บทที่ 4

แผนภาพกระแสข้อมูล

(Data Flow Diagram)

อาจารย์ ศิริรัตน์ ตรงวัฒนาวุฒิ

เนื้อหา

- ความหมายของแผนภาพกระแสข้อมูล
- สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล
- กฎที่ใช้ในการเขียนแผนภาพกระแสข้อมูล
- ประเภทของแผนภาพกระแสข้อมูล
- การพัฒนาแผนภาพกระแสข้อมูล

วัตถุประสงค์

1) ทราบความสำคัญและวัตถุประสงค์ของแผนภาพกระแสข้อมูล (Data Flow Diagram)
2) เข้าใจสัญลักษณ์ต่างๆ และขั้นตอนการเขียนแผนภาพกระแสข้อมูล
3) สามารถวิเคราะห์และเขียนแผนภาพกระแสข้อมูลได้

เอกสาร

แบบฟอร์ม

รายงาน

บทสรุปการสัมภาษณ์

สร้างแบบจำลองเชิงตรรกะ

(Logical Model)

ความหมายของแผนภาพกระแสข้อมูล (Data Flow Diagram)

แผนภาพกระแสข้อมูล (Data Flow Diagram: DFD) หรือเรียกอีกอย่างหนึ่งว่า แผนภาพการไหลของข้อมูล

เป็นเครื่องมือที่ใช้เพื่อแสดงการไหลของข้อมูลและการประมวลผลต่างๆ ในระบบ สัมพันธ์กับแหล่งเก็บข้อมูลที่ใช้

เป็นสื่อที่ช่วยให้การวิเคราะห์เป็นไปได้โดยง่าย และมีความเข้าใจตรงกันระหว่างผู้วิเคราะห์ระบบเอง หรือระหว่างผู้วิเคราะห์ระบบกับโปรแกรมเมอร์ หรือระหว่างผู้วิเคราะห์ระบบกับผู้ใช้ระบบ

ประโยชน์ที่ได้จากการใช้แผนภาพกระแสข้อมูล

1) มีความอิสระในการใช้งาน โดยไม่ต้องมีเทคนิคอื่นมาช่วย เนื่องจากสามารถใช้สัญลักษณ์ต่างๆ แทนสิ่งที่วิเคราะห์มา

2) เป็นสื่อที่ง่ายต่อการแสดงความสัมพันธ์ระหว่างระบบใหญ่และระบบย่อย ซึ่งจะทำให้เข้าใจความสัมพันธ์ต่าง ๆ ได้ดี

3) เป็นสื่อที่ช่วยในการวิเคราะห์ระบบให้เป็นไปได้ง่าย และมีความเข้าใจตรงกันระหว่างผู้วิเคราะห์ระบบ หรือระหว่างผู้วิเคราะห์ระบบกับโปรแกรมเมอร์ หรือระหว่างผู้วิเคราะห์ระบบกับผู้ใช้

4) ช่วยในการวิเคราะห์ระบบให้สะดวก โดยสามารถเห็นข้อมูลและขั้นตอนต่างๆ เป็นแผนภาพ

สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล ประกอบด้วย 4 สัญลักษณ์ คือ

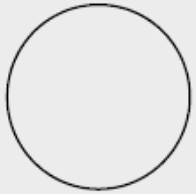
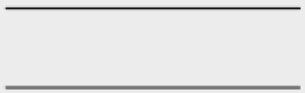
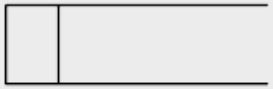
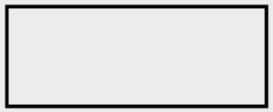
1) สัญลักษณ์การประมวลผล (Process Symbol)

2) สัญลักษณ์กระแสข้อมูล (Data Flow Symbol)

3) สัญลักษณ์แหล่งเก็บข้อมูล (Data Store Symbol)

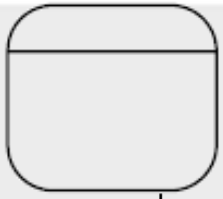
4) สัญลักษณ์สิ่งที่อยู่ภายนอก (External Entity Symbol)

สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

ชื่อสัญลักษณ์	DeMarco & Yourdon symbols	Gane & Sarson symbols
การประมวลผล (Process)		
แหล่งเก็บข้อมูล (Data Store)		
กระแสข้อมูล (Data Flow)		
สิ่งที่อยู่ภายนอก (External Entity)		

