

Analyzing Information System

01418321 System Analysis and Design
Chalothon Chootong (Ph.D.)

Department of Computer Science and Information, Faculty of
Science at Sriracha, Kasetsart University Sriracha Campus

chootong.c@ku.th



System Development Life Cycle

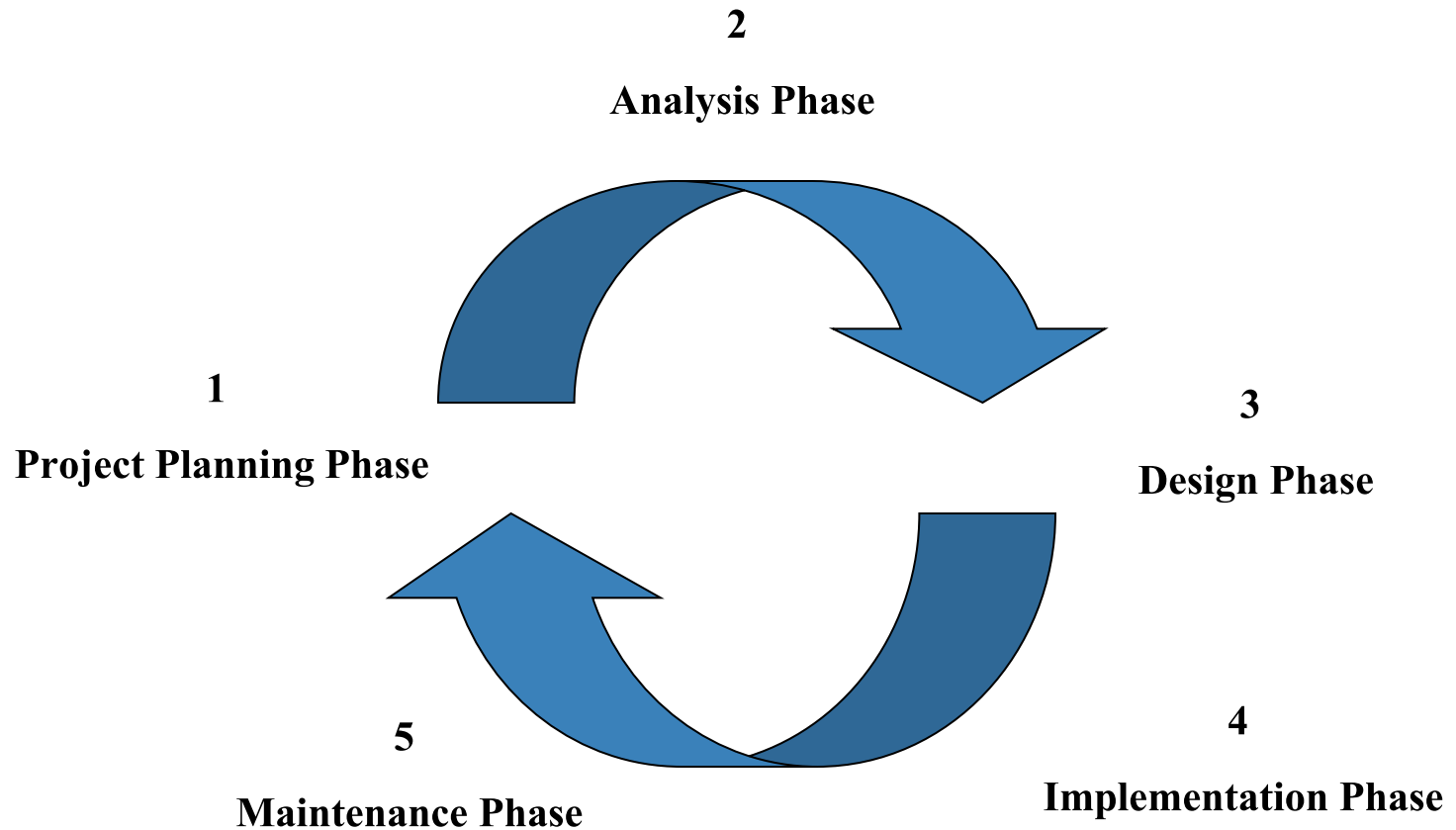
- ▷ การพัฒนาซอฟต์แวร์ ตามปกติจะประกอบไปด้วยกลุ่มกิจกรรม 3 ส่วนหลัก คือ
 - การวิเคราะห์ (**Analysis**)
 - การออกแบบ (**Design**)
 - การนำไปใช้ (**Implementation**)
- ▷ กิจกรรมทั้งสามนี้สามารถใช้งานได้ดีกับโครงการซอฟต์แวร์ที่มีขนาดเล็ก

