



ครั้งที่ 6

การออกแบบระบบ

การออกแบบระบบคืออะไร

- การเปลี่ยนปัญหาไปเป็นรายละเอียดของการแก้ปัญหา
- ผลของการออกแบบอาจอยู่ในรูปของขั้นตอนของการประมวลผล (process) ที่สามารถแก้ปัญหาตามที่ลูกค้าต้องการได้

การออกแบบระบบนั้นสามารถแบ่งการประมวลผล เป็น 2 ส่วน คือ

■ Conceptual system design

เป็นการออกแบบระบบโดยบรรยายถึง
หน้าที่ต่างๆ(function) ที่
ระบบสามารถกระทำได้

■ Technical design

เป็นการออกแบบระบบทางด้านเทคนิค
ซึ่งบรรยายถึงรูปแบบ(form) ของ
ระบบที่สามารถกระทำได้ ในส่วน
ของสถาปัตยกรรมระบบ ,ซอฟต์แวร์
ที่ต้องการ, การติดต่อสื่อสาร,
ข้อมูลเข้าและออกของระบบ,
รูปแบบของรายงาน, โครงสร้างของ
ข้อมูล

Technical design

system architecture

บรรยายถึงองค์ประกอบหลัก
ทางฮาร์ดแวร์ และ หน้าที่ที่
กระทำในระบบ

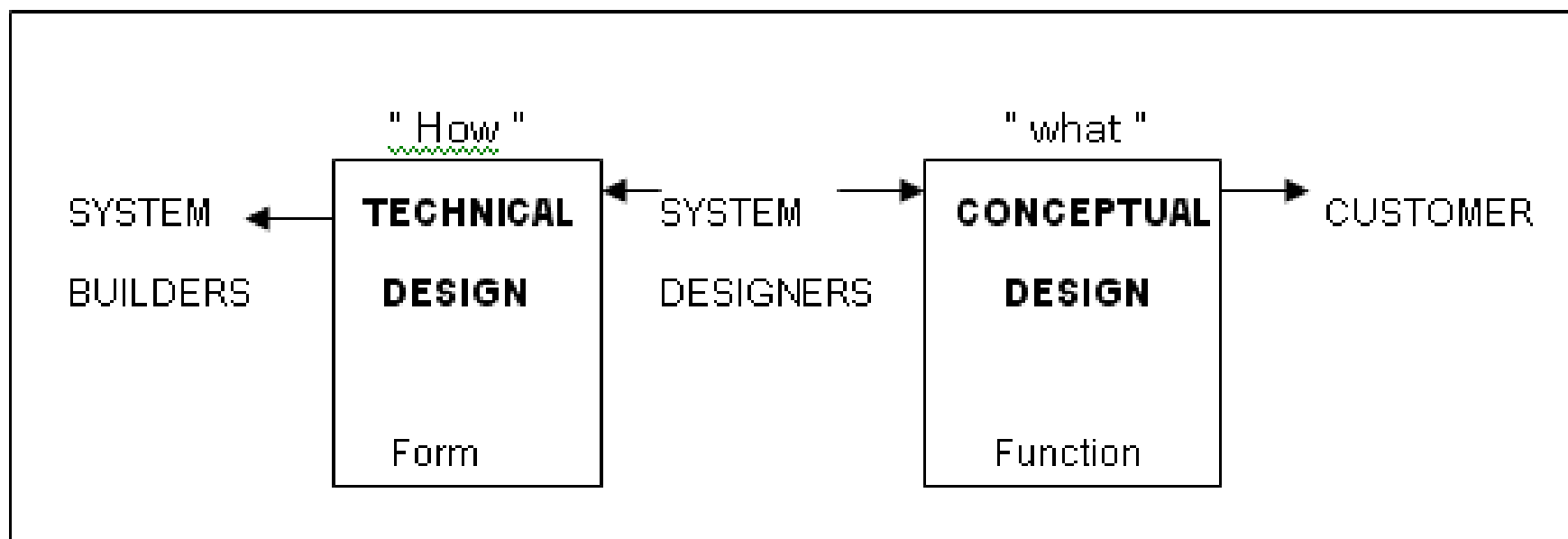
system software structure

บรรยายถึงองค์ประกอบต่างๆของ
ซอฟต์แวร์หน้าที่, ลำดับชั้น ,
ความเกี่ยวพันของหน้าที่ต่างๆ

