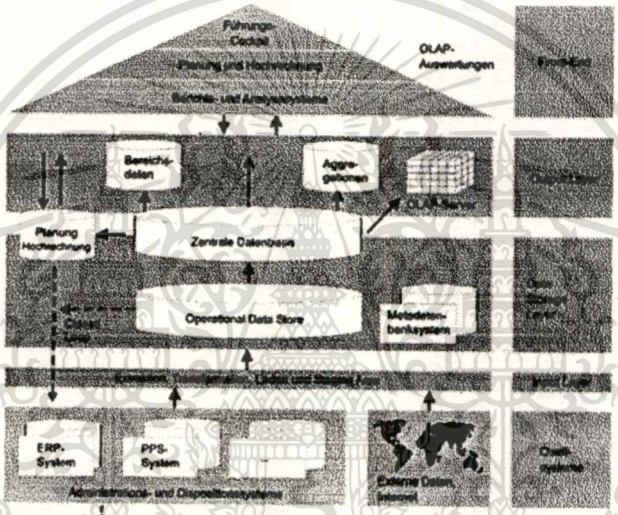
ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.



ตัดสินใจด้านการบริหารจำเป็นต้องมีข้อมูลเปรียบเทียบในแต่ละช่วงเวลา แต่ละจุดของข้อมูลจะเกี่ยวข้องกับจุดของเวลาและข้อมูลแต่ละจุดสามารถเปรียบเทียบกันได้ตามแกนของเวลา

3. ความเสถียรของข้อมูล (Nonvolatile) หมายถึง ข้อมูลในคลังข้อมูลจะไม่เปลี่ยนแปลง ไม่ว่าจะเป็นการเพิ่มเติมข้อมูลใหม่ หรือการปรับปรุงแก้ไขข้อมูลเดิมที่บรรจุอยู่แล้ว ผู้ใช้ทำได้เพียงการเข้าถึงข้อมูลเท่านั้น

## 2.3 สถาปัตยกรรมคลังข้อมูล (Data Warehouse Architecture - DWA) และ ระบบดาต้ามาร์ท (Data Mart)




รูปที่ 2.5 ภาพแสดงสถาปัตยกรรมคลังข้อมูล

DWA เป็นโครงสร้างมาตรฐานที่ใช้อธิบาย เพื่อให้เข้าใจแนวคิด และกระบวนการของคลังข้อมูลนั้นๆ ซึ่งโดยทั่วไปแล้วคลังข้อมูลแต่ละระบบอาจจะมีรูปแบบที่ไม่เหมือนกันได้ เพื่อให้เหมาะสมกับองค์กรนั้นๆ ทั้งนี้ส่วนประกอบต่างๆ ภายใน DWA ที่สำคัญ ได้แก่

1. Operational Database หรือ external database layer ทำหน้าที่จัดการกับข้อมูลในระบบงานปฏิบัติการหรือแหล่งข้อมูลภายนอกองค์กร

2. Information access layer เป็นส่วนที่ผู้ใช้ปลายทางติดต่อผ่านโดยตรงประกอบด้วย ฮาร์ดแวร์และ ซอฟต์แวร์ ที่ใช้ในการแสดงผลเพื่อการวิเคราะห์ โดยมีเครื่องมือช่วย เป็นตัวกลางที่ผู้ใช้ใช้ติดต่อกับคลังข้อมูล โดยในปัจจุบันเครื่องมือที่ได้รับความนิยมเพิ่มขึ้นอย่างรวดเร็วนั่นคือ Online Analytical Processing Tool หรือ OLAP tool ซึ่งเป็นเครื่องมือที่มีความสามารถในการวิเคราะห์ที่ซับซ้อน และแสดงข้อมูลในรูปแบบหลายมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Data access layer เป็นส่วนต่อประสานระหว่าง Information access layer กับ operational layer

4. Data director (metadata) layer เพื่อให้เข้าถึงข้อมูลได้ง่ายขึ้น และเป็นการเพิ่มความเร็วในการเรียกและดึงข้อมูลของคลังข้อมูล

5. Process management layer ทำหน้าที่จัดการกระบวนการทำงานทั้งหมด

6. Application messaging layer เป็นมิดเดิลแวร์ทำหน้าที่ในการส่งข้อมูลภายในองค์กรผ่านทางเครือข่าย

7. Data warehouse (physical) layer เป็นแหล่งเก็บข้อมูลของทั้ง Information data และ external data ในรูปแบบที่ง่ายแก่การเข้าถึงและยืดหยุ่นได้

8. Data staging layer เป็นกระบวนการแก้ไข และดึงข้อมูลจาก external database

## 2.4 การเคลื่อนที่ของข้อมูลในคลังข้อมูล

ข้อมูลที่จะจัดเก็บภายในคลังข้อมูลมีการเคลื่อนที่ของข้อมูล (Information flow) 5 ประเภท สามารถอธิบายได้ดังต่อไปนี้

1. Inflow คือการนำข้อมูลจากฐานข้อมูลอื่นเข้าสู่คลังข้อมูลทั้งฐานข้อมูลภายในและภายนอกองค์กร โดยในขั้นนี้อาจมีการเปลี่ยนแปลงโครงสร้างข้อมูล การทำ denormalize การลบหรือเพิ่มฟิลด์เพื่อให้ข้อมูลทั้งหมดอยู่ในเนื้อหาที่สนใจเดียวกัน ในขั้นตอนนี้อาจใช้เครื่องมือ ที่เรียกว่า data warehouse tool

2. Upflow เมื่อข้อมูลที่เราต้องการอยู่ในคลังข้อมูลแล้วในบางครั้งอาจต้องมีการเพิ่มคุณค่าให้กับข้อมูลด้วยเพื่อให้ข้อมูลอยู่ในรูปแบบที่เป็นประโยชน์มากที่สุดต่อการนำเครื่องมือมาใช้ ซึ่งได้แก่การจัดกลุ่มข้อมูลหาค่าทางสถิติที่ซับซ้อน จัดข้อมูลให้อยู่ในรูปแบบหรือเทมเพลตมาตรฐาน

3. Downflow เป็นขั้นตอนของการปรับปรุงเปลี่ยนแปลงข้อมูลที่เก่า และไม่อยู่ในเนื้อหาที่องค์กรสนใจออกไปจากคลังข้อมูลขององค์กร

4. Outflow เป็นขั้นที่ผู้ใช้เรียกใช้ข้อมูลในคลังข้อมูลผ่านเครื่องมือต่างๆ โดยการเรียกใช้อาจมีเพียงขอเรียกเป็นครั้งคราวเป็นประจำทุกวัน/เดือน หรือแม้กระทั่งต้องการแบบทันที

5. Metafolw ข้อมูลที่จัดเก็บในคลังข้อมูลจะถูกทำข้อมูลไว้อีกชุดหนึ่ง เป็นแหล่งที่มาของข้อมูลนั้น หรือแม้กระทั่งที่อยู่ของข้อมูลนั้นในคลังข้อมูลและข้อมูลอื่นที่เกี่ยวข้อง

## 2.5 วิธีการออกแบบฐานข้อมูลสำหรับคลังข้อมูล และ ดาต้ามาร์ท

วิธีการนี้ถูกเสนอโดย Kimball ในปี 1996 เรียกว่า ระเบียบวิธี 9 ขั้น หรือ Nine-Step Methodology โดยวิธีการนี้เริ่มจากการออกแบบจากส่วนย่อยที่แสดงถึงแต่ละระบบงานขององค์กร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือเรียกอีกอย่างหนึ่งว่าดาต้ามาร์ท (data mart) โดยเมื่อออกแบบแต่ละส่วนสำเร็จแล้ว จึงนำมารวมกันเป็นคลังข้อมูล ขององค์กรในขั้นสุดท้าย ซึ่งขั้นตอนทั้ง 9 ขั้นตอน มีรายละเอียดดังนี้

1. กำหนดดาต้ามาร์ท คือการเลือกว่าจะสร้างดาต้ามาร์ทของระบบงานใดบ้าง และระบบงานใดเป็นระบบงานแรกโดยองค์กรจะต้องสร้าง E-R model ที่รวมระบบงานทุกระบบขององค์กรไว้ แสดงการเชื่อมโยงของแต่ละระบบงานอย่างชัดเจน และสิ่งที่ต้องคำนึงถึงในการเลือกระบบงานที่จะเป็นดาต้ามาร์ทแรกนั้น มี 3 ปัจจัยที่เกี่ยวข้อง ได้แก่ จะต้องสามารถพัฒนาออกมาได้ทันตามเวลาที่ต้องการ โดยอยู่ในงบประมาณที่กำหนดไว้และต้องตอบปัญหาทางธุรกิจให้แก่องค์กรได้ ดังนั้น ดาต้ามาร์ทแรกควรจะเป็นของระบบงานที่นำรายได้เข้ามาสู่องค์กรได้ เช่น ระบบงานขาย เป็นต้น

2. กำหนด Fact table ของดาต้ามาร์ท คือการกำหนดเนื้อหาหลักที่ควรจะเป็นของดาต้ามาร์ท โดยการเลือกเอนทิตีหลักและกระบวนการที่เกี่ยวกับเอนทิตีนั้นๆ ออกมาจาก E-R model ขององค์กร นั่นหมายถึงว่าจะทำให้เราทราบถึง dimension table ที่ควรจะมีด้วย

3. กำหนดแอตทริบิวต์ ที่จำเป็นในแต่ละ dimension table คือการกำหนดแอตทริบิวต์ที่บอกหรืออธิบายรายละเอียดของ dimension ได้ ทั้งนี้แอตทริบิวต์ที่เป็น primary key ควรเป็นค่าที่คำนวณได้ กรณีที่มีดาต้ามาร์ทมากกว่าหนึ่งดาต้ามาร์ทมี dimension เหมือนกัน นั่นหมายถึงว่าแอตทริบิวต์ใน dimension นั้นจะต้องเหมือนกันทุกประการ แต่นั่นก็ไม่อาจจะแก้ไขปัญหาการจัดเก็บข้อมูลซ้ำซ้อน อันจะนำมาสู่ความแตกต่างกันของข้อมูลชุดเดียวกัน ปัญหานี้จึงเป็นการดีที่จะมีการใช้ dimension table ร่วมกันในแต่ละ fact table ที่จำเป็นต้องมี dimension ดังกล่าว โดยเรียก dimension table ลักษณะแบบนี้ว่า conformed และเรียก fact table ว่า fact constellation

4. กำหนดแอตทริบิวต์ที่จำเป็นใน fact table โดยแอตทริบิวต์หลักใน fact table จะมาจาก primary key ในแต่ละ dimension table นอกจากนี้แล้ว ยังสามารถมีแอตทริบิวต์ที่จำเป็นอื่นๆ ประกอบอยู่ด้วย เช่น แอตทริบิวต์ที่ได้จากการคำนวณค่าเบื้องต้นที่จำเป็นสำหรับการคงอยู่ของแอตทริบิวต์อื่นใน fact table เรียกอีกอย่างหนึ่งว่า measure การกำหนดแอตทริบิวต์นี้ไม่ควรจะเลือกแอตทริบิวต์ที่คำนวณค่าไม่ได้ เช่น เป็นตัวหนังสือหรือไม่ใช่ตัวเลข เป็นต้น และไม่ควรเลือกแอตทริบิวต์ที่ไม่เกี่ยวข้องกับเนื้อหาของ fact table ที่เราสนใจด้วย

5. จัดเก็บค่าการคำนวณเบื้องต้นใน fact table คือการจัดเก็บค่าที่ได้จากการคำนวณให้เป็นแอตทริบิวต์หนึ่งใน fact table ถึงแม้ว่าจะสามารถหาค่าได้จากแอตทริบิวต์อื่นๆ ก็ตาม ทั้งนี้เพื่อให้การสอบถามมีประสิทธิภาพมากขึ้น สามารถทำงานด้วยความเร็วที่เพิ่มขึ้น เนื่องจากไม่ต้องคำนวณค่าใหม่ทั้งหมด ถึงแม้ว่าจะเกิดความซ้ำซ้อนของข้อมูลในการจัดเก็บบ้างก็ตาม

6. เขียนคำอธิบายของ dimension table ทั้งนี้ก็เพื่อให้ผู้ใช้สามารถใช้งานดาต้ามาร์ทได้อย่างมีประสิทธิภาพเพราะเกิดความเข้าใจอย่างดีในส่วนต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. กำหนดระยะเวลาในการจัดเก็บข้อมูลในฐานข้อมูล โดยอาจจะเป็นการจัดเก็บเพียงช่วงระยะเวลา 1-2 ปี หรือ นานกว่านั้น ขึ้นอยู่กับความต้องการขององค์กร เนื่องจากองค์กรแต่ละประเภทมีความต้องการในการจัดเก็บข้อมูลต่างช่วงเวลากัน ทั้งนี้ขึ้นอยู่กับความจำเป็นหรือข้อกำหนดในการดำเนินธุรกิจ

8. การติดตามปัญหาการเปลี่ยนแปลงของ dimension อย่างช้าๆ คือ การเปลี่ยนเอาแอตทริบิวต์ของ Dimension table เก่ามาใช้แล้วส่งผลกระทบต่อข้อมูลปัจจุบันของ dimension table โดยสามารถแบ่งประเภทของปัญหาที่เกิดได้เป็น 3 ประเภท คือ เกิดการเขียนทับข้อมูลใหม่โดยข้อมูลเก่า เกิดเรคอร์ดใหม่ๆ ขึ้นใน dimension เกิดเรคอร์ดที่มีทั้งค่าเก่าและใหม่ปนกันไป

9. กำหนดคิวรี เป็นการออกแบบด้านกายภาพเพื่อให้ผู้ใช้เกิดความสะดวกในการใช้งานและสามารถทำงานได้อย่างมีประสิทธิภาพ

เมื่อดำเนินการทั้ง 9 ขั้นตอนสำหรับแต่ละดาต้ามาร์ทเสร็จแล้ว จึงจะนำทั้งหมดมารวมกันเป็นภาพของคลังข้อมูลขององค์กรต่อไป

## 2.6 การแปลงข้อมูลเข้าสู่ดาต้ามาร์ท

เมื่อเราออกแบบฐานข้อมูลสำหรับแต่ละดาต้ามาร์ทเสร็จแล้ว ขั้นตอนต่อไปที่สำคัญยิ่งก็คือการนำข้อมูลจากแหล่งข้อมูลไปแปลงให้อยู่ในแพลตฟอร์มของฐานข้อมูลที่ได้ออกแบบไว้ นั่นก็คือการแปลงข้อมูล หรือ Extraction Transformation and Loading (ETL) นั่นเอง โดยที่คุณภาพของการแปลงข้อมูลเป็นสิ่งที่สำคัญมากสำหรับการสร้างคลังข้อมูล ความซับซ้อนของการแปลงข้อมูลและโครงสร้างของข้อมูลจะแตกต่างกันไปตามคลังข้อมูลที่แต่ละองค์กรต้องการ โดยที่การแปลงข้อมูลหมายรวมตั้งแต่การวิเคราะห์แหล่งข้อมูล กำหนดการส่งข้อมูลรวบรวมหรือสร้างข้อมูลภายนอก วางแผนและสร้างรูทีนของการแปลงข้อมูล และตรวจสอบความถูกต้องของข้อมูลที่ได้สามารถสรุปเป็นขั้นตอนได้ดังนี้

1. วิเคราะห์แหล่งข้อมูล เช่น ปริมาณของข้อมูล จำนวนและชนิดของการเข้าถึงแหล่งข้อมูล แพลตฟอร์มและภาษาโปรแกรมที่ใช้ เป็นต้น

2. ย้ายข้อมูลที่ต้องการจากระบบเดิม ไว้ในบริเวณที่ใช้ปรับแต่งข้อมูล หรือเรียกบริเวณนี้ว่า staging area เพื่อนำมาเลือกเฉพาะส่วนที่ต้องการแปลงข้อมูลและตรวจสอบความถูกต้อง หรือการทำความสะอาดข้อมูล

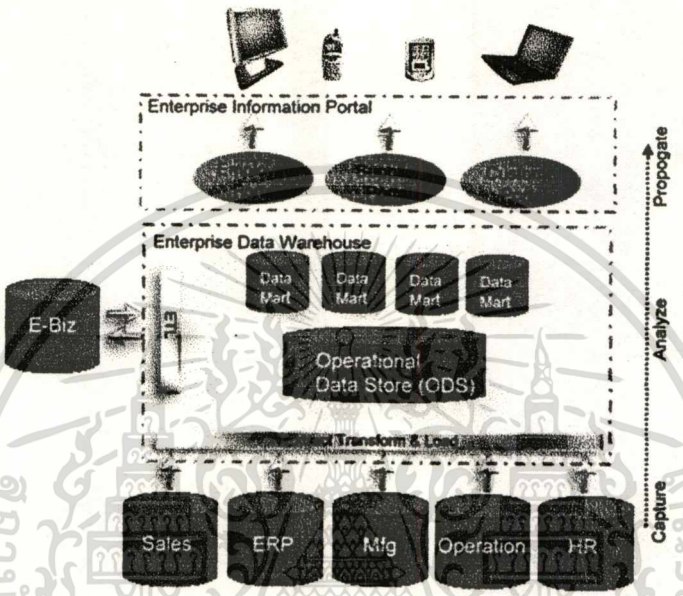
3. กำหนด primary key ของ fact table และ dimension table และกำหนด foreign key ระหว่าง fact table กับ dimension table

4. ย้ายข้อมูล ที่ทำความสะอาดแล้วจาก staging area ลงสู่เซิร์ฟเวอร์ของดาต้ามาร์ท

5. สร้าง metadata ของแต่ละดาต้ามาร์ท โดยเก็บรายละเอียดของข้อมูลการอัปเดตและส่งออกไปไว้ในดาต้ามาร์ท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ตรวจสอบความถูกต้องของข้อมูล ซึ่งจะต้องกระทำตลอดทั้งกระบวนการแปลงข้อมูล โดยตรวจสอบผลรวมทั้งหมดของจำนวนข้อมูลที่ดึงมาจากแหล่งข้อมูลที่เพิ่มเข้าไป ตรวจแก้ข้อมูลในระบบเดิมของแหล่งข้อมูล หรือในรูทีนของการแปลง ซึ่งควรจะเก็บข้อมูลในการตรวจแก้ไว้ใน metadata ของการแปลงข้อมูลด้วย ตรวจสอบค่าของข้อมูลให้ถูกต้องในกระบวนการรวบรวมข้อมูล ตรวจสอบผลรวมของข้อมูลหลังจากการย้ายข้อมูลลงสู่คาต้ามาร์ทแล้ว




รูปที่ 2.6 ภาพแสดงสถาปัตยกรรมของการนำเอา ข้อมูลจากคาต้ามาร์ทมาใช้

## 2.7 แนวคิดของเทคโนโลยี OLAP

### 2.7.1 ลักษณะงานการประมวลผลธุรกรรมออนไลน์และการประมวลผลเชิงวิเคราะห์ออนไลน์

ระบบ OLTP โดยทั่วไปแล้วนั้น จำเป็นจะต้องสามารถดำเนินการกับข้อมูลธุรกรรมได้อย่างมีประสิทธิภาพงานที่ทำกับข้อมูลได้แก่การปรับค่าของข้อมูลให้เป็นปัจจุบันและการเพิ่มข้อมูลลงไปในฐานข้อมูล ข้อมูลเหล่านี้อาจจะมีจำนวนมากและเพิ่มขึ้นตลอดเวลา ณ เวลาใดเวลาหนึ่งอาจจะมีการประมวลผลข้อมูลจำนวนนับแสนเรคอร์ดได้ เช่น ณ สนามบินแต่ละแห่งจะมีผู้โดยสารเข้ามารับบัตรที่นั่งของสายการบินต่างๆ เป็นจำนวนนับหมื่นๆ คน คอมพิวเตอร์ของสายการบินจะต้องตรวจสอบการสำรองที่นั่งต้องบันทึกเลขที่นั่งและเที่ยวบินรวมทั้งอาจจะต้องมีการปรับเปลี่ยนโยกย้ายข้อมูลจากเที่ยวบินหนึ่งไปอีกเที่ยวบินหนึ่งได้ด้วย หรือในกรณีของศูนย์การค้าและ ซูเปอร์มาร์เก็ต จะมีการบันทึกเรคอร์ดการขายเพิ่มเข้าไปในฐานข้อมูลการขายตลอดเวลา รวมแล้ววันละเป็นหมื่นๆ รายการ การออกแบบระบบ OLTP แบบนี้จำเป็นต้องหาทางให้ระบบสามารถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำงานได้อย่างถูกต้องรวดเร็วตลอดเวลา เอื้ออำนวยให้ผู้ใช้จำนวนมากสามารถใช้ระบบได้พร้อมกัน อีกทั้งยังต้องสามารถแก้ไขฟื้นสภาพให้กลับดีดังเดิมได้หากเกิดความขัดข้องเสียหาย

การที่จะจัดทำระบบ OLTP ให้มีความสามารถในแบบนี้ได้ต้องคำนึงถึงปัจจัยต่อไปนี้

1. ขนาดและตำแหน่งของ rollback segment
2. ดัชนี การจัดกลุ่ม และ การคำนวณตำแหน่งที่อยู่ (hashing)
3. การออกแบบข้อมูลธุรกรรมให้เหมาะกับงานประยุกต์
4. หน่วยเก็บและเนื้อที่ว่างสำหรับการเก็บข้อมูลใหม่
5. ความเข้าใจลักษณะงานประยุกต์และการเขียนคำสั่งสำหรับค้นคืนข้อมูล
6. การปรับปรุงสมรรถนะของระบบอย่างต่อเนื่อง

ระบบ OLTP ที่พัฒนาขึ้นโดยใช้เทคนิคด้านฐานข้อมูลตามปกติมักจะไม่สามารถรับกับปริมาณข้อมูลที่เพิ่มขึ้นอย่างมากมายเป็นประจำทุกวันได้ การนำระบบเช่นนี้มาใช้จึงมีความเสี่ยงที่จะเกิดความผิดพลาดเสียหายขึ้น วิธีการแก้ไขก็คือการแยกฐานข้อมูลออกมาเป็นส่วน ๆ ให้เหมาะกับการใช้งาน

งานที่เกี่ยวข้องกับฐานข้อมูลอีกอย่างหนึ่งก็คืองานที่เรียกว่า *การประมวลผลเชิงวิเคราะห์ออนไลน์ (On-Line Analytical; OLAP)* ระบบ OLTP ที่กล่าวไปแล้วนั้นเน้นที่การบันทึกเก็บข้อมูลใหม่ๆ เพิ่มเข้าไว้ในฐานข้อมูล ส่วนระบบ OLAP นั้นเน้นที่การค้นคืนข้อมูลที่มีอยู่แล้วจากฐานข้อมูลเพื่อนำมาวิเคราะห์อย่างละเอียด ผู้ใช้ระบบ OLAP ส่วนใหญ่คือผู้บริหาร นักวิจัยตลาด นักสถิติ หรือ ผู้ใช้อื่นๆ ดังนั้นปัจจัยสำคัญสำหรับความสำเร็จของระบบ OLAP ก็คือระบบจะต้องทำงานได้รวดเร็ว สามารถค้นหาข้อมูลจากฐานข้อมูลขนาดใหญ่มาคำนวณได้อย่างครบถ้วนไม่ตกหล่น ในขณะเดียวกันระบบก็จะต้องมีความมั่นคง ไม่ผิดพลาดได้ง่ายระหว่างการใช้งานปัจจัยที่จะทำให้ได้ตามที่กล่าวนี้คือ จะต้องมีระบบจัดคำสั่งค้นคืนข้อมูลให้ทำงานได้รวดเร็วที่สุด (query optimization) การจัดดัชนี จัดกลุ่มข้อมูล และ การคำนวณตำแหน่งที่อยู่ข้อมูล การประมวลผลคำสั่งค้นคืนในแบบขนาน โดยเฉพาะเมื่อใช้หน่วยเก็บแบบ RAID ·

แม้ว่าระบบ OLTP และ OLAP นี้จะเกี่ยวข้องกับข้อมูลธุรกรรมเหมือนกันแต่ก็มีความแตกต่างมากในกระบวนการทำงานที่เกี่ยวกับข้อมูล หากพบว่าการอ่านข้อมูลจากฐานข้อมูลมาประมวลผลมีช่วงเวลาโต้ตอบ (response time) ช้ามากและต้องการปรับการเก็บโดยการจัดทำดัชนีเพิ่มเติมให้การค้นคืนข้อมูลได้สะดวกขึ้น จะส่งผลให้การบันทึกข้อมูลกลับต้องช้าลงเพราะต้องเสียเวลาเนินการกับดัชนีมากขึ้นกว่าระบบเดิม ด้วยเหตุนี้จึงเป็นเรื่องยากที่เราจะปรับระบบทั้งสองให้มีสมรรถนะดีมากขึ้นพร้อมกัน

ปัจจุบันนี้แนวทางแก้ไขปัญหาข้างต้นก็คือการแยกระบบ OLTP และระบบ OLAP ออกจากกันให้เป็นคนละระบบ โดยให้ระบบ OLTP สามารถจัดเก็บข้อมูลจำนวนมากได้อย่างรวดเร็วมีประสิทธิภาพ และระบบ OLAP ก็สามารถค้นคืนและวิเคราะห์ข้อมูลตามความต้องการของผู้ใช้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่างรวดเร็ว ระบบ OLTP นั้นปกติยังคงปล่อยให้เป็นแบบเดิม หากใช้คอมพิวเตอร์ขนาดใหญ่เช่น เครื่องเมนเฟรมและใช้ระบบจัดการฐานข้อมูลขนาดใหญ่อยู่แล้วก็เพียงแต่ปรับให้สามารถบันทึกจัดเก็บข้อมูลให้เร็วขึ้น จากนั้นก็จัดทำระบบขึ้นใหม่ให้แยกข้อมูลพื้นฐานออกจากฐานข้อมูลในระบบเดิมแล้วนำข้อมูลมาจัดทำดัชนีใหม่เพื่อให้ผู้บริหารวิเคราะห์ อย่างไรก็ตามทั้งระบบ OLTP และระบบ OLAP ก็อาจจะยังไม่เหมาะที่เราจะนำมาใช้ในการวิเคราะห์ทางธุรกิจหรือช่วยผู้บริหารสำหรับการตัดสินใจ (Decision Support System) ทางธุรกิจได้เพราะต้องใช้เวลาในการประมวลผลที่นานพอสมควรและส่งผลกระทบไปถึงระบบการทำงานของเครื่องที่ใช้งานประจำวัน