System Development Life Cycle

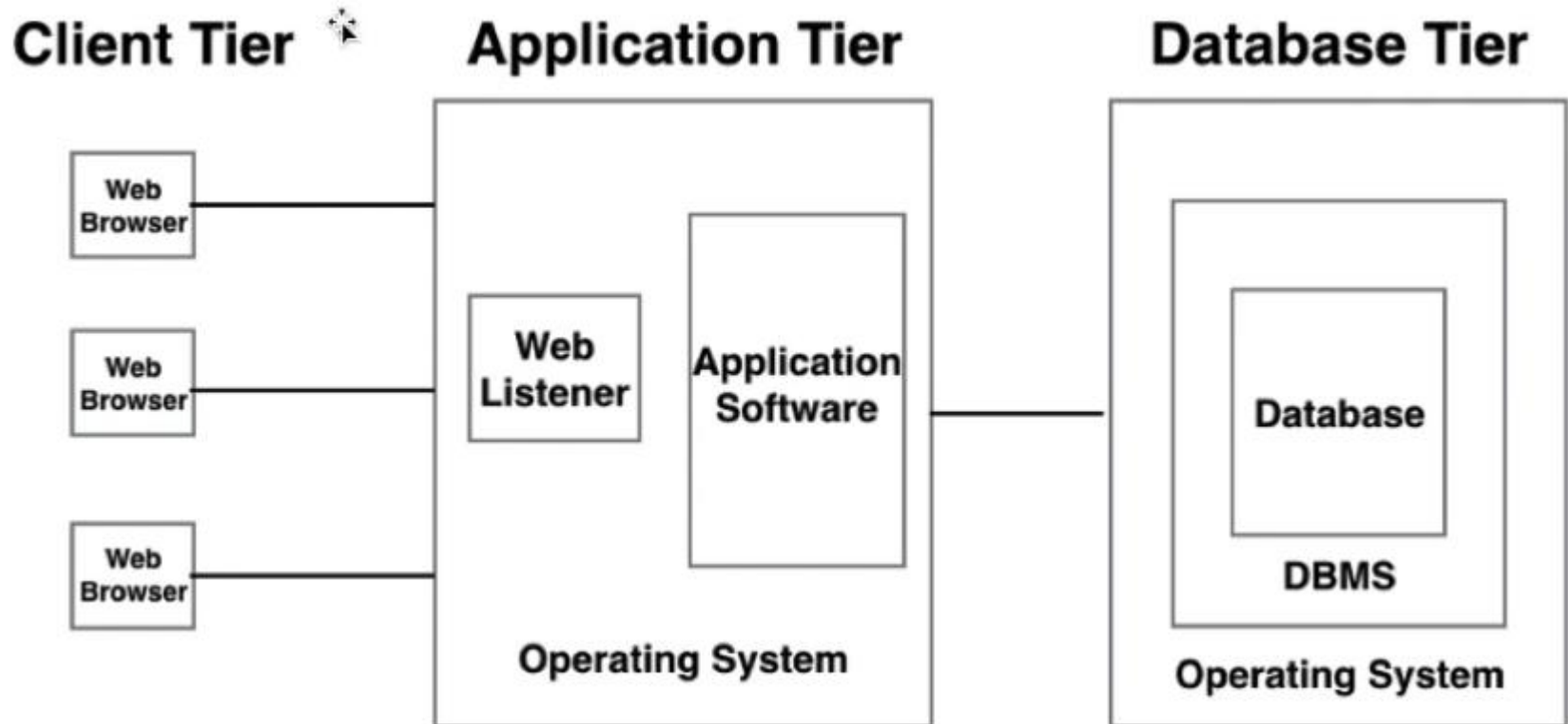
▷ โครงการซอฟต์แวร์ขนาดใหญ่ จำเป็นต้องใช้แบบแผนการพัฒนาซอฟต์แวร์ตามแนวทางของ **SDLC** จนครบทุกกิจกรรม ซึ่งประกอบด้วยระยะ

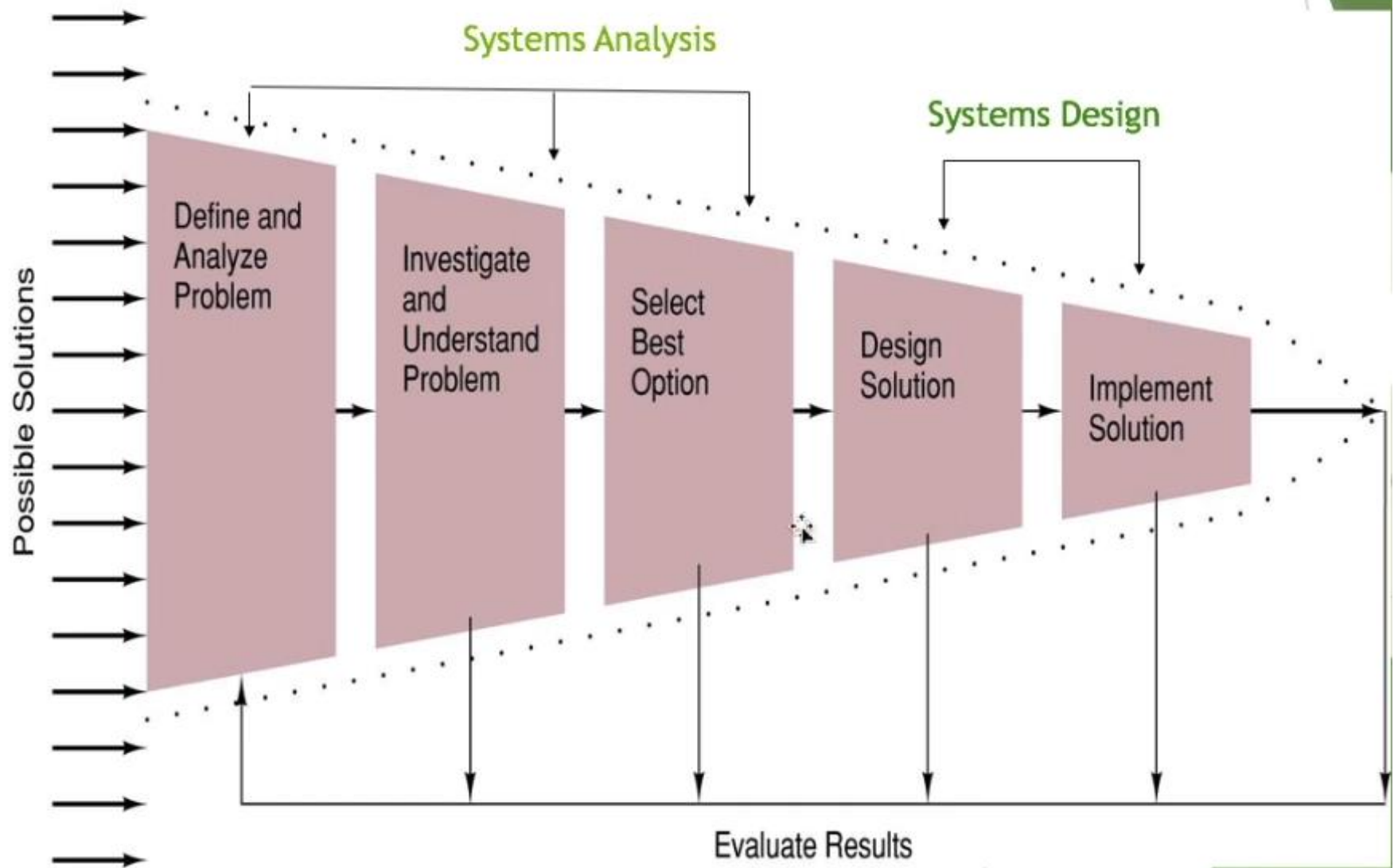
- ระยะที่ 1 : การวางแผนโครงการ (Project Planning Phase)
- ระยะที่ 2 : การวิเคราะห์ (Analysis Phase)
- ระยะที่ 3 : การออกแบบ (Design Phase)
- ระยะที่ 4 : การนำไปใช้ (Implementation Phase)
- ระยะที่ 5 : การบำรุงรักษา (Maintenance Phase)