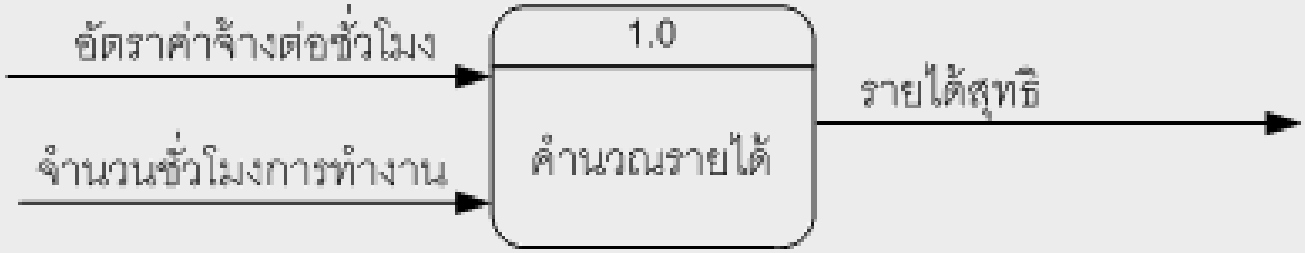
สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์การประมวลผล (Process Symbol)



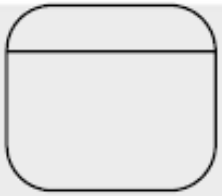
การประมวลผล (Process) เป็นการเปลี่ยนแปลงข้อมูลจากรูปแบบหนึ่ง (Input) ไปเป็นอีกรูปแบบหนึ่ง (Output)

เช่น การคำนวณรายได้สุทธิของลูกจ้างรายวัน จะต้องประกอบด้วยข้อมูลนำเข้าที่เป็น ”อัตราค่าจ้างต่อชั่วโมง” และ “จำนวนชั่วโมงการทำงาน” เมื่อผ่านการประมวลผลแล้วจะได้ “รายได้สุทธิ”



สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

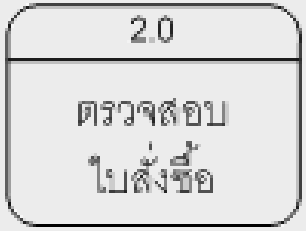
สัญลักษณ์การประมวลผล (Process Symbol)



<u>ตัวอย่างการประมวลผล</u> ได้แก่

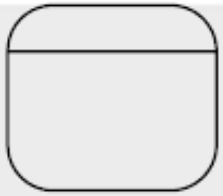
1) คำนวณค่าคอมมิชชัน
2) ตรวจสอบใบสั่งซื้อ
3) ลงทะเบียน

เป็นต้น



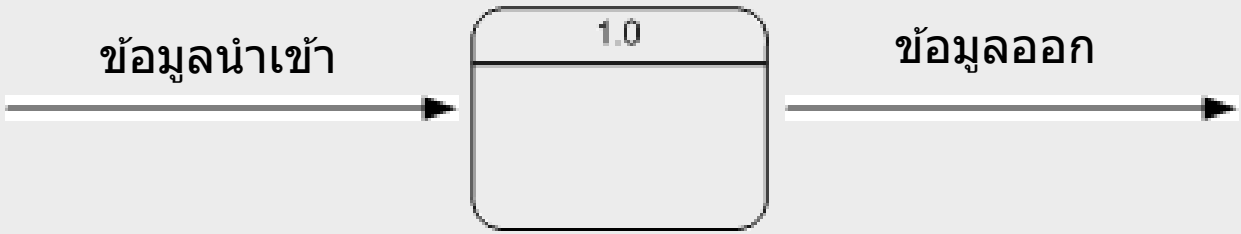
สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์การประมวลผล (Process Symbol)



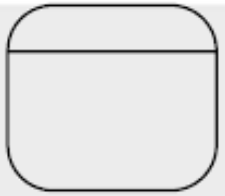
การใช้สัญลักษณ์การประมวลผล

1) ต้องใช้สัญลักษณ์การประมวลผล (Process) คู่กับสัญลักษณ์ กระแสข้อมูล (Data Flow) เสมอ โดยที่ถ้าลูกศรชี้เข้าหมายถึงเป็นข้อมูลนำเข้า ถ้าลูกศรชี้ออกหมายถึงเป็นข้อมูลออกจากการประมวลผล ซึ่ง 1 Process สามารถมีข้อมูลนำเข้ามากกว่า 1 เส้น หรือข้อมูลออกมากกว่า 1 เส้นได้



สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์การประมวลผล (Process Symbol)