เราจะมีวิธีการอย่างไรเพื่อที่จะทำให้ข้อมูลที่เรามีอยู่สามารถนำมาใช้ตอบสนองความต้องการทางธุรกิจได้อย่างรวดเร็วและมีประสิทธิภาพ ดังนั้นจึงได้นำเอาแนวความคิดระบบ*คลังข้อมูล (Data warehouse)* มาใช้ร่วมกันเพื่อตอบสนองงานในรูปแบบของคลังเก็บข้อมูลสำหรับการบริหารและหากองค์กรใดสามารถที่จะนำข้อมูลที่มีอยู่มาใช้อย่างมีประสิทธิภาพย่อมที่จะทำให้องค์กรประสบความสำเร็จเหนือคู่แข่ง

ข้อมูลส่วนมากที่จัดเก็บในคลังข้อมูลนั้นปกติจะมีน้อยกว่าข้อมูลในฐานข้อมูลของระบบ OLTP เพราะเป็นข้อมูลที่ได้นำมาจัดกลุ่มให้เหมาะสมแก่การค้นคืนแล้ว ข้อมูลเหล่านี้จะมีลักษณะ consistent กล่าวคือ ข้อมูลทุกรายการที่แสดงเรื่องเดียวกันจะต้องเขียนให้เหมือนกัน สะกดแบบเดียวกัน หรือ มีรหัสเดียวกัน หากข้อมูลมีลักษณะแตกต่างกันแล้วจะวิเคราะห์ข้อมูลได้ยาก หรืออาจทำให้ได้ผลลัพธ์ที่ไม่ถูกต้อง ในหน่วยงานและบริษัทขนาดใหญ่นั้นโอกาสที่ข้อมูลทั้งหมดจะ "สะอาด" นั้นเป็นเรื่องที่ยาก ดังนั้นจึงจำเป็นจะต้องมีผู้ทำหน้าที่กลั่นกรองและควบคุมคุณภาพของข้อมูลด้วย

## 2.8 คำจำกัดความของเทคโนโลยี OLAP

OLAP หรือ Online analytical processing เป็นเทคโนโลยีที่นำมาใช้สร้างสารสนเทศทางธุรกิจ ด้วยวิธีการเปลี่ยนรูปให้ธุรกิจแข็งแกร่งขึ้น โดยการวิเคราะห์ข้อมูลทางธุรกิจเพื่อสนับสนุนการตัดสินใจ OLAP ประกอบด้วย เครื่องมือที่ช่วยดึงและ นำเสนอข้อมูลหลายมิติ (Multidimensional) จากหลาย ๆ มุมมองโดยออกแบบมสำหรับผู้ใช้ในระดับผู้บริหาร หรือ ส่วนงานที่เกี่ยวข้องกับการวิเคราะห์ข้อมูลเพื่อใช้ประกอบ การตัดสินใจ สำหรับโครงสร้างข้อมูล OLAP เป็นแบบลำดับชั้น (Hierarchical) ช่วยให้ผู้ใช้สามารถเข้าใจภาพรวมของข้อมูลที่มีความเกี่ยวข้องกัน ในเรื่องฟังก์ชันการทำงาน เทคโนโลยี OLAP สนับสนุนการวิเคราะห์ข้อมูลระดับรายละเอียดสามารถสร้างข้อมูลสรุป และ แสดงข้อมูลเชิงเปรียบเทียบในมุมมองต่างๆ

อย่างไรก็ตามในปัจจุบัน ยังไม่มีคำอธิบายที่ชัดเจนว่า OLAP หมายถึงอะไร ไม่ได้ชี้ชัดว่าทำไมจึงต้องใช้เครื่องมือ OLAP หรือ เครื่องมือเกี่ยวกับ OLAP ที่มีอยู่ในปัจจุบันมีของค่ายใดบ้าง ในปี ค.ศ. 1993 E.F.Codd และ Associate ได้จัดพิมพ์บทความหนึ่ง โดยใช้ชื่อบทความว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"Providing OLAP to User-Analysis" โดยมีทั้งหมด 12 กฎ รวมทั้งมีคุณลักษณะเพิ่มเติมอีก 18 กฎ Codd rules กลับกลายมาเป็นทางที่ไม่เหมาะสมของการค้นหา 'OLAP compliance' ดังนั้นคณะกลุ่มทำงานกลุ่มหนึ่งจึงได้สรุปคำจัดความของ OLAP โดยเหลือเพียง 5 คำจำกัดความ ซึ่งนคำจำกัดความที่มีเหตุผลและเข้าใจง่ายสำหรับจุดประสงค์ของ OLAP คือ Fast Analysis of Shared Multidimensional Information หรือ FASMI โดยคำจำกัดความนี้เริ่มใช้ครั้งแรกในปี 1995

1. FAST หมายความว่า ระบบจะตอบโต้ไปยังผู้ใช้ในเวลาประมาณ 5 วินาที ด้วยการวิเคราะห์อย่างง่ายใช้เวลาไม่เกิน 1 วินาทีและไม่เกิน 20 วินาที จากการวิจัยอิสระในประเทศเนเธอร์แลนด์ได้แสดงว่า end-user คิดว่า process นั้น fail ถ้าไม่ได้ผลการทำงานภายใน 20 วินาที และพวกเขาจะทำการกดปุ่ม 'Alt + Ctrl + Delete' นอกจากระบบจะเตือนพวกเขาว่า report ต้องใช้ระยะเวลา แต่ในดานของการวิเคราะห์ ความเร็วนี้เป็นการยากที่จะได้รับผลในงานที่มีข้อมูลเป็นปริมาณมากๆ โดยเฉพาะอย่างยิ่ง ถ้า on-the-fly และ ad hoc calculation ถูกเรียกใช้ Vendor ใช้เทคนิคหลายเทคนิคเพื่อบรรลุเป้าหมายนี้ รวมถึง specialized forms ของการเก็บข้อมูล ขยายก่อนการคำนวณและความต้องการ รวมถึง hardware เฉพาะทาง แต่พวกเราไม่ได้คำนึงว่า product ของเราเป็น product ที่ดีที่สุด เราคาดหวังเพียงว่าสิ่งนี้จะอยู่ใน developing technology โดยเฉพาะ pre-calculation ที่ยอมรับกับข้อมูลขนาดใหญ่ application ส่วนน้อยคือฐานข้อมูลจะใหญ่เกินไป (ปัญหา database explosion) เท่าที่การทำทุกอย่างเป็น on-the-fly จะเป็นการช้าเกินไปกับฐานข้อมูลขนาดใหญ่ โดยถ้าฮาร์ดแวร์ผิดปกติถูกใช้อยู่ มันดูเหมือนเป็นสิ่งอัศจรรย์ในครั้งแรกถ้า report นั้นใช้เวลาน้อย เพราะ user จะเบื่อการรอคอย และมันจะส่งผลให้ project สำเร็จช้ากว่าการมีการโต้ตอบแบบทันทีทันใด

2. ANALYSIS หมายถึง ระบบสามารถจัดการกับตรรกะและการวิเคราะห์สถิติทางธุรกิจที่สัมพันธ์กันระหว่าง application และ user และง่ายพอสำหรับ user กลุ่มเป้าหมาย ถึงแม้ว่าก่อนการเขียนโปรแกรมอาจเป็นสิ่งจำเป็น เราไม่ได้คิดให้มันได้รับการยอมรับ ถ้าคำจำกัดความ application ทั้งหมดถูกใช้โดย professional มันเป็น necessary ที่อนุญาตให้ user ทำการ define ad hoc calculations ใหม่ให้เป็นส่วนหนึ่งของการวิเคราะห์และรายงานข้อมูลในด้านที่ต้องการ โดยปราศจากโปรแกรมและเราจะปฏิเสธผลิตภัณฑ์ที่ไม่มีความยืดหยุ่นเพียงพอต่อผู้ใช้

3. SHARED หมายถึง ระบบถูก implement ขึ้นด้วยความต้องการความมั่นใจทางด้านความปลอดภัย และจำเป็นในการที่ให้หลากหลายผู้ใช้เข้าถึงการเขียนข้อมูลได้ เช่นการ update การ lock ประเภทความสำคัญของผู้ใช้ ไม่ใช่ให้ application ทั้งหมดให้ผู้ใช้เข้าไปทำการเขียนได้ทุกคน ระบบควรจะสามารถจัดการกับผู้ใช้หลากหลาย priority ซึ่งจุดนี้เป็นจุดอ่อนของผลิตภัณฑ์ OLAP จำนวนมากซึ่งคิดเอาเองว่า OLAP ทั้งหมดจะสามารถอ่านได้เพียงอย่างเดียว (read-only) กับระบบความปลอดภัยอย่างง่าย ผลิตภัณฑ์ที่สามารถให้ multi-user ทำการอ่านและเขียนข้อมูลได้นั้นปกติจะมี security model ตัวอย่างเช่นMicrosoftOLAPServices

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. MULTIDIMENSIONAL เป็น key หลักที่มีความต้องการ ถ้าเรามีคำจำกัดความหนึ่งคำของ OLAP multidimension ก็คือคำจำกัดความนั้นด้วย ระบบจะต้องคลอบคลุมแนวคิดการแสดงข้อมูลแบบหลายมิติ รวมถึง support สำหรับ hierarchies และ multiple hierarchies นี้คือทางตรรกะที่จะวิเคราะห์ธุรกิจและองค์กร

5. INFORMATION คือ ข้อมูลทั้งหมดและได้มาจากความจำเป็นของสารสนเทศ เราจัด capacity ของหลายผลิตภัณฑ์ในด้านของความสามารถในการจัดการข้อมูล ไม่ใช่จำนวน Gigabyte ที่สามารถเก็บข้อมูลได้ capacities ของผลิตภัณฑ์มีความแตกต่างกัน ผลิตภัณฑ์ OLAP ที่ใหญ่ที่สุดสามารถเก็บข้อมูลได้อย่างน้อยเป็นพันครั้ง โดยเก็บข้อมูลได้มากเท่าที่จะเก็บได้และรวมถึง data duplication, ความต้องการ RAM, การใช้ disk space ให้เป็นประโยชน์, ประสิทธิภาพ, การผสมผสานกับ data warehouse เป็นต้น

## 2.9 OLAP เครื่องมือสำหรับการวิเคราะห์เพื่อสนับสนุนการตัดสินใจ

### 2.9.1 ลักษณะสำคัญของเทคโนโลยี OLAP

เทคโนโลยี OLAP เป็นที่นิยมไม่ใช่แค่เพียงเพราะว่า เป็นเครื่องมือวิเคราะห์สถานการณ์ทางธุรกิจเท่านั้น แต่ยังเป็นเครื่องมือที่สร้างการเปลี่ยนแปลงสำหรับการมองข้อมูลในมุมมองใหม่ ๆ เพื่อใช้วิเคราะห์ข้อมูลทางธุรกิจ บนสถานการณ์ที่แตกต่างกัน อีกด้วย ด้วยเหตุนี้เทคโนโลยี OLAP จึงเป็นที่นิยมสำหรับองค์กร และ หน่วยงานต่าง ๆ อย่างแพร่หลาย สามารถแจกแจงคุณสมบัติได้ดังนี้

1. สามารถแสดงข้อมูลซึ่งอยู่ในมุมมองหลายมิติได้
2. สามารถคำนวณหาผลรวมของข้อมูลที่ต้องการได้ล่วงหน้า และ รวบรวมผลลัพธ์เหล่านั้นเก็บไว้ที่เดียวกัน มีความสามารถเรื่องการวิเคราะห์และ การคิวรี
3. สนับสนุนการทำงานของนักวิเคราะห์ธุรกิจ
4. สนับสนุนการแสดงข้อมูลแบบระดับสูง (Roll-up) ระดับรายละเอียด (Drill-down) และระดับตามขวาง (Drill-across)
5. มีการสนับสนุนการสร้างแบบจำลองเชิงวิเคราะห์ และ การทำนายผลในอนาคต

### 2.9.2 เทคโนโลยีแสดงข้อมูลด้วยมุมมองหลายมิติ

การประมวลผลเชิงวิเคราะห์จะนำข้อมูลที่อยู่ภายในคลังข้อมูล ซึ่งผ่านการรวบรวมมาแล้วมาจัดให้อยู่ในรูปของข้อมูลหลายมิติ (Multidimensional) ซึ่งเป็นแบบจำลองข้อมูล ซึ่งเปรียบเสมือนกับรูปลูกบาศก์ แต่ละมุมมองจะทำให้เกิดการคิวรีข้อมูลจากคลังข้อมูลเรียกว่า คิวบ์ (Cube) ประกอบด้วยองค์ประกอบที่สำคัญ คือ Dimension และ Measure การผสมผสานของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Dimension ต่าง ๆ ของคิวบ์จะช่วยสร้างผลลัพธ์หลาย ๆ แบบซึ่งได้มาจากการพิจารณามุมมองแต่ละด้าน สามารถอธิบายได้ดังนี้

1. Roll Up และ Drill Down

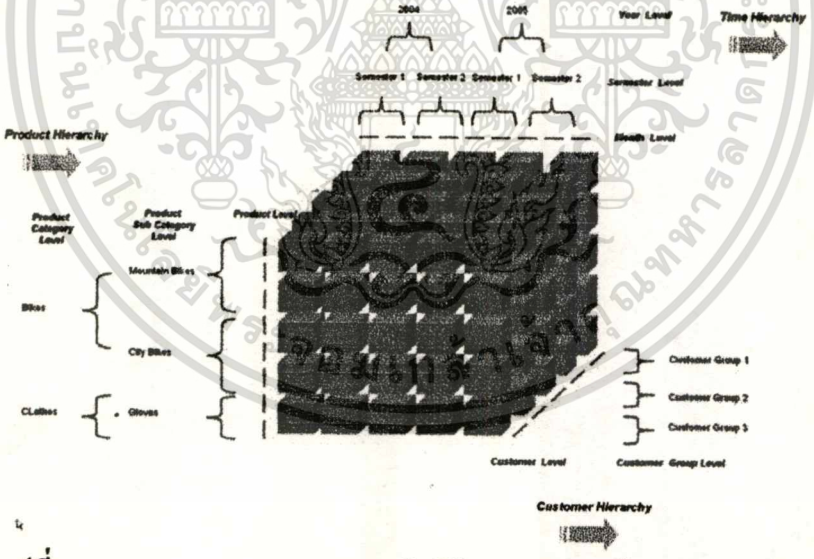
Roll Up และ Drill Down คือ การเปลี่ยนแปลงระดับความละเอียดของการพิจารณาข้อมูลซึ่งหมายถึง การเปลี่ยนแปลงมุมมอง ไปตามลำดับชั้นของ Dimension ที่กำกับ Measure ที่สนใจอยู่ Drill down หมายถึงการเพิ่มความละเอียดในการพิจารณาข้อมูลจากระดับที่หยาบไปสู่ระดับที่ละเอียดมากขึ้น ในทางกลับกันการ Roll Up เป็นกระบวนการที่ตรงกันข้ามกับการ Drill down เพราะจะหมายถึงการเปลี่ยนแปลงระดับความละเอียดของการพิจารณาข้อมูล จากระดับที่ละเอียดขึ้นมาสู่ระดับที่หยาบมากขึ้น

2. Slice

การเลือกพิจารณาผลลัพธ์บางส่วนที่เราสนใจ โดยการเลือกเฉพาะค่าที่ถูกกำกับโดยบางค่าที่เป็นไปได้ของ Dimension

3. Dice

คือ กระบวนการพลิกแกนหรือมุมมอง ให้ตรงตามความต้องการของผู้ใช้งาน




**รูปที่ 2.7** ภาพแสดงมุมมองของการคิวรีข้อมูลจากคลังข้อมูลเรียกว่า คิวบ์

**2.9.3 สถาปัตยกรรมของ OLAP**

สถาปัตยกรรมของ OLAP ประกอบด้วยส่วนประกอบการทำงาน ที่สำคัญ ๆ 3 ส่วน คือ การบริการการแสดง (Presentation Service) การบริการเทคโนโลยี OLAP (OLAP Service) และการบริการฐานข้อมูล (Database Service) สามารถแสดงรายละเอียดได้ดังนี้

1. บริการการแสดง (Presentation Service)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กล่าวได้ว่า OLAP เป็นเครื่องมือที่เชื่อมโยงระหว่างข้อมูลกับธุรกิจเข้าไว้ด้วยกัน ดังนั้นสารสนเทศเป็นข้อมูล ซึ่งสามารถนำมาทำการวิเคราะห์ และนำมาใช้กันอย่างมากแพร่หลาย สำหรับประกอบการตัดสินใจในบริบทของธุรกิจ ดังนั้นข้อมูล มีความจำเป็นอย่างยิ่งที่ต้องนำเสนอให้อยู่ในรูปซึ่งคนทำงานธุรกิจ สามารถที่จะเข้าใจเพื่อนำไปพัฒนาต้นแบบได้ ดังนั้นการบริการการแสดงผล ต้องสามารถใช้งานง่าย นอกจากนี้ยังรวมถึงจะต้องซ่อนสถาปัตยกรรมที่ซับซ้อน และกระบวนการทำงานของระบบไว้เบื้องหลัง รวมทั้ง การบริการการแสดงผลต้องมีความยืดหยุ่น และ ปรับตัวได้เพราะความแตกต่างของผู้ใช้ซึ่งมีทักษะ และความถนัดในการใช้งานระบบที่แตกต่างกัน

2. การบริการ OLAP (OLAP Service)

สนับสนุนการซักถามด้วยมุมมองหลายมิติ สามารถรวมความต้องการในการประมวลผลเชิงวิเคราะห์ทั้งหมดได้ ไม่ว่าจะเป็นความสามารถในการซักถามจากคำถามที่ง่ายไปยังคำถามที่มีความซับซ้อน และ ความสามารถในการออกรายงาน จากการรายงานผลแบบง่าย ๆ ไปยัง การออกรายงานโดยอ้างเหตุผลประกอบ อาจกล่าวได้ว่า เทคโนโลยี OLAP ได้รวมเอาบริการต่าง ๆ ไม่ว่าจะเป็นการซักถาม การออกรายงาน และ การวิเคราะห์ข้อมูล เอาไว้เพื่อสนับสนุนการทำงานของผู้ใช้

3. การบริการฐานข้อมูล (Database Service)

ซึ่งสถาปัตยกรรมของ OLAP สนับสนุนการทำงานฐานข้อมูล 2 ลักษณะคือ Conventional relational database ซึ่งใช้เครื่องมือ ROLAP และ Proprietary multidimensional database ใช้เครื่องมือ MOLAP ดังนั้น ฐานข้อมูล OLAP สามารถแบ่งได้เป็น 3 ประเภทหลัก ๆ คือ Multidimensional OLAP และ Relational OLAP นอกจากนี้ยังมี Hybrid OLAP เป็นเทคโนโลยีที่เกิดจากการประสมทั้ง 2 แบบเข้าด้วยกัน ซึ่งมีรายละเอียดดังนี้

3.1 MOLAP เป็นวิธีที่นิยมใช้ในกรทำ OLAP ข้อมูลแบบนี้เก็บอยู่ใน Multidimensional OLAP Server เป็นรูปแบบที่มีประสิทธิภาพสูงสุดในการทำงาน ช่วยให้ดิวรีเร็วที่สุด ดีที่สุด เนื่องจากคำนวณหาผลรวมของข้อมูลที่ต้องการไว้ ล่วงหน้า และ เก็บผลลัพธ์เหล่านั้นไว้ในดิสก์ เหมาะสำหรับระบบที่มีข้อมูลปานกลาง

3.1 ROLAP โครงสร้างนี้ยังใช้ข้อมูลที่อยู่ใน Relational Database และสร้าง Database อีกชุดหนึ่งเพื่อเก็บข้อมูลแบบยอดรวมไว้ โครงสร้างแบบนี้ ซึ่งเหมาะสำหรับระบบที่มีข้อมูล ปริมาณมาก และมีความถี่ในการใช้งานน้อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 HOLAP โครงสร้างนี้จะผสมผสานระหว่าง โครงสร้างแบบ MOLAP และ ROLAP เข้าด้วยกัน โดยยังใช้ข้อมูลที่มีอยู่ในรีเลชั่นนัลดาต้าเบสแต่สร้างยอดรวมไว้ในคิวบ์

## 2.10 การประยุกต์ใช้ OLAP ในระดับองค์กร

แอพพลิเคชันของ OLAP นั้น มีทั้งทำงานบนเดสก์ทอปและบนเซิร์ฟเวอร์ ถ้าเป็น OLAP ที่ทำงานบนเดสก์ทอปนั้น ฟังก์ชันการวิเคราะห์ข้อมูลจะเก็บและทำงานอยู่บนคอมพิวเตอร์ไคลเอ็นต์ โดยติดต่อกับฐานข้อมูลที่อยู่บนเซิร์ฟเวอร์โดยการส่งคิวรี SQL และรับผลลัพธ์กลับมา จากนั้นแอพพลิเคชันก็จะวิเคราะห์ เปรียบเทียบและนำเสนอข้อมูลตามรูปแบบที่ผู้ใช้ต้องการ ซึ่งจากการทำงานนี้จะเห็นได้ค่อนข้างชัดเจนว่าแอพพลิเคชัน OLAP ที่ทำงานบนเดสก์ทอปถึงแม้ว่าติดตั้งและใช้งานได้สะดวก แต่ก็มีปัญหากับเรื่องของการขยายขนาด ในทางกลับกันข้าม แอพพลิเคชัน OLAP ที่ทำงานบนฝั่งเซิร์ฟเวอร์ จะมีความสามารถในการเก็บข้อมูลไว้ในตัวเองรวมถึงการเชื่อมต่อกับฐานข้อมูลภายได้ ทำให้มีความสามารถในการขยายขนาดที่ดีกว่า รวมทั้งยังสนับสนุนฟังก์ชันการวิเคราะห์ที่ซับซ้อนมากกว่าแอพพลิเคชันที่ทำงานบนเดสก์ทอปเนื่องด้วย ทั้งนี้ก็มาจากการที่ไม่มีข้อจำกัดของทรัพยากรนั่นเอง

ในตลาดของเซิร์ฟเวอร์ OLAP นั้นมีผลิตภัณฑ์ที่โด่งดังอยู่ 4 ตัวประกอบด้วย Microsoft SQL Server Analysis Services, Hyperion Essbase, Oracle Express และ MicroStrategy นอกจากนี้ก็ยังมีผลิตภัณฑ์จากผู้ขายรายย่อยอีกเป็นจำนวนไม่น้อยให้เลือกใช้งานตามความเหมาะสม

เซิร์ฟเวอร์ OLAP นั้นสนับสนุนการบราวซ์และคิวรีข้อมูลพื้นฐาน มีฟังก์ชันการวิเคราะห์ข้อมูลที่มีความซับซ้อน และมีประสิทธิภาพของการคิวรีที่ยอดเยี่ยมด้วยการใช้คุณสมบัติของ Aggregation ที่ได้มีการคำนวณไว้ก่อนหน้าแล้ว (Pre-Computed Aggregation) เนื่องด้วยฟังก์ชันในการวิเคราะห์ของ OLAP ทำงานอยู่บนเซิร์ฟเวอร์ ดังนั้นการติดต่อระหว่างผู้ใช้กับเซิร์ฟเวอร์จึงจำเป็นต้องมีภาษาในการคิวรีอื่นที่เหมาะสมมากกว่า SQL (ที่สนับสนุนเฉพาะการดึงและค้นหาข้อมูลบนพื้นฐานของเซ็ตเท่านั้น) ภาษาที่ใช้ใน OLAP ก็ได้แก่ MDX หรือ Calc Scripts โดยภาษาในการคิวรีของ OLAP ได้รับการออกแบบให้มีประสิทธิภาพในการทำงานด้วยการเก็บคำนิยามของการคำนวณที่ซับซ้อนต่างๆ ไว้ที่เซิร์ฟเวอร์ ทำให้ฟังก์ชันในการคำนวณทั้งที่เป็นพื้นฐาน และที่ซับซ้อนสามารถให้บริการกับผู้ใช้ที่ต้องการได้อย่างกว้างขวาง

นอกจากนี้ด้วยการปรับปรุงอย่างต่อเนื่องทำให้บรรดาผู้พัฒนาแอพพลิเคชันของ OLAP ได้เริ่มปรับเปลี่ยนการใช้แหล่งข้อมูลของตนเอง ซึ่งในอดีตเป็นการเก็บข้อมูลในไฟล์หรือในฐานข้อมูลเฉพาะมาเป็นฐานข้อมูลแบบสัมพันธ์ซึ่งเป็นมาตรฐานและมีผลิตภัณฑ์รองรับที่เปิดกว้างมากกว่าแทน นับเป็นปัจจัยสำคัญที่ทำให้ผลิตภัณฑ์ OLAP กำลังกลายเป็นสิ่งที่มีความสำคัญในการตัดสินใจเลือกโซลูชันขององค์กร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ด้วยเหตุที่ว่า แอพพลิเคชันชนิดนี้ยังเล็กอยู่เมื่อเทียบกับเซิร์ฟเวอร์ฐานข้อมูล มีสาเหตุอยู่หลายประการได้แก่ การทำตลาด ความสามารถในการขยายขนาด และความยืดหยุ่น สำหรับในส่วนของการทำตลาด เป็นปัญหาเนื่องจากผู้ใช้ยังไม่ยอมรับผลิตภัณฑ์ที่มีอยู่ ซึ่งก็เกิดขึ้นทั้งจากความไม่คุ้นเคยของตัวผู้ใช้เองไปจนถึงผลิตภัณฑ์ที่ทำความเข้าใจ ติดตั้งและใช้งานได้ยาก จนถึงขณะนี้เซิร์ฟเวอร์ OLAP ยังคงไม่มีข้อตกลงในเรื่องของ API เพื่อให้ไคลเอ็นต์เข้าใช้เลย

อย่างไรดี ปัญหานี้กำลังได้รับการแก้ไขด้วยการเกิดขึ้นของเทคโนโลยีอย่าง XML ซึ่งกำลังได้รับการยอมรับจากบรรดาผู้ขายเซิร์ฟเวอร์ OLAP นั่นก็เป็นสัญญาณที่บ่งชี้ให้เห็นว่าในอนาคตอันใกล้นี้มาตรฐานการเข้าใช้ OLAP จากไคลเอ็นต์คงจะได้รับการโมเดลให้อยู่ในรูปของเอกสาร XML อย่างแน่นอน

ในส่วนความสามารถในการขยายขนาดนั้น ในอดีตที่ผ่านมา ระบบ OLAP นั้นยังทำได้ไม่ดีเท่ากับระบบฐานข้อมูลแบบสัมพันธ์ ถึงแม้ว่าในวันนี้ปัญหาในเรื่องของความสามารถในการขยายขนาดจะยังคงมีอยู่ในผลิตภัณฑ์ OLAP ส่วนใหญ่ แต่บรรดาผู้ที่เกี่ยวข้องก็เริ่มที่จะแก้ปัญหานี้บ้างแล้ว ดังจะเห็นได้จาก Case Study และเอกสารเกี่ยวกับการใช้ OLAP ที่ทำงานกับข้อมูลระดับเทอระไบต์จากผู้ผลิตหลายๆ รายที่มีอยู่ในปัจจุบัน (โดยเฉพาะอย่างยิ่ง Micro Strategy และ Microsoft)