System Development Life Cycle



Basic Business Software Architecture





► Systems Analysis

- Problem analysis (what)
- Information gathering (where & why)
- Decision making (how)
 - Establish objectives
 - Determine feasibility
 - Choose best solution

► Systems Design (Input, Process, Output, Procedures, Control)

- Logical design

Systems Design

- ▶ Logical design (what will the system do?)
 - ▶ Input: content, format, source, volume, frequency, timing
 - ▶ Process: rule, model, formula, timing
 - ▶ Output: content, format, organization, volume, freq., timing
 - ▶ Storage: data, format, organization, relationship, volume
 - ▶ Procedure: manual activities, rule, sequence, timing, location
 - ▶ Control: security, accuracy, validity, supervision
- ▶ Physical design (how the system will work?)
 - ▶ Input: keyboard, voice, scanner
 - ▶ Process: PC, operating system, software
 - ▶ Output: print-outs, files, audio
 - ▶ Storage: tape, CD
 - ▶ Procedure: batching, backup, auditing, data entry
 - ▶ Control: batch control, password, audit logs
- ▶ Implementation (coding, testing, training)



Planning Phase

Project Planning Phase

- ▷ การวางแผนโครงการ จะประกอบไปด้วยกิจกรรมต่างๆ ดังนี้
 - กำหนดปัญหา (**Problem Definition**)
 - จัดทำตารางกำหนดเวลาโครงการ (**Project scheduling**)
 - จัดตั้งทีมงานโครงการ (**Staff the project**)
 - ดำเนินการโครงการ (**Launch the project**)



Analysis Phase

Analysis Phase

- ▷ ระยะการวิเคราะห์จะต้องมีคำตอบเกี่ยวกับคำถามว่าใคร (Who) เป็นผู้ใช้ระบบ
- ▷ ระบบที่จะจัดทำขึ้นมีอะไรบ้าง (What)
- ▷ นักวิเคราะห์ระบบจะต้องดำเนินการในขั้นตอนการวิเคราะห์ระบบงานปัจจุบัน (Current System) เพื่อนำมาพัฒนาแนวความคิดสำหรับระบบใหม่ (New System)
- ▷ วัตถุประสงค์หลักของระยะการวิเคราะห์คือ จะต้องศึกษาและทำความเข้าใจในความต้องการต่างๆที่ได้รวบรวมมา