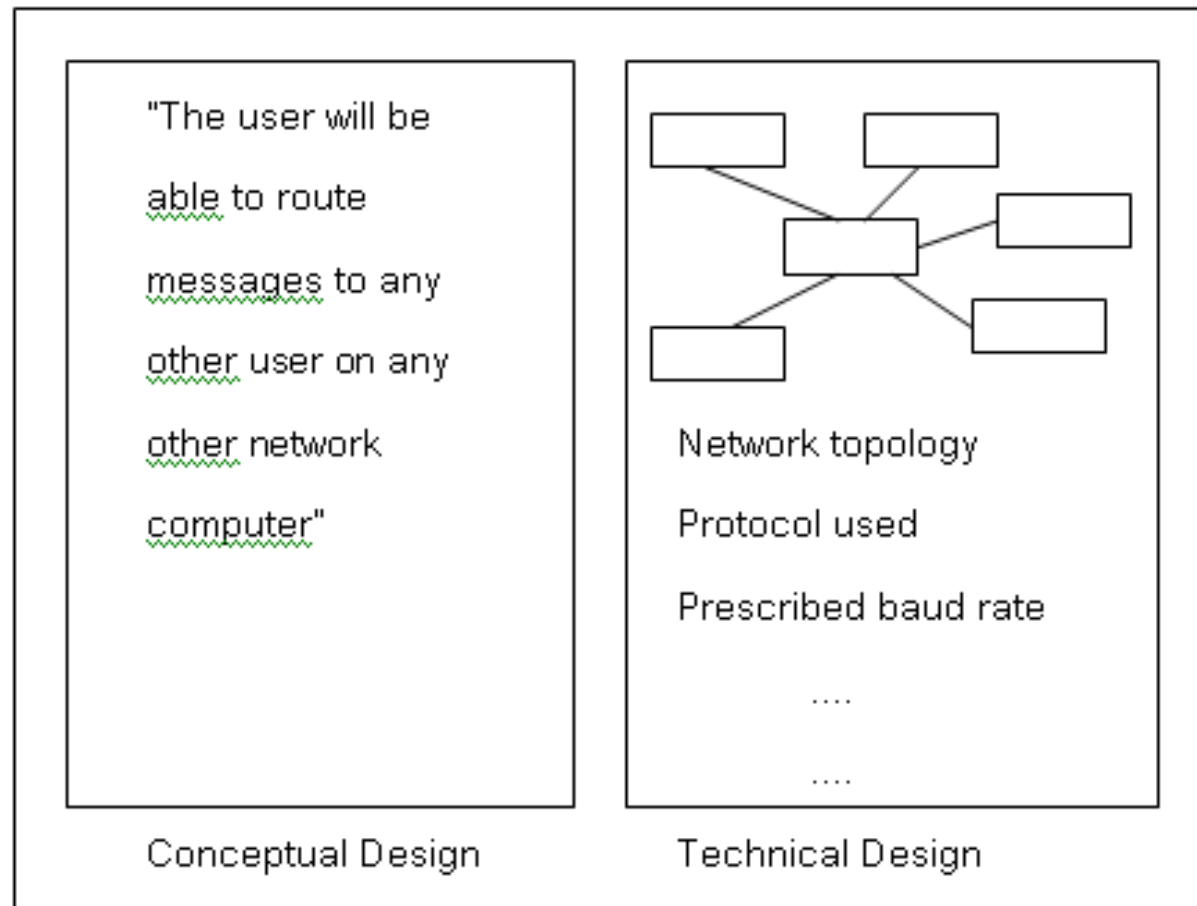
data

บรรยายถึงโครงสร้างข้อมูล และ
การไหลของข้อมูลในระบบ

รูปภาพที่ 4.1 Two Parts of System Design

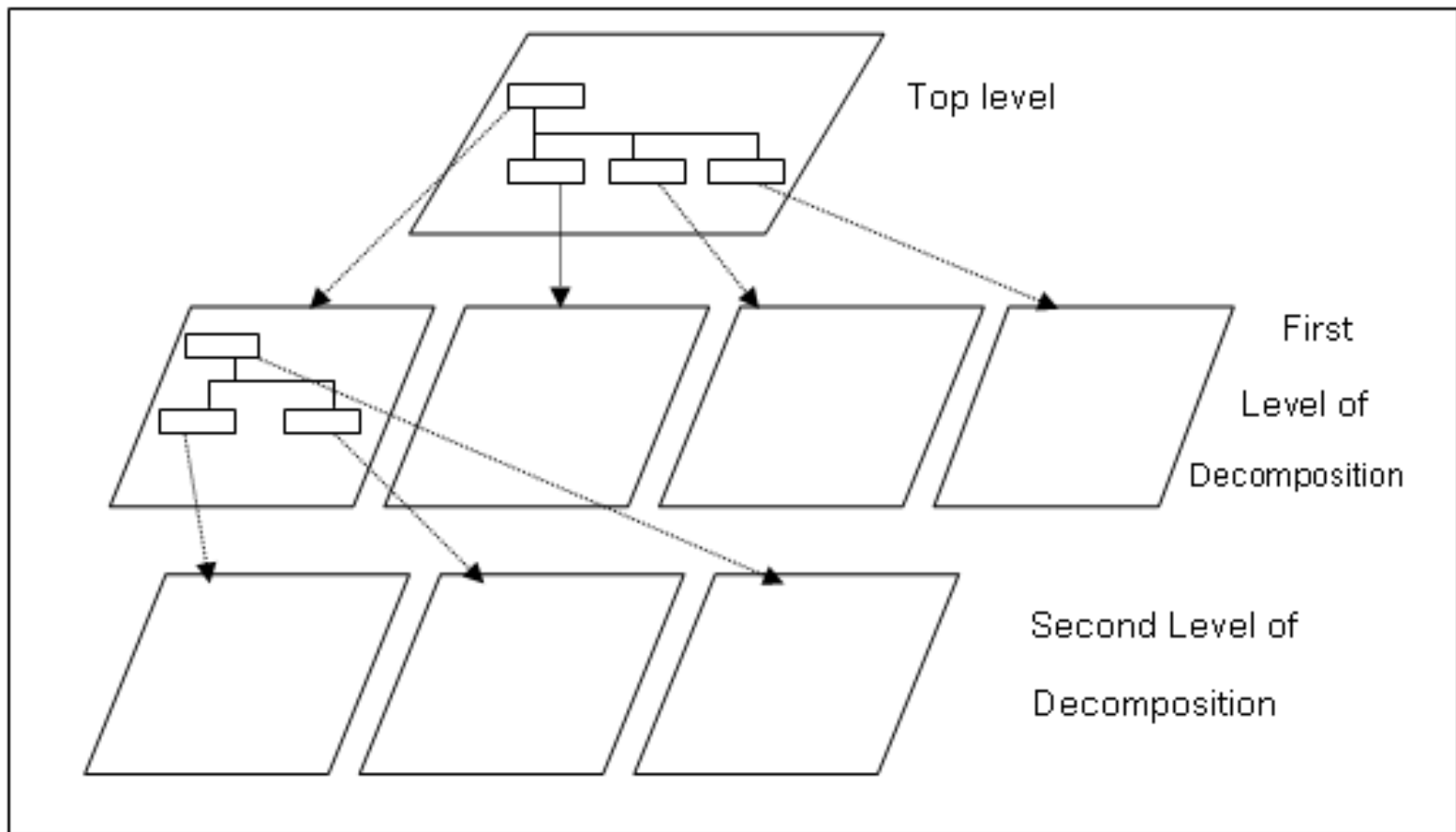


รูปภาพที่ 4.2 Conceptual Design vs. Technical Design



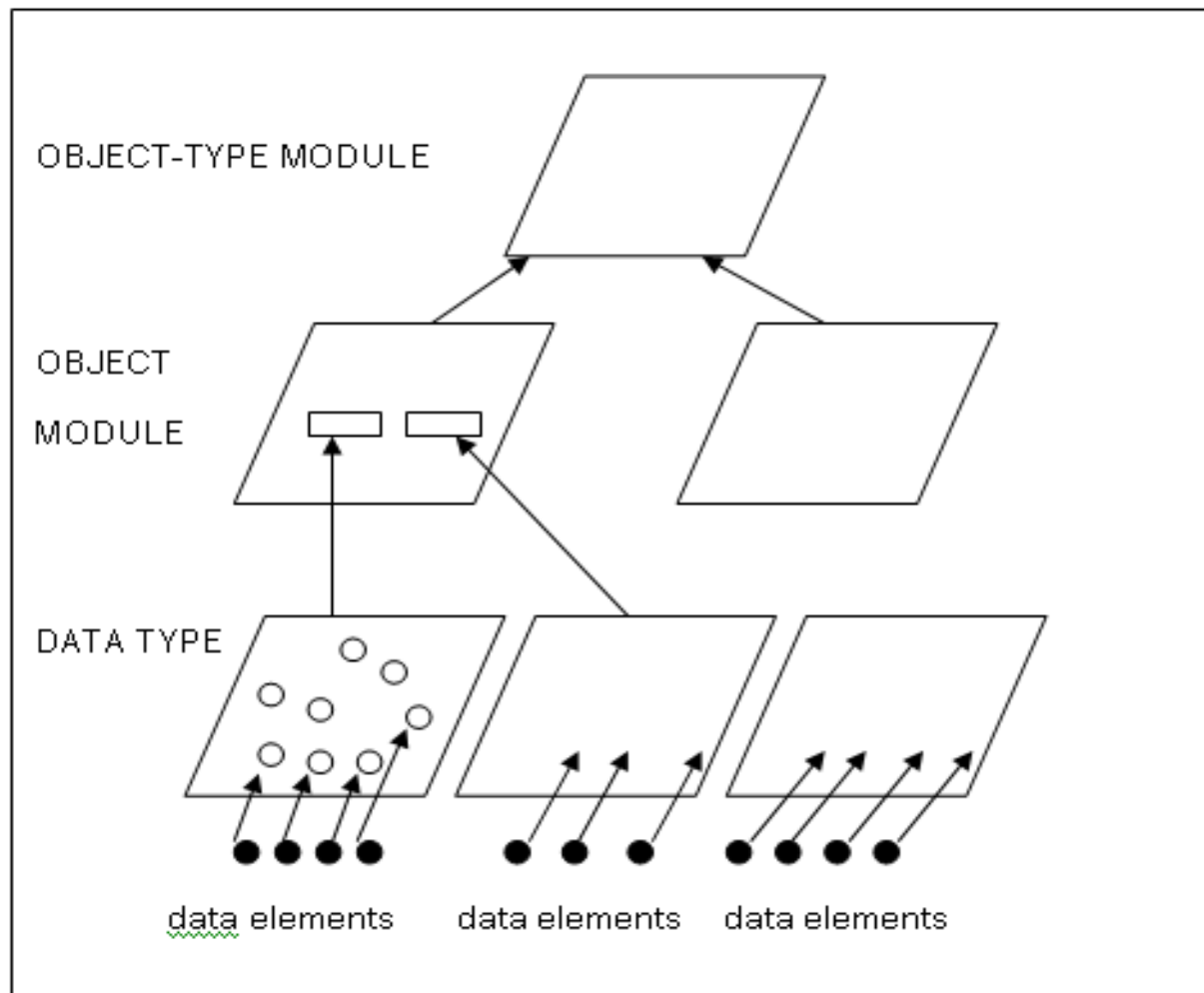
วิธีการออกแบบระบบ

รูปภาพที่ 4.3 Decomposition of System into Levels



วิธีการออกแบบระบบ

รูปภาพที่ 4.4 Composition of a System from Data



คุณลักษณะของการออกแบบระบบที่ดี

■ Modularity

- แบ่งปัญหาของระบบออกเป็น โมดูลย่อยๆ โดยแต่ละ โมดูลให้มีความเกี่ยวพันกันน้อยที่สุด
- ให้ผลลัพธ์การทำงานของแต่ละ โมดูลมีเพียงวัตถุประสงค์เดียว
- ทำให้เราสามารถแก้ปัญหาแต่ละ โมดูลย่อยได้ง่าย

คุณลักษณะของการออกแบบระบบที่ดี

■ Levels of Abstraction

- จัดระดับของโมดูล ซึ่งโมดูลในระดับต่ำกว่าเป็นรายละเอียดที่กำหนดจากโมดูลในระดับที่สูงกว่า ทำให้เข้าใจถึงปัญหาที่ซ่อนอยู่ซึ่งสามารถติดตามได้เมื่อเลื่อนไปยังระดับที่ต่ำลงมา ทำให้ง่ายต่อการเปลี่ยนแปลงแก้ไข
- สำหรับโมดูลในระดับสูงสุดจะสำคัญที่สุดเพราะเราสามารถหารายละเอียดของปัญหาที่ซ่อนอยู่ในระดับที่ต่ำลงมาได้

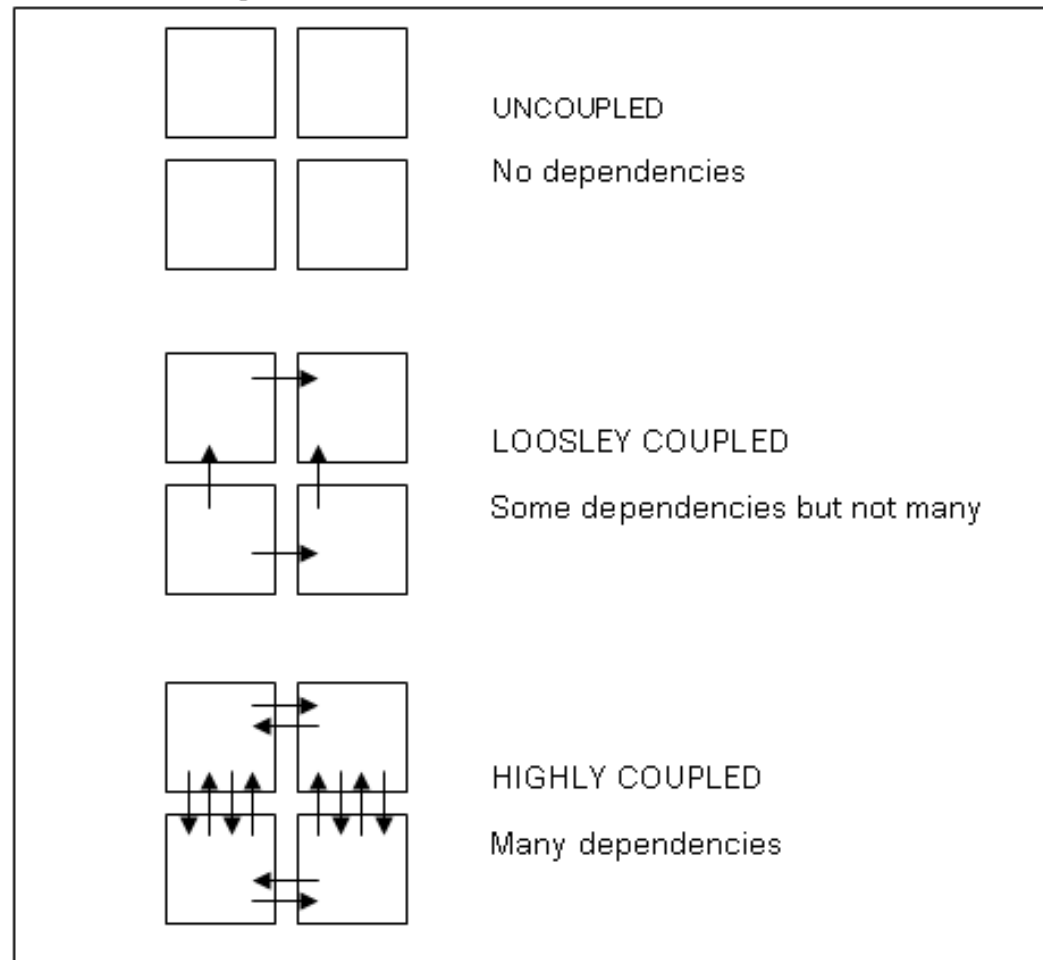
คุณลักษณะของการออกแบบระบบที่ดี

■ Coupling

- เป็นการวัดความสัมพันธ์ระหว่างโมดูล การออกแบบที่ดีนั้น ความสัมพันธ์ระหว่างโมดูลนั้นต้องมีความเกี่ยวพันกันน้อยที่สุด