ในส่วนของความยืดหยุ่นนั้น ถึงแม้ว่า OLAP จะยังมีความยืดหยุ่นไม่เท่ากับฐานข้อมูลแบบสัมพันธ์ แต่ก็นับได้ว่าในปัจจุบันเซิร์ฟเวอร์ OLAP ได้รับการพัฒนาให้รองรับกับความต้องการของผู้ใช้ โปรแกรมเมอร์และฝ่ายบริหารได้อย่างหลากหลายมากยิ่งขึ้น

ด้านของราคา ถึงแม้ว่าจะเป็นผลิตภัณฑ์ที่ดูว่าจำเป็นต้องใช้เทคโนโลยีชั้นสูง ก็น่าจะส่งผลให้ราคาของผลิตภัณฑ์แพงขึ้นตามไปด้วย แต่ความจริงก็คือ เซิร์ฟเวอร์ OLAP ที่มีอยู่ส่วนใหญ่ยังคงเป็นเพียงฟังก์ชันที่เพิ่มเติมขึ้นมาจากเซิร์ฟเวอร์ฐานข้อมูลแบบสัมพันธ์อยู่แล้ว ทำให้ค่าใช้จ่ายที่เกิดขึ้นเป็นเพียงการจ่ายเงินเพิ่มจากการซื้อเซิร์ฟเวอร์ฐานข้อมูล ซึ่งก็มีจำนวนไม่สูงมากอย่างที่คิด

### 2.10.1 แนวคิดของ Dimensional ที่เหมือนกัน

สำหรับผู้ออกแบบและโมเดลข้อมูลแบบ Dimensional ที่มีประสบการณ์ สามารถทำงานได้สะดวกมากขึ้น เนื่องจากเอกสารเกี่ยวกับผลิตภัณฑ์เซิร์ฟเวอร์ OLAP ส่วนใหญ่มักจะมีการพูดถึงแนวคิดและใช้คำศัพท์ที่คุ้นเคยไม่ว่าจะเป็น Dimensional, Hierarchies และ Facts หรือ Measures ซึ่งผลิตภัณฑ์เกือบทั้งหมดใช้แนวคิดของ Cube ซึ่งสามารถเปรียบเทียบได้โดยตรงกับ Table Schema ในฐานข้อมูลแบบสัมพันธ์ ซึ่งมีความสัมพันธ์กับ Dimension และ Aggregate Tables

Dimension ของ OLAP นั้นประกอบด้วย Hierarchies หรือ Summarization Paths โดยความสัมพันธ์ระหว่าง Data Hierarchy กับ OLAP Dimension นั้นเป็นสิ่งที่มีความสำคัญมาก มีผลิตภัณฑ์ OLAP จำนวนไม่น้อยที่สามารถให้ผู้ใช้กำหนด Hierarchies หลายๆ ตัวได้บน Dimension ตัวอย่างเช่น ข้อมูลยอดขายประจำปีกับปฏิทินมาตรฐาน ถ้าผู้ใช้สร้างแอพพลิเคชัน OLAP ขึ้นมาจากดาต้าแวร์เฮาส์ที่สนับสนุน Relational Dimensional ในกรณีนี้แอพพลิเคชันที่สร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ ทำซ้ำ ดัดแปลง เผยแพร่ หรือแสวงหาผลประโยชน์ทางการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขึ้นก็สามารถใช้ Surrogate Key ซึ่งเป็นคุณสมบัติที่ดีของ OLAP เมื่อเทียบกับฐานข้อมูลแบบสัมพันธ์ได้ด้วย

เช่นเดียวกับฐานข้อมูลแบบสัมพันธ์ เซิร์ฟเวอร์ OLAP นั้นมี Physical และ Calculated Facts อย่างไรก็ตามเอนจิ้นในการวิเคราะห์ของเซิร์ฟเวอร์ OLAP นั้นสนับสนุน Calculated Facts ที่หลากหลายมากกว่า รวมทั้งยังสนับสนุนการกำหนดคีย์ซึ่งเป็นประโยชน์ต่อการสร้างแอพพลิเคชัน OLAP ของผู้ใช้ให้ทำได้ง่ายขึ้นอีกด้วย

ความแตกต่างที่สำคัญระหว่าง Dimension ของ OLAP กับ Relational Dimension พื้นฐานก็คือ บทบาทกลางของ Hierarchies ของ OLAP ใน Dimension ของ OLAP นั้นเป็นโครงสร้างที่เข้มงวดมากกว่าโครงสร้างของ Hierarchies รวมทั้ง Metadata ของ Cube Definition ก็ประกอบอยู่ในระดับต่างๆ ของ Hierarchical ซึ่งนับได้ว่าเป็นจุดแข็งหนึ่งของ OLAP ส่วนเครื่องมือในการคิวรีของ OLAP ก็มีความสามารถในการดึงข้อมูลเกี่ยวกับโครงสร้างแบบ Hierarchical จากเซิร์ฟเวอร์ OLAP รวมทั้งยังสามารถนำเสนอโครงสร้างเดียวกันนี้ให้กับผู้ใช้ในรูปแบบที่ทำความเข้าใจและใช้งานได้ง่ายอีกด้วย

ที่มีความสำคัญไม่แพ้กันก็คือ เครื่องมือ OLAP นั้นใช้ Hierarchies ในการกำหนด Aggregation เซิร์ฟเวอร์ OLAP จะบังคับความถูกต้องของการอ้างอิงระหว่างระดับใน Hierarchy โดยการใช้ Dimension ในขณะที่ระบบเก็บข้อมูลแบบสัมพันธ์ปกติจะไม่มีการเก็บและบังคับความถูกต้องของ Hierarchies โดยเฉพาะอย่างยิ่งถ้าผู้ใช้ใช้การออกแบบ Dimension แบบ Star แทนที่จะเป็น Snowflake

ความแตกต่างอีกประการระหว่าง OLAP กับฐานข้อมูลแบบสัมพันธ์ก็คือ การสร้าง ชนิดที่ต่างกัน (Different Type) ของ Dimension ที่มีการเปลี่ยนแปลง ใน Dimension ชนิด 2 (Type 2) ซึ่งเกี่ยวข้องกับการติดตามประวัติโดยการเพิ่มสมาชิกของ Dimension เข้ามาใหม่ ได้ง่าย เมื่อเทียบกับที่การสร้างระบบให้มีคุณสมบัติเดียวกันในฐานข้อมูลแบบสัมพันธ์ อย่างไรก็ตามในทางตรงกันข้าม Dimension ชนิด 1 (Type 1) ซึ่งเกี่ยวข้องกับการแก้ไขสถานะของประวัติโดยการอัพเดตนั้น กลับเป็นปัญหาที่ยากสำหรับผู้ใช้ใน OLAP ซึ่งถ้าคุณเคยสร้างและดูแล Aggregate Table บน Relational Schema โดยใช้ Dimension ชนิด 1 แล้ว คุณจะสามารถบอกได้ทันทีว่าปัญที่เกิดขึ้นคืออะไร นั่นคือเมื่อลูกค้าจำเป็นต้องแก้ไข Schema แล้วก็จะส่งผลให้ Aggregate จำเป็นต้องได้รับการแก้ไข นั่นหมายความว่าวิธีเดียวที่คุณทำได้ก็คือจะต้องลบและสร้าง Aggregate Table ขึ้นมาใหม่ทั้งหมด หรือไม่ก็จะต้องมีการแบ่งว่าส่วนใดของ Schema ที่ได้รับผลกระทบจาก Aggregate จากนั้นจึงลงมือแก้ไข ผลิตภัณฑ์เซิร์ฟเวอร์ OLAP กำลังหาทางแก้ไขปัญหานี้อยู่ ซึ่งถึงแม้ว่าขั้นตอนทั้งหมดสามารถทำได้โดยไม่จำเป็นต้องให้ผู้ใช้ทราบ อย่างไรก็ตาม ผลที่ตามมาก็คือการประมวลผลที่มากขึ้นซึ่งหมายถึงเวลา และทรัพยากรของระบบที่จะต้องเสียไปนั่นเอง สำหรับการเปลี่ยนแปลง Dimension ชนิด 1 นั้นนับว่าเป็นงานที่ใช้ทรัพยากรของเซิร์ฟเวอร์ OLAP ค่อนข้างมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**2.10.2 OLAP เทคโนโยลีสำหรับการวิเคราะห์ข้อมูลระดับองค์กร**

แนวโน้มของตลาด OLAP แสดงให้เห็นว่าผลิตภัณฑ์เซิร์ฟเวอร์ OLAP กำลังมีราคาลดลง มีประสิทธิภาพมากขึ้น สนับสนุนการขยายขนาด เพิ่มฟังก์ชันการทำงานโดยเฉพาะอย่างยิ่งการวิเคราะห์ข้อมูลทางธุรกิจ รวมทั้งยังมีส่วนขยายเพื่อทำงานร่วมกับระบบอื่น อย่างเช่น Data Mining ได้อีกด้วย แนวโน้มที่เกิดขึ้นกับ OLAP จะยังคงอยู่ต่อไปอีกหลายปีข้างหน้า ดังจะเห็นได้จากผู้ขายฐานข้อมูลรายใหญ่หลายรายได้เริ่มลงทุนและผลักดันผลิตภัณฑ์เซิร์ฟเวอร์ OLAP ของตนไปข้างหน้า ด้วยการเพิ่มลงในผลิตภัณฑ์เซิร์ฟเวอร์ฐานข้อมูลแบบสัมพันธ์ที่เป็นเรื่องของตัวเอง เซิร์ฟเวอร์ OLAP นั้นทำหน้าที่นำเสนอข้อมูลในรูปแบบ Dimensional ด้วยวิธีการทำความเข้าใจได้ง่าย ช่วยให้ผู้ใช้สามารถเข้าใช้ฟังก์ชันในการวิเคราะห์ข้อมูลขององค์กรได้อย่างมีประสิทธิภาพ

OLAP นั้นเป็นโมเดลการจัดโครงสร้างข้อมูลแบบ Dimension ที่ได้สืบทอดคุณสมบัติบางส่วนมาจากฐานข้อมูลแบบสัมพันธ์ โดยมีการเพิ่มความสามารถในการกำหนดความสัมพันธ์และการคำนวณที่มีประโยชน์ไว้บนเซิร์ฟเวอร์ การทำงานบนเซิร์ฟเวอร์มีข้อดีที่สำคัญก็ประสิทธิภาพของการคิวรีสูงขึ้นรวมทั้งยังสามารถใช้เครื่องมือในการคิวรีและวิเคราะห์ข้อมูลได้อย่างหลากหลาย คุณไม่ควรคิดว่าเซิร์ฟเวอร์ OLAP เป็นคู่แข่งของดาต้าแวร์เฮาส์ที่ทำงานบนของฐานข้อมูลแบบสัมพันธ์ หากแต่เป็นส่วนขยายที่ช่วยเพิ่มการใช้ประโยชน์จากเซิร์ฟเวอร์ฐานข้อมูลให้คุ้มค่ามากยิ่งขึ้น ผู้ใช้ต้องเข้าใจและทราบจุดเด่นของแต่ละผลิตภัณฑ์ จุดเด่นที่ถือว่าเป็นมาตรฐานซึ่งองค์กรต่าง ๆ ต่างก็เลือกใช้เทคโนโลยีOLAPคือ

1. OLAP มีส่วนติดต่อกับผู้ใช้ที่ง่ายสำหรับการบราวซ์ข้อมูล
2. OLAP มีประสิทธิภาพในการคิวรีข้อมูลที่โดดเด่น ทั้งนี้ขึ้นอยู่กับการจัดโครงสร้างของ
3. Aggregates และ Partition ด้วยว่าทำไว้ดีและเหมาะสมเพียงใด
4. โครงสร้างของ Dimension แบบ Parent-Child นั้นง่ายต่อการสร้างและใช้งาน
5. OLAP มีกฎที่กำหนดไว้บนเซิร์ฟเวอร์สำหรับรองรับ Measure แบบ Semiadditive และ Nonadditive

ในกรณี ระบบจัดการยอดคงเหลือของคลังสินค้า ขอหยิบยกตัวอย่าง เกี่ยวกับยอดคงเหลือของคลังสินค้าสำหรับเดือนมกราคมและกุมภาพันธ์นั้น ไม่ใช่ผลรวมของสินค้าในคลังทั้งหมดในเดือนมกราคมและกุมภาพันธ์ คุณควรแนะนำผู้ใช้ว่าไม่ควรรวมยอดคงเหลือของสินค้าคงคลังตลอดระยะเวลาที่ต้องการ แต่ปัญหาก็คือผู้ใช้จะจำได้หรือไม่ และผู้ใช้ทั้งหมดจะใช้วิธีของ Aggregation เดียวกัน อย่างเช่น การหายอดคงเหลือท้ายสุดหรือเฉลี่ยแทนหรือไม่ อย่างไรก็ตามระบบ OLAP สามารถจัดการปัญหานี้ได้โดยอัตโนมัติ

ระบบ OLAP สามารถใช้การคำนวณที่มีความซับซ้อนสูงที่เซิร์ฟเวอร์ได้ ในขณะที่ภาษา SQL นั้นมีข้อจำกัดเนื่องจากไม่ได้เป็นภาษาที่ออกแบบมาสำหรับการวิเคราะห์ รวมทั้งยังไม่เหมาะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับการสร้างรายงานอีกด้วย ในกรณีนี้คุณต้องการเซิร์ฟเวอร์ซึ่งทำหน้าที่วิเคราะห์ เซิร์ฟเวอร์นี้สนับสนุนการวิเคราะห์ทางสถิติ, อัลกอริทึมของ Data Mining หรือการคำนวณทางธุรกิจตามกฎพื้นฐาน อย่างเช่น การจัดสรรและการวิเคราะห์ทรัพยากรโดยไม่ต้องสนใจว่าข้อมูลและคิวรีเหล่านี้ถูกเก็บและมีวิธีการทำงานเป็นอย่างไร

คิวรีและการคำนวณที่มีการทำงานบนเซิร์ฟเวอร์ประสิทธิภาพสูงสามารถทำงานร่วมกับ Fact Table หรือ Cube หลายๆ ตัวได้พร้อมกัน ด้วยการใช้ข้อมูลจาก Fact Table หลายๆ ตัวนับเป็นงานที่ยากสำหรับฐานข้อมูลแบบสัมพันธ์ (นึกถึงการ Join ระหว่างตาราง ยิ่งมีจำนวนตารางมากเท่าใดก็จะทำให้โปรแกรมมีความซับซ้อนและประสิทธิภาพก็จะตกลงอย่างฉับพลัน) แต่สำหรับงานเดียวกับบนเซิร์ฟเวอร์OLAPนับเป็นสิ่งที่ทำได้ง่ายและมีประสิทธิภาพสูงอีกด้วย

การคำนวณสามารถกำหนดขึ้นมาเพื่อใช้งานเพียงครั้งเดียวหรือหลายครั้งก็คือถึงแม้จะมีจำนวนการคำนวณมากเท่าใด คุณก็สามารถเก็บไว้บนเซิร์ฟเวอร์ซึ่งช่วยให้การจัดการทำได้ง่ายกว่าและมีความยืดหยุ่นต่อผู้ใช้ในการนำมาใช้งานเพื่อประมวลผลข้อมูลที่ดีกว่า ด้วยเครื่องมือที่ใช้งานได้ง่าย ช่วยให้การเข้าใช้การวิเคราะห์ข้อมูลที่ซับซ้อนซึ่งถูกเก็บอยู่บนเซิร์ฟเวอร์ OLAP ไม่ใช่เรื่องยากอีกต่อไป ความสามารถนี้นับว่าเป็นจุดเด่นซึ่งไม่มีในฐานข้อมูลแบบสัมพันธ์ และแน่นอนว่าสำหรับผู้ใช้ที่ต้องการกำหนดการคำนวณที่ซับซ้องเองก็สามารถทำได้บนเซิร์ฟเวอร์ซึ่งผู้ใช้คนอื่นก็สามารถเข้าใช้งานได้ด้วยเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 3
# การวิเคราะห์ออกแบบระบบดาต้ามาร์ทสำหรับสินค้าคงคลัง กรณีศึกษา บริษัทคาโปรแลคตรัมไทย จำกัด(มหาชน)

เนื้อหาของบทนี้จะกล่าวถึงการหาความต้องการของผู้ใช้งานข้อมูล เพื่อนำมาเป็นพื้นฐานของการวิเคราะห์ด้าต้ามาร์ต สำหรับการบริหารสินค้าคงคลัง ด้วย แบบจำลองฐานข้อมูล ออกแบบด้าต้ามาร์ต ด้วยแบบจำลองข้อมูลเชิงมิติ(Dimensional Data Model) แสดงลักษณะเค้าร่างข้อมูลแบบสตาร์(Star Schema) การวิเคราะห์ระบบดาต้ามาร์ท โดยใช้ Use Case Diagram เพื่อหาความต้องการใช้งานใช้ Class Diagram เพื่อหาสิ่งที่สนใจจะนำมาใช้ภายในขอบเขตของระบบ ตลอดจนหาการมีปฏิสัมพันธ์กันของฟังก์ชันการทำงาน โดยใช้ Sequence Diagram

## 3.1 ข้อมูลพื้นฐานของกระบวนการทำงานปัจจุบัน การบริหารสินค้าคงคลัง

เนื่องจากการจัดการการผลิตในปัจจุบันจะต้องคำนึงถึงความสำคัญด้านคุณภาพเป็น หลักการบริการลูกค้าที่ดีเป็นส่วนหนึ่งของการสร้างคุณภาพที่ทำให้ลูกค้ามีความพึงพอใจสูงสุด ฉะนั้น บริษัทคาโปรแลคตรัมไทย จำกัด มหาชน จึงให้ความสำคัญต่อ การจัดการสินค้าคลังที่มีประสิทธิภาพซึ่งมีผลกระทบต่อผลกำไรจากการประกอบการโดยตรง เพื่อให้บริษัทสามารถตอบสนองความต้องการของลูกค้าในแต่ละช่วงเวลา ยังคงรักษาการผลิตให้มีอัตราคงที่สม่ำเสมอ ช่วยทำให้ธุรกิจได้ส่วนลดปริมาณจากการจัดซื้อครั้งละมาก ๆ สามารถป้องกันการเปลี่ยนแปลงราคา และที่สำคัญที่สุดคือ เพื่อให้กระบวนการผลิตสามารถดำเนินการต่อเนื่องอย่างราบรื่น โดยไม่มีการหยุดชะงักเพราะวัตถุดิบขาดจนเกิดความเสียหายแก่กระบวนการผลิต

### 3.1.1 การทำงานของระบบงานในปัจจุบัน

ด้วยสภาวะทางเศรษกิจประเภทอุตสาหกรรมด้านปิโตรเคมีของประเทศมีอัตราการเติบโตที่เพิ่มขึ้นทุก ๆ ปีอย่างต่อเนื่อง บริษัทคาโปรแลคตรัมไทย จำกัด (มหาชน) ได้ก่อตั้งขึ้นเมื่อวันที่ 31 ธันวาคม 2533 เพื่อเป็นแหล่งของการผลิตคาโปรแลคตรัมที่มีคุณภาพสูง ซึ่งถือว่าเป็นธาตุที่เป็นส่วนประกอบสำคัญของหลักของการทำไนลอนพลาสติก รวมทั้งรองรับความต้องการของอุตสาหกรรมสิ่งทอ ของประเทศไทยอีกด้วย เนื่องจากประเทศไทยพยายามพัฒนานวัตกรรมใหม่เพื่อลดวัตถุดิบนำเข้าผลิตภัณฑ์ปิโตรเคมีจากต่างประเทศ และเป็นหนึ่งในแรงขับเคลื่อนสำคัญที่จะนำอุตสาหกรรมปิโตรเคมีไปสู่อนาคต โดยผลักดันให้ประเทศไทยเป็นศูนย์กลางการผลิตชิ้นส่วนของรถยนต์ในภูมิภาค บริษัท จึงได้ร่วมมือกับพันธมิตร โดยมี บริษัท UBE Group จำกัด เพื่อร่วมกันพัฒนาศักยภาพทางด้านเทคโนโลยีชั้นสูงเกี่ยวกับกระบวนการผลิตธาตุคาโปรแลคตรัม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งถือว่าเป็นธาตุที่เป็นส่วนประกอบสำคัญของหลักของการทำไนลอนพลาสติก รวมทั้งรองรับความต้องการของอุตสาหกรรมสิ่งทอ ของประเทศไทยอีกด้วย เนื่องจากประเทศไทยพยายามพัฒนานวัตกรรมใหม่เพื่อลดวัตถุดิบนำเข้าผลิตภัณฑ์ปิโตรเคมีจากต่างประเทศ และเป็นหนึ่งในแรงขับเคลื่อนสำคัญที่จะนำอุตสาหกรรมปิโตรเคมีไปสู่อนาคต โดยผลักดันให้ประเทศไทยเป็นศูนย์กลางการผลิตชิ้นส่วนของรถยนต์ในภูมิภาค บริษัท จึงได้ร่วมมือกับพันธมิตร โดยมี บริษัท UBE Group จำกัด เพื่อร่วมกันพัฒนาศักยภาพทางด้านเทคโนโลยีชั้นสูงเกี่ยวกับกระบวนการผลิตของอุตสาหกรรม ตลอดจนการเอาเทคโนโลยีสารสนเทศเข้ามาใช้เพื่อสนับสนุน การทำงานต่าง ๆ เพื่อลดข้อผิดพลาดจากการทำงาน หรือ แม้แต่เสริมสร้างความถูกต้องต่อการทำงานมากขึ้น

3.1.2 สภาพปัญหาในปัจจุบัน

จากการศึกษาระบบงานต่าง ๆ ที่นำมาสนับสนุนการทำงาน และ การตัดสินใจแก่ผู้บริหารของบริษัทดาโปรแลคตรัมไทย จำกัด (มหาชน) พบว่า ปัญหาที่เกิดขึ้น เกิดจากรูปแบบของการเก็บ และ ใช้ข้อมูลจาก ฐานข้อมูลประจำวัน โดยผ่านทางโปรแกรม Microsoft Business Solution Navision ซึ่งเป็นระบบสารสนเทศระดับเอนทราไพส์ ซึ่งเข้าไปควบคุมกระบวนการทำงานในทุก ๆ แผนกที่เกี่ยวข้องเข้ามาสู่ฐานข้อมูลกลาง โดยสื่อสารผ่านเครือข่ายของบริษัท ข้อมูลดังกล่าวเกิดจากการทำงานประจำวันมีเป็นจำนวนมาก จากแหล่งข้อมูลที่หลากหลาย โดยแผนกเทคโนโลยีสารสนเทศ เป็นหน่วยงานที่รับผิดชอบเรื่องการทำกระบวนการแปลง และ จัดเก็บรายการข้อมูลลงในค้าต้าแวร์มาร์ต (Data Mart) นำมาวิเคราะห์โดยสร้างความสัมพันธ์ ระหว่างข้อมูลในมุมมองหลายมิติ โดยใช้เทคโนโลยี OLAP เป็นเครื่องมือในการวิเคราะห์มุมมองหลายมิติ แต่เนื่องจากผลลัพธ์ที่ได้จากการวิเคราะห์เป็นคำตอบที่ไม่ถูกต้อง ผู้บริหาร ไม่สามารถนำไปข้อมูลดังกล่าวไปใช้ให้เกิดประโยชน์ได้ สาเหตุดังกล่าวมาจาก การจัดเก็บข้อมูลที่เก็บในดาต้ามาร์ต ไม่ได้ผ่านกระบวนการแปลง สกัด และ การย้ายข้อมูลที่ถูกต้อง หากจะนำข้อมูลดังกล่าวมาใช้เพื่อสนับสนุนการตัดสินใจ หรือ วางแผนแนวโน้มต่าง ๆ เพื่อสร้างแผนกลยุทธ์ในการขับเคลื่อนทางธุรกิจในเรื่องของ การบริหารสินค้าคงคลัง และ การทำงานในด้านอื่น ๆ อาจจะก่อให้เกิดผลกระทบต่าง ๆ จากการทำงานในปัจจุบัน และ อนาคต ซึ่งปัญหาดังกล่าว สามารถแจกแจงได้ดังต่อไปนี้

1. การจัดเก็บข้อมูลของบริษัทต่างๆ ในกลุ่มโรงงานเป็นข้อมูลที่มีรูปแบบแตกต่างกัน
2. การจัดเก็บข้อมูลคลังสินค้าไม่ได้ผ่านกระบวนการ ETL และ การทำความสะอาดข้อมูล ส่งผลให้ข้อมูลที่จัดเก็บอยู่ในดาต้าแวร์มาร์ต และ ข้อมูลที่ได้จากการใช้เทคโนโลยี OLAP ซึ่งเป็นข้อมูลที่ไม่ถูกต้อง ยังส่งผลต่อผลลัพธ์ จากการวิเคราะห์ข้อมูลด้วย
3. เนื่องจากข้อมูลที่ได้จากการเก็บในดาต้าแวร์เฮาส์เป็นข้อมูลที่ไม่ครอบคลุมในส่วนของสินค้าคงคลัง ซึ่งถูกนำมาใช้เป็นปัจจัยในการสนับสนุนการตัดสินใจ ทำให้ไม่สามารถตอบสนองความต้องการของผู้บริหารได้อย่างถูกต้อง และ ครบถ้วน
4. การใช้เทคโนโลยี OLAP เชิงวิเคราะห์คงสามารถทำได้ยากเนื่องจากระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฐานข้อมูลที่ใช้อยู่ในปัจจุบัน มีความซับซ้อน และ มีขนาดใหญ่ รวมทั้งไม่มี คาต้ามาร์ตใน ส่วนของการสินค้าคงคลัง