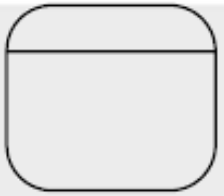
การใช้สัญลักษณ์การประมวลผล

2) การตั้งชื่อของ Process ควรเป็นวลีเดียวที่อธิบายการทำงานทั้งหมดได้ และควรอธิบายการทำงานอย่างใดอย่างหนึ่งโดยเฉพาะมากกว่าที่จะอธิบายการทำงานอย่างกว้างๆ เช่น หากแสดงถึงการประมวลผล “ตรวจสอบรายการ” ควรจะระบุว่าเป็น “การตรวจสอบรายการถอนเงิน” หรือ “ตรวจสอบรายการค่าใช้จ่ายรายสัปดาห์” เป็นต้น

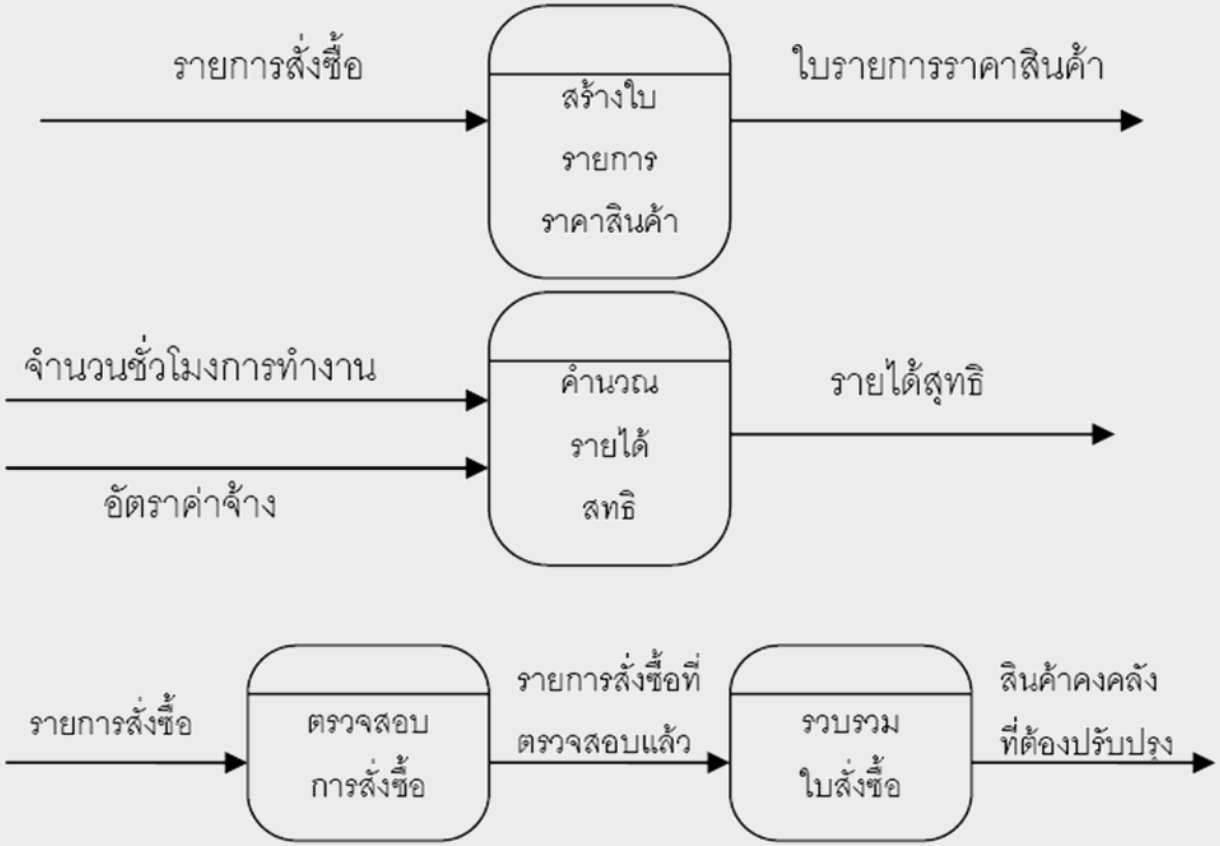
3) แต่ละ Process จะมีแต่ข้อมูลเข้าอย่างเดียว หรือออกอย่างเดียวไม่ได้

สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์การประมวลผล (Process Symbol)

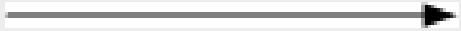


ตัวอย่างการใช้สัญลักษณ์การประมวลผล (Process)



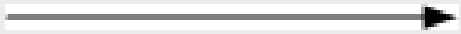
สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์กระแสข้อมูล (Data Flow Symbol)