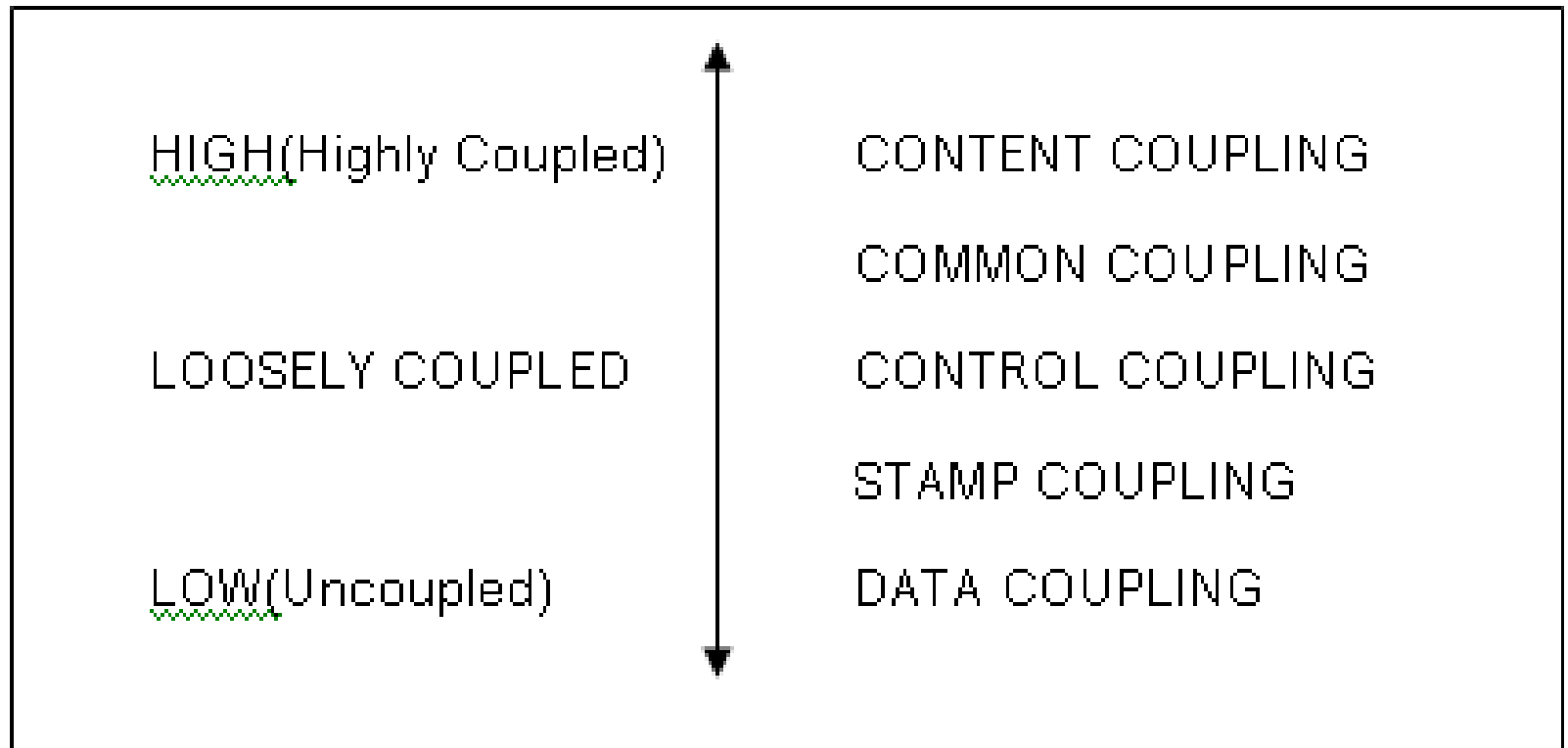
Degree of complexity

รูปภาพที่ 4.5 Independent vs. Dependent Modules



ความเกี่ยวพันของ โมดูลนั้นเราสามารถแบ่งออกเป็น 5 ชนิด

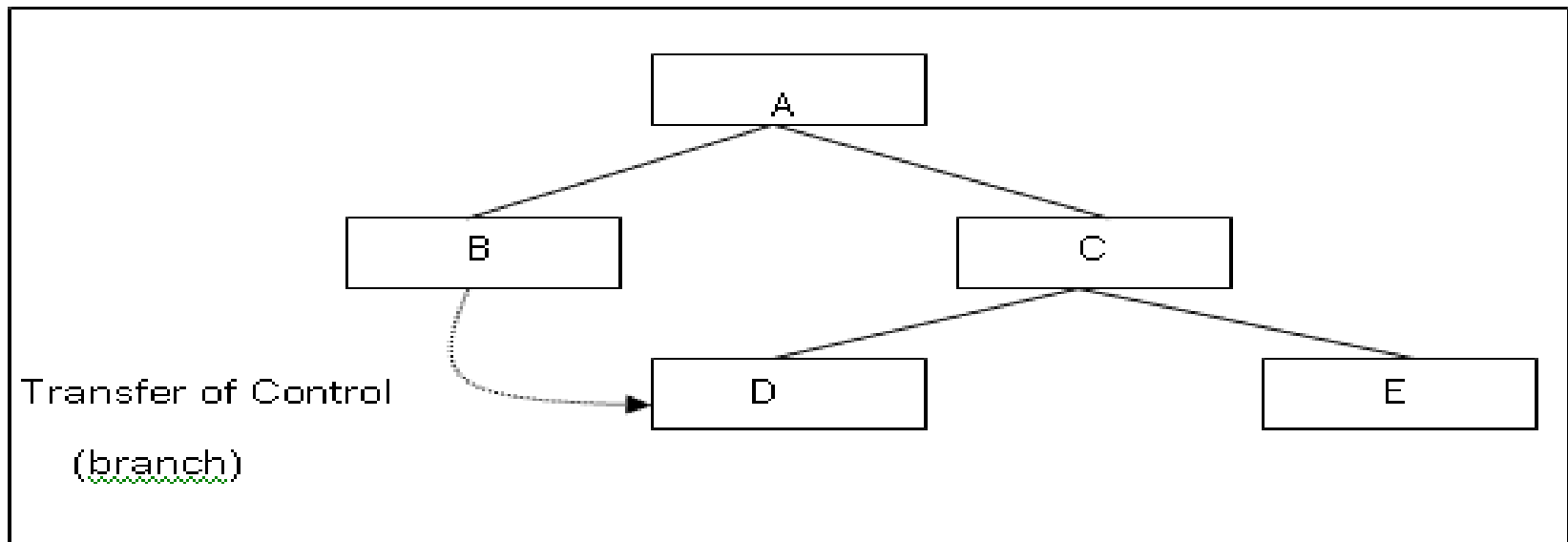
รูปภาพที่ 4.6 Types of Coupling



ความเกี่ยวพันของโมดูล

- **Content coupling** มีความเกี่ยวพันระหว่างโมดูลสูงมาก อาจเป็นเหตุการณ์ที่มีการแก้ไขในโมดูลหนึ่งจะส่งผลต่อการปฏิบัติงานในโมดูลอื่นๆด้วย