5. เวลาในการเรียกดูข้อมูลจากคาต้าแวร์เฮาส์ ใช้เวลานาน

6. ผู้พัฒนาระบบไม่มีความเชี่ยวชาญในกระบวนการพัฒนาระบบ Business Intelligence

จากปัญหาทั้งหมดที่ได้กล่าวมาแล้วในข้างต้น บริษัท คาโปรแลคตรัมไทย จำกัด (มหาชน) ได้พยายามหาแนวทางในการแก้ปัญหา และ พัฒนาประสิทธิภาพโดยการวิเคราะห์ ออกแบบ และพัฒนาคาต้ามาร์ต รวมทั้งนำเอาเทคโนโลยี OLAP มาใช้เพื่อการวิเคราะห์ความสัมพันธ์ของข้อมูลของสินค้าคงคลัง ในแง่ของการมีสินค้าไว้บริการลูกค้าในปริมาณที่เพียงพอ และ ทันต่อความต้องการของลูกค้า ตลอดจนการจำแนกสินค้าคงคลังออกเป็นประเภทต่าง ๆ สำหรับการควบคุมสินค้าคงคลัง รวมทั้งเพื่อให้ดูแนวโน้มต่าง ๆ ที่จะสามารถลดระดับการลงทุนในสินค้าคงคลังให้ต่ำที่สุดเพื่อทำให้ต้นทุนการผลิตต่ำโดยอาศัยกระบวนการพัฒนาตามขั้นตอนพัฒนาระบบสารสนเทศเพื่อการสนับสนุนการตัดสินใจให้เป็นไปอย่างถูกต้อง รวมทั้งทำการศึกษาปัญหาที่เกิดขึ้น และวิเคราะห์ข้อมูลที่มีอยู่ เพื่อหาความต้องการที่แท้จริงของผู้บริหาร จนได้เป็นข้อมูลที่มีคุณค่าสามารถนำมาใช้งานและประเมิณผลการใช้งานได้ในอนาคต

## 3.2 ความต้องการ ของการใช้งานระบบ

จากการศึกษาระบบงานของบริษัทคาโปรแลคตรัมไทย จำกัด(มหาชน) ทำให้ทราบถึงวัตถุประสงค์ ของการพัฒนาระบบคาต้ามาร์ต สำหรับสินค้าคงคลัง เพื่อตอบสนองความต้องการข้อมูล เพื่อนำมาวิเคราะห์ ดังต่อไปนี้

1. ต้นทุนของสินค้าแต่ละรายการ และในช่วงเวลาต่าง ๆ (Cost Among)
2. ปริมาณสินค้าที่ต้องการแต่ละรายการ แต่ละ ไซท์งาน ตามช่วงเวลาต่าง ๆ (Quantity)
3. ปริมาณสินค้าที่ยังคงมีอยู่ในแต่ละ Location ตามช่วงเวลาต่าง ๆ (Remaining quantity)
4. แนวโน้มปริมาณสินค้าที่ต้องการในใบสั่งซื้อ ตามช่วงเวลาต่าง ๆ (Invoiced quantity)
5. ต้นทุนสินค้าต่อหน่วยในช่วงเวลาต่าง ๆ ตามแต่ละไซท์งาน (Unit cost)
6. ต้นทุนสินค้าเฉลี่ยแต่ละรายการสินค้า ตามไซท์งานต่าง ๆ (Avgcost)
7. รายการที่มีปริมาณของสินค้าซึ่งต่ำที่สุดของแต่ละไซท์ (Minimum Quantity)
8. รายการที่มีปริมาณของสินค้าซึ่งสูงที่สุดของแต่ละไซท์ (Maximum Quantity)
9. รายการที่มีต้นทุนรวมของสินค้าซึ่งต่ำที่สุดของแต่ละไซท์ (Minimum Cost Amount)
10. รายการที่มีต้นทุนรวมของสินค้าซึ่งสูงที่สุดของแต่ละไซท์ (Maximum Cost Amount)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ด้วยสภาวะการแข่งขันทางธุรกิจในยุคเทคโนโลยีดอทคอมทวีความรุนแรงขึ้นทุกขณะ ความสำเร็จขององค์กรธุรกิจโดยส่วนใหญ่ มักขึ้นกับความสามารถในการวิเคราะห์ข้อมูลการประเมิน สถานการณ์ และการตัดสินใจของผู้บริหาร ด้วยเหตุนี้การได้รับข้อมูลที่อัพเดท รวดเร็ว และทันการณ์จะช่วยให้ผู้บริหารสามารถพลิกสถานการณ์ให้เกิดความได้เปรียบในการแข่งขันได้ ทั้งในแง่ความพึงพอใจของลูกค้า ความเข้มแข็งด้านการขาย และระบบงานภายในที่มีประสิทธิภาพ

ดังนั้น บริษัท คาโปรแลคตรัมไทย จำกัด (มหาชน) ได้พยายามนำเอา ระบบธุรกิจอัจฉริยะ (Business Intelligence) ซึ่งระบบที่ช่วยในการวิเคราะห์ และการตัดสินใจทางธุรกิจของผู้บริหาร เข้ามาประยุกต์ใช้ กับการบริหารคงคลัง โดยที่เทคโนโลยี BI สามารถเรียกดูข้อมูลที่เก็บไว้ในดาต้ามาร์ตขององค์กรอาทิ ข้อมูลการมีอยู่ของสินค้าคงคลัง ข้อมูลการรับสินค้าตลอดจนข้อมูลเกี่ยวกับลูกค้า ดึงข้อมูลที่เกี่ยวข้องกับสิ่งที่ผู้บริหาต้องการมาแสดงในรูปแบบของรายงานกราฟตารางและภาพกราฟิก โดยเป็นการวิเคราะห์ข้อมูลด้านสถิติ การหา ความสัมพันธ์ของข้อมูล หรือการจำลองสถานการณ์ต่างๆ สามารถแสดงข้อมูลเป็นภาพรวม และเจาะลึกเฉพาะส่วนได้ เพื่อนำไปสู่การค้นหาสาเหตุของปัญหาได้ตรงจุด การแก้ไขที่ตรงประเด็น และการวางแผนอนาคตได้ถูกทางสอดคล้องกับ KPI ขององค์กร

## 3.3 การวิเคราะห์และออกแบบดาต้าแวร์มาร์ตในส่วนของการบริหารสินค้าคงคลัง

การออกแบบฐานข้อมูลสำหรับคลังข้อมูลอาศัยหลักการของ Kimball ซึ่ง เรียกว่าระเบียบวิธี 9 ขั้น หรือ Nine-Step Methodology โดยเลือกออกแบบจากส่วนย่อยที่แสดงถึงแต่ละระบบงานขององค์กร หรือเรียกอีกอย่างหนึ่งว่าดาต้ามาร์ท (data mart) โดยเมื่อออกแบบแต่ละส่วนสำเร็จแล้ว จึงนำมารวมกันเป็นคลังข้อมูล ขององค์กรในขั้นสุดท้าย ในการพัฒนาระบบดาต้ามาร์ต จึงเลือกออกแบบในส่วนของสินค้าคงคลัง การดำเนินการมีรายละเอียดดังนี้

### 3.3.1 การวิเคราะห์ดาต้าแวร์เฮาส์ส่วนการบริหารสินค้าคงคลัง

1. กระบวนการดำเนินการทางธุรกิจ:

พิจารณาการมีอยู่ของสินค้าในคงคลัง

2. ลักษณะของการดำเนินงาน (Granularity):

การนำสินค้าคงคลังไปใช้ผลิตเป็นผลิตภัณฑ์ทุก ๆวัน ดำเนินการโดย ที่เก็บสินค้าคงคลัง

3. มุมมองที่ต้องการมองหลายมิติ (Dimension):

Date, Item, Location, Site

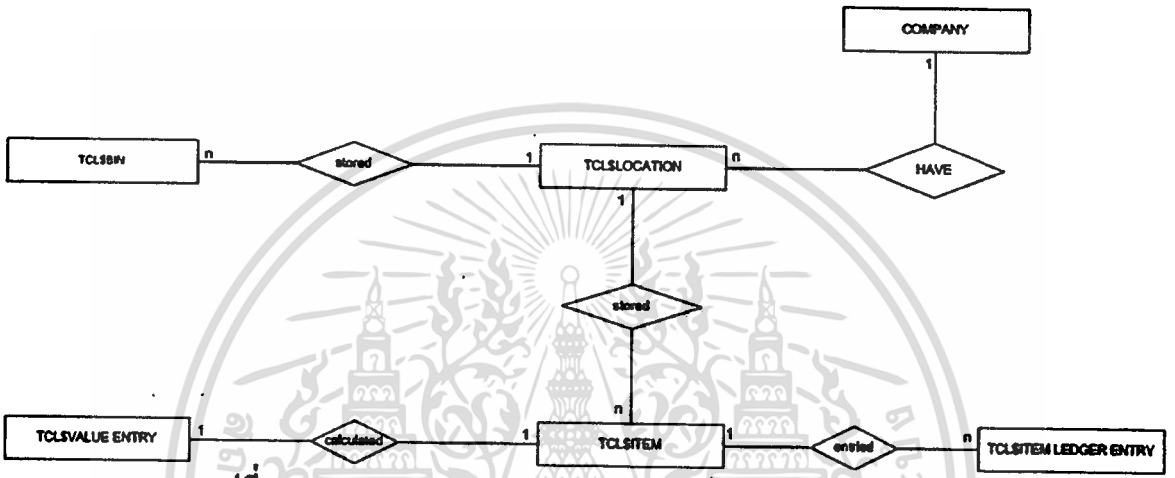
4. ความจริงที่ใช้ตอบคำถามเชิงธุรกิจ (Fact):

Quantity on hand, Remain Quantity, Invoiced Quantity, Unit Cost, Average Cost, Cost Amount

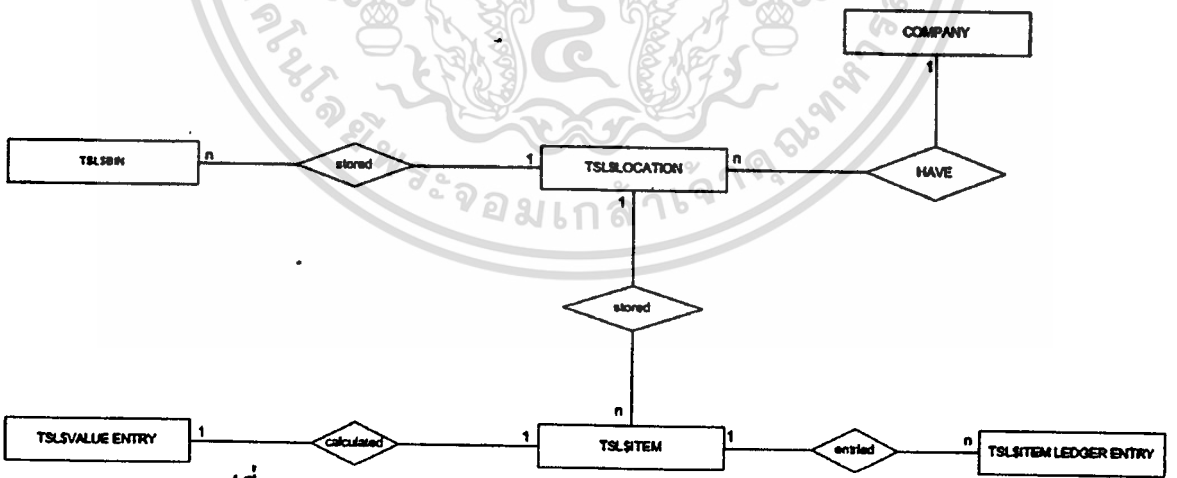
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.2 การวิเคราะห์ข้อมูลนำเข้าจากที่พักข้อมูล (Data Stage)

การวิเคราะห์ข้อมูลจากฐานข้อมูลของบริษัท คาโปรแลคตรัมไทย จำกัด (มหาชน) ซึ่งประกอบไปด้วยข้อมูลต่าง ๆ จาก 3 บริษัทได้แก่ TCL, UNT และ TSL สามารถแสดงความสัมพันธ์ของข้อมูลโดยใช้แบบจำลองความสัมพันธ์ข้อมูลดังรูปต่อไปนี้

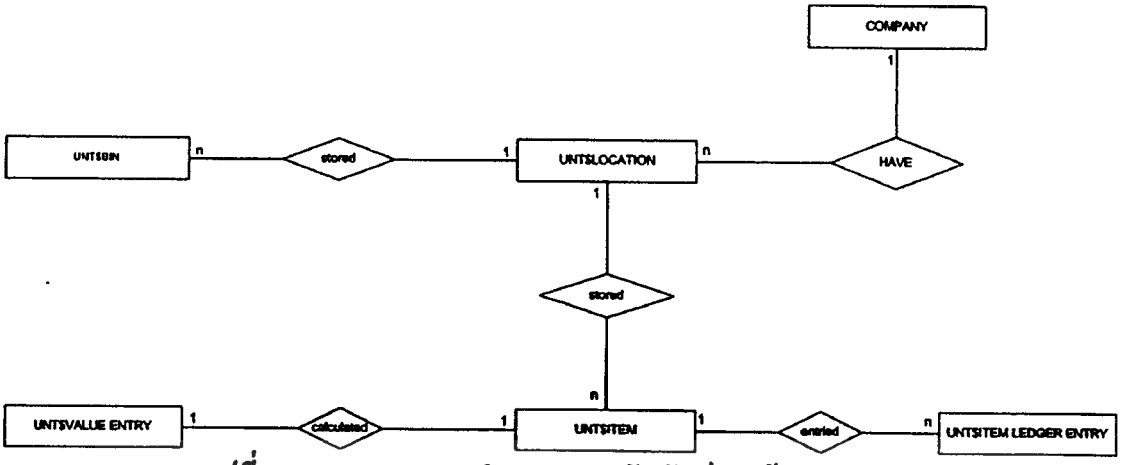



**รูปที่ 3.1 ภาพแสดงแบบจำลองความสัมพันธ์ของข้อมูล บริษัท TCL**




**รูปที่ 3.2 ภาพแสดงแบบจำลองความสัมพันธ์ของข้อมูล บริษัท UNT**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 3.3** ภาพแสดงแบบจำลองความสัมพันธ์ของข้อมูล บริษัท UNT

### 3.3.3 การออกแบบระบบดาต้ามาร์ตสำหรับสินค้าคงคลังด้วยแบบจำลองเชิงมิติ

โครงร่างแบบดาว (Star Schema) และ อธิบายแบบจำลองโดยการแสดงรายละเอียดของ Dimension ต่างๆ ในลักษณะข้อมูลของข้อมูล (Meta data) ซึ่งสามารถอธิบาย ได้ดังต่อไปนี้

**ตารางที่ 3.1** ตารางแสดง Meta Data: Item Dimension

| Data Item | Data Type | Brief Description | Note |
|---|---|---|---|
| ITEMID | varchar(30) | รหัสของสินค้าตามรายการ | PK |
| ITEM_NO | varchar(30) | รหัสของสินค้า | |
| DESCRIPTION | varchar(220) | ชื่อของสินค้า | |
| BASE UNIT OF MEASURE | varchar(10) | หน่วยสินค้า | |
| PROD_POSTING_GROUP | varchar(20) | กลุ่มของสินค้า | |
| POSTING_GROUP_DES | varchar(50) | ชื่อกลุ่มสินค้า | |
| BIN CODE | varchar(20) | รหัสที่พักสินค้า | |

**ตารางที่ 3.2** ตารางแสดงคำอธิบายข้อมูล Date Dimension

| Data Item | Data Type | Brief Description | Note |
|---|---|---|---|
| DATEID | varchar(20) | รหัสวันที่ตามรายการ | PK |
| DATE KEY | varchar(20) | รหัสวันที่ | |
| DAY OF WEEK | varchar(20) | ชื่อวัน | |
| MONTH | varchar(20) | ชื่อเดือน | |
| YEAR | Smallint | ปี | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 3.2** (ต่อ)

| Data Item | Data Type | Brief Description | Note |
|---|---|---|---|
| QUARTER | Tinyint | ไตรมาส | |
| DAY OF YEAR | Int | จำนวนวันใน 1 ปี | |
| HOLIDAY | char(5) | วันหยุดราชการและ วันหยุดนักขัตฤกษ์ | |
| WEEKEND | char(5) | วันหยุดสัปดาห์ | |
| YEARMONTH | varchar(20) | แสดงข้อมูลปี-เดือน | |
| WEEK OF YEAR | Int | ลำดับที่ของอาทิตย์ ในปีนั้น | |

**ตารางที่ 3.3** ตารางแสดงคำอธิบายข้อมูล Site Dimension

| Data Item | Data Type | Brief Description | Note |
|---|---|---|---|
| SITE_ID | char(5) | รหัสไซท์งาน | PK |
| SITE_NAME | varchar(50) | ชื่อของไซท์งาน | |
| HEADADDRESS | varchar(20) | ที่อยู่ของสำนักงานใหญ่ | |
| HEADPHONE_NO | varchar(20) | โทรศัพท์ของสำนักงานใหญ่ | |
| HEADFAX_NO | varchar(20) | แฟกซ์ของสำนักงานใหญ่ | |
| RYADDRESS | varchar(170) | ที่อยู่ของสำนักงานระยอง | |

**ตารางที่ 3.4** ตารางแสดงคำอธิบายข้อมูล Location Dimension

| Data Item | Data Type | Brief Description | Note |
|---|---|---|---|
| LOCATIONID | Varchar(20) | รหัสที่อยู่สินค้าตามรายการ | PK |
| LOCATION_CODE | varchar(20) | รหัสที่อยู่สินค้า | |
| LOCATION_NAME | varchar(50) | ชื่อที่อยู่สินค้า | |
| ADDRESS | varchar(170) | ที่อยู่ | |
| PHONE_NO | varchar(30) | เบอร์ติดต่อที่อยู่ของสินค้า | |
| FAX_NO | varchar(30) | เบอร์แฟกซ์ที่อยู่สินค้า | |
| E-MAIL | varchar(80) | E-mail | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.5 ตารางแสดงคำอธิบายข้อมูล Fact Table

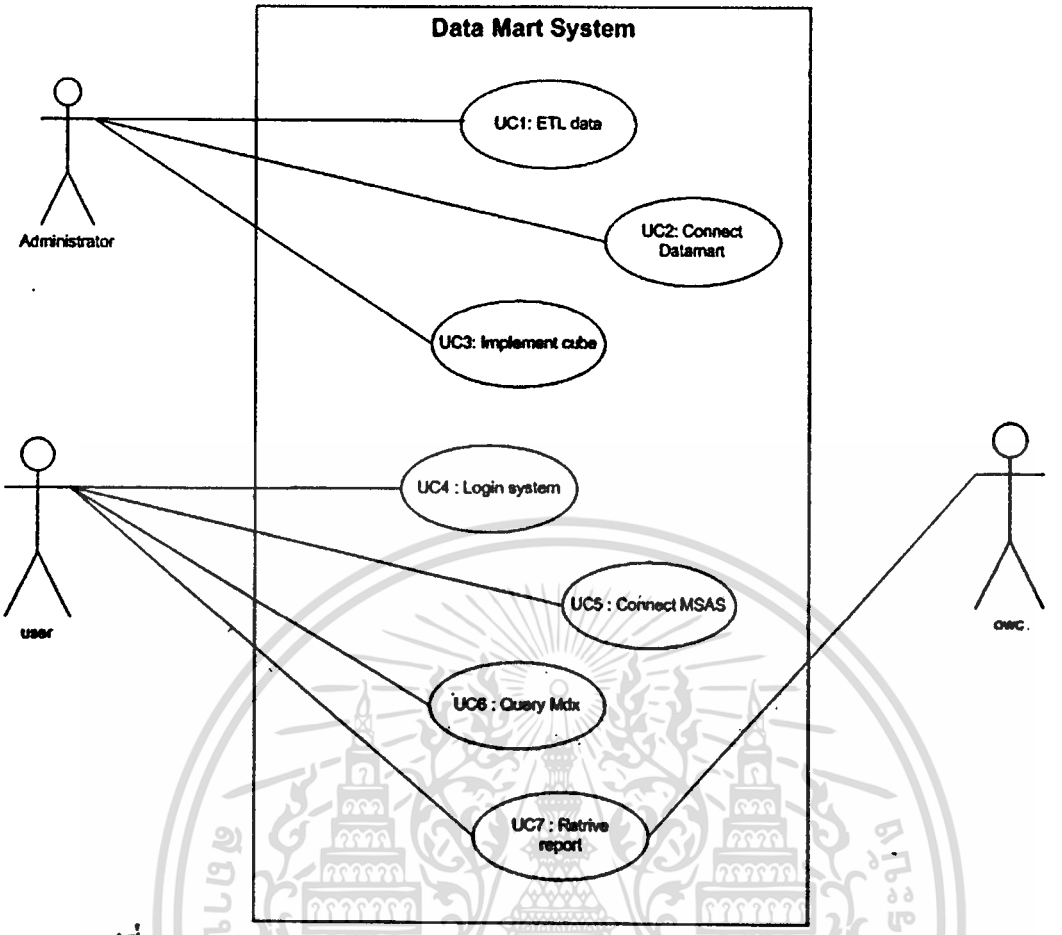
| Data Item | Data Type | Brief Description | Note |
|---|---|---|---|
| DATEID | varchar(20) | รหัสวันที่ตามรายการ | FK |
| ITEMID | varchar(-0) | หมายเลขของสินค้า | FK |
| LOCATIONID | varchar(20) | รหัสที่อยู่สินค้า | FK |
| SITE_ID | char(5) | รหัสของไซท์งาน | FK |
| QUANTITY | Decimal(32, 20) | ปริมาณที่ต้องการสินค้า | |
| REMAINING_QUANTITY | Decimal(32, 20) | ปริมาณที่ถือครองสินค้า | |
| INVOICE_QUANTITY | Decimal(32, 20) | ปริมาณที่ต้องการในใบสั่งซื้อ | |
| UNIT_COST | Decimal(32, 20) | ต้นทุนสินค้าต่อหน่วย | |
| LASTED_COST | Decimal(32, 20) | ต้นทุนสินค้าราคาสุดท้าย | |
| AVGCOST | Decimal(32, 20) | ต้นทุนเฉลี่ย | |
| COST_AMOUNG | Decimal(32, 20) | จำนวนรวมของต้นทุนสินค้า | |

## 3.4 ความต้องการเชิงฟังก์ชัน (Functional Requirements)

### 3.4.1 Use Case Diagram

พิจารณาความต้องการ และกระบวนการทำงานของระบบดาต้ามาร์ทรวมทั้งใช้เทคโนโลยี OLAP นำมาศึกษา วิเคราะห์ และหาแนวโน้มของการจัดการสินค้า โดยสามารถอธิบายการทำงานของระบบค้าตามาร์ตโดยพิจารณาจาก *มุมมองของผู้ใช้ระบบ (User's View)* อาศัยภาษา UML มาอธิบายถึงการปฏิสัมพันธ์ระหว่างระบบงานกับสิ่งที่อยู่นอกระบบ ด้วย Use Case Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 3.4** แสดง Use Case Diagram แสดงกระบวนการทำงานทำงานของระบบค้าตามาร์ท

### 3.4.2 Use Case Description

**ตารางที่ 3.6** แสดง Use Case Description สำหรับ ฟังก์ชัน ETL

| Use Case Name | UC1: ETL data |
|---|---|
| Brief Description | Use case นี้ใช้ในการอธิบายว่า Administrator สกัดข้อมูลจากแหล่งข้อมูลต่าง ๆ ได้อย่างไร โดยการเลือกข้อมูล Table และ Attribute จากนั้นถ่ายโอนข้อมูลจากแหล่งเพื่อทำความสะอาดโดยจัดรูปแบบข้อมูลให้อยู่เป็นมาตรฐานเดียวกัน เมื่อได้รับการตรวจสอบว่าถูกต้องจาก MSAS แล้วจึงยอมให้ข้อมูลไปเก็บที่ปลายทางของข้อมูล หรือ ที่พักข้อมูล |
| Primary Actor | Administrator |
| Extending | - |
| Stakeholder | Microsoft server analysis service, Sql Server 2005 |
| Preconditions | 1) Administrator ต้องได้รับอนุญาตเข้าใช้ข้อมูลของบมจ. คาโปรแลคตรัมไทย จำกัด |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 3.6 (ต่อ)**

| Use Case Name | UC1: ETL data |
|---|---|
| Alternative Flows | 2a) ข้อมูลไม่สามารถโอนย้ายเพื่อจัดรูปแบบได้<br>2a1) ระบบแสดงข้อความเตือน<br>2a2) กลับไปทำข้อ 1<br>2b) ข้อมูลบาง record เกิด config<br>2a1) ระบบแสดงข้อความเตือน<br>2a2) กลับไปทำยังข้อ 1<br>5) ระบบไม่สามารถทำงานได้ 100%<br>5a1) ระบบแสดงข้อความเตือน<br>5a2) กลับไปดูตั้งแต่ ข้อ 1 2 3 และ 4<br>5a3) ทำข้อที่ 5 อีกครั้ง |
| Post conditions | 1) ระบบมีข้อมูลอยู่ในที่พักข้อมูล |

**ตารางที่ 3.7 แสดง Use Case Description สำหรับ ฟังก์ชัน Connect Data mart**

| Use Case Name | UC2: Connect Data mart |
|---|---|
| Brief Description | Use case นี้ใช้ในการอธิบายว่า Administrator เข้ามาสร้างการติดต่อจาก Data mart ได้อย่างไร โดยการป้อนชื่อ Server ชื่อฐานข้อมูล และ ชื่อ Table ที่ต้องการใช้จากนั้นคลิก next เมื่อได้รับการตรวจสอบว่าถูกต้องจาก Database Server แล้วจึงจะยอมสร้างการติดต่อระหว่าง administrator และ Data mart ได้ |
| Primary Actor | Administrator |
| Extending | - |
| Stakeholder | Microsoft Sql Analysis Service |
| Preconditions | 1) Administrator ต้องได้รับอนุญาตเข้าใช้ข้อมูลของบมจ. คาโปรแลคตรัมไทย จำกัด |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 3.7** (ต่อ)

| | |
|---|---|
| Basic Flows | 1) Administrator ป้อนข้อมูลการติดต่อกับ Data Mart<br>• ชื่อ Server<br>• ชื่อฐานข้อมูล<br>• ชื่อตารางข้อมูล<br>• ชื่อของการติดต่อ<br>3) Administrator กำหนดสิทธิ์ในการติดต่อ<br>4) MSAS Execute package สร้างการติดต่อฐานข้อมูล |
| Alternative Flows | 4a) MSAS ไม่สามารถสร้างการติดต่อฐานข้อมูลได้<br>4a1) ระบบแสดงข้อความเตือน<br>4a2) กลับไปทำข้อ 1 ข้อ 2 ข้อ 3 และ 4 ได้ตามลำดับ |
| Postconditions | 1) Administrator สามารถติดต่อฐานข้อมูลได้สำเร็จ |