เป็นเส้นทางในการไหลของข้อมูลจากส่วนหนึ่ง ไปยังอีกส่วนหนึ่งของระบบสารสนเทศ โดยจะมีลูกศรแสดงถึงการไหลจากปลายลูกศร ไปยังหัวลูกศร ซึ่งข้อมูลที่ปรากฏบนเส้นนี้จะเป็นได้ทั้งข้อความ ตัวเลข รายการเรคคอร์ดที่ระบบคอมพิวเตอร์สามารถนำไปประมวลผลได้

สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์กระแสข้อมูล (Data Flow Symbol) 

<u>ตัวอย่างกระแสข้อมูล ได้แก่</u>

1) ใบสั่งซื้อสินค้า

2) ใบเสร็จรับเงิน

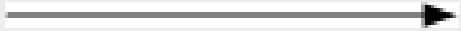
3) เกรดของนักศึกษา

4) ใบส่งของที่ผ่านการตรวจสอบแล้ว

เป็นต้น

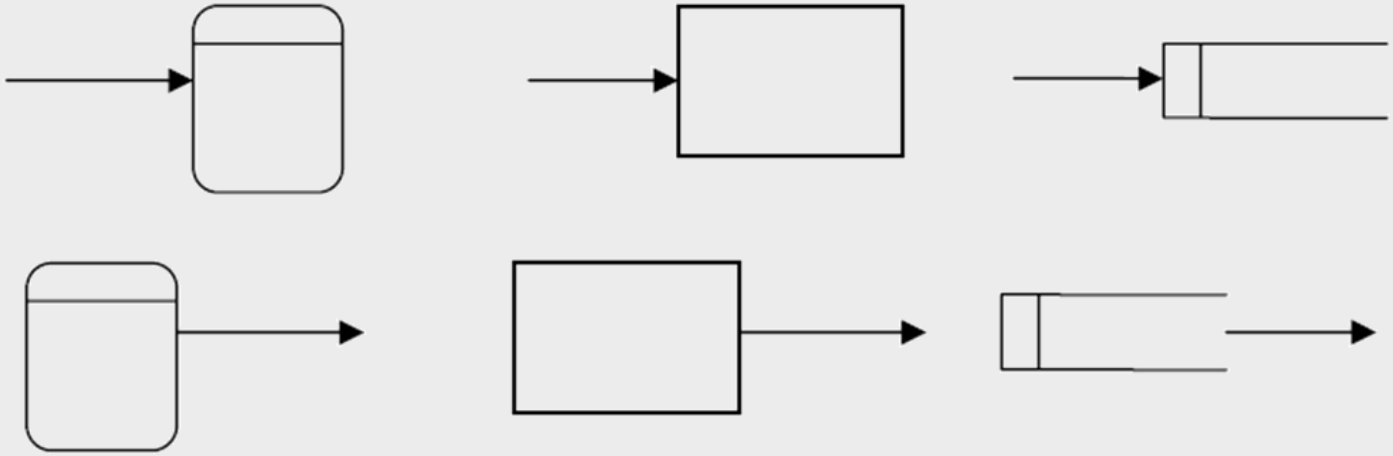
สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์กระแสข้อมูล (Data Flow Symbol)



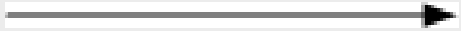
การใช้สัญลักษณ์กระแสข้อมูล

1) กระแสข้อมูลสามารถใช้คู่กับการประมวลผล (Process), สิ่งที่อยู่นอกระบบ (External Entities) หรือแหล่งเก็บข้อมูล (Data Store) ก็ได้ ขึ้นอยู่กับระบบงานว่า ข้อมูลนั้นจะนำไปไว้ที่ไหน หรือข้อมูลนั้นจะนำออกจากส่วนใด



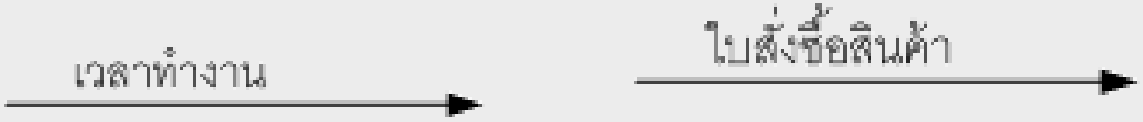
สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์กระแสข้อมูล (Data Flow Symbol)