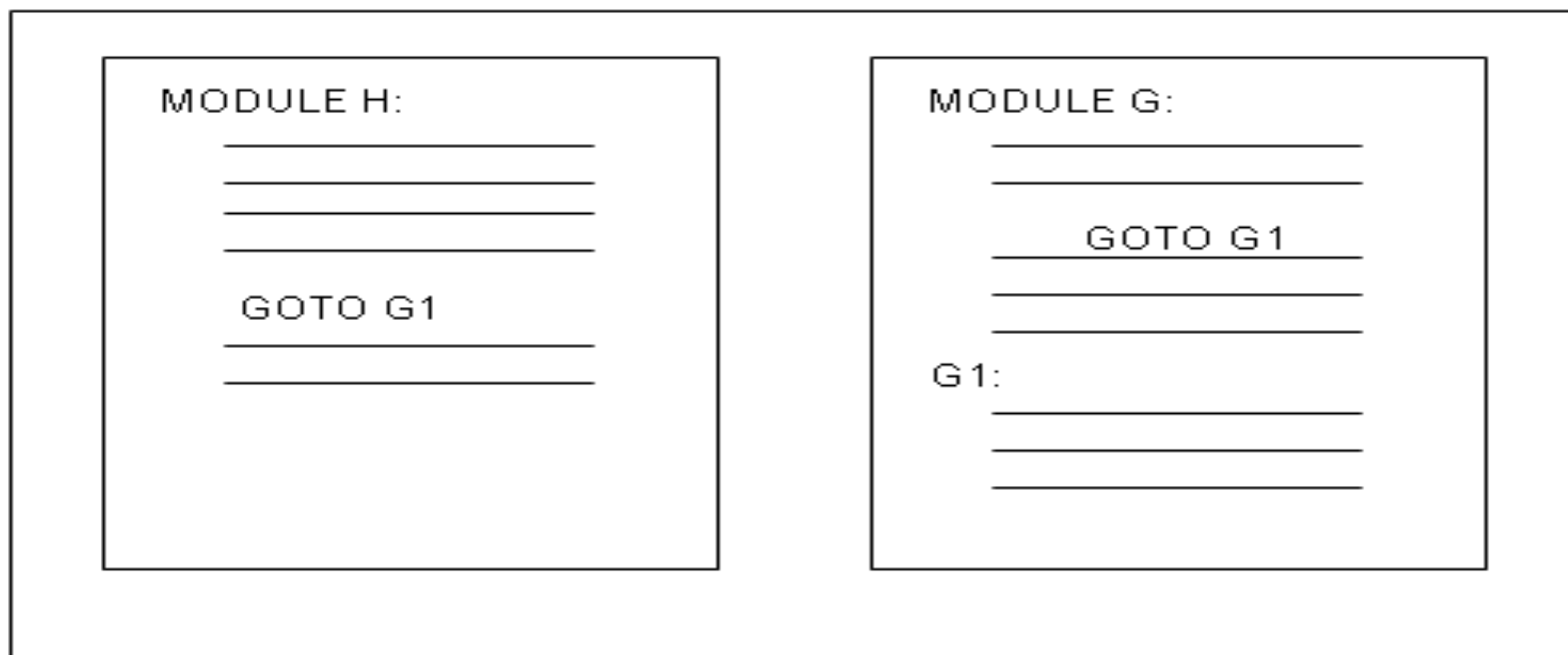
รูปภาพที่ 4.7 Content Coupling



ความเกี่ยวพันของโมดูล

- **Content coupling** มีความเกี่ยวพันระหว่างโมดูลสูงมาก อาจเป็นเหตุการณ์ที่มีการแก้ไขในโมดูลหนึ่งจะส่งผลต่อการปฏิบัติงานในโมดูลอื่นๆด้วย

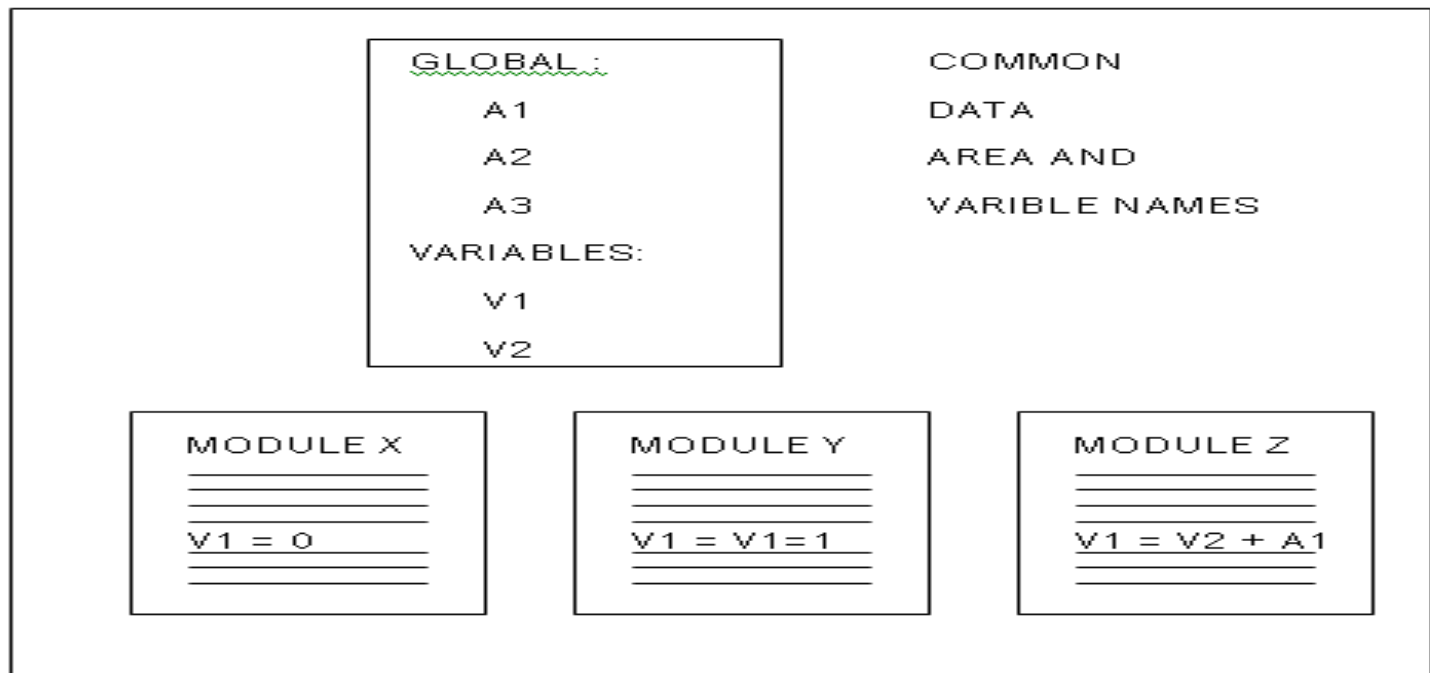
รูปภาพที่ 4.8 Content Coupling Involving Flow of Control



ความเกี่ยวพันของโมดูล

- **Common coupling** เราสามารถออกแบบเพื่อลดความเกี่ยวพันกันระหว่างโมดูลลงได้โดยนำข้อมูลที่มีการใช้งานร่วมกันไปไว้ในส่วนของ **global** หรือ **common data area**

รูปภาพที่ 4.9 Common Coupling



ความเกี่ยวพันของโมดูล

- **Control coupling** เป็นความเกี่ยวพันที่โมดูลหนึ่งส่งสัญญาณเพื่อไปควบคุมกิจกรรมในโมดูลอื่นๆ
- ควรออกแบบให้ปริมาณของการควบคุมน้อยที่สุดเท่าที่จะเป็นไปได้

ความเกี่ยวพันของโมดูล

- **Stamp coupling** เป็นความเกี่ยวพันระหว่างโมดูลเมื่อโมดูลหนึ่งมีการส่งผ่านโครงสร้างข้อมูลไปให้แก่โมดูลอื่น อาจเป็นกลุ่มของสัญญาณควบคุม หรือ โครงสร้างข้อมูลก็ได้

ความเกี่ยวพันของโมดูล

- **Data coupling** เป็นการเกี่ยวพันที่โมดูลหนึ่งส่งผ่านเพียงข้อมูล หรือรูปแบบพื้นฐาน
- ซึ่งถือว่ามีคุณสมบัติ **Coupling** ที่ดีที่สุดเพราะเกิดข้อผิดพลาดได้น้อยมากและถ้ามีความผิดพลาดเกิดขึ้น สามารถติดตามได้โดยง่าย

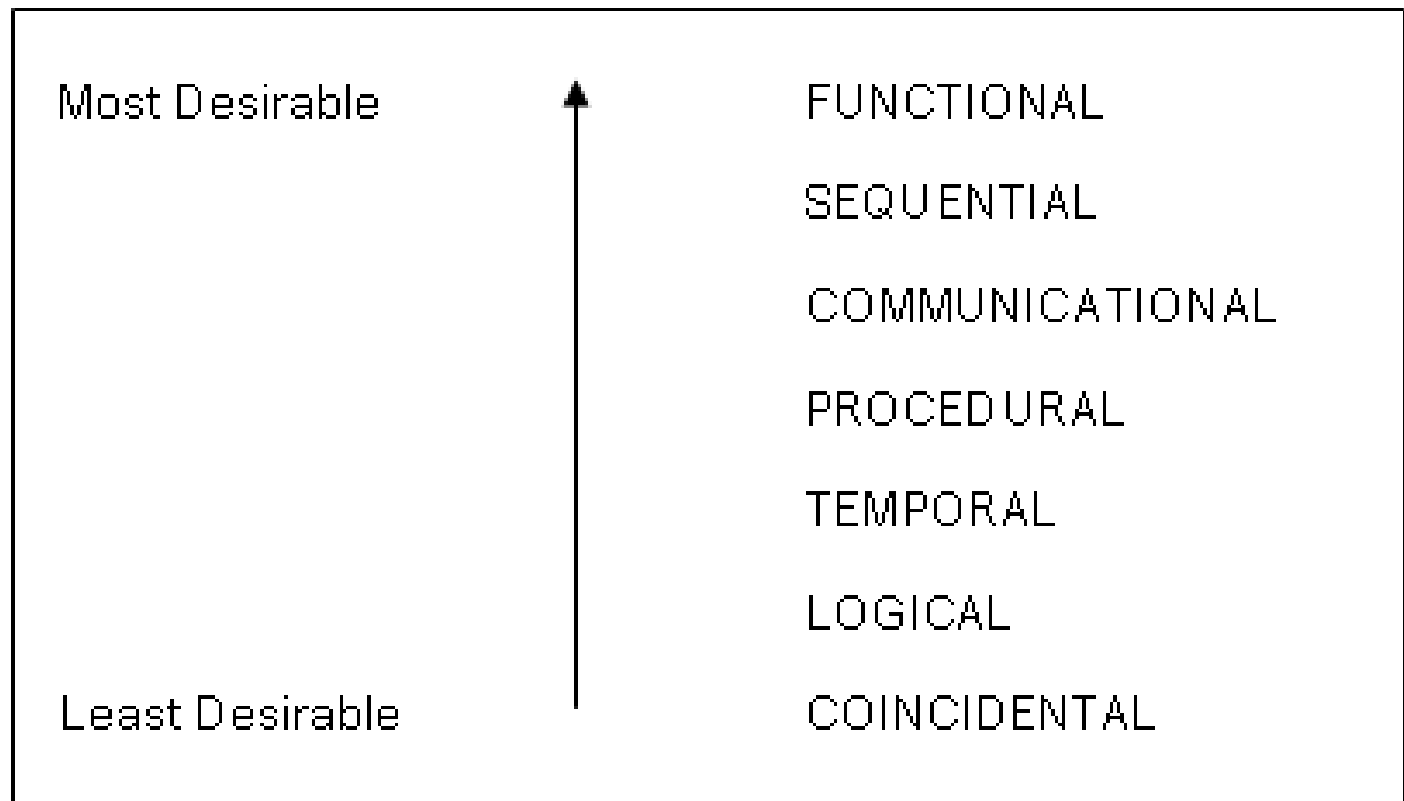
คุณลักษณะของการออกแบบระบบที่ดี

■ Cohesion

- เป็นการวัดความเกี่ยวข้องกันภายในโมดูล
- การออกแบบที่ดีนั้นต้องออกแบบให้ภายใน โมดูลมีความเกี่ยวข้องของคำสั่งหรือการประมวลผลต่างๆเป็นไปเพื่อวัตถุประสงค์เดียวกัน