Analysis Phase

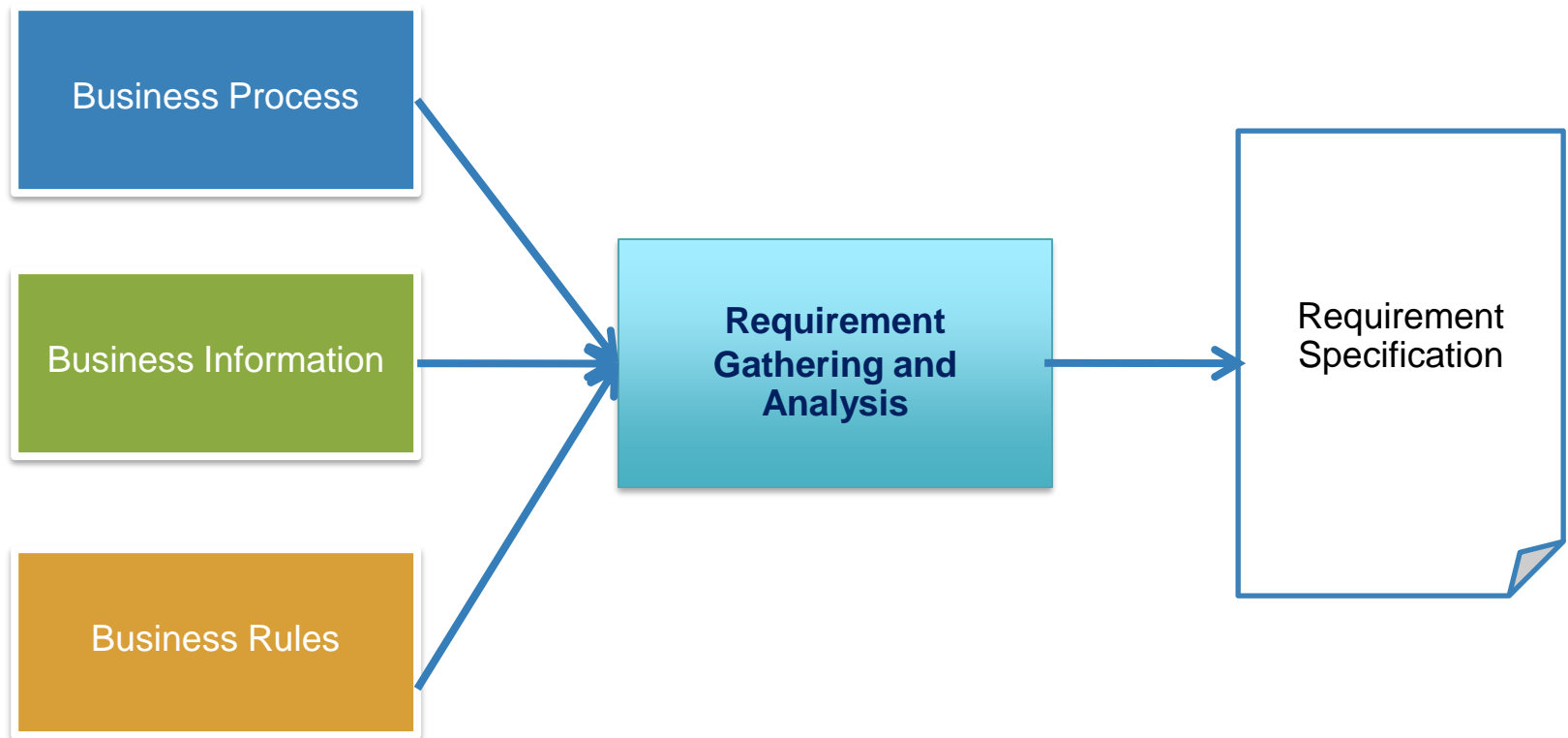
Requirement Gathering



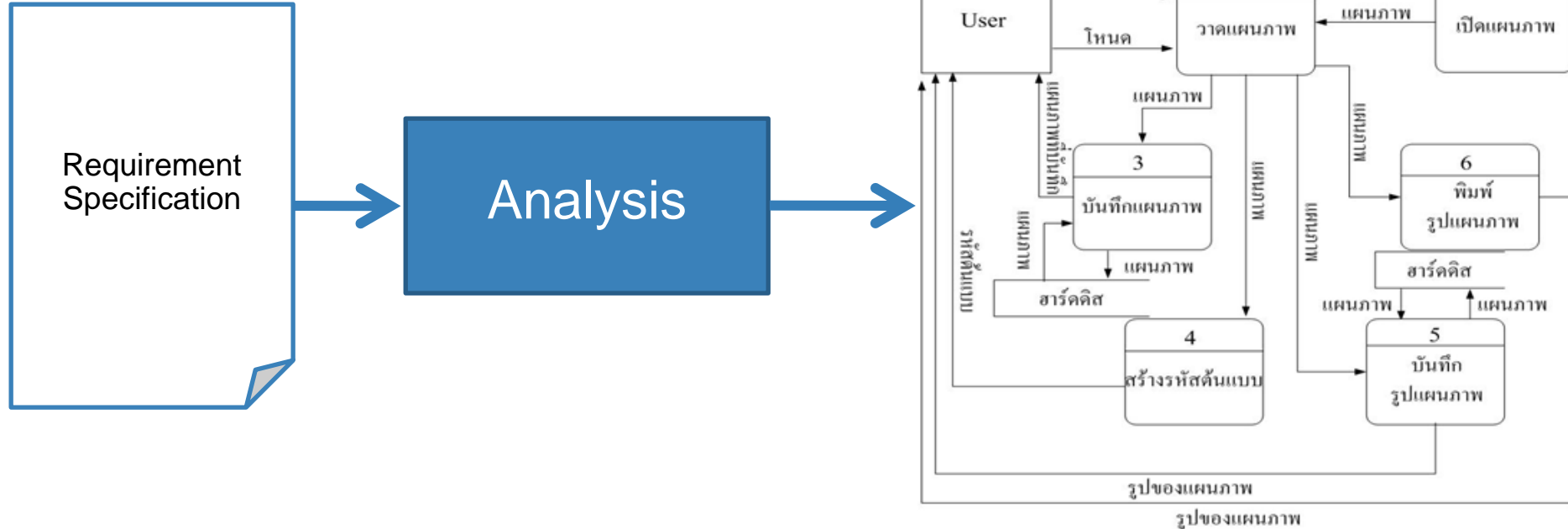
Analysis Phase

หากนักวิเคราะห์ระบบมิได้เอาใจใส่กับการรวบรวมความต้องการจากผู้ใช้ แต่มีการกำหนดความต้องการขึ้นเองโดยใช้ความคิดส่วนตัวของตนเองเป็นหลัก หรือประเมินความต้องการของผู้ใช้ระบบไม่ตรงตามวัตถุประสงค์ และหากมีการดำเนินการพัฒนาระบบต่อไปจนเสร็จสิ้น ระบบงานที่ได้ก็จะไม่ตรงกับความต้องการของผู้ใช้ระบบอย่างแท้จริง ทำให้ต้องมีการปรับแก้ หรือเปลี่ยนแปลงอยู่เสมอ