**ตารางที่ 3.8** แสดง Use Case Description สำหรับ ฟังก์ชัน Implement cube

| | |
|---|---|
| Use Case Name | UC3: Implement cube |
| Brief Description | Use case นี้ใช้ในการอธิบายว่า Administrator จะสร้างการติดต่อ และ จะสร้างคิวบ์ขึ้นมาได้อย่างไร โดยสร้างคิวบ์และโครงสร้างคิวบ์จากแหล่งข้อมูลจากนั้นตรวจหา dimension และ Fact Table เมื่อได้รับการตรวจสอบว่า Dimension, Fact Table และ Measure จาก MSAS ว่าถูกต้องแล้วจึงดำเนินการสร้างคิวบ์ |
| Primary Actor | Administrator |
| Extending | - |
| Stakeholder | Microsoft server analysis service, Sql Server 2005 |
| Preconditions | 1) Administrator ต้องได้รับอนุญาตเข้าใช้ข้อมูลของบมจ. คาโปรแลคตรัมไทย จำกัด |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 3.8 (ต่อ)**

| | |
|---|---|
| Basic Flows | 1) Administrator ป้อนข้อมูล<br>● ชื่อของแหล่งข้อมูล<br>● ประเภทของโครงสร้างคิวบ์<br>2) Administrator กำหนดตารางข้อมูลที่ต้องการใช้งาน<br>● Dimension Table<br>● Fact Table<br>3) Administrator กำหนดส่วนที่ใช้วัดผล<br>● Measures<br>4) Administrator คลิก Execute เพื่อให้ MSAS ประมวลผลการทำงาน<br>5) ระบบแสดง meta data, hierarchies |
| Alternative Flows | 5a) ระบบไม่สามารถแสดง Meta data, hierarchies<br>5a1) ระบบแสดงข้อความเตือน<br>5a2) กลับไปทำข้อ 1 2 3 และ 4 ตามลำดับ |
| Post conditions | 1) ระบบแสดงส่วนประกอบของคิวบ์ และคิวบ์ ที่ใช้งาน |

**ตารางที่ 3.9** ตารางแสดง ฟังก์ชัน Login System

| | |
|---|---|
| Use Case Name | UC1: Login system |
| Brief Description | Use case นี้ใช้ในการอธิบายว่า User เข้ามาใช้งานแอพพลิเคชั่นได้อย่างไร โดยการป้อนข้อมูลชื่อผู้ใช้ และ รหัสผ่าน จากนั้นคลิก login เมื่อได้รับการตรวจสอบว่าถูกต้องจาก Database Server แล้วจึงจะยอมให้เข้าใช้งานระบบ |
| Primary Actor | User ผู้มีสิทธิ์ |
| Extending | - |
| Stakeholder | Administrator, Manager, IT cooperation |
| Preconditions | 1) User ต้องเป็นพนักงานของ บมจ. ไทยคาโปรแลคตรัม<br>2) User ต้องได้รับสิทธิ์ในการเข้าใช้งาน |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 3.9** (ต่อ)

| Use Case Name | UC1: Login system |
|---|---|
| Basic Flows | 3) User ป้อนข้อมูลการเข้าใช้งานระบบ<br>• ชื่อผู้ใช้<br>• รหัสผ่าน<br>4) ระบบสั่งให้ Database Server ตรวจสอบข้อมูลผู้ใช้ว่าถูกต้อง และ เป็นผู้ใช้ประเภทใดของระบบ<br>5) ระบบแสดงฟังก์ชันการทำงาน |
| Alternative Flows | 3a) ข้อมูลผู้ใช้ไม่ถูกต้อง<br>3a1) ระบบแสดงข้อความเตือน<br>3a2) กลับไปทำข้อ 1 |
| Post conditions | 1) ผู้ใช้ใช้ฟังก์ชันการทำงานในระบบได้ |

**ตารางที่ 3.10** ตารางแสดง Use Case connect MSAS

| Use Case Name | UC2: Connect MSAS |
|---|---|
| Brief Description | Use case นี้ใช้ในการอธิบายว่า User เข้ามาติดต่อ เครื่องมือวิเคราะห์ข้อมูล (Microsoft Analysis Service) ได้อย่างไร โดยการป้อนข้อมูล Database server, Catalog, Cube จากนั้นคลิก Establish Connection เมื่อได้รับการตรวจสอบว่าถูกต้องจาก MSAS แล้วจึงจะสร้างการติดต่อให้ |
| Primary Actor | User ผู้มีสิทธิ์ |
| Extending | - |
| Stakeholder | Administrator, Manager, IT cooperation |
| Preconditions | 1) User ต้องเป็นพนักงานของ บมจ. ไทยคาโปรแลคตรัม<br>2) User ต้องได้รับสิทธิ์ในการเข้าใช้งาน<br>3) User login เข้าระบบมาเรียบร้อยแล้ว |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 3.10** (ต่อ)

| Use Case Name | UC2: Connect MSAS |
|---|---|
| Basic Flows | 1) User ป้อนข้อมูลการติดต่อ MSAS<br>• Database server location<br>• Catalog<br>• Cube<br>2) ระบบสร้างใช้คำสั่ง Connection String เพื่อสร้างการติดต่อ Database และ MSAS ที่ผู้ใช้ต้องการเรียกใช้บริการ<br>3) ระบบแสดงคิวบ์ Dimension, Measure |
| Alternative Flows | 2a) ข้อมูลผู้ใช้กรองไม่ครบถ้วน<br>2a1) ระบบแสดงข้อความเตือน<br>2a2) กลับไปทำข้อ 1<br>3a) ระบบไม่สามารถสร้างการติดต่อ MSAS<br>3a1) ระบบแสดงข้อความเตือน<br>3b2) ตรวจเช็ค Microsoft Analysis Server และ Microsoft SQL Server 2005<br>3b3) กลับไปทำข้อ 1 |
| Post conditions | 1) ผู้ใช้สามารถดู และ ควบคุม คิวบ์ได้ |

**ตารางที่ 3.11** ตารางแสดง Use Case Description สำหรับฟังก์ชัน Query MDX

| Use Case Name | UC3: Query MDX |
|---|---|
| Brief Description | Use case นี้ใช้ในการอธิบายว่า Admin สามารถสร้าง query โดยผ่านทาง API ของระบบได้อย่างไร โดยการป้อนคำสั่งภาษา mdx จากนั้นคลิก Execute เมื่อได้รับการตรวจสอบว่าภาษาที่เรียกดูข้อมูลจากคิวบ์ เป็น syntax ที่ถูกต้องระบบ run คำสั่งและแสดงผล |
| Primary Actor | Administrator |
| Extending | - |
| Stakeholder | Manager, IT cooperation |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 3.11 (ต่อ)**

| Use Case Name | UC3: Query MDX |
|---|---|
| Preconditions | 1) User ต้องเป็นพนักงานของ บมจ. ไทยคาโปรแลคตรัม<br>2) User ต้องได้รับสิทธิ์เป็น Administrator<br>3) User login เข้าระบบมาเรียบร้อยแล้ว |
| Basic Flows | 1) User คลิกที่ เขียนคำสั่ง mdx<br>2) ระบบปรากฏหน้าต่างสำหรับการเขียนคำสั่งเพื่อเรียกดูข้อมูลจากคิวบ์<br>3) ระบบตรวจสอบไวยกรณ์ของภาษา mdx<br>4) ค้นข้อมูลตามคำสั่ง และ แสดงผล |
| Alternative Flows | 3a) ผู้ใช้เขียนคำสั่งไม่ถูกต้อง<br>3a1) ระบบแสดงข้อความเตือน กลับไปทำข้อ 2 |
| Post conditions | 1) ผู้ใช้ดูรายงานข้อมูลได้ |

**ตาราง 3.12 ตารางแสดงฟังก์ชันการทำงาน Retrieve report**

| Use Case Name | UC4: Retrieve report |
|---|---|
| Brief Description | Use case นี้ใช้ในการอธิบายว่า User เข้ามาดูรายงาน เพื่อวิเคราะห์ข้อมูล (Microsoft Analysis Service) ได้อย่างไร โดยการเลือกรายงานที่ต้องการดู |
| Primary Actor | User ผู้มีสิทธิ์ |
| Extending | - |
| Stakeholder | Administrator, Manager, IT cooperation |
| Preconditions | 1) User ต้องเป็นพนักงานของ บมจ. ไทยคาโปรแลคตรัม<br>2) User ต้องได้รับสิทธิ์ในการเข้าใช้งาน<br>3) User login เข้าระบบมาเรียบร้อยแล้ว |
| Basic Flows | 1) User เลือกรายงานที่ต้องการจะดู<br>2) ระบบสร้างการติดต่อกับ excel OWC version 1.0<br>3) Excel OWC upload แสดง Pivot Table บนเว็บบราวเซอร์ |
| Alternative Flows | - |
| Post conditions | 1) ผู้ใช้สามารถดู Pivot Table และ Graph ได้<br>2) ผู้ใช้สามารถ drill down, drill up เพื่อดูรายละเอียด |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

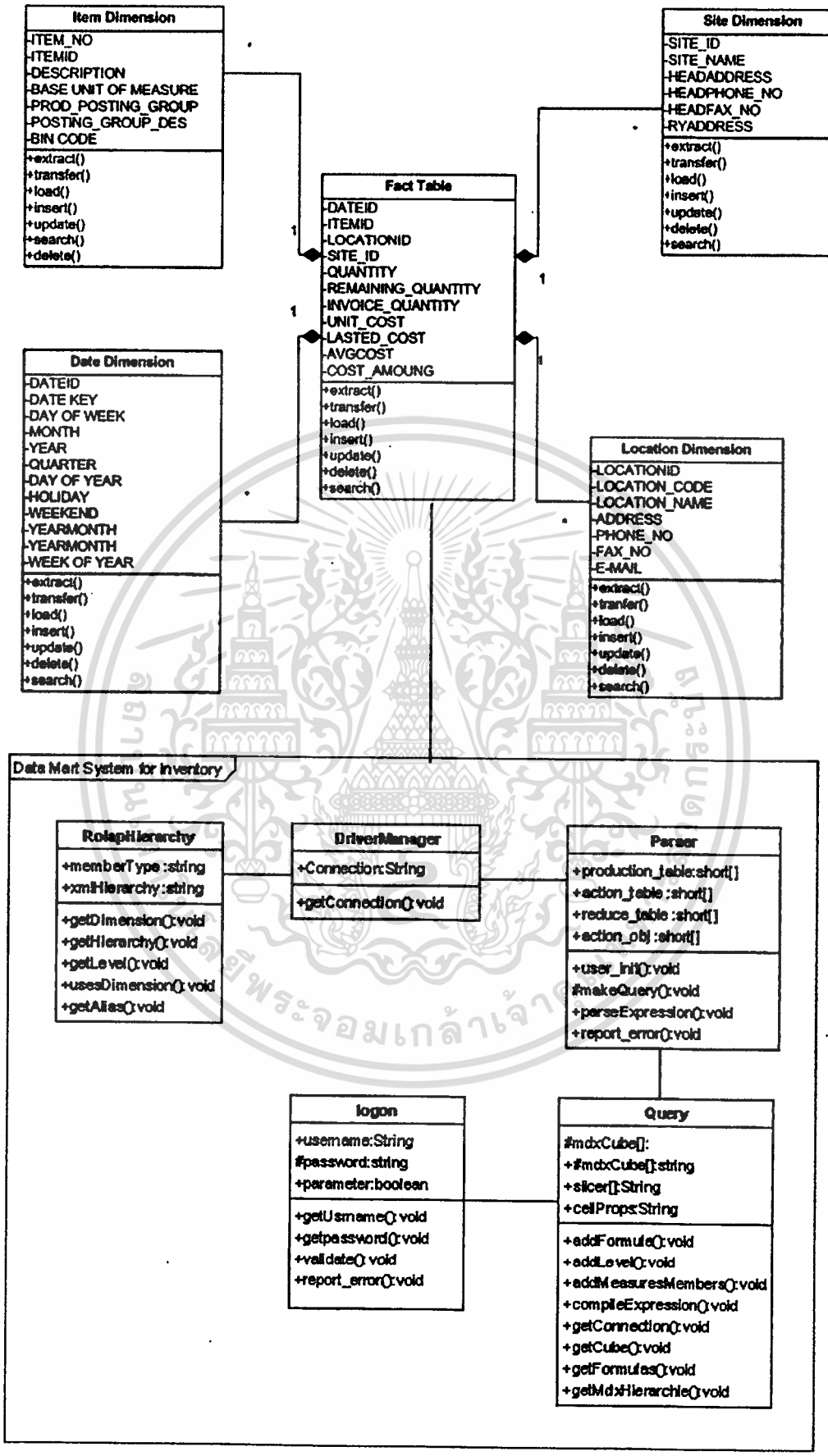
### 3.5 ความต้องการเชิงประสิทธิภาพ(Non Functional Requirements)

1. สินค้าที่มีอยู่ในระบบต้องมีไว้บริการลูกค้าในปริมาณที่เพียงพอ
2. สินค้าที่มีอยู่ในระบบต้องคงไว้ต่อความต้องการของลูกค้าเสมอ
3. กระบวนการทำงานภายในระบบสามารถสร้างยอดขาย และสามารถรักษาส่วนแบ่งทางการตลาดได้
4. กระบวนการทำงานของการบริหารสินค้าคงคลังสามารถลดระดับการลงทุนการเก็บสินค้าให้ต่ำที่สุดได้
5. การบริหารสินค้าคงคลังต้องสามารถทำให้กระบวนการผลิตดำเนินการได้อย่างต่อเนื่อ
6. สามารถสร้างความได้เปรียบทางการแข่งขันภายใต้ความกดดันของธุรกิจประเภทเดียวกัน

### 3.6 การวิเคราะห์และออกแบบสิ่งที่เราสนในด้วย Class Diagram

วิเคราะห์และออกแบบความสัมพันธ์ของสิ่งที่เราสนใจจะนำมาวิเคราะห์เพื่อให้รู้ และสามารถกำหนดว่าสิ่งที่เรามีนั้นมีคุณสมบัติ และพฤติกรรมอย่างไรต่อระบบงาน รวมถึงเป็นการอธิบายถึงสิ่งต่าง ๆ ในมุมมองที่เราสนใจ สามารถแสดง Class ที่สนใจดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

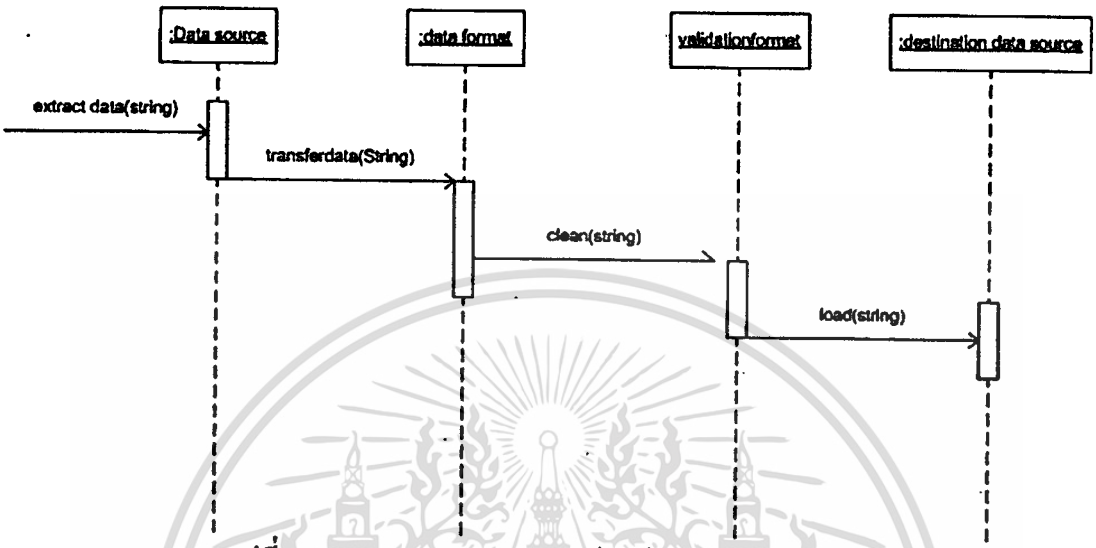



**รูปที่ 3.5** Class Diagram แสดงคลาสต่างๆ และความสัมพันธ์ระหว่างคลาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

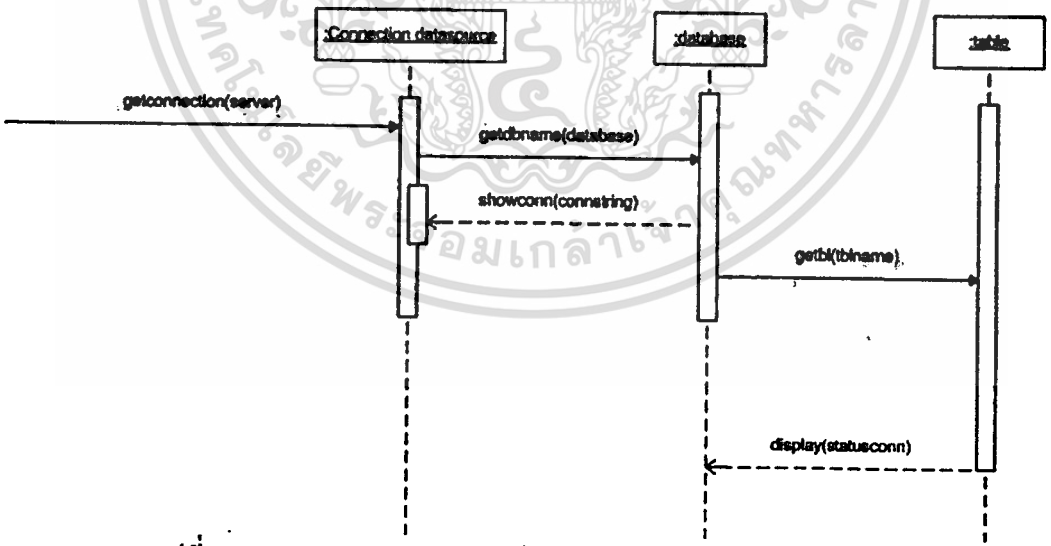
## 3.7 การวิเคราะห์และออกแบบความสัมพันธ์ระหว่างฟังก์ชันการทำงานของระบบ โดย Sequence Diagram

1. Sequence Diagram แสดงความสัมพันธ์ระหว่างฟังก์ชัน ETL




**รูปที่ 3.6** ภาพแสดงความสัมพันธ์ฟังก์ชัน ETL ของ Item

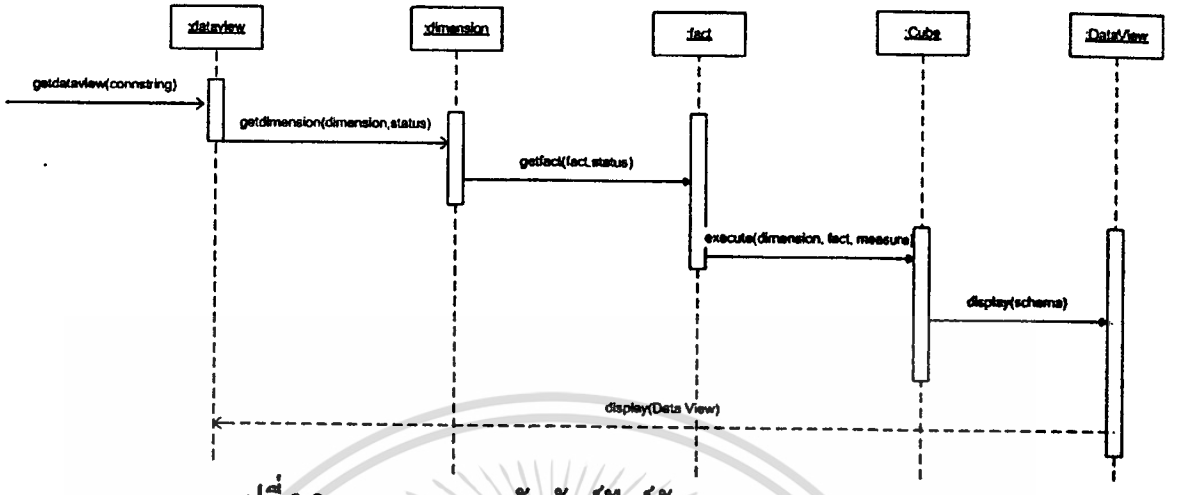
2. Sequence Diagram แสดงความสัมพันธ์ระหว่างฟังกชันการติดต่อดาต้ามาร์ท




**รูปที่ 3.7** ภาพแสดงความสัมพันธ์ฟังก์ชัน Connection Data source

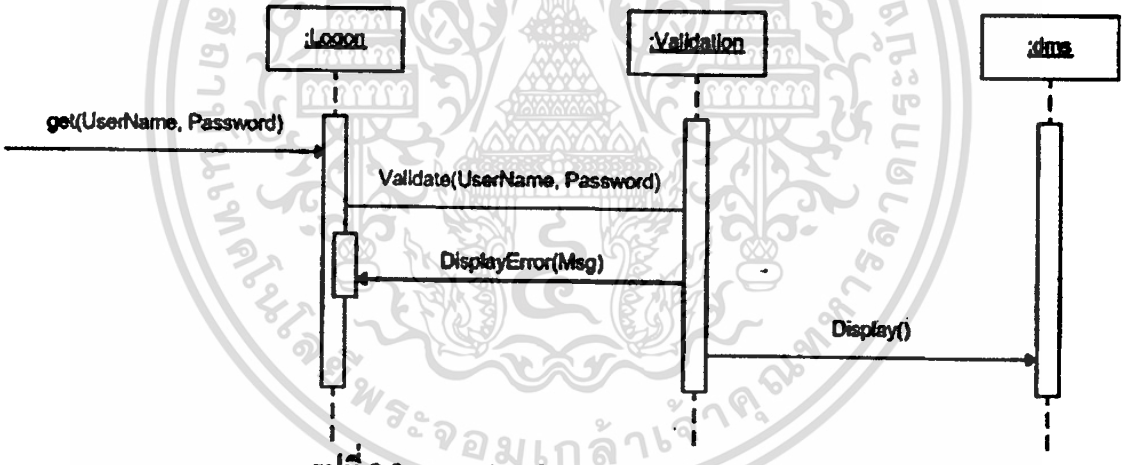
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Sequence Diagram แสดงความสัมพันธ์ระหว่างฟังกชันการคิดต่อดาต้ามาร์ท




**รูปที่ 3.8** ภาพแสดงความสัมพันธ์ฟังก์ชัน Implement Cube

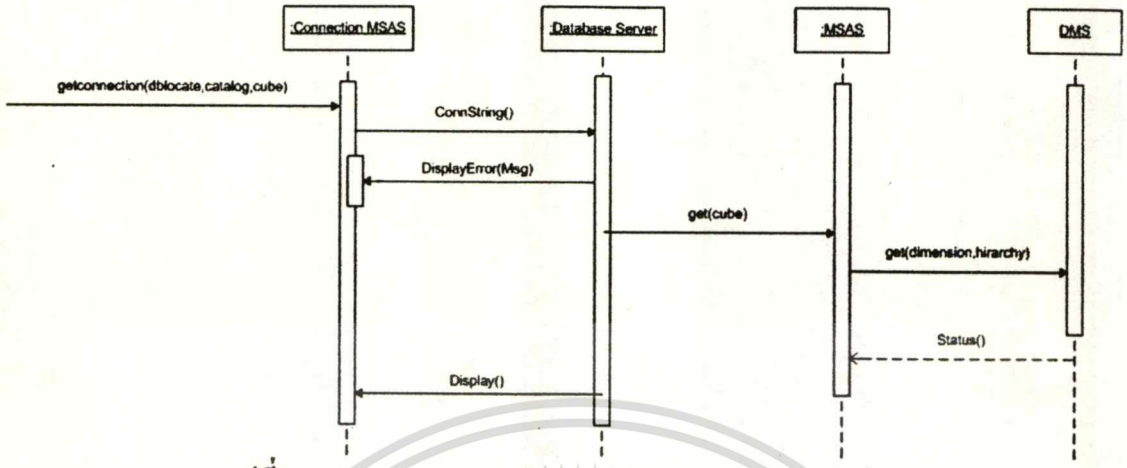
4. Sequence Diagram แสดงความสัมพันธ์ระหว่างฟังก์ชันการทำงาน Login system




**รูปที่ 3.9** ภาพแสดง Sequence Diagram Logon

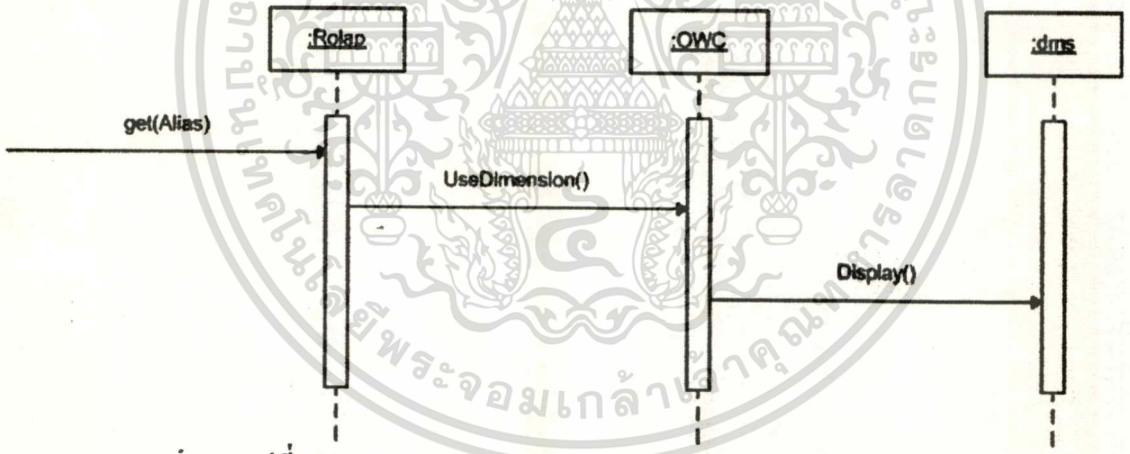
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. Sequence Diagram แสดงความสัมพันธ์ระหว่างฟังก์ชันการทำงาน Connect MSAS