Cohesion

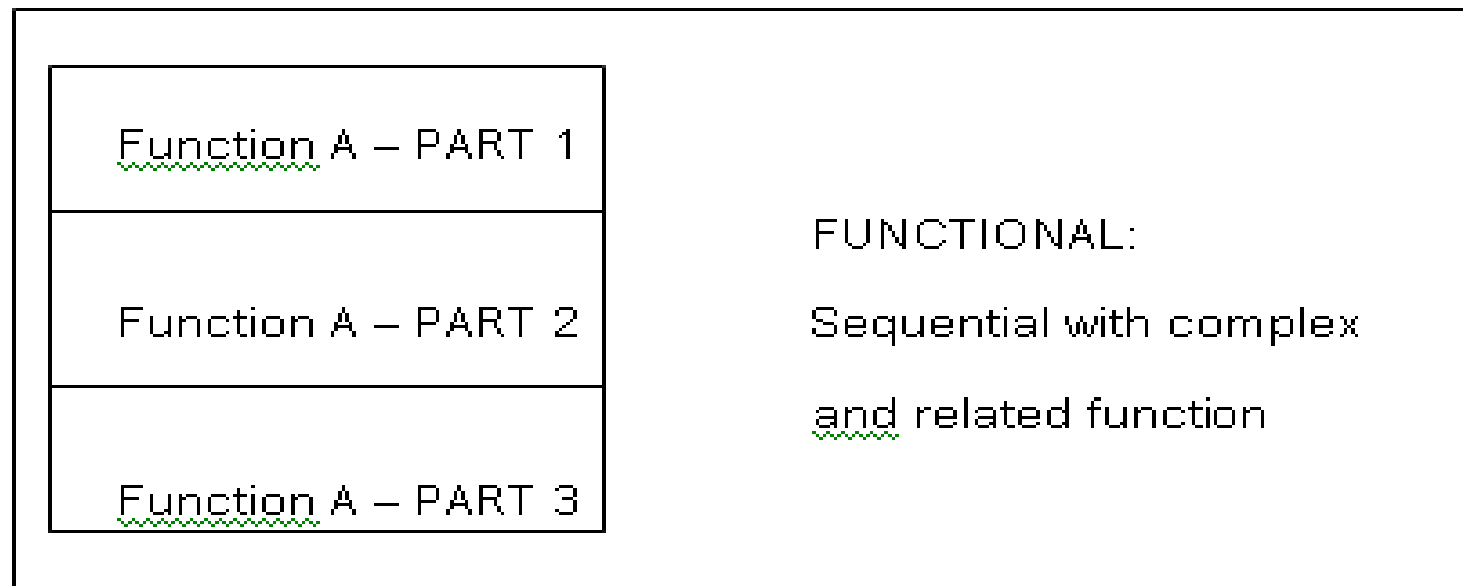
รูปภาพที่ 4.10 Types of Cohesion



Cohesion

- **Functional cohesion** เป็นการออกแบบที่หน้าที่ต่างๆ ภายในโมดูลมีความเกี่ยวข้องกันมาก
- **Module** ที่ออกแบบสามารถกระทำกิจกรรมได้เพียงอย่างเดียวเท่านั้น

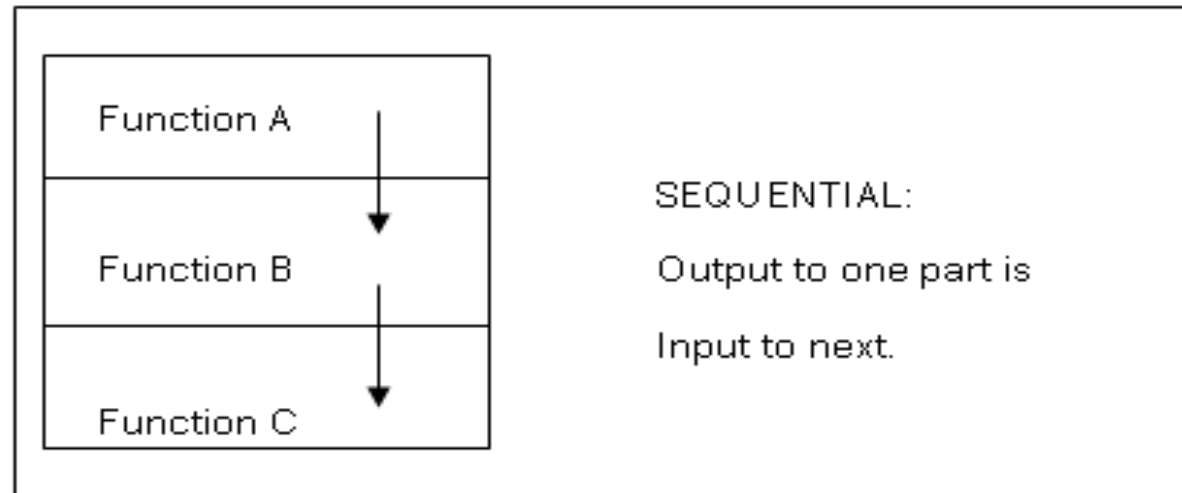
รูปภาพ 4.11 Functional cohesion



Cohesion

- **Sequential cohesion** เป็นความเกี่ยวพันกันภายในโมดูลที่มีหลายฟังก์ชัน และผลลัพธ์ของฟังก์ชันหนึ่งจะเป็นข้อมูลเข้าของอีกฟังก์ชันหนึ่ง เป็นลำดับของการทำงาน ซึ่งการทำงานอยู่ภายใต้วัตถุประสงค์เดียวกัน

รูปภาพ 4.12 Sequential cohesion

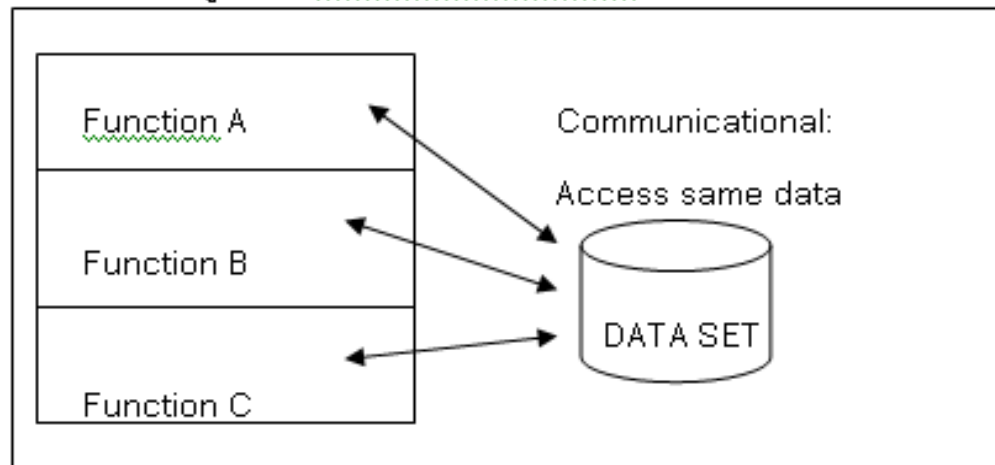


Cohesion

■ Communication cohesion

- เป็นความเกี่ยวพันที่ **Module** ประกอบด้วยหลายฟังก์ชันที่ทำหน้าที่แยกจากกัน แต่ใช้โครงสร้างของข้อมูลเดียวกัน เช่น โมดูลที่ประกอบด้วยฟังก์ชันการจัดเก็บข้อมูล , ฟังก์ชันการค้นหาข้อมูล , ฟังก์ชันการลบข้อมูลซึ่งใช้กลุ่มของข้อมูลชุดเดียวกัน

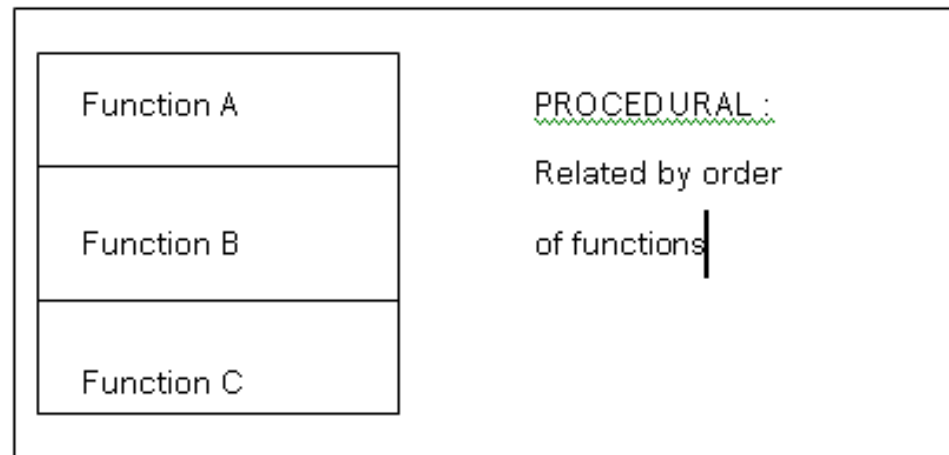
รูปภาพ 4.13 Communication cohesion



Cohesion

- **Procedural cohesion** มักจะพบในกรณีของการแบ่งโมดูลขนาดใหญ่ให้เป็นฟังก์ชันย่อยไปตามสายการควบคุมโดยไม่คำนึงถึงกิจกรรมที่ทำ เจตนาเพื่อลดภาระการดูแลโมดูลขนาดใหญ่ให้เล็กลงเพื่อความสะดวกในการตรวจสอบโปรแกรม ซึ่งกิจกรรมที่กระทำเป็นไปตามลำดับขั้นตอน การออกแบบในลักษณะนี้แต่ละ **Module** ยังคงอยู่ในลักษณะที่ยังมีความสัมพันธ์กันอยู่มาก