Analysis Phase



Analysis Phase



Analysis Phase

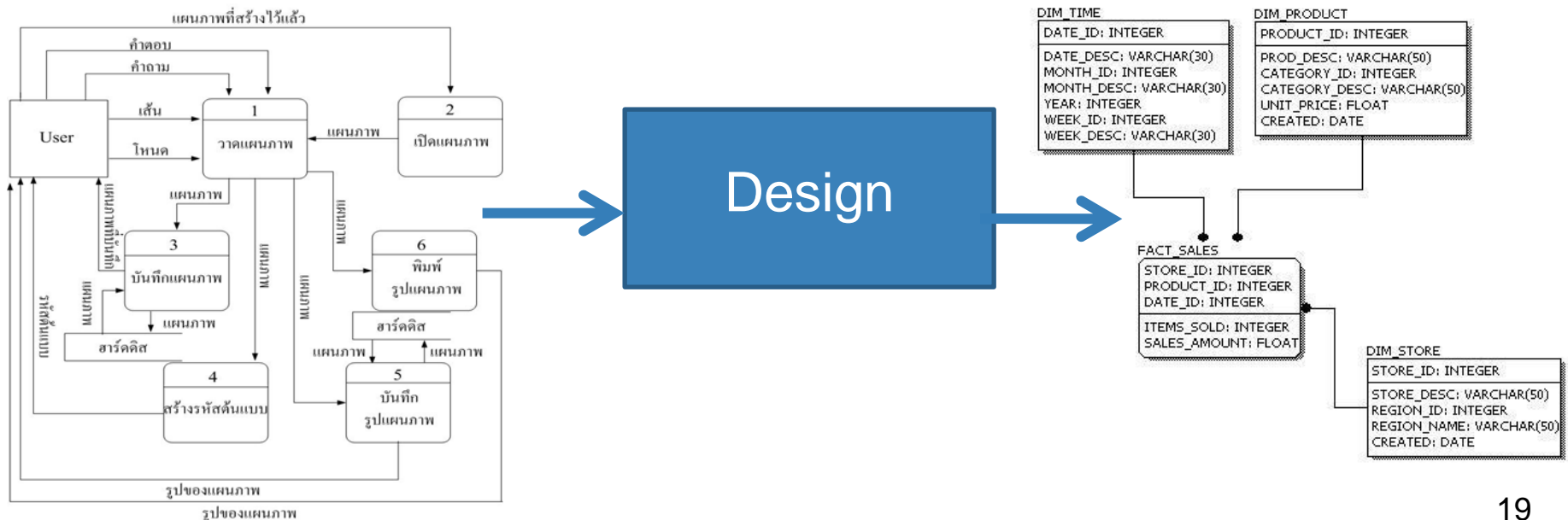
- สรุประยะของการวิเคราะห์ จะประกอบด้วยกิจกรรมต่างๆ ดังนี้
 - วิเคราะห์ระบบงานปัจจุบัน
 - รวบรวมความต้องการในด้านต่างๆ และนำมาวิเคราะห์เพื่อสรุปเป็นข้อกำหนดที่ชัดเจน
 - นำข้อกำหนดมาพัฒนาออกมาเป็นความต้องการของระบบใหม่
 - สร้างแบบจำลองกระบวนการทำงานของระบบใหม่ด้วยการวาดแผนภาพกระแสข้อมูล (Data Flow Diagram : DFD)
 - สร้างแบบจำลองข้อมูล ด้วยการวาดอีอาร์ไดอะแกรม (Entity Relationship Diagram : ERD)



Design Phase

Design Phase

- ระยะเวลาออกแบบ เป็นการพิจารณาว่า ระบบจะดำเนินการไปได้ได้อย่างไร (How)
- เกี่ยวข้องกับยุทธวิธีการออกแบบที่ว่าด้วยการตัดสินใจว่าจะพัฒนาระบบใหม่ด้วยแนวทางใด เช่น พัฒนาขึ้นเอง ซื้อมาโปรแกรมสำเร็จรูป หรือว่าจ้างบริษัทพัฒนาระบบให้



Design Phase

- พัฒนาขึ้นเอง
- ซื่อโปรแกรมสำเร็จรูป
- ว่าจ้างบริษัทพัฒนาระบบให้

การวิเคราะห์ มุ่งเน้นการแก้ปัญหาอะไร (What)

การออกแบบ มุ่งเน้นการแก้ปัญหายังไง (How)

Design Phase

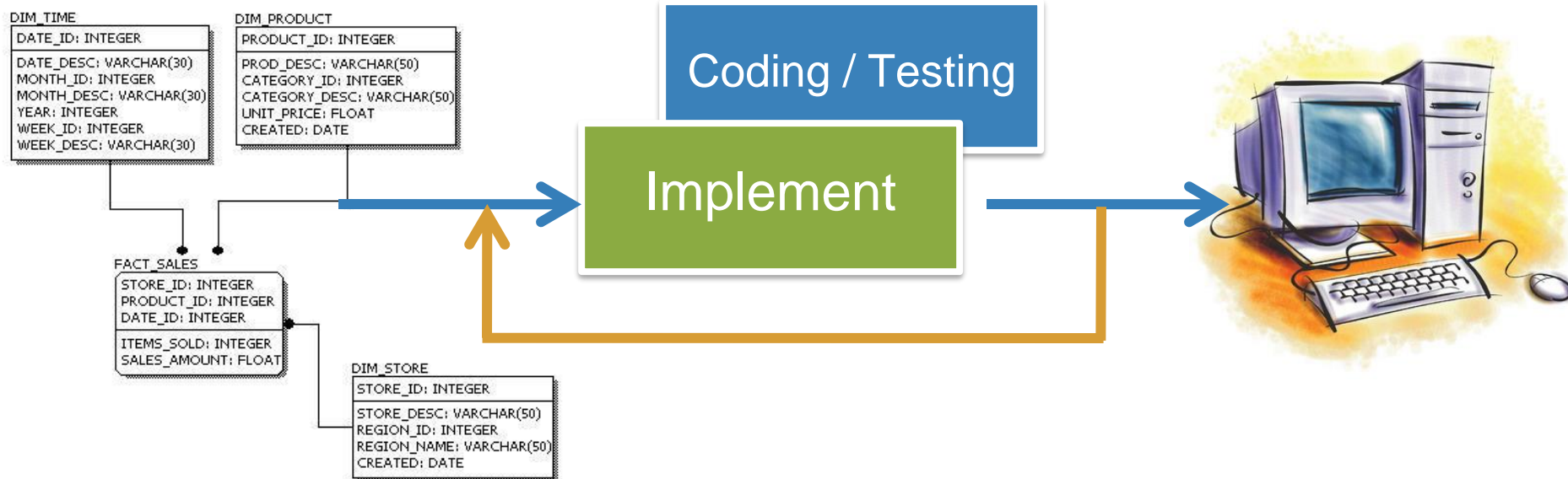
- การออกแบบทางสถาปัตยกรรม (Architecture Design) ที่เกี่ยวข้องกับอุปกรณ์ฮาร์ดแวร์ ซอฟต์แวร์ เครือข่าย
- การออกแบบรายงาน (Output Design)
- การออกแบบหน้าจอเพื่อปฏิสัมพันธ์กับผู้ใช้ (User Interface)
- การออกแบบอินพุต
- การออกแบบเอาต์พุต
- ฐานข้อมูล (Database)
- จัดทำต้นแบบ (Prototype)