**รูปที่ 3.10** ภาพแสดง Sequence Diagram Connection MSAS

6. Sequence Diagram แสดงความสัมพันธ์ระหว่างฟังก์ชันการทำงานการดึงข้อมูลจาก OLAP




**รูปที่ 3.11** ภาพแสดง Sequence Diagram Retrieve Report

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

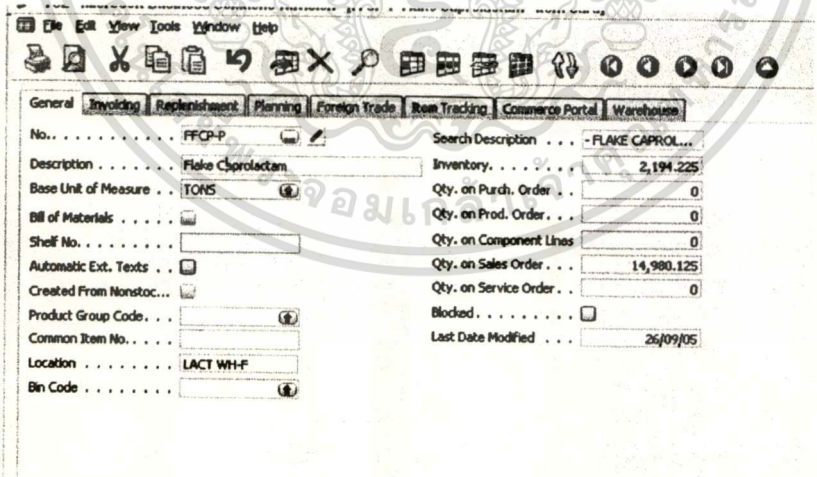
# บทที่ 4

# การพัฒนาระบบดาต้ามาร์ท สำหรับสินค้าคงคลัง กรณีศึกษา บริษัทคาโปรแลคตรัมไทย จำกัด (มหาชน)

เนื้อหาของบทนี้จะกล่าวถึง ขั้นตอนของการพัฒนาระบบดาต้ามาร์ท โดยเริ่มต้นสกัดข้อมูลจากแหล่งต่าง ๆ นำมาจัดรูปแบบให้เป็นมาตรฐานเดียวกัน การโอนย้ายข้อมูลเพื่อทำความสะอาด รวมทั้งการนำข้อมูลที่ผ่านการจัดรูปแบบ เรียบร้อยแล้วนำไปใส่ที่ฐานข้อมูลปลายทาง หรือ ที่พักข้อมูล ตลอดจนการพัฒนาแอพพลิเคชันสำหรับการเรียกดูข้อมูลจาก ดาต้ามาร์ท เพื่อศึกษา หรือวิเคราะห์ แนวโน้มของข้อมูล ซึ่งอยู่ในรูปของรายงาน สามารถเรียกใช้ข้อมูลได้ทั้งระดับต่ำ และระดับรายละเอียด

## 4.1 การเตรียมข้อมูลเข้าสู่ดาต้ามาร์ท

ข้อมูลที่ถูกระบุ และนำเข้าสู่ระบบดาต้ามาร์ท นั้นเป็นข้อมูลที่ได้จากการทำรายการในแต่ละวันของผู้ใช้งานระบบ Microsoft Business Navision Solution เพียงระบบเดียวเท่านั้น แหล่งข้อมูลที่เกี่ยวข้องสามารถแสดงได้ดังนี้




**รูปที่ 4.1** ภาพแสดงส่วนติดต่อระหว่างผู้ใช้กับระบบ Microsoft Business Navision Solution

Table ที่เป็นแหล่งข้อมูลสามารถแสดงได้ดังต่อไปนี้

1. TCL$ITEM LEDGER ENTRY
2. TCL$ITEM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

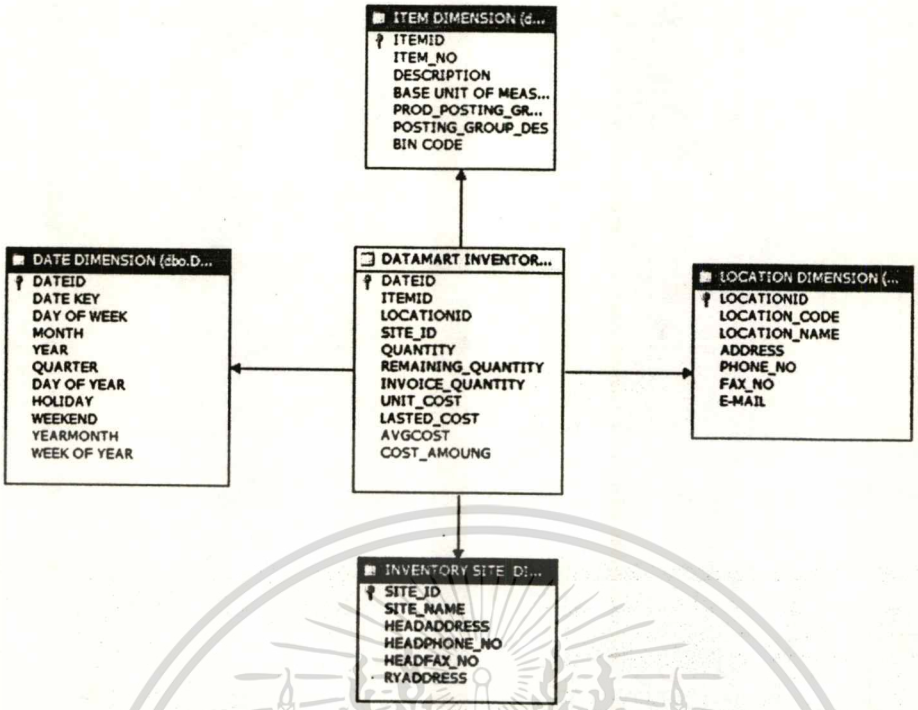
3. TCL$LOCATION
4. TCL$COMPANY
5. TCL$VALUE ENTRY
6. UNT$ITEM LEDGER ENTRY
7. UNT$ITEM
8. UNT$LOCATION
9. UNT$VALUE ENTRY
10. TSL$ITEM LEDGER ENTRY
11. TSL$ITEM
12. TSL$LOCATION
13. TSL$VALUE ENTRY

## 4.2 การแปลงข้อมูลเข้าสู่ดาต้ามาร์ท

หลังจากการออกแบบฐานข้อมูลสำหรับแต่ละดาต้ามาร์ทเสร็จแล้ว ขั้นตอนต่อไปที่สำคัญยิ่งก็คือการวิเคราะห์ และ ออกแบบ การนำข้อมูลจากแหล่งข้อมูลไปแปลงให้อยู่ในแพลตฟอร์มของฐานข้อมูลที่ได้ออกแบบไว้ นั่นก็คือการแปลงข้อมูล หรือ Extraction Transformation and Loading (ETL) นั่นเอง โดยที่คุณภาพของการแปลงข้อมูลเป็นสิ่งสำคัญมากสำหรับการสร้างคลังข้อมูล จะแตกต่างกันไปตามคลังข้อมูลซึ่งในแต่ละองค์กรมีความต้องการโดยที่การแปลงข้อมูลหมายรวมตั้งแต่การวิเคราะห์แหล่งข้อมูล กำหนดการส่งข้อมูลรวบรวมหรือสร้างข้อมูลภายนอก วางแผนและสร้างรูทีนของการแปลงข้อมูล และตรวจสอบความถูกต้องของข้อมูลที่ได้โดยเลือกใช้แอพพลิเคชันสำหรับการแปลงข้อมูล (ETL) SQL Server Business Intelligence Development Studio ซึ่งเป็นผลิตภัณฑ์ของบริษัท Microsoft ขั้นตอนของการแปลงข้อมูลสามารถสรุปเป็นขั้นตอนได้ดังนี้

1. วิเคราะห์แหล่งข้อมูล เช่น ปริมาณของข้อมูล จำนวนและชนิดของการเข้าถึงแหล่งข้อมูลแพลตฟอร์มและภาษาโปรแกรมที่ใช้ เป็นต้น
2. ย้ายข้อมูลที่ต้องการจากระบบเดิมมาไว้ในบริเวณที่ใช้ปรับแต่งข้อมูล หรือเรียกบริเวณนี้ว่า Staging area เพื่อนำมาเลือกเฉพาะส่วนที่ต้องการแปลงข้อมูล และตรวจสอบความถูกของต้องหรือการทำความสะอาดข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 4.2 ภาพแสดงโครงร่างข้อมูลแบบดาว (Star Schema)**

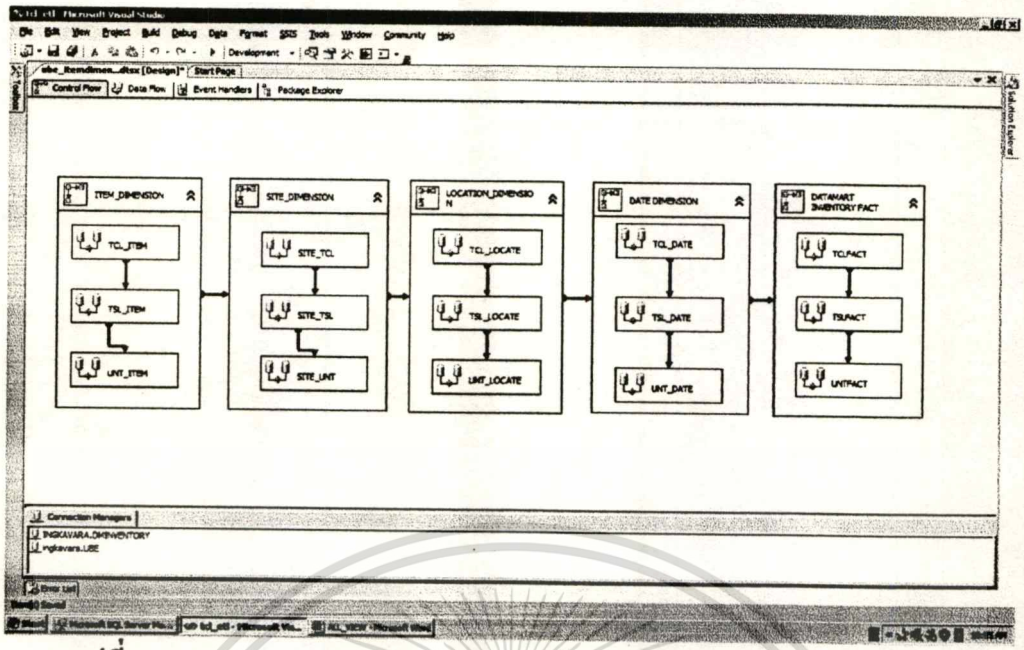
3. กำหนด primary key ของ fact table และ dimension table และกำหนด foreign key ระหว่าง Fact table กับ dimension table

4. ย้ายข้อมูลที่ทำความสะอาดแล้วจาก Staging area ลงสู่เซิร์ฟเวอร์ของดาต้ามาร์ท สามารถ แสดงขั้นตอนการแปลงข้อมูลทั้ง 3 ไซท์งานคือ

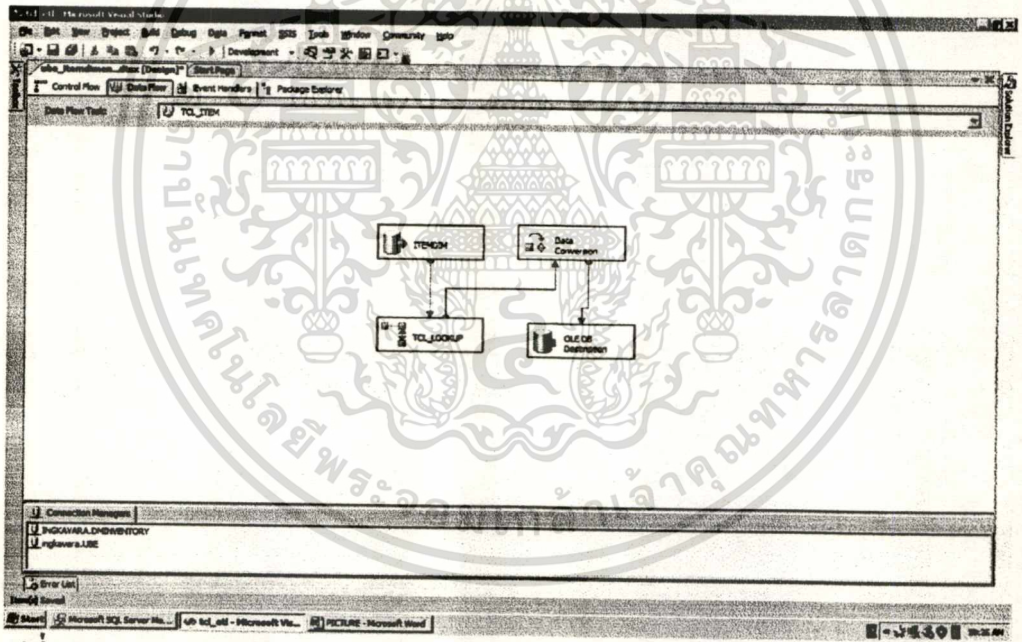
- TCL : Thai Caprolactam Public Company
- TSL : Thai Synthetic Rubbers Co., Ltd.
- UNT : Ube Nylon (Thailand) Ltd.

สามารถแสดงขั้นตอนการทำงานได้ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

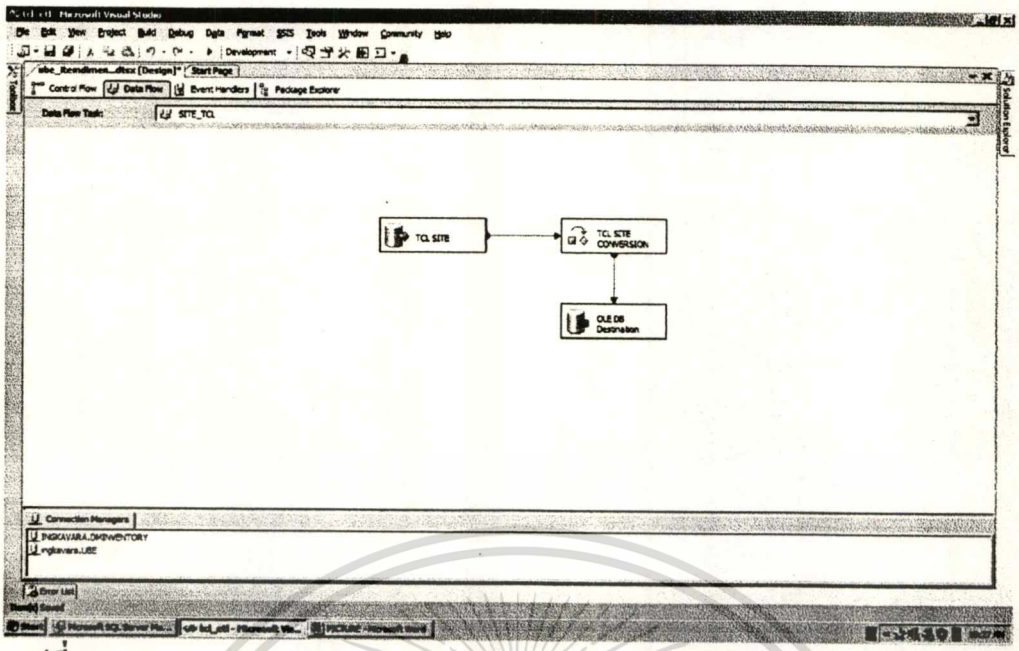



**รูปที่ 4.3** ภาพแสดงการออกแบบการแปลงข้อมูลจากแหล่งข้อมูลไปยังปลายทาง

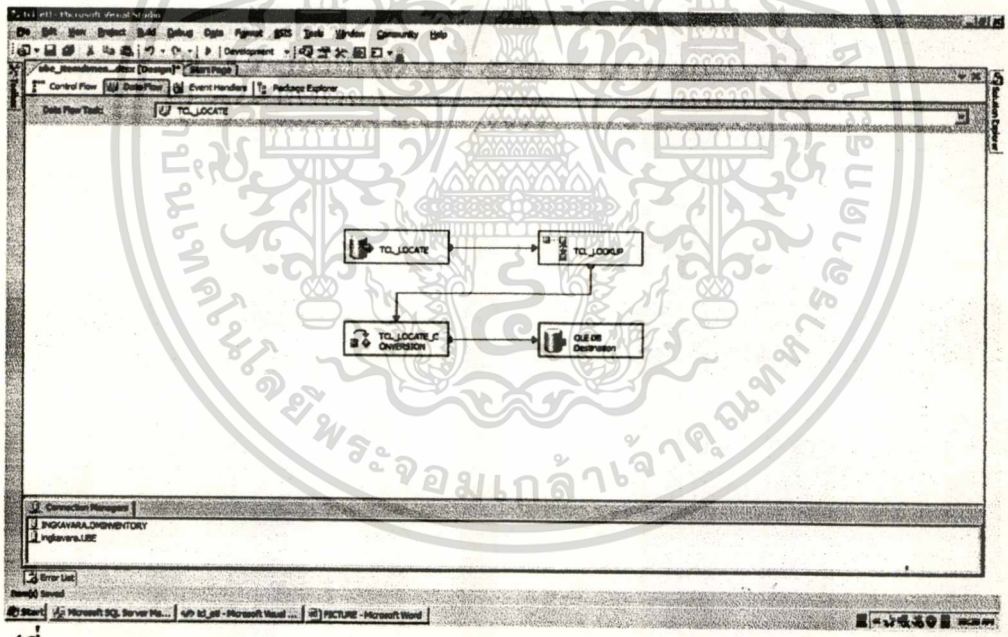



**รูปที่ 4.4** ภาพการออกแบบการทำความสะอาดข้อมูล โดยแปลงรูปแบบข้อมูลจากต้นทางไปยังปลายทางของ Item Dimension

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

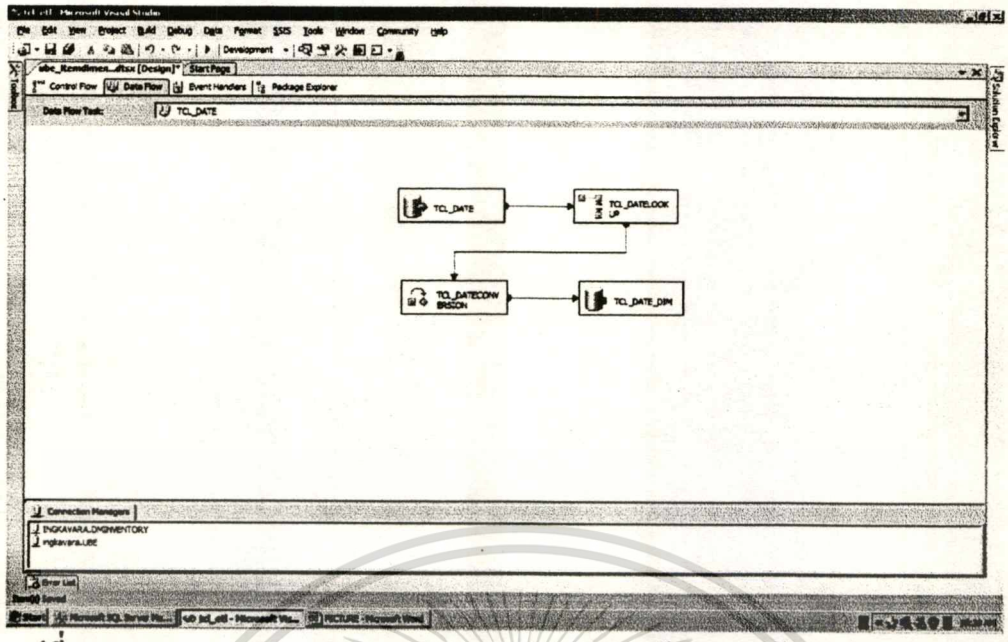



**รูปที่ 4.5** ภาพแสดงการออกแบบการทำความสะอาด แปลงรูปแบบข้อมูลจากต้นทางไปยังปลายทางของ Site Dimension

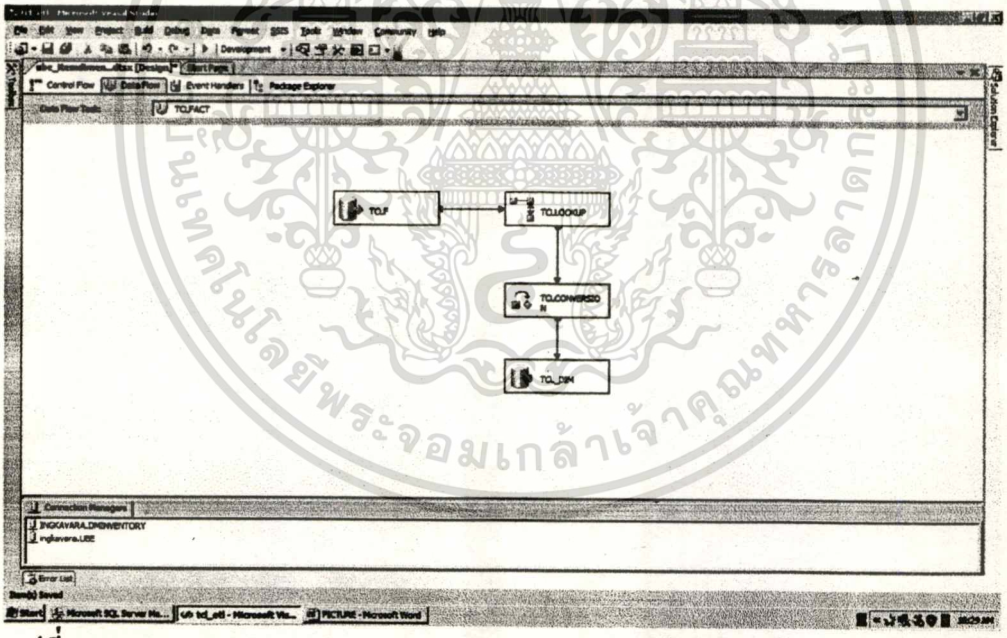



**รูปที่ 4.6** ภาพการออกแบบการทำความสะอาด แปลงรูปแบบข้อมูลจากต้นทางไปยังปลายทางของ Location Dimension

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

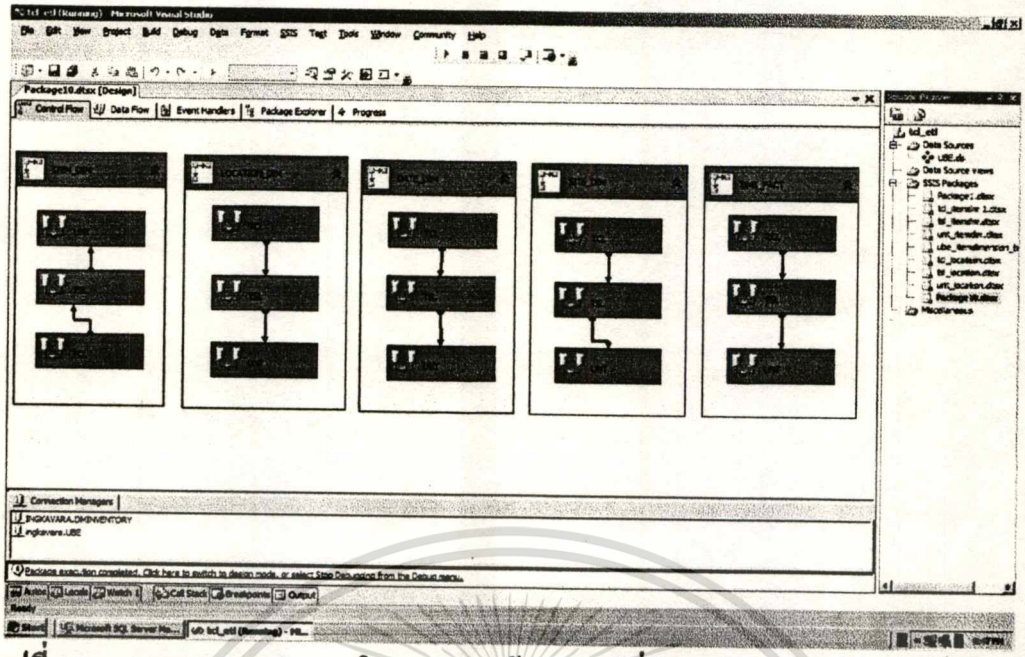



**รูปที่ 4.7** ภาพออกแบบการทำความสะอาด แปลงรูปแบบข้อมูลจากต้นทางไปยังปลายทาง ของ Date Dimension




**รูปที่ 4.8** ภาพออกแบบการทำความสะอาด แปลงรูปแบบข้อมูลจากต้นทางไปยังปลายทาง ของ Data Mart Inventory Fact Dimension

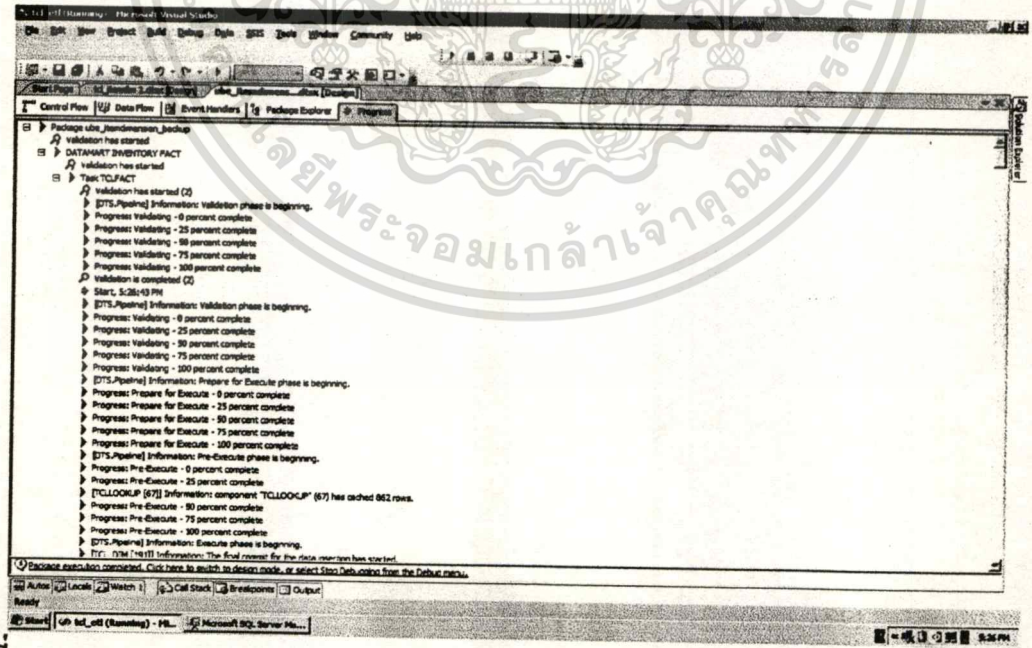
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 4.9** ภาพแสดงกระบวนการในการแปลงข้อมูลกรณีที่มีกระบวนการทำงานถูกต้อง 100 %

5. สร้าง metadata ของแต่ละดาต้ามาร์ท โดยเก็บรายละเอียดของข้อมูลการอัปเดตและส่งออกไปไว้ในดาต้ามาร์ท

6. ตรวจสอบความถูกต้องของข้อมูล ซึ่งจะต้องกระทำตลอดทั้งกระบวนการแปลงข้อมูล สามารถทำได้ดังนี้