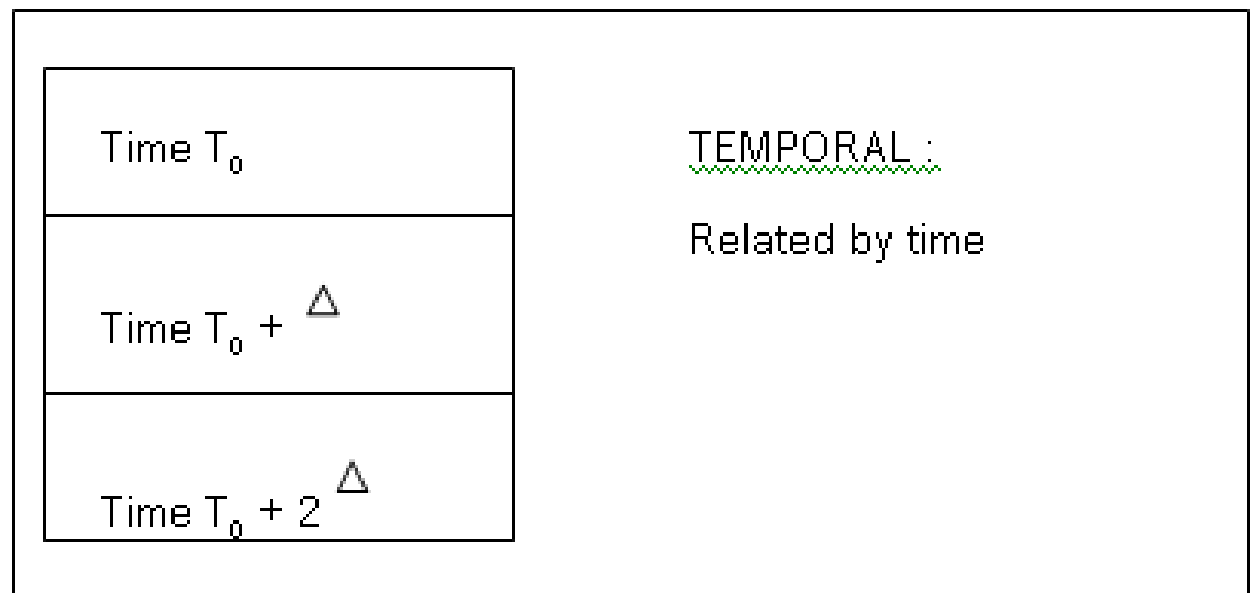
รูปภาพ 4.14 Procedural cohesion



Cohesion

- **Temporal cohesion** เป็นการออกแบบโมดูลที่การทำงานไม่สัมพันธ์กันแต่ต้องประมวลผลในเวลาเดียวกัน เช่น โมดูลในการกำหนดค่าเริ่มต้นของระบบหรือตัวแปรต่างๆ ซึ่งในเวลาหนึ่งต้องมีการกระทำในฟังก์ชันต่างๆพร้อมๆกัน

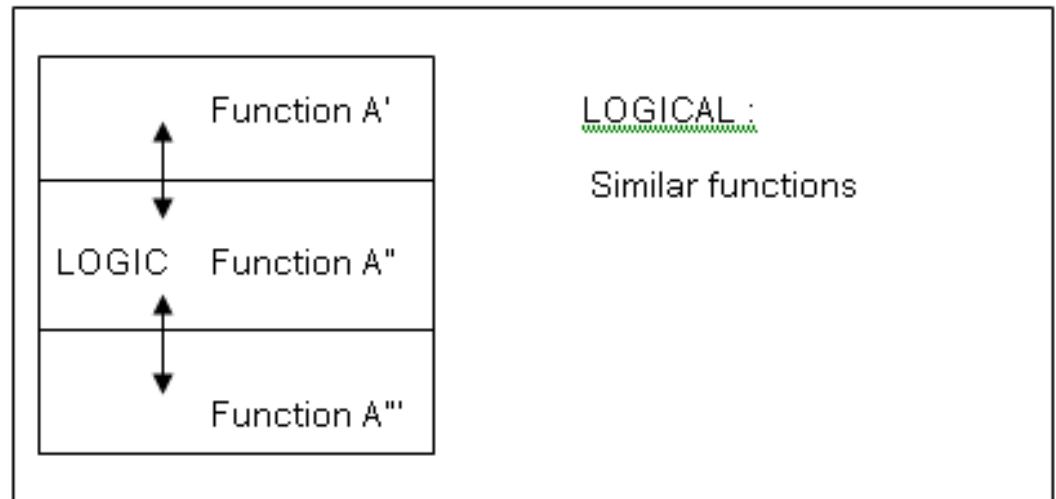
รูปภาพ 4.15 Temporal cohesion



Cohesion

- **Logical cohesion** เป็นการออกแบบโมดูลซึ่งไม่เป็นที่ยอมรับ เนื่องจากฟังก์ชันต่างๆภายใน โมดูลมีตรรกเหมือนกันหรือคล้ายกัน เช่น โมดูลการอ่านข้อมูลซึ่งประกอบด้วยฟังก์ชันในการอ่านข้อมูลจากเทปแม่เหล็ก, ฟังก์ชันการอ่านข้อมูลจากดิสก์, ฟังก์ชันการอ่านข้อมูลจากเครื่องอ่านบาร์โค้ด

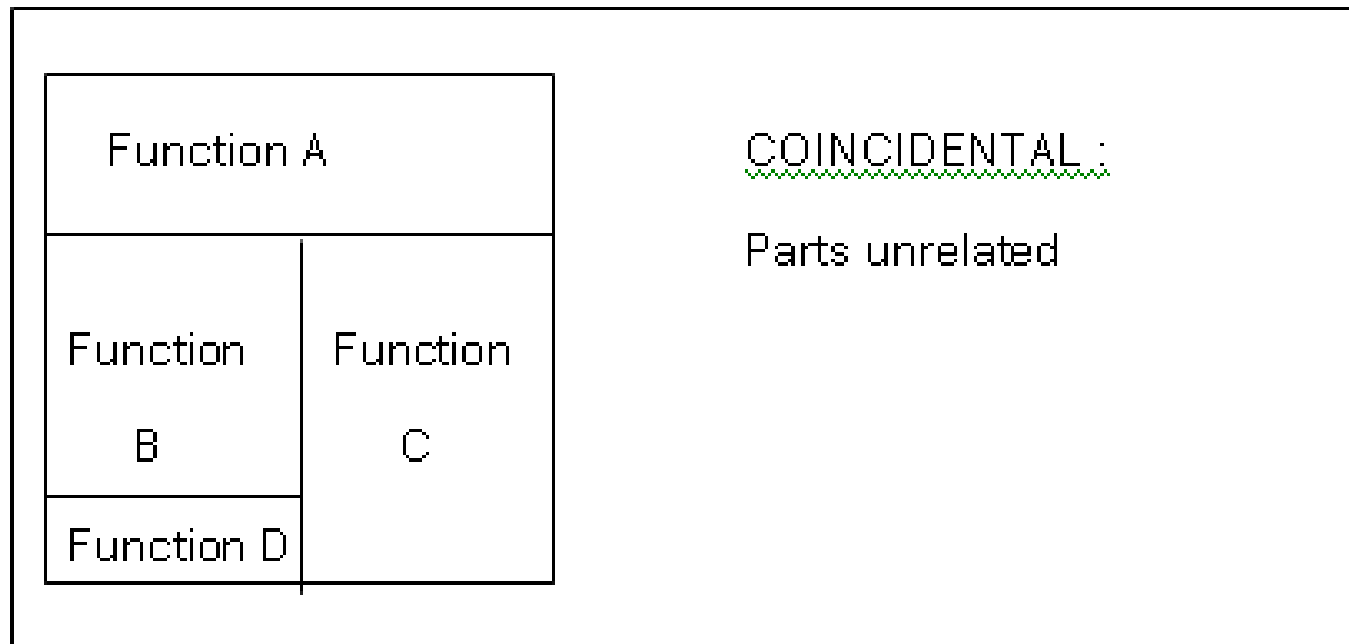
รูปภาพ 4.16 Logical cohesion



Cohesion

- **Coincidental cohesion** เป็นการออกแบบที่แย่ที่สุด เนื่องจากโมดูลประกอบด้วยฟังก์ชันที่ไม่มีความเกี่ยวข้องกันเลย