Implementation Phase

Implementation Phase

- จะทำให้เกิดผลขึ้นมาด้วยการสร้างระบบ ทดสอบระบบ และการติดตั้งระบบ



Implementation Phase

- วัตถุประสงค์หลักของกิจกรรมในระยะนี้
 - ความน่าเชื่อถือของระบบ
 - ระบบต้องสามารถทำงานได้ดี
 - ผู้ใช้ต้องได้รับการฝึกอบรมเพื่อใช้งานระบบ
 - ความคาดหวังในองค์กรที่ต้องการผลตอบแทนในด้านดีกับการใช้ระบบใหม่



Maintenance Phase

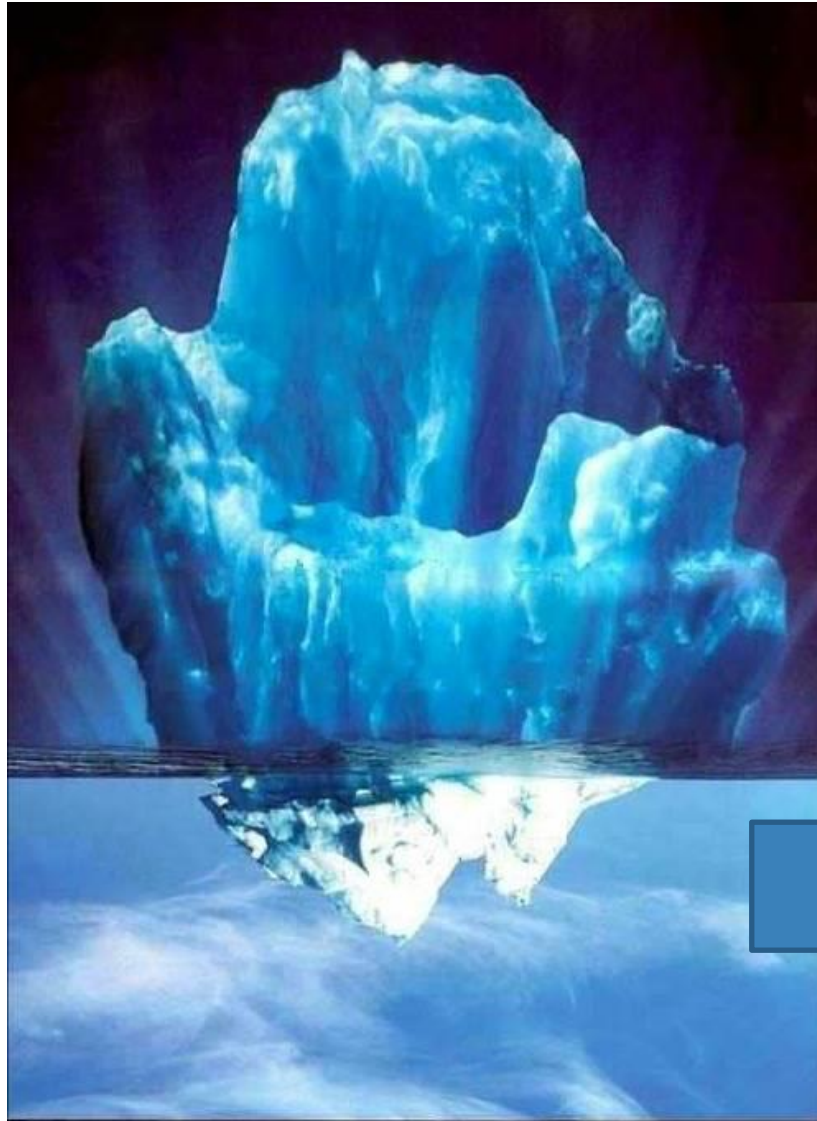
Maintenance Phase

- ☐ ระยะการบำรุงรักษา จะไม่นำเข้าไปรวมกับในส่วนของ SDLC จนกระทั่งหลังจากที่ระบบได้มีการติดตั้งเพื่อใช้งานแล้วเท่านั้น
- ☐ ระยะนี้จะใช้เวลายาวนานที่สุดเมื่อเทียบกับระยะอื่นๆ เนื่องจากระบบจะต้องได้รับการบำรุงรักษาตลอดระยะเวลาที่มีการใช้ระบบ
- ☐ สิ่งที่คาดหวังของหน่วยงานก็คือ ต้องการให้ระบบสามารถใช้งานได้ยาวนานหลายปี ระบบมีความสามารถรองรับเทคโนโลยีใหม่ๆ ในอนาคตได้
- ☐ ในช่วงระยะการบำรุงรักษา สามารถทำการเพิ่มเติมคุณสมบัติระบบให้มีประสิทธิภาพสูงขึ้น รวมถึงการแก้ไขปรับปรุงโปรแกรม ในกรณีที่เกิดข้อผิดพลาดที่พบบ่อย และการเขียนโมดูลการทำงานเพิ่มเติม