การใช้สัญลักษณ์กระแสข้อมูล

2) การตั้งชื่อกระแสข้อมูล โดยทั่วไปจะตั้งชื่อด้วยคำเพียงคำเดียว ที่มีความหมายชัดเจนและเข้าใจง่าย ควรกำกับชื่อบนเส้นด้วย คำนาม เช่น “เวลาทำงาน”, ใบสั่งซื้อสินค้า” เป็นต้น



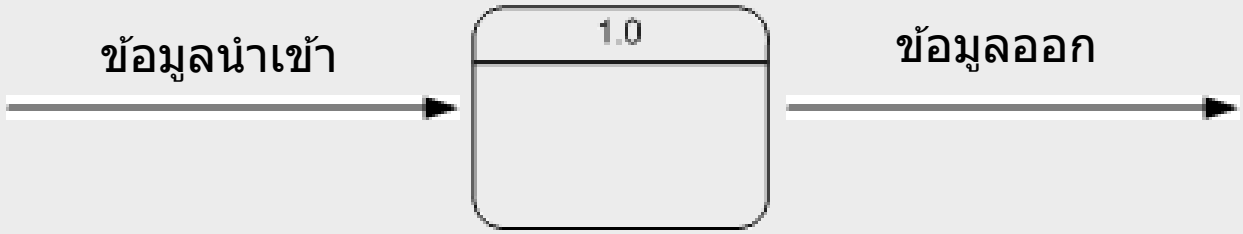
สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์กระแสข้อมูล (Data Flow Symbol)

การใช้สัญลักษณ์กระแสข้อมูล

3) ควรตั้งชื่อกระแสข้อมูล ตามข้อมูลที่ได้เปลี่ยนแปลงไปแล้วหลังจากออกจากการประมวลผล เนื่องจากการประมวลผลหรือ Process ใช้แสดงถึงการเปลี่ยนข้อมูล หรือการส่งผ่านข้อมูล

ดังนั้น Data Flow ที่ออกจาก Process มักจะมีการเขียนชื่อกำกับให้แตกต่างออกไปจาก Data Flow ที่เข้ามาใน Process เสมอ



สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์แหล่งเก็บข้อมูล (Data Store Symbol)

เป็นส่วนที่ใช้แทนชื่อแฟ้มข้อมูลที่เก็บข้อมูล เพราะมีการประมวลผลหลายแบบ ที่จะต้องมีการเก็บข้อมูลไว้เพื่อที่จะได้นำไปใช้ภายหลัง

ซึ่งแหล่งเก็บข้อมูลจะต้องมีทั้งข้อมูลเข้าและข้อมูลออก

สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์แหล่งเก็บข้อมูล (Data Store Symbol)

D1	

<u>ตัวอย่างแหล่งเก็บข้อมูล</u> ได้แก่

1) แฟ้มคนไข้

D1	แฟ้มคนไข้

2) แฟ้มพนักงาน

D2	แฟ้มพนักงาน

เป็นต้น

สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

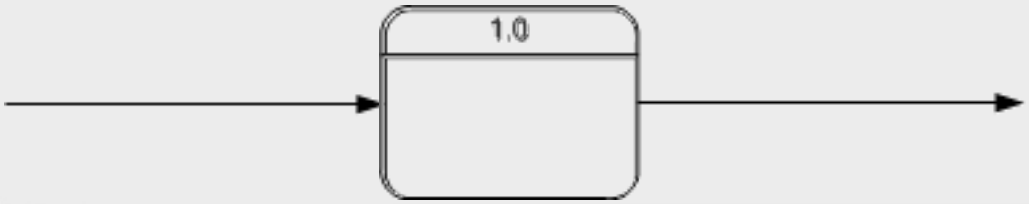
สัญลักษณ์แหล่งเก็บข้อมูล (Data Store Symbol)

การใช้สัญลักษณ์แหล่งเก็บข้อมูล

1) ต้องใช้สัญลักษณ์แหล่งเก็บข้อมูล (Data Store) คู่กับสัญลักษณ์ กระแสข้อมูล (Data Flow) เสมอ

โดยที่

- ถ้าลูกศรชี้เข้าหมายถึง เป็นข้อมูลนำเข้าไปเก็บยังแหล่งเก็บ (เพิ่ม-ลบ แก้ไข)
- ถ้าลูกศรชี้ออกหมายถึง อ่านข้อมูลจากแหล่งเก็บข้อมูลไปใช้ในการประมวลผล



สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์แหล่งเก็บข้อมูล (Data Store Symbol)

การใช้สัญลักษณ์แหล่งเก็บข้อมูล

2) Data Store ต้องเชื่อมต่อการประมวลผล (Process) เสมอโดยเชื่อมผ่านกระแสข้อมูล (Data Flow)