รูปภาพ 4.17 Coincidental cohesion

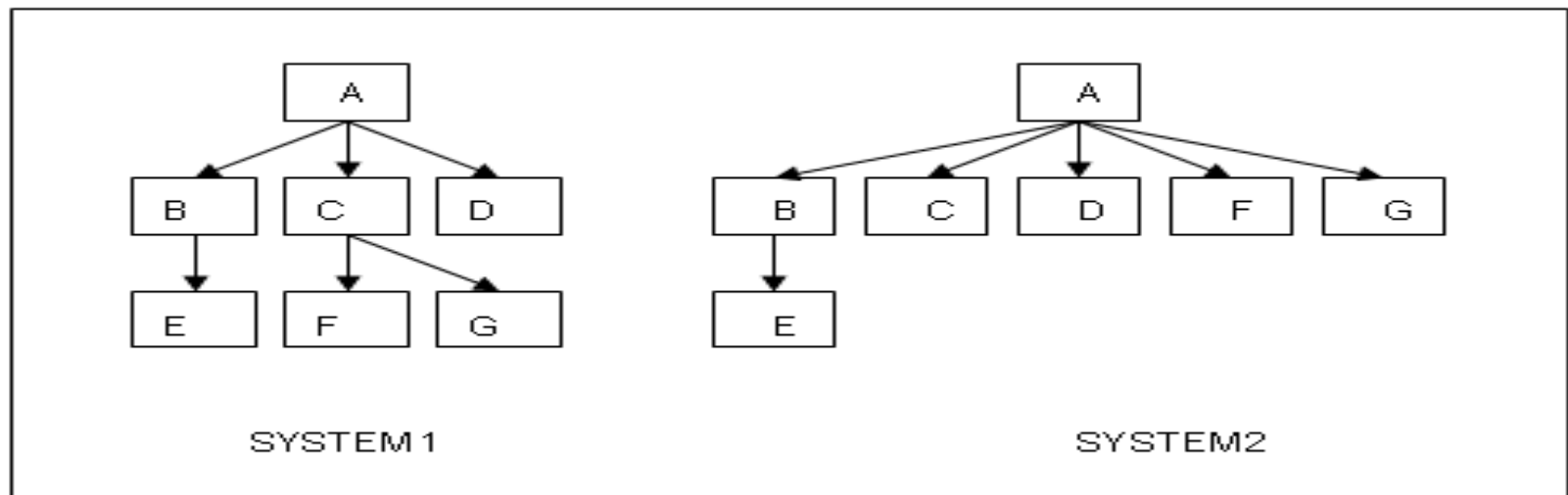


คุณลักษณะของการออกแบบระบบที่ดี

■ Control Issues

- คุณภาพของการออกแบบอีกประการหนึ่งคือวัดจากการควบคุมระหว่างโมดูล ระบบใดก็ตามที่มีการออกแบบให้มีการควบคุมระหว่างโมดูลมีจำนวนน้อยกว่า จะมีคุณภาพที่ดีกว่า

รูปภาพที่ 4.18 ตัวอย่างของ Fan-In และ Fan-Out



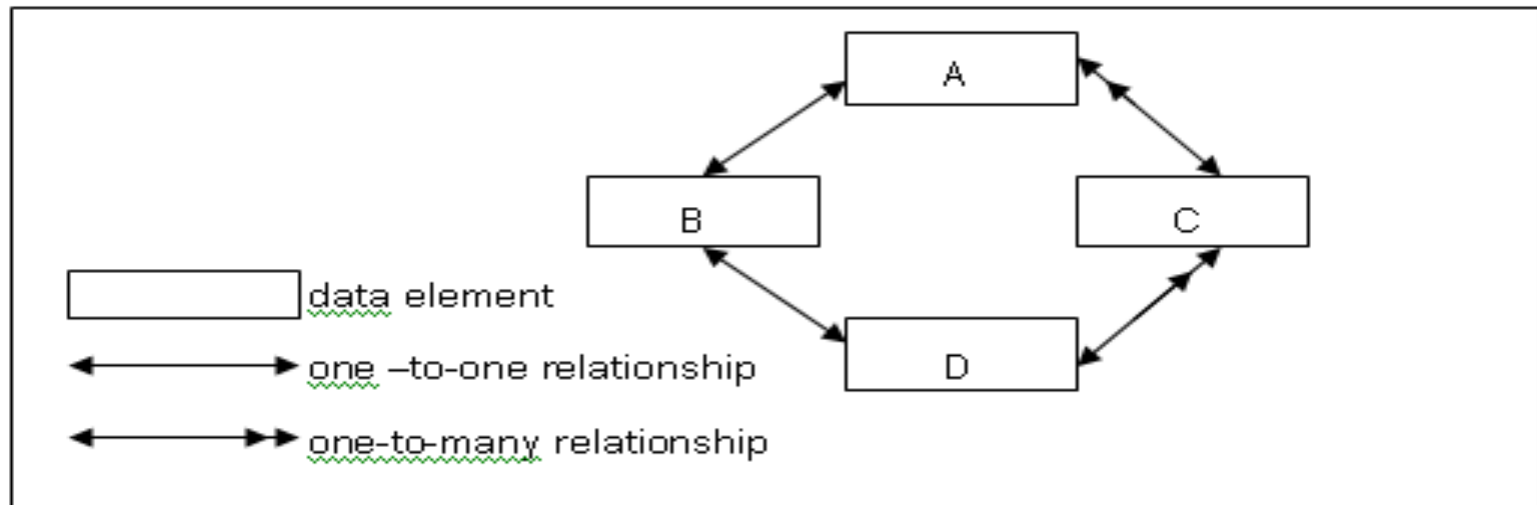
กลยุทธ์ในการออกแบบระบบ

- **Low coupling** มีความเกี่ยวพันระหว่างโมดูลต่ำ
- **Highly cohesive**
มีความเกี่ยวพันของฟังก์ชันต่างๆภายในโมดูลสูง
- **Minimal number of modules with fan-out**
มีจำนวนของโมดูลที่ถูกควบคุมจำนวนน้อย
- **Scope of effect of a module limited to its scope of control** ออกแบบให้ขอบเขตของผลกระทบต่อโมดูลนั้นสามารถควบคุมได้ในขอบเขตที่จำกัด

เทคนิคและเครื่องมือในการออกแบบระบบ

- **Data dictionary** นั้นเพียงแต่มีการแสดงถึงข้อมูลต่างๆที่ใช้ในระบบแต่ไม่ได้แสดงถึงความสัมพันธ์ระหว่างข้อมูล **Bachman** ได้เสนอแนวความคิดในการแสดงความสัมพันธ์ของข้อมูลเป็นไคอะแกรม โดยแทนรูปสี่เหลี่ยมด้วยข้อมูล ใช้ลูกศรในการเชื่อมโยงสี่เหลี่ยมต่างๆ

รูปภาพที่ 4.20 Data Structure Diagram



เทคนิคและเครื่องมือในการออกแบบระบบ

■ Data Abstraction

- ถ้าเราใช้นามธรรมข้อมูลเป็นเครื่องมือในการระบุรายละเอียดความต้องการ เราสามารถใช้นามธรรมข้อมูลนี้เพื่อออกแบบระบบได้ โดยเริ่มจากพิจารณาแนวความคิดของนามธรรมต่างๆเหล่านั้น

DO WHILE I is between 1 and (length of L) - 1

Set LOW to next of smallest value in $L(I), \dots, L(\text{length of } L)$

Interchange $L(I)$ and $L(\text{LOW})$

ENDDO

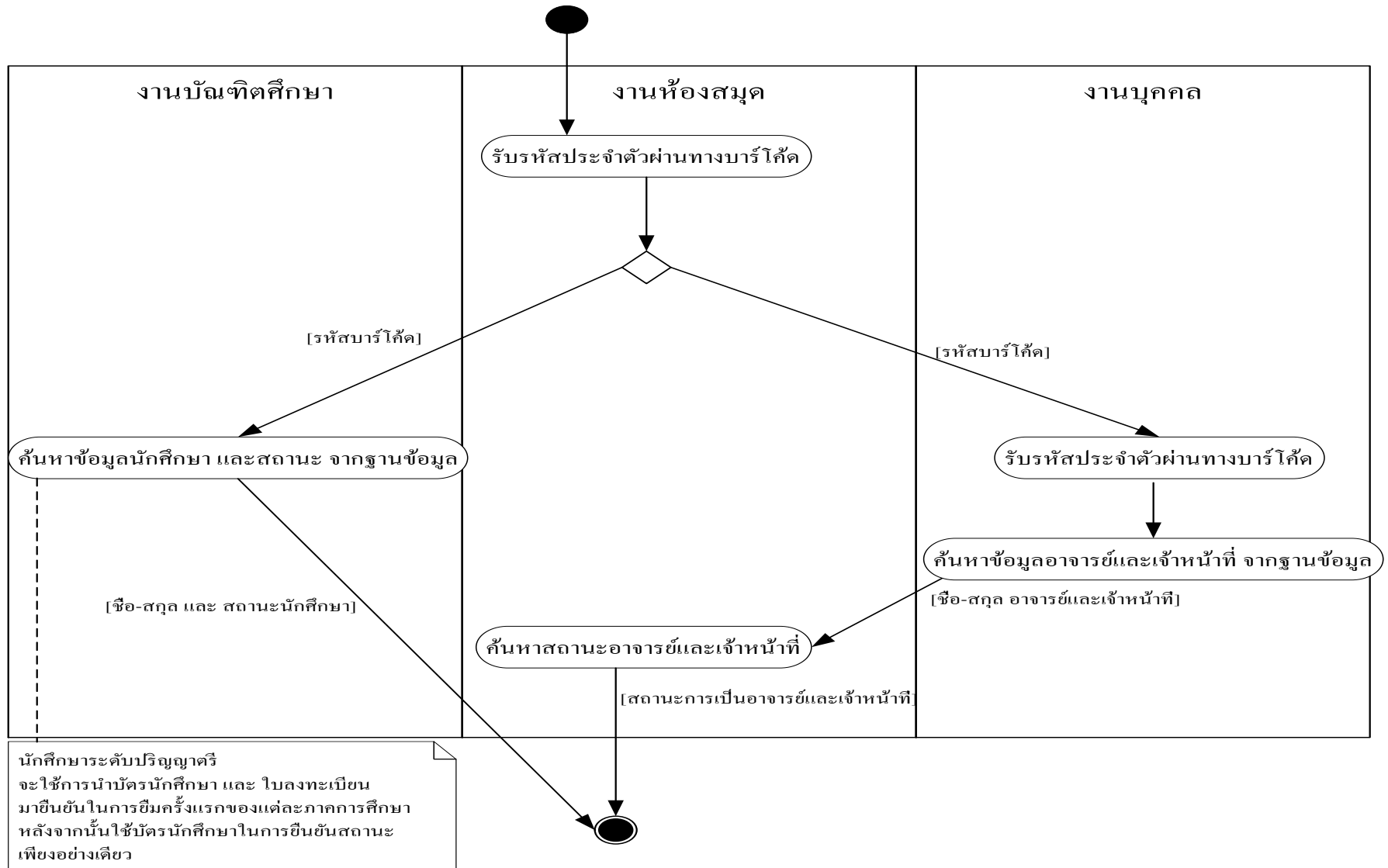
เทคนิคและเครื่องมือในการออกแบบระบบ

- **UML(Unified Modeling Language)**
- เป็นแนวคิดและสร้างระบบงานในลักษณะโลกของความเป็นจริง โดยมองสิ่งต่างๆเป็นวัตถุหรือออบเจกต์ ซึ่งออบเจกต์ต่างๆจะมีความเป็นอิสระไม่ขึ้นต่อกัน แต่มีการทำงานร่วมกันโดยติดต่อถึงกันโดยการส่งเมสเสจ (message)

ข้อดี UML (Unified Modeling Language)

- UML รวมข้อดีของโมเดลต่างๆเอาไว้ ได้แก่ Data Model , Business Model , Object Model , Component Model
- เป็นภาษาที่เป็นมาตรฐานเปิด โปรแกรมทุกภาษาในปัจจุบันสนับสนุน UML
- ภาษา UML ครอบคลุมทุกส่วนในวัฏจักรพัฒนาระบบ

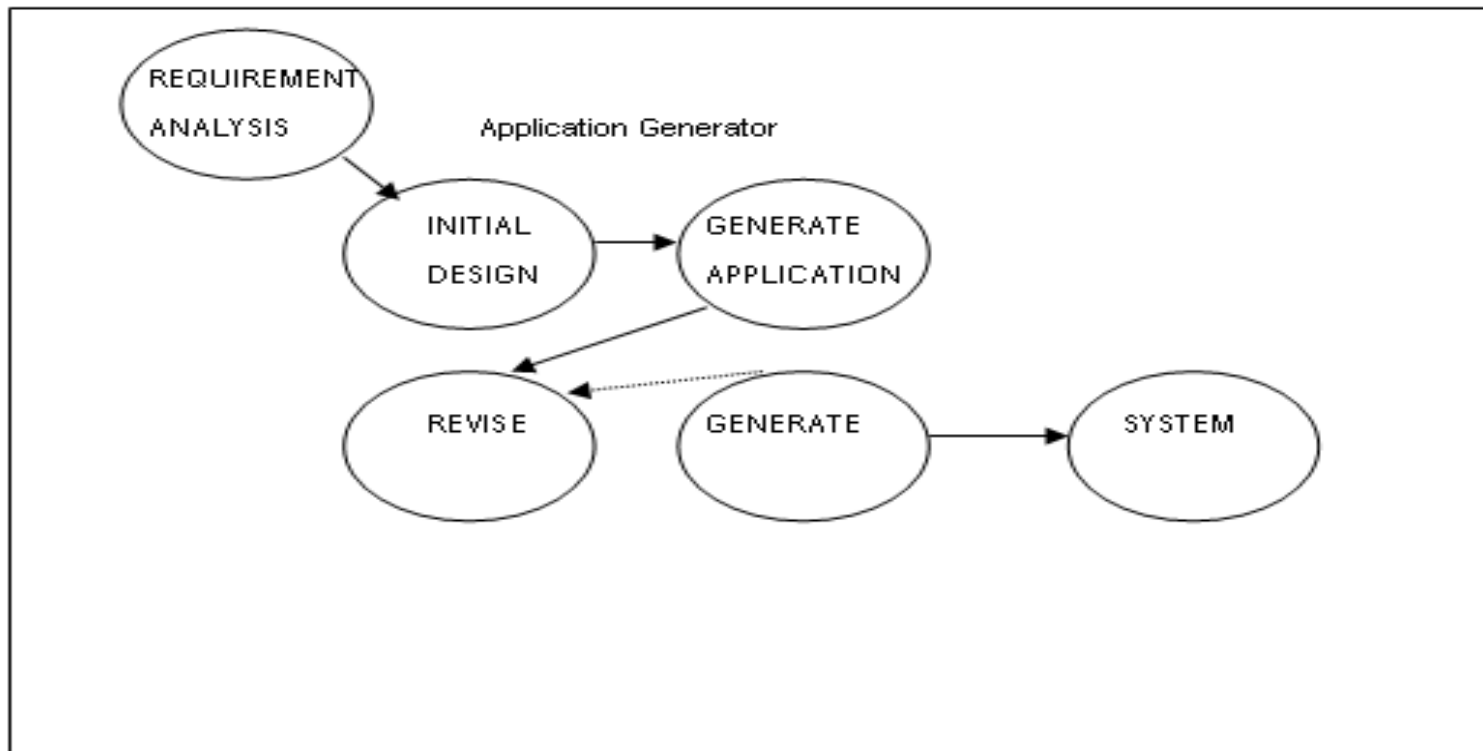
Activity Diagram ของการตรวจสอบผู้ใช้บริการ



เทคนิคและเครื่องมือในการออกแบบระบบ

- เป็นโปรแกรมที่แปลความต้องการของระบบให้เป็นระบบโดยอัตโนมัติ โดยสามารถสร้างเป็นโปรแกรมประยุกต์ที่ระบุได้ตามความต้องการ

รูปภาพที่ 4.28 Application Generator in Iteration development



การทวนสอบและการคำนวณการออกแบบ

■ Mathematical Validation

- ตามสามัญสำนึกวิธีตรวจสอบความถูกต้องของระบบก็คือทดลองใส่ข้อมูลเข้าไปในระบบแล้วดูผลลัพธ์ที่ได้ ถ้าถูกต้องแสดงว่าระบบทำงานได้ถูกต้องถ้าไม่ถูกต้องแสดงว่าระบบทำงานไม่ถูกต้อง

การทดสอบและการคำนวณการออกแบบ

■ Weighted Matrices

- การวัดคุณภาพของการออกแบบนั้น
เราจะวัดจากคุณลักษณะต่างๆอัน
ได้แก่ Modularity ,
Testability , Maintainability
, Efficiency , Ease of
Understanding , Ease of
Modification , Consistency

ตารางที่ 4.4 Weighted Design Criteria

Characteristic	Weight
Modularity	5
Testability	4
Maintainability	3
Efficiency	2
Ease of Understanding	3
Ease of Modification	3
Consistency	4

การทดสอบและการคำนวณการออกแบบ

ตารางที่ 4.5 Weighted Design Matrix

Characteristic	Weight	Design	Design	Design
		1	2	3
Modularity	5	8	6	7
Testability	4	7	7	10
Maintainability	3	9	10	8
Efficiency	2	4	6	9
Ease of Understanding	3	10	8	8
Ease of Modification	3	9	9	8
Consistency	4	6	6	7

การทดสอบและการคำนวณการออกแบบ

รูปภาพที่ 4.29 Computing a Design's Score

Characteristic	Weight	Design		
		1	2	3
Modularity	5 → 40 ← 8	6	7	
Testability	4 → 28 ← 7	7	10	
Maintainability	3 → 27 ← 9	10	8	
Efficiency	2 → 8 ← 4	6	9	
Ease of Understanding	3 → 30 ← 10	8	8	
Ease of Modification	3 → 27 ← 9	9	8	
Consistency	4 → 24 ← 6	6	7	
		= 184		

การทดสอบและการคำนวณการออกแบบ

ตารางที่ 4.6 Scores for Weighted Design Matrix

Characteristic	Weight	Design	Design	Design
		1	2	3
Modularity	5	8	6	7
Testability	4	7	7	10
Maintainability	3	9	10	8
Efficiency	2	4	6	9
Ease of Understanding	3	10	8	8
Ease of Modification	3	9	9	8
Consistency	4	6	6	7
SCORE:		<u>184</u>	<u>170</u>	<u>193</u>

Design Reviews

- **Preliminary Design Review** เป็นการพบกับลูกค้าเพื่อตรวจสอบความถูกต้องของการออกแบบแนวคิด(conceptual design) โดยกลุ่มบุคลากรที่เกี่ยวข้องในการตรวจสอบประกอบด้วย
 - ลูกค้า
 - นักวิเคราะห์ระบบ
 - ผู้ใช้ระบบ
 - นักออกแบบระบบ
 - ผู้ควบคุมการประชุม
 - ผู้จัดบันทึกการประชุม

- **Critical Design Review** การประชุมปรึกษาหารือกันในส่วนของการออกแบบเทคนิค (technical design) โดย ผู้ร่วมประชุมประกอบด้วย
 - นักวิเคราะห์ระบบ
 - นักออกแบบระบบ
 - ผู้ควบคุมการประชุม
 - ผู้จัดบันทึกการประชุม
 - นักออกแบบโปรแกรม
 - ผู้พัฒนาระบบอื่นๆที่ไม่มีส่วนในการพัฒนาระบบที่ผ่านมา

เอกสาร

- เมนู และ รูปแบบของหน้าจอ โดยอธิบายเป็นไดอะแกรมเป็นลำดับชั้นหรือเครื่องมือที่ให้เห็นภาพรวมของระบบ, การไหลของข้อมูล, ความสัมพันธ์ระหว่างโมดูลต่างๆ
- เครื่องมือที่ใช้ปฏิสัมพันธ์กับผู้ใช้ อาทิเช่น ฟังก์ชันคีย์, touch screen,เมาส์ ฯลฯ
- คุณลักษณะทั่วไปของกิจกรรมต่างๆในระบบ รูปแบบของรายงาน การประมวลผลของโหนดในรายงานต่างๆ การจัดสรรทรัพยากรต่างๆให้กับผู้ใช้ในรายงาน, ข้อกำหนดในเรื่องเวลาในการทำงานและการประมวลผล , บุรณภาพของข้อมูลที่ใช้ในรายงาน และระเบียบในการปฏิบัติเมื่อระบบล้มเหลว
- ข้อกำหนดถึงประสิทธิภาพของระบบ เช่นความเร็วในการปฏิบัติงาน , ความสามารถในการตรวจสอบความผิดพลาด , ระบบรักษาความปลอดภัย
- กระบวนการในการจัดการข้อมูล