Maintenance Phase

ระยะการบำรุงรักษา มีค่าใช้จ่ายค่อนข้างสูง ดังนั้น หากระบบที่ไม่ได้รับการวางแผนที่ดีตั้งแต่เริ่มต้น จนกระทั่งได้ระบบที่พัฒนา
มาแล้วเสร็จ ทำให้เกิดปัญหาในด้านของระบบไม่สามารถ
ตอบสนองความต้องการของผู้ใช้งานอย่างแท้จริง ทำให้สูญเสีย
เวลาไปกับการปรับแก้ ทำให้เกิดค่าใช้จ่ายสูงขึ้นหลายเท่าตัว
รวมถึงการสูญเสียเวลา และการสูญเสียโอกาส

Maintenance Phase



- Planning
- Analysis
- Design
- Implement

Maintenance

Maintenance

System Development Methodology

- ▷ นักวิเคราะห์ระบบ สามารถนำเครื่องมือต่างๆ มาประยุกต์ใช้กับการวิเคราะห์และออกแบบ เรียกว่า “Methodology”
- ▷ เป็นแนวทางที่มีการนำ
 - โมเดล
 - เครื่องมือ
 - เทคนิค

Models

- ☐ ผังงาน (Flowchart)
- ☐ แผนภาพกระแสข้อมูล (Data Flow Diagram)
- ☐ อีอาร์ไคอะแกรม (ER Diagram)
- ☐ ยูสเคสไคอะแกรม (Use Case Diagram)
- ☐ คลาสไคอะแกรม (Class Diagram)
- ☐ ซีควนส์ไคอะแกรม (Sequence Diagram)
- ☐ แผนภาพแกนต์/เพิร์ต (Gantt Chart/PERT)
- ☐ โมเดลวิเคราะห์ทางการเงิน เช่น NPV , ROI

Tools

- ▷ ประกอบด้วยซอฟต์แวร์หรือโปรแกรมที่ใช้สนับสนุนการใช้งานในการพัฒนาระบบ
- ▷ สามารถนำมาใช้เพื่อสร้างแบบจำลองหรือโมเดลต่างๆ รวมถึงส่วนประกอบอื่นๆ
 - โปรแกรมจัดการโครงการ
 - โปรแกรม/เครื่องมือช่วยวาด
 - โปรแกรมประมวลผลคำ หรือโปรแกรมเอดิเตอร์
 - โปรแกรมจัดการฐานข้อมูล
 - โปรแกรมแปลงไดอะแกรมเป็นรหัสคำสั่ง

Techniques

- ▷ คือกลุ่มแนวทางที่ช่วยในการชี้แนะ (Guidelines) เพื่อให้นักวิเคราะห์ระบบสามารถนำเทคนิคไปดำเนินการพัฒนาระบบเพื่อให้เกิดความสมบูรณ์มากยิ่งขึ้น เช่น
- เทคนิคการบริหารโครงการ
 - เทคนิคการสัมภาษณ์
 - เทคนิคการสร้างแบบจำลองข้อมูล
 - เทคนิคการออกแบบฐานข้อมูลเชิงสัมพันธ์
 - เทคนิคการทดสอบซอฟต์แวร์
 - เทคนิคการวิเคราะห์และออกแบบระบบเชิงวัตถุ