**รูปที่ 4.10** ภาพแสดง SQL Server Business Intelligence Development แสดงการตรวจสอบความถูกต้องของข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตรวจสอบความถูกต้องของข้อมูล และ แสดงกระบวนการทำงานของ SSID มีขั้นตอนดังนี้

6.1 ตรวจสอบผลรวมทั้งหมดของจำนวนข้อมูลที่ดึงมาจากแหล่งข้อมูลต่าง ๆ ที่ได้ทำการเพิ่มเข้าไป

6.2 ตรวจแก้ข้อมูลในระบบเดิมของแหล่งข้อมูล หรือในรูทีนของการแปลง ซึ่งควรจะเก็บข้อมูลในการตรวจแก้ไว้ใน metadata ของการแปลงข้อมูลด้วย

6.3 ตรวจสอบค่าของข้อมูลให้ถูกต้องในกระบวนการรวบรวมข้อมูล

6.4 ตรวจสอบผลรวมของข้อมูลหลังจากย้ายข้อมูลลงสู่ดาต้ามาร์ทแล้ว

## 4.3 การกำหนดโครงสร้างของคิวบ์

มีส่วนประกอบต่าง ๆ ของคิวบ์ ซึ่งประกอบไปด้วย

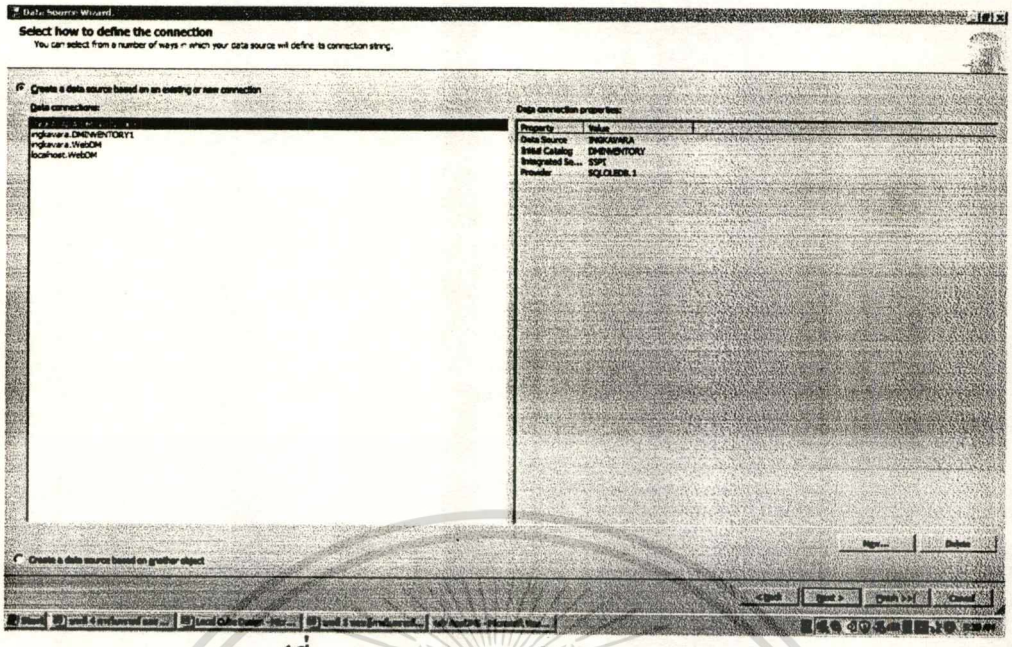
1. Dimensions ซึ่งภายใน dimension จะมีชุดของค่าต่าง ๆ ที่อยู่ในแกนของคิวบ์ ดังนั้น Dimension จึงเปรียบเสมือนเป็นมุมมองทั้งหมดของผู้ใช้งาน
2. Member คือ ค่าทั้งหมดที่ถูกเก็บอยู่แต่ละ dimension
3. Cells คือ ชุดที่ประกอบไปด้วย Member จาก แต่ละ dimension ซึ่ง cell สามารถมี Measure ได้มากกว่า 1 ซึ่งเกิดจากการ intersection กัน ของ dimension ของ cube
4. Hierarchies จะถูกเก็บอยู่ในแต่ละ dimension เป็นกลุ่มซึ่งมี member ที่หลากหลาย เช่น กลุ่มของวันทีอยู่ในเดือน อยู่ใน Quarter อยู่ใน ปี
5. Level เมื่อมีการระบุ Hierarchies แล้ว Level ก็มักจะถูกระบุไปพร้อมกันด้วย โดยเป็นการอธิบายถึง Member ต่าง ๆ จากการสรุป level ของตัวข้อมูลในระดับรายละเอียด

## 4.4 กระบวนการสร้างคิวบ์

มีขั้นตอนในการสร้างดังต่อไปนี้

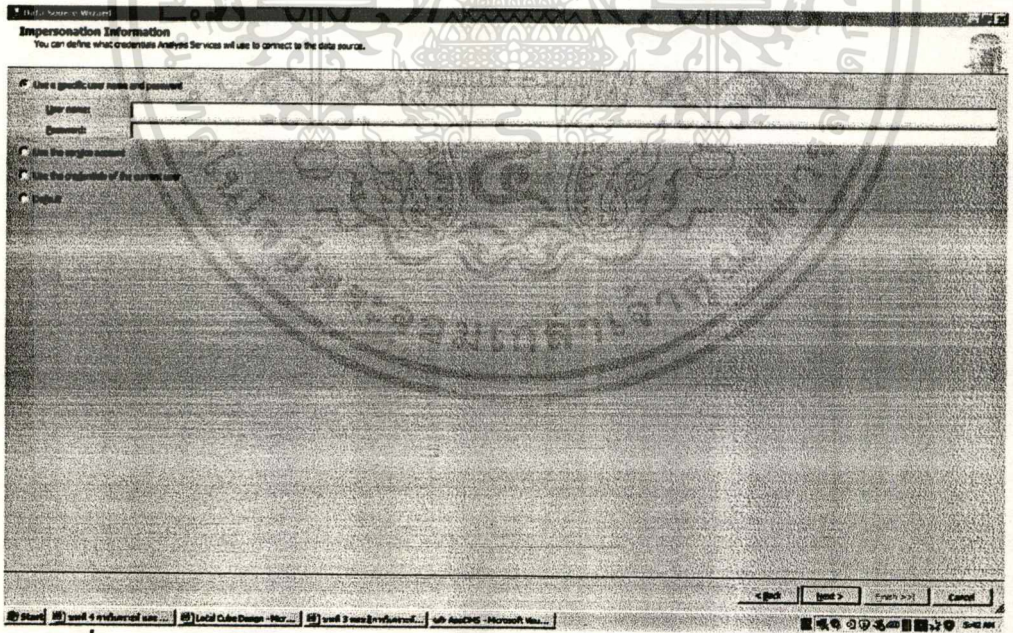
1. เปิดโปรแกรม Sql Server Business Intelligence Development เพื่อเรียกใช้ Microsoft Analysis Service
2. คลิก Create New Project เลือกสร้างโปรเจค Analysis Service เพื่อการวิเคราะห์ข้อมูลจากดาต้ามาร์ท หรือ ดาต้าแวร์เฮาส์
3. ปรากฏหน้าต่างซึ่งประกอบไปด้วยเครื่องมือในการสร้างคิวบ์ โดยสามารถสร้าง และดำเนินการกับคิวบ์โดย Solution Explorer
4. กำหนด Data Source เพื่อสร้างการติดต่อระหว่างดาต้ามาร์ตกับคิวบ์ โดยการ คลิกขวา เลือก New Data Source ปรากฏดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 4.11** ภาพแสดงการติดต่อ Data Source

5. เลือก Data Source ที่ต้องการ คลิก next ระบุสิทธิ์ในการเข้าใช้เครื่องมือสำหรับการวิเคราะห์ ข้อมูลผู้ใช้ปรากฏดังรูป

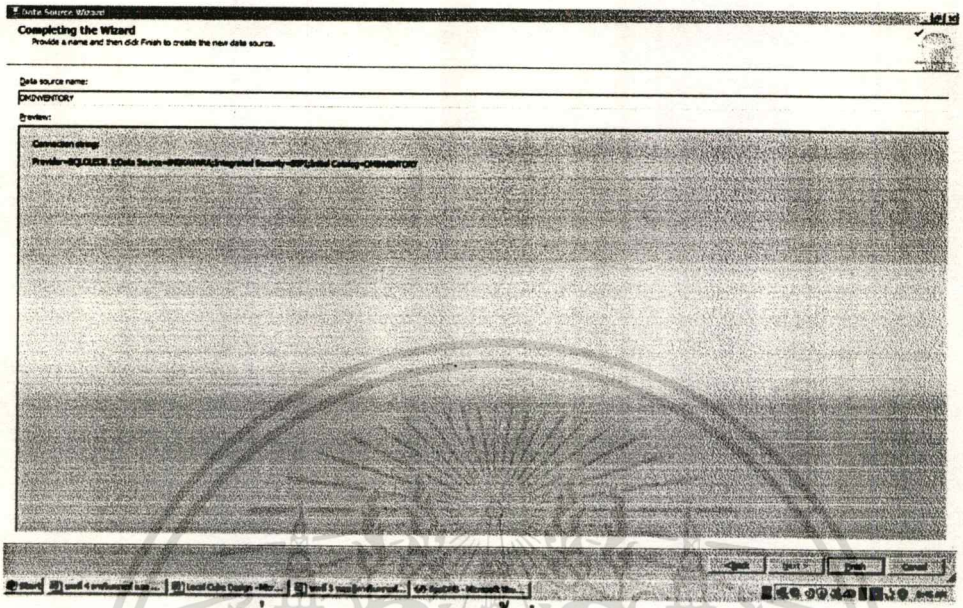



**รูปที่ 4.12** ภาพแสดงการกำหนดสิทธิ์การเข้าใช้บริการเครื่องมือสำหรับการวิเคราะห์ข้อมูล

6. ตั้งชื่อให้ Data Source จะปรากฏ Connection String ในการติดต่อกัน ระหว่าง ดาต้ามาร์ท และ Analysis Service ซึ่งเป็นเครื่องมือในการวิเคราะห์ โดยมีคำสั่งในการติดต่อระหว่างดาต้ามาร์ท ดังนี้

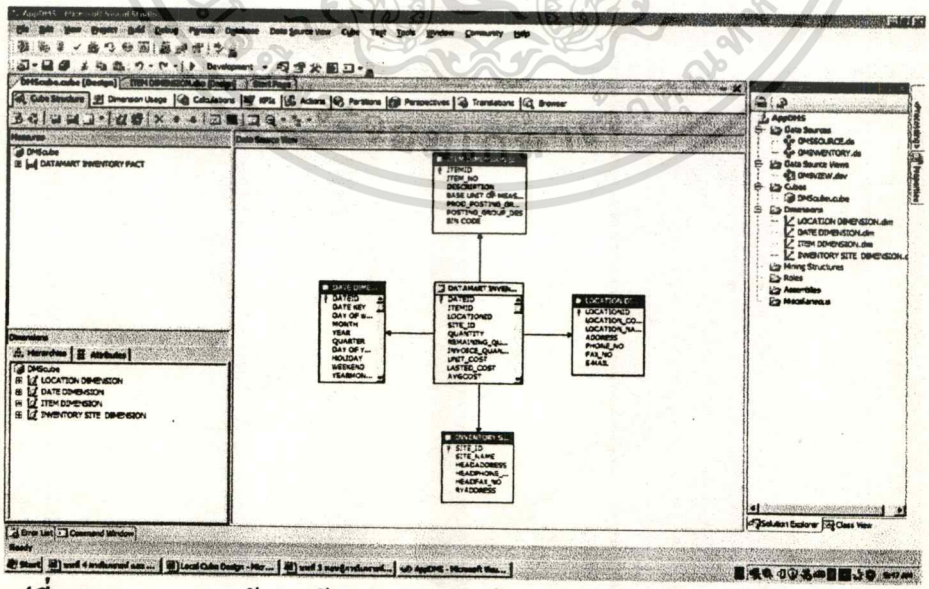
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"Provider=SQLOLEDB.1;DataSource=INGKAVARA;Integrated Security=SSPI;Initial Catalog=DMINVENTORY" แสดงได้ดังรูป




**รูปที่ 4.13** ภาพแสดงการตั้งชื่อของ Data Source

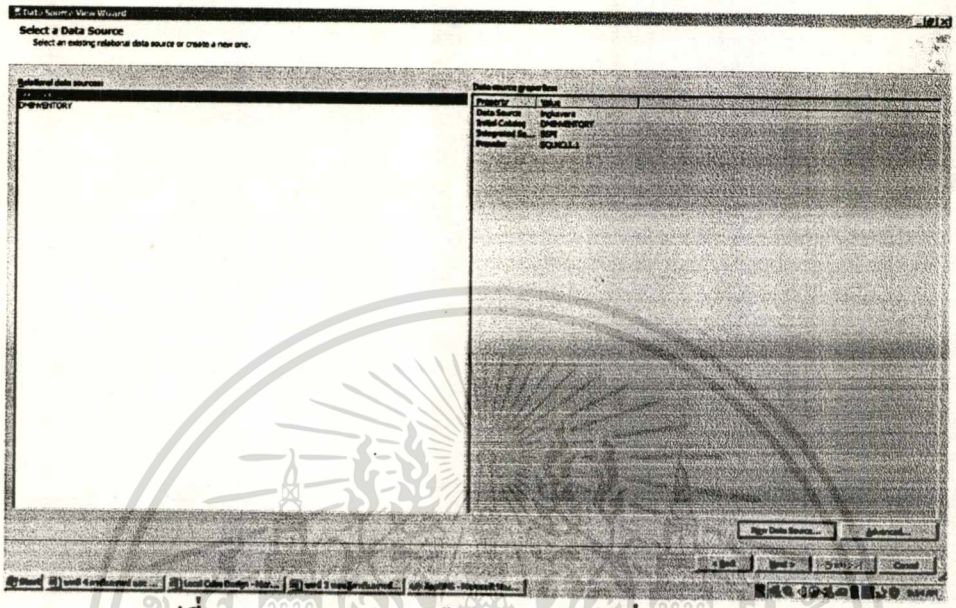
7. หลังจากนั้น คลิกที่ finish ระบบจะทำการติดต่อดาต้ามาร์ตที่ต้องการทำการวิเคราะห์ โดยระบบจะทำการสร้าง Dimension, Measure, Hierarchies, Level ให้อัตโนมัติ ปรากฏดังรูป



**รูปที่ 4.14** ภาพแสดงเค้าร่างข้อมูลแบบสตาร์ หลังจากติดต่อกับ Data Source สำเร็จ

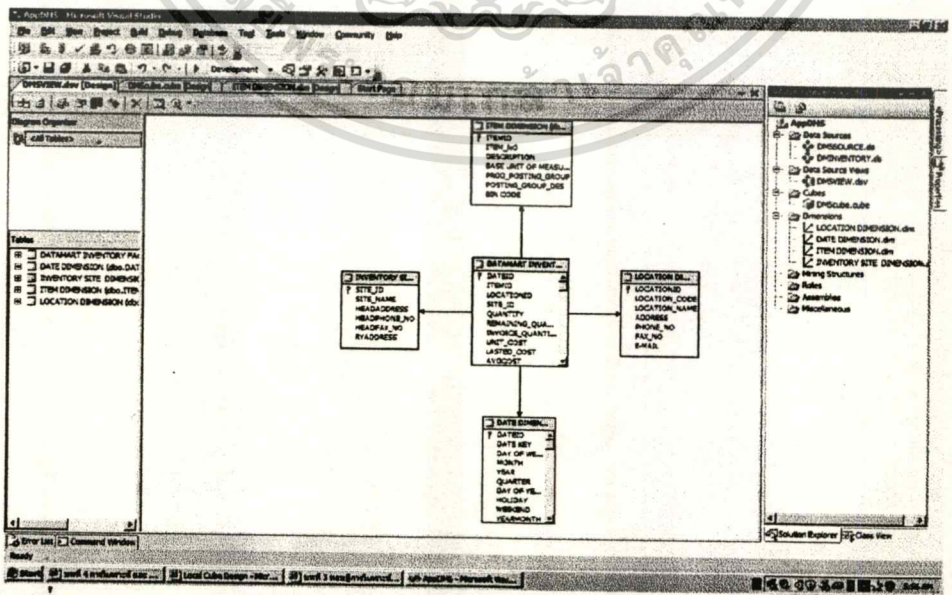
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. กำหนดมุมมองของ Data Source เป็นการกำหนดเค้าร่างของข้อมูล เพื่อระบุ Dimension และ Fact ที่จะถูกนำไปสร้างเป็น Cube มีขั้นตอนการทำงานดังนี้ คลิกขวาที่คำสั่ง Data source view เลือก new data source view ปรากฏดังรูป




**รูปที่ 4.15 ภาพแสดงการเลือกแหล่งข้อมูลที่ต้องการติดต่อ**

9. เลือก Data Source ที่ต้องการนำ dimension และ fact มาใช้งาน หรือหากไม่ต้องการที่จะเลือก data source ที่สร้างไว้แล้ว ก็คลิกที่ new data source เพื่อทำการติดตั้ง data source ตัวใหม่หลังจากนั้น คลิก Next ปรากฏดังรูป

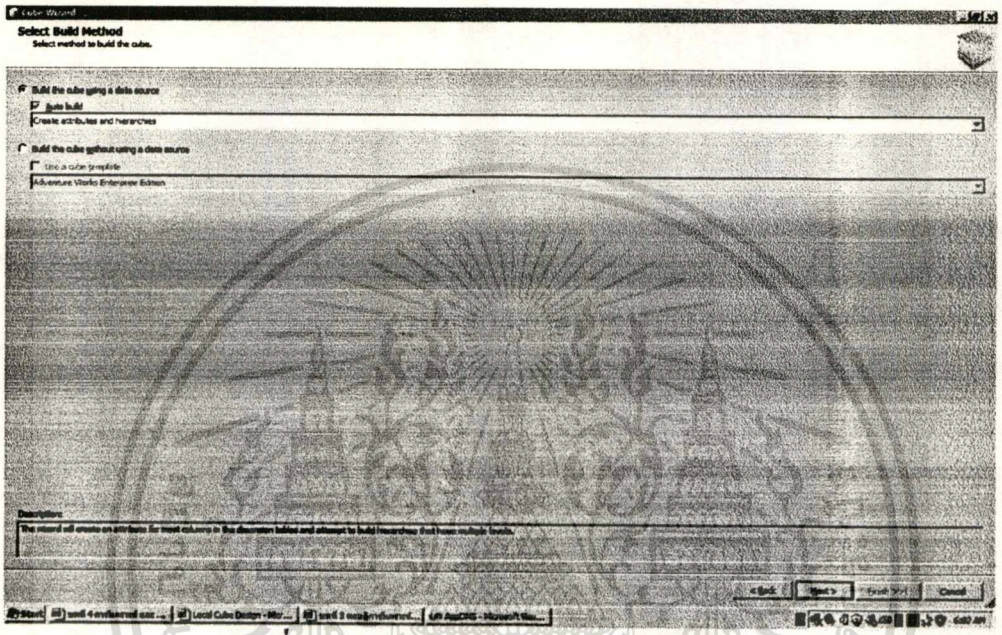


**รูปที่ 4.16 ภาพแสดงเค้าร่างข้อมูลแบบสตาร์ หลังจากติดต่อกับ Data Source สำเร็จ**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. ปรากฏ dimension table และ fact table ที่ถูกเลือก จากการกำหนดเค้าร่างของคิวบ์ เพื่อดำเนินการสร้างคิวบ์ต่อไปดังขั้นตอนต่อไปนี้

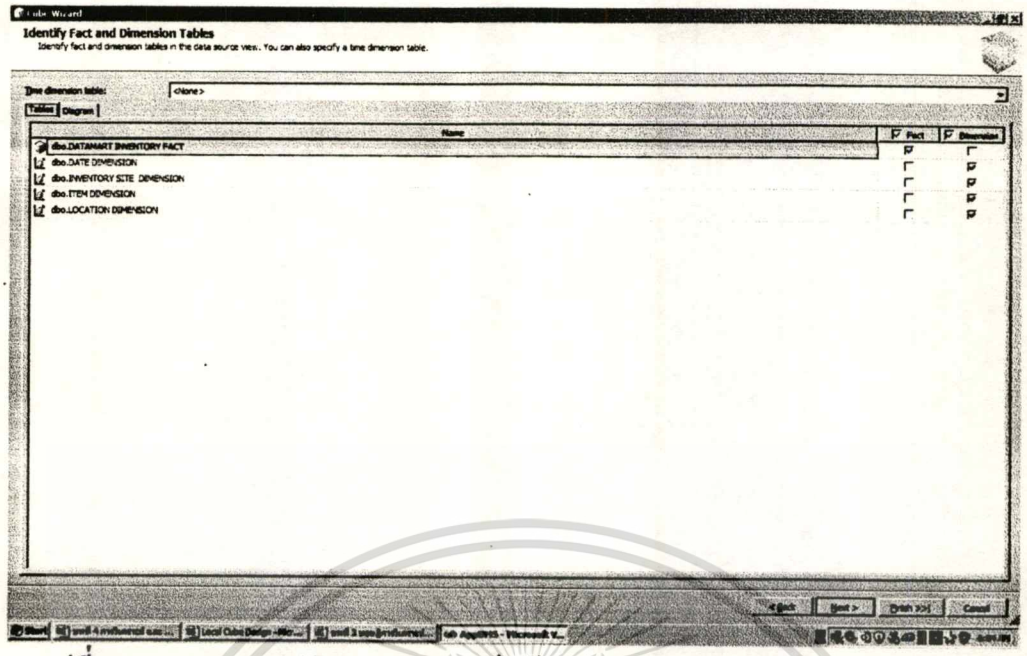
- คลิกขวาที่ Cubes เลือก new Cube ปรากฏหน้าต่างเพื่อกำหนดวิธีในการสร้าง Cube โดยเลือกที่คำสั่ง Build The cube using data source หลักจากนั้นสร้างคิวบ์ เพื่อใช้ในการวิเคราะห์ข้อมูลหลายมิติ คลิกที่ next ดังรูป



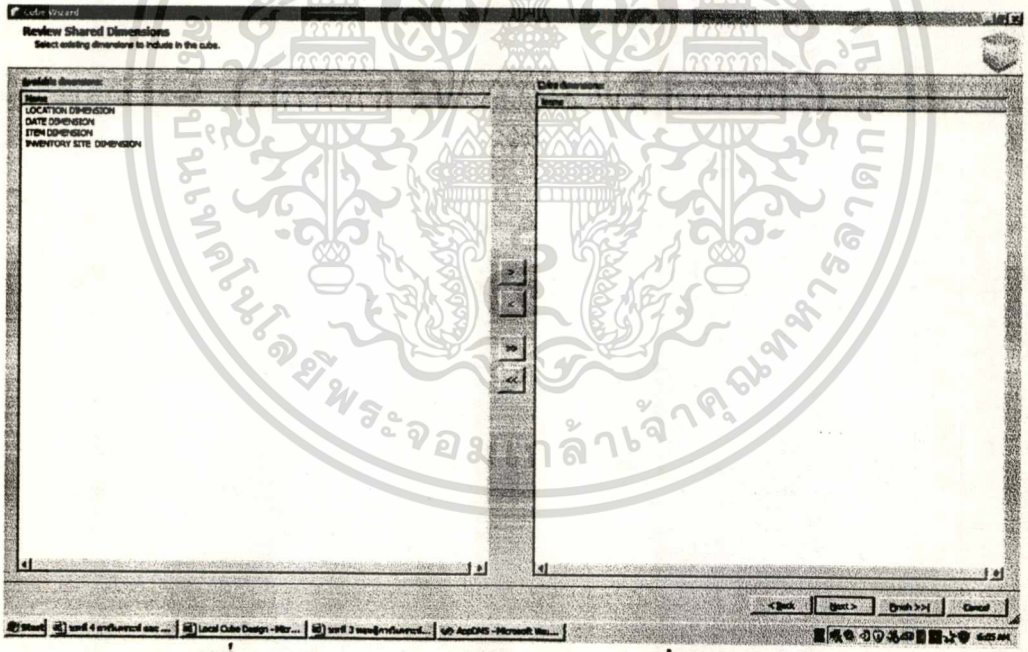

**รูปที่ 4.17 ภาพแสดงรูปแบบของการสร้างคิวบ์**

- ระบุ dimension และ fact หลังจากนั้นคลิก next เพื่อเลือก dimension และ fact ต้องการนำมาใช้วิเคราะห์ข้อมูล สามารถเลือกที่แท็บ Diagram เพื่อดูแบบจำลองข้อมูลแบบเค้าร่างสตาร์ ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