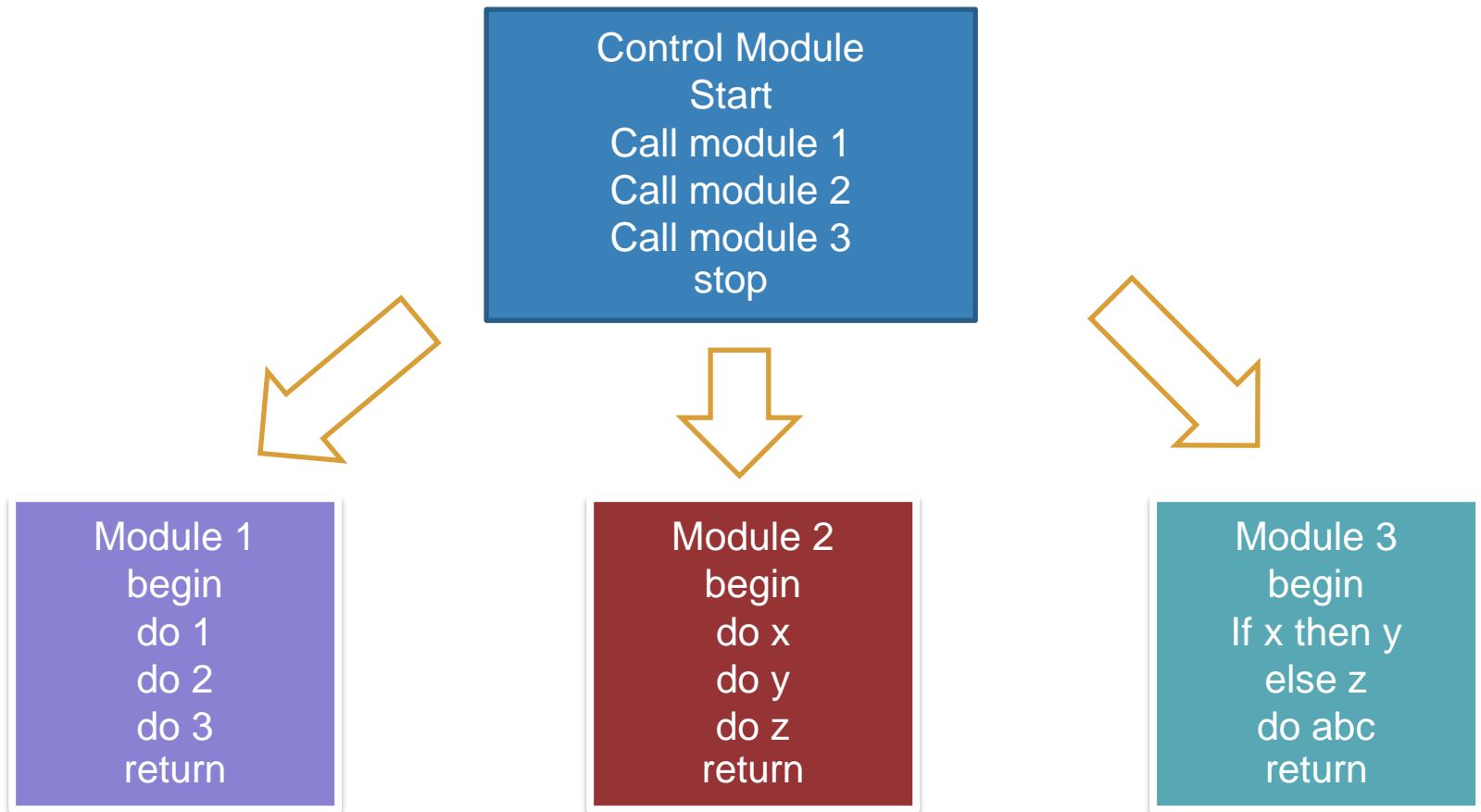
วิธีการพัฒนาระบบ

- ▷ วิธีการพัฒนาระบบโดยทั่วไป จะประกอบด้วย 2 วิธีด้วยกัน คือ
 - วิธีการพัฒนาระบบแบบดั้งเดิม (The Traditional Approach)
 - วิธีการพัฒนาระบบเชิงวัตถุ (The Object-Oriented Approach)

The Traditional Approach

- ▷ การพัฒนาระบบเชิงโครงสร้าง (Structured System Development)
- ▷ แนวทางในการโปรแกรมเชิงโครงสร้าง ประกอบด้วย
 - ชุดคำสั่งที่เรียงเป็นลำดับ (Sequence)
 - ชุดคำสั่งที่มีการตัดสินใจ (Decision)
 - ชุดคำสั่งที่มีการทำงานเป็นลูปหรือการทำซ้ำ (Repetition)
- ▷ นอกจากนี้ยังมีแนวความคิดที่สัมพันธ์กับการโปรแกรมเชิงโครงสร้าง เรียกว่า Top-Down Programming

The Traditional Approach



Structured Design

- ▷ ระบบงานในปัจจุบันได้มีความซับซ้อนมากขึ้น
- ▷ เทคนิคการออกแบบเชิงโครงสร้างได้พัฒนาขึ้นเพื่อเป็นแนวทางในการตัดสินใจว่า โปรแกรมควรมีลำดับชั้นอย่างไร และมีโมดูลอะไรบ้าง รวมถึงมีการระบุข้อมูลที่ต้องมีการส่งผ่านกันระหว่างโมดูล
- ▷ มีการอธิบายโดยใช้ **ผังโครงสร้าง (Structure Chart)**

The Object-Oriented Approach

- ▷ เป็นวิธีใหม่ของการพัฒนาระบบ
- ▷ ในการวิเคราะห์และออกแบบที่เป็นแบบโครงสร้างมีการแบ่งส่วนของข้อมูลและส่วนของกระบวนการแยกออกจากกัน
- ▷ เทคนิควิธีการพัฒนาระบบเชิงวัตถุ จะเป็นการมองวัตถุหรือออบเจกต์หนึ่ง เป็นแหล่งรวมของข้อมูล และกระบวนการ

The Object-Oriented Approach

- ▷ ประกอบไปด้วย 3 แนวทางคือ
 - การวิเคราะห์ระบบด้วยวิธีเชิงวัตถุ
(Object-Oriented Analysis : OOA)
 - การออกแบบระบบด้วยวิธีเชิงวัตถุ
(Object-Oriented Design : OOD)
 - การเขียนโปรแกรมเชิงวัตถุ
(Object-Oriented Programming : OOP)

ซอฟต์แวร์ที่มีคุณภาพประกอบด้วย

- ▷ มีความถูกต้อง (Correctness) ตรงกับความต้องการ
- ▷ มีความน่าเชื่อถือ (Reliability) ความน่าเชื่อถือในผลลัพธ์
- ▷ ใช้งานง่าย (User Friendliness)
- ▷ บำรุงรักษาง่าย (Maintainability) ในอนาคตปรับปรุงง่าย
- ▷ สามารถนำกลับมาใช้งานใหม่ได้ (Reusability)
- ▷ มีความคงทน (Robustness) สามารถทำงานได้ตามปกติ แม้เกิดปัญหา
- ▷ มีประสิทธิภาพ (Efficiency) การทำงานดีขึ้นกว่าเดิม
- ▷ มีความสะดวกในการเคลื่อนย้าย (Portability)
- ▷ มีความปลอดภัย (Security/Safety)

Thanks!

Any questions?