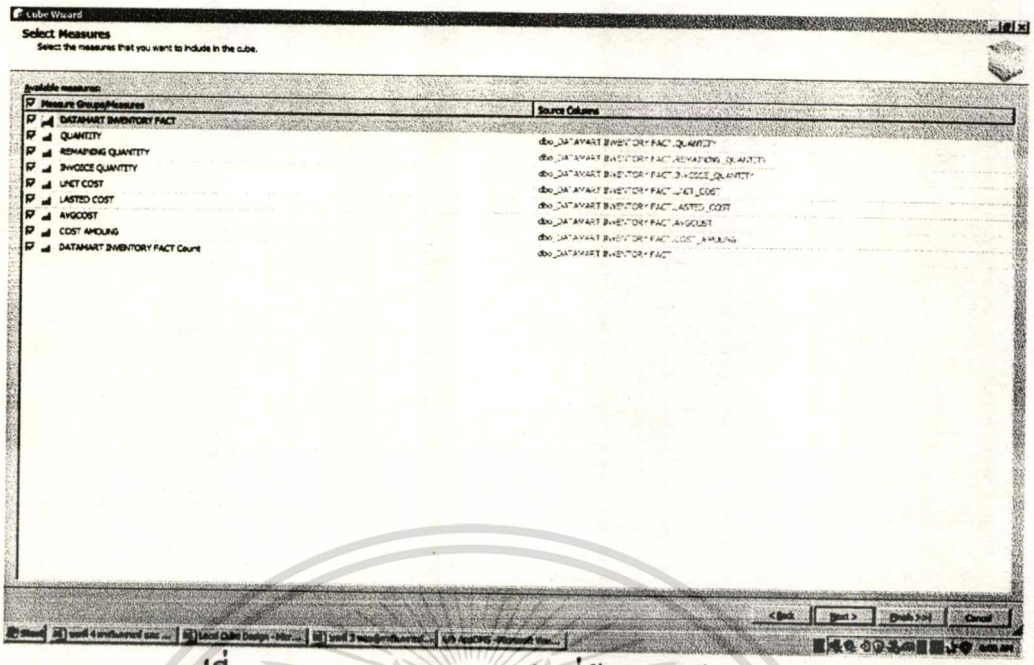

**รูปที่ 4.18** ภาพแสดงรูปแบบของตารางเก็บข้อมูลระหว่าง Fact table และ Dimension Table




**รูปที่ 4.19** ภาพแสดงการเลือก Dimension ที่ต้องการใช้งาน

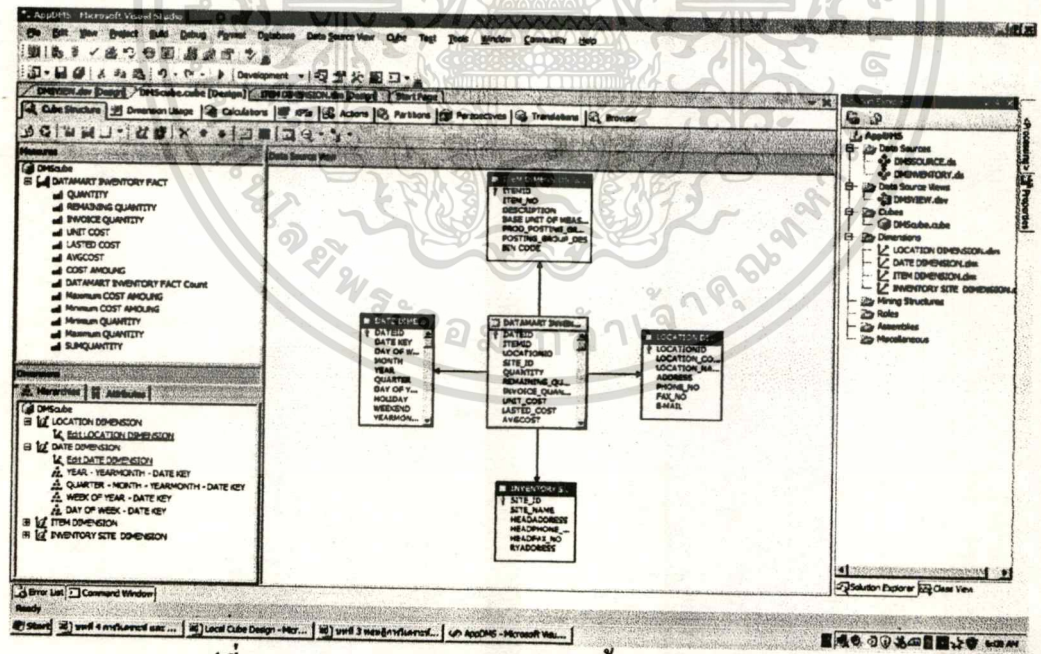
- เลือก Dimension ที่ต้องการใช้งานในคิวบ์ คลิก Next จะปรากฏหน้าต่างเลือก measures

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 4.20 ภาพแสดง Measure ต่างที่ต้องการใช้งาน**

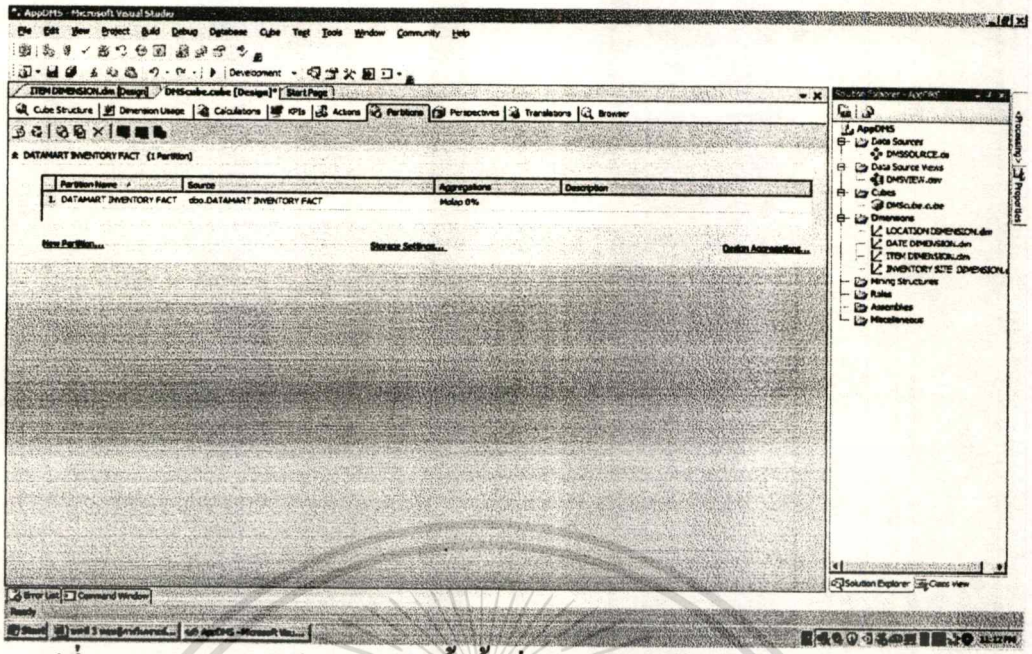
- ตั้งชื่อให้แก่ คิวบ์ Microsoft Analysis Service ดำเนินการสร้าง คิวบ์ให้อัตโนมัติ Microsoft Analysis Service สร้าง dimension, member, hierarchies



**รูปที่ 4.21 ภาพแสดงหลังจากการติดตั้งค่าต่าง ๆ ให้กับคิวบ์**

- กำหนด การแบ่งพื้นที่ในการจัดเก็บข้อมูล ข้อมูลการคำนวณแบบ Aggregations โดยเลือกที่ Partitions Tab เลือก คำสั่ง storage setting ดังรูป

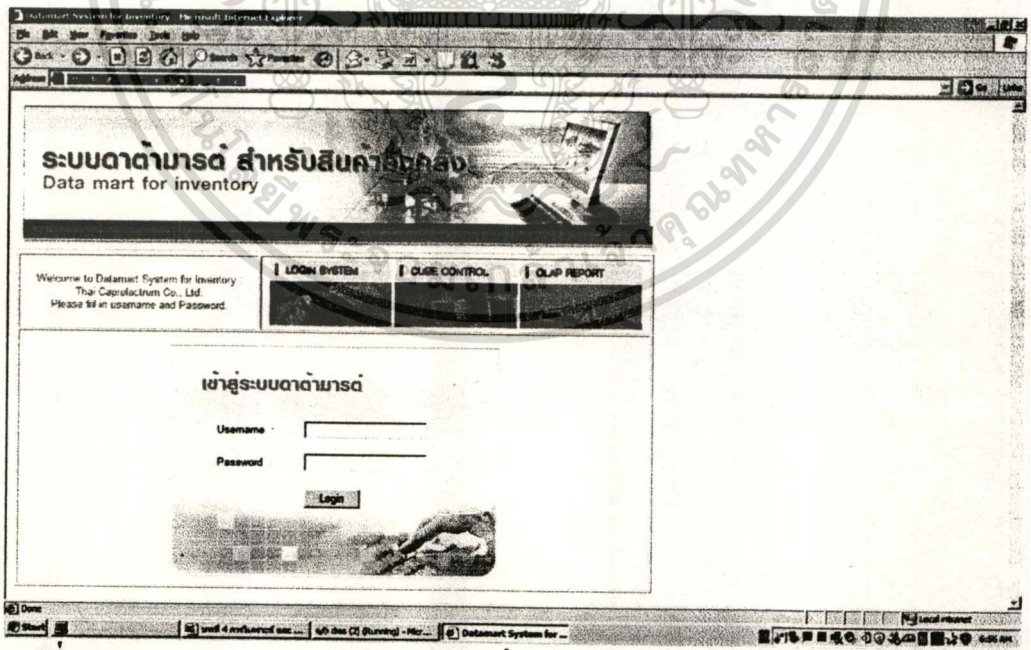
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 4.22** ภาพแสดงหน้าต่างการติดตั้งพื้นที่ สำหรับการคำนวณแบบ Aggregation

## 4.5 ขั้นตอนการใช้งานระบบ ในส่วนของแอพพลิเคชัน

1. เปิด โปรแกรม Internet Explorer เพื่อเปิดระบบดาต้ามาร์ท ผู้ใช้งานระบบกรอก Username และ Password หลังจากนั้นระบบจะตรวจสอบบัญชีรายชื่อผู้ใช้งาน จากฐานข้อมูล



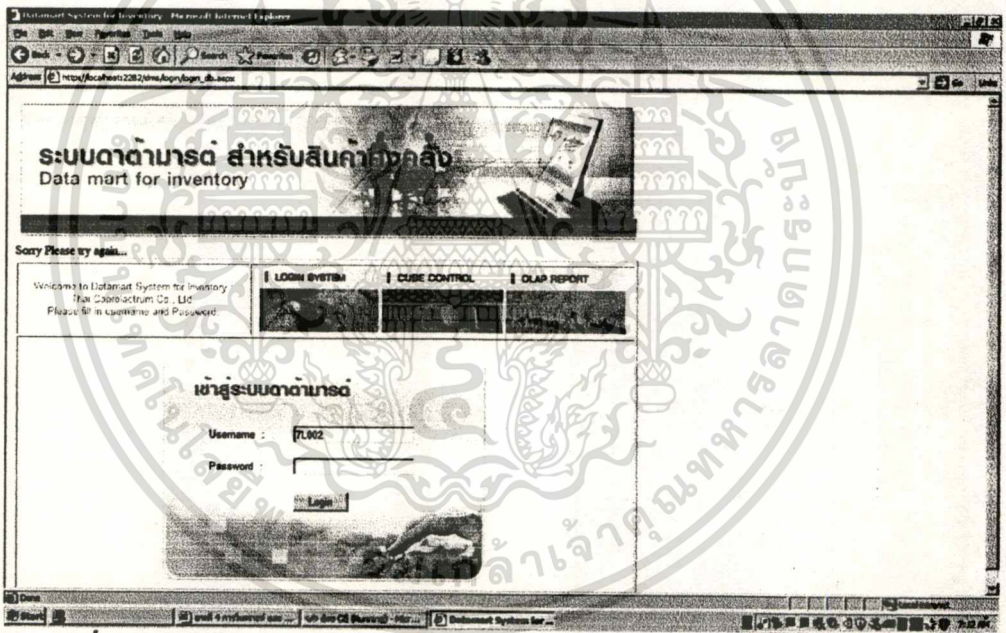

**รูปที่ 4.23** ภาพแสดงหน้าจอสำหรับผู้ใช้ใส่ชื่อ และ รหัสผ่านเพื่อเข้าใช้งานระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมนูที่ปรากฏในหน้าจอ Login ประกอบไปด้วยฟังก์ชันการทำงาน 3 ฟังก์ชันคือ

- Login System มีสถานะของ Object เป็น Enable คือ user สามารถเลือกใช้ได้ เนื่องจากเป็นครั้งแรกของการใช้งาน
- Cube Control มีสถานะของ Object เป็น disable คือ user ไม่สามารถเลือกใช้ได้ เนื่องจากต้องผ่านการตรวจสอบการเข้าใช้งานก่อนถึงจะสามารถใช้งานได้
- Olap Report มีสถานะของ Object เป็น disable คือ user ไม่สามารถเลือกใช้ได้ เนื่องจากต้องผ่านการตรวจสอบการเข้าใช้งานก่อนถึงจะสามารถใช้งานได้

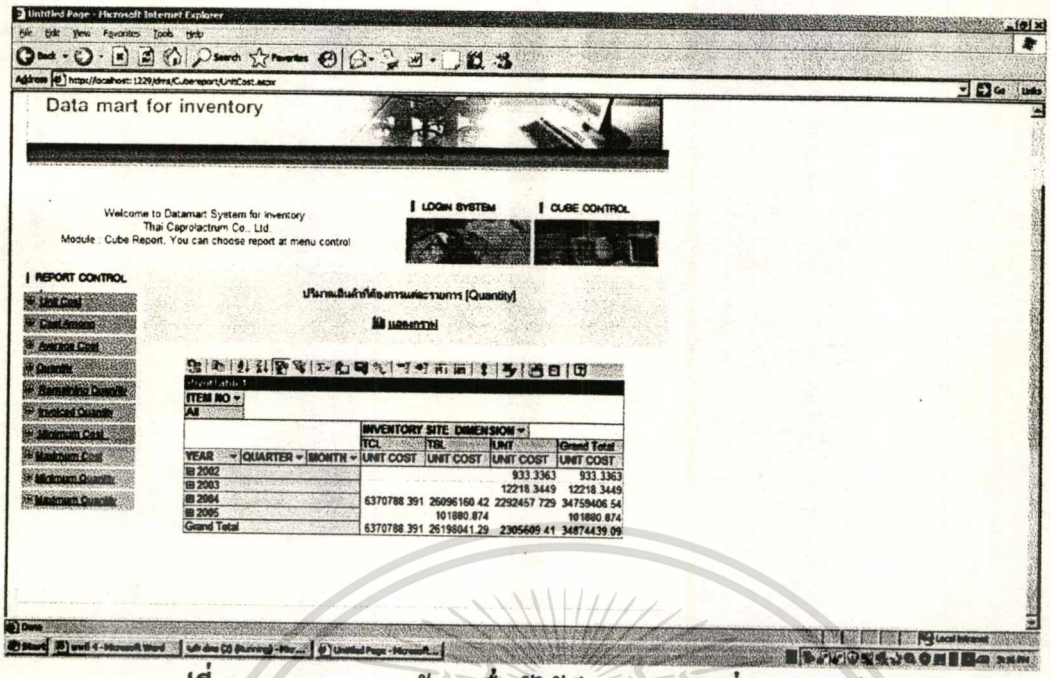
2. ในกรณีที่ user name หรือ password ผิด จะ แสดงข้อความเตือนผู้ใช้งานว่า ไม่ถูกต้อง รวมทั้ง ปรากฏหน้าจอเดิมอีกครั้งเพื่อให้ ผู้ใช้งานระบบใส่ชื่อผู้ใช้ และ รหัสผ่านอีกครั้ง ปรากฏได้ดังรูป




**รูปที่ 4.24** ภาพแสดงหน้าจอสำหรับผู้ใช้ใส่ชื่อ และ รหัสผ่านเพื่อเข้าใช้งานระบบ

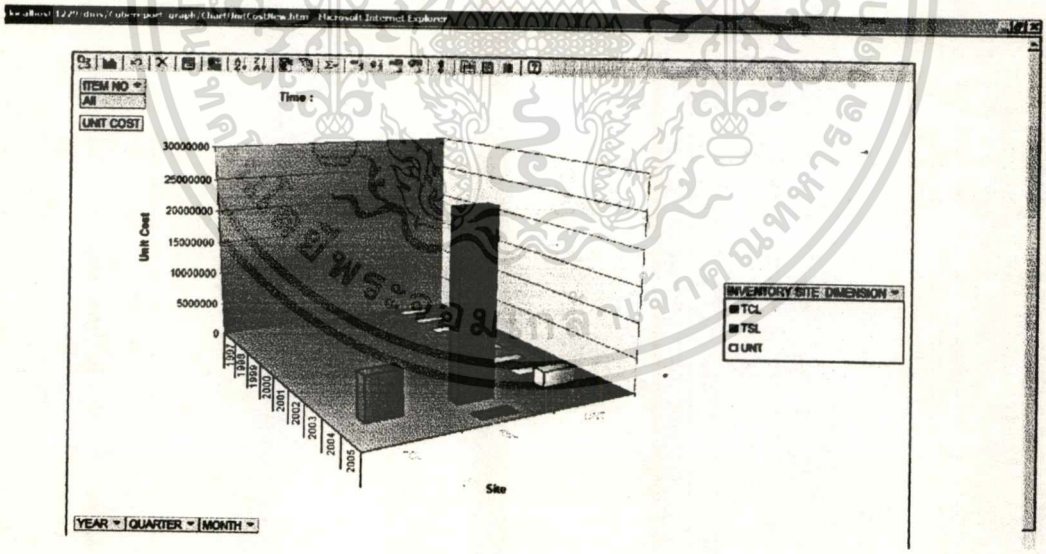
3. ในกรณีที่ user login ผ่านเข้ามาเรียบร้อยแล้ว ฟังก์ชัน Olap report เป็นฟังก์ชันการทำงานเพื่อเรียกดูข้อมูลซึ่งอยู่ในรูปของ Pivot Table และ Graph ซึ่งในเมนูด้านซ้ายมือ เป็น Report ต่าง ๆ Administrator และ User สามารถเรียกดูได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 4.25** ภาพแสดงหน้าจอเมื่อผู้ใช้เรียกดูรายงานที่แสดงผลเป็นตัวเลข

เมื่อผู้ใช้ต้องการดูข้อมูลสำหรับการวิเคราะห์ด้วยกราฟ สามารถคลิกที่ แสดงกราฟ ข้อมูลจะสามารถแสดงให้อยู่ในรูปดังต่อไปนี้




**รูปที่ 4.26** ภาพแสดงหน้าจอเมื่อผู้ใช้เรียกดูรายงานที่อยู่ในรูปของกราฟ

รายงานที่แสดงนั้น สร้างขึ้นเพื่ออำนวยความสะดวกแก่ผู้ใช้ ที่ต้องการทราบข้อมูลในมิติต่าง ๆ ดังนั้นรายงาน สามารถดูข้อมูลในระดับต่างๆ โดยกร Drill down เพื่อข้อมูลเชิงลึก และ drill up เพื่อเรียกดูข้อมูลเชิงตื้น หากต้องการวิเคราะห์ข้อมูลด้วยก็ย่อมสามารถทำได้เช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 5
# การสรุปผลการศึกษาวิเคราะห์ออกแบบและพัฒนา ระบบดาต้ามาร์ท สำหรับสินค้าคงคลัง กรณีศึกษาบริษัทคาโปรแลคตรัมไทย จำกัด(มหาชน)

เนื้อหาของบทนี้จะกล่าวถึงการสรุปผลการศึกษาและการพัฒนาระบบ โดยกล่าวถึงการสรุปผลการพัฒนาระบบหาอุปสรรคและปัญหาที่เกิดขึ้น ตลอดจนข้อเสนอแนะของการพัฒนาระบบ

## 5.1 สรุปผลการพัฒนาระบบ

สืบเนื่องมาจากการตระหนักถึงเทคโนโลยีสารสนเทศซึ่งมีบทบาทสำคัญในการบริหารงานและจัดการในองค์กรต่างๆ ทั้งในภาครัฐและเอกชน ทุกหน่วยงานได้พัฒนาระบบสารสนเทศในหน่วยงานของตนให้สามารถจัดเก็บและประมวลผลข้อมูล ซึ่งช่วยเพิ่มประสิทธิภาพการทำงานในองค์กรให้ทันกับความต้องการของตลาด ด้วยเหตุนี้ในปัจจุบันฝ่ายต่างๆ ในองค์กรจึงมีการจัดเก็บข้อมูลไว้มากมายอยู่ในรูปแบบเอกสารหรือตารางต่างๆ ในฐานข้อมูล แต่เนื่องจากข้อมูลเหล่านี้อาจจะกระจัดกระจายอยู่ในแต่ละฝ่ายในองค์กร นับวันข้อมูลเหล่านี้ก็จะมีปริมาณเพิ่มขึ้นเรื่อยๆ โดยไม่ได้นำไปใช้ให้เกิดประโยชน์ ในฝ่ายงานการบริหารสินค้าคงคลังของบริษัทคาโปรแลคตรัม จำกัด(มหาชน) ก็เช่นกัน มีการจัดเก็บและประมวลผลปริมาณสินค้าคงคลังในแต่ละวันหรือแต่ละเดือน เมื่อเวลาผ่านไปข้อมูลนี้ก็จะถูกเก็บไว้ต่างหากโดยไม่ได้นำมาใช้อีกทั้งๆ ที่ข้อมูลนี้อาจจะมีประโยชน์กับผู้บริหารในแง่ของการวางแผน เพื่อให้การดำเนินงานบรรลุวัตถุประสงค์ของการมีสินค้าไว้บริการลูกค้าในปริมาณที่เพียงพอทันต่อความต้องการของลูกค้า และสามารถลดระดับการลงทุนในสินค้าคลังให้ต่ำที่สุด นอกจากนี้อาจจะมีประโยชน์ในการกำหนดนโยบายหรือทิศทางการดำเนินงานของบริษัทอีกด้วย

จากการศึกษาวิเคราะห์ออกแบบและพัฒนาระบบ ได้สรุปผลการดำเนินงานดังนี้

1. นำกระบวนการที่ใช้ในการวิเคราะห์ข้อมูลเชิงธุรกิจ(Business Intelligence) มาใช้เพื่อศึกษาความต้องการในการใช้ข้อมูลตั้งแต่ระดับล่างสุด เพื่อนำมาจัดการเรียบเรียงวิเคราะห์ออกแบบ และสกัดสารสนเทศจากแหล่งข้อมูลต่างๆ มาแปลงเป็นข้อมูลชนิดเดียวกัน(Extract Transform and Load) โดยอาศัยโปรแกรม Sql Server Business Intelligence Development ให้อยู่ในรูป ด้าต้ามาร์ทสำหรับสินค้าคงคลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. สร้างเครื่องมือสำหรับการวิเคราะห์ข้อมูล ได้แก่ Cube, Dimension, Measures, Hierarchies โดยใช้ Microsoft Analysis Service.
3. วิเคราะห์หาคำตอบเชิงธุรกิจโดยอาศัยเทคโนโลยี OLAP เป็นเครื่องมือในการประมวลผล
4. ข้อมูลโดยการสร้างความสัมพันธ์ระหว่างมุมมองต่างๆ เพื่อศึกษาเหตุการณ์ปัจจุบัน และนำข้อมูลมาวิเคราะห์ทางสถิติ ค้นหาข้อมูล และทำรายงานเพื่อแสดงผล (Statistical Analysis Querying and Reporting) โดยใช้ Excel OWC version 1.0 แสดง Pivot Table
5. พัฒนาระบบคาต้ามาร์ทสำหรับสินค้าคงคลัง กรณีศึกษาบริษัทคาโปรแลคตรัมไทย จำกัด(มหาชน) โดยใช้เทคโนโลยีเว็บแอพพลิเคชั่นภาษา Microsoft Visual Studio .NET (C#)

## 5.2 อุปสรรคหรือปัญหาที่เกิดขึ้น

1. การให้ข้อมูลเชิงลึกของ บมจ. คาโปรแลคตรัมไทย เนื่องจากข้อมูลบางส่วนเป็นข้อมูลที่เป็นความลับไม่สามารถให้รายละเอียดได้ครบถ้วน ข้อมูลบางตัวจึงขาดความชัดเจน
2. ความเข้าใจที่คาดเคลื่อนกันระหว่างผู้ให้ข้อมูลของ บมจ. คาโปรแลคตรัมไทยกับผู้พัฒนา ทำให้การพัฒนาล่าช้าไปกว่ากำหนด
3. เครื่องคอมพิวเตอร์ที่ใช้ในการพัฒนาไม่เหมาะสมต่อแอพพลิเคชั่นด้านการทำคาต้ามาร์ทหรือคาต้าแวร์เฮาส์ เนื่องจากทรัพยากรหน่วยความจำต่ำไม่เพียงพอต่อการประมวลผล
4. การเข้าไปเก็บข้อมูลและหาความต้องการของผู้ใช้งานไม่สะดวก เนื่องจากบางครั้งไม่ค่อยได้รับความร่วมมือ
5. การนำเสนอเนื้อหาไม่สามารถนำมาแสดงได้ทั้งหมด เนื่องจากข้อมูลดังกล่าวเป็นข้อมูลซึ่งเป็นความลับของบริษัท ทำให้ไม่มีข้อมูลบางส่วนนำมาเป็นข้อมูลที่ละเอียดในรายงาน

## 5.3 ข้อเสนอแนะ

1. ก่อนการพัฒนาระบบ ควรจะศึกษาข้อมูลอย่างละเอียด รวมทั้งศึกษา ข้อดี ข้อเสียของการนำเทคโนโลยีที่เหมาะสมกับโปรเจคมาใช้ด้วย
2. ควรเลือกใช้ Hardware ที่เหมาะสมต่อลักษณะงานที่ต้องการพัฒนา เนื่องจากจะส่งผลกระทบโดยตรง ประสิทธิภาพ และประสิทธิผล ของผลที่ได้จากการพัฒนาแอพพลิเคชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ผู้พัฒนาควรเตรียมตัวศึกษาในเรื่องที่จะทำการพัฒนาเป็นอย่างดี และควรให้เวลาอย่างเต็มที่ เพื่อให้เกิดผลงานที่มีคุณภาพสูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บรรณานุกรม

กิตติพงศ์ กลมกล่อม. 2546. **การออกแบบและพัฒนาคลังข้อมูล.** กรุงเทพฯ : เคทีพี.

พิรพร หมุนสนิท และคณะ. 2549. **คัมภีร์ Visual Basic 2005.** กรุงเทพฯ : เคทีพี.

สุรสิทธิ์ คิวประสพศักดิ์ และนันทนี แขวงโสภา. 2546. **อินไซท์ Visual Basic .NET ฉบับสมบูรณ์.** กรุงเทพฯ : โปรวิชัน.

อำไพ สินลิขิตกุล. 2544. **อินไซด์ SQL Server 7.** กรุงเทพฯ : โปรวิชั่น.

Mattison, R. 1996. **Data Warehousing.** New York, NY : McGraw-Hill.

Microsoft Corporation. 2005. **Microsoft SQL Server 2005.** [Online]. Available: http://www.microsoft.com.

Gunderloy, M and Sneath, T. 2003. **SQL Server Developer's Guide to OLAP with Analysis Service.** Sybex, Amlamed.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ประวัติผู้เขียน

| | |
|---|---|
| ชื่อ-นามสกุล | นางสาว พิศสมร อิงควระ |
| ที่อยู่ | 318/76 สถานคุ้มครองสวัสดิภาพเด็กภาคตะวันออก ต.ห้วยโป่ง อ.เมือง จ.ระยอง 21150 |
| เบอร์ที่ติดต่อได้ | 089-8144866 |
| การศึกษา | มหาวิทยาลัยวลัยลักษณ์<br>สารสนเทศศาสตร์บัณฑิต (ระบบสารสนเทศเพื่อการจัดการ) |
| ประวัติการทำงาน | 2550 – ปัจจุบัน: บริษัท พีทีที โพลีเมอร์ มาร์เก็ตติ้ง จำกัด<br>2548 - 2549: บริษัท แอดวานซ์ รีเสิร์จ กรุ๊ป (AR Group)<br>2546 – 2549: บริษัท อนิลาเยม อินเตอร์เนชันแนล จำกัด |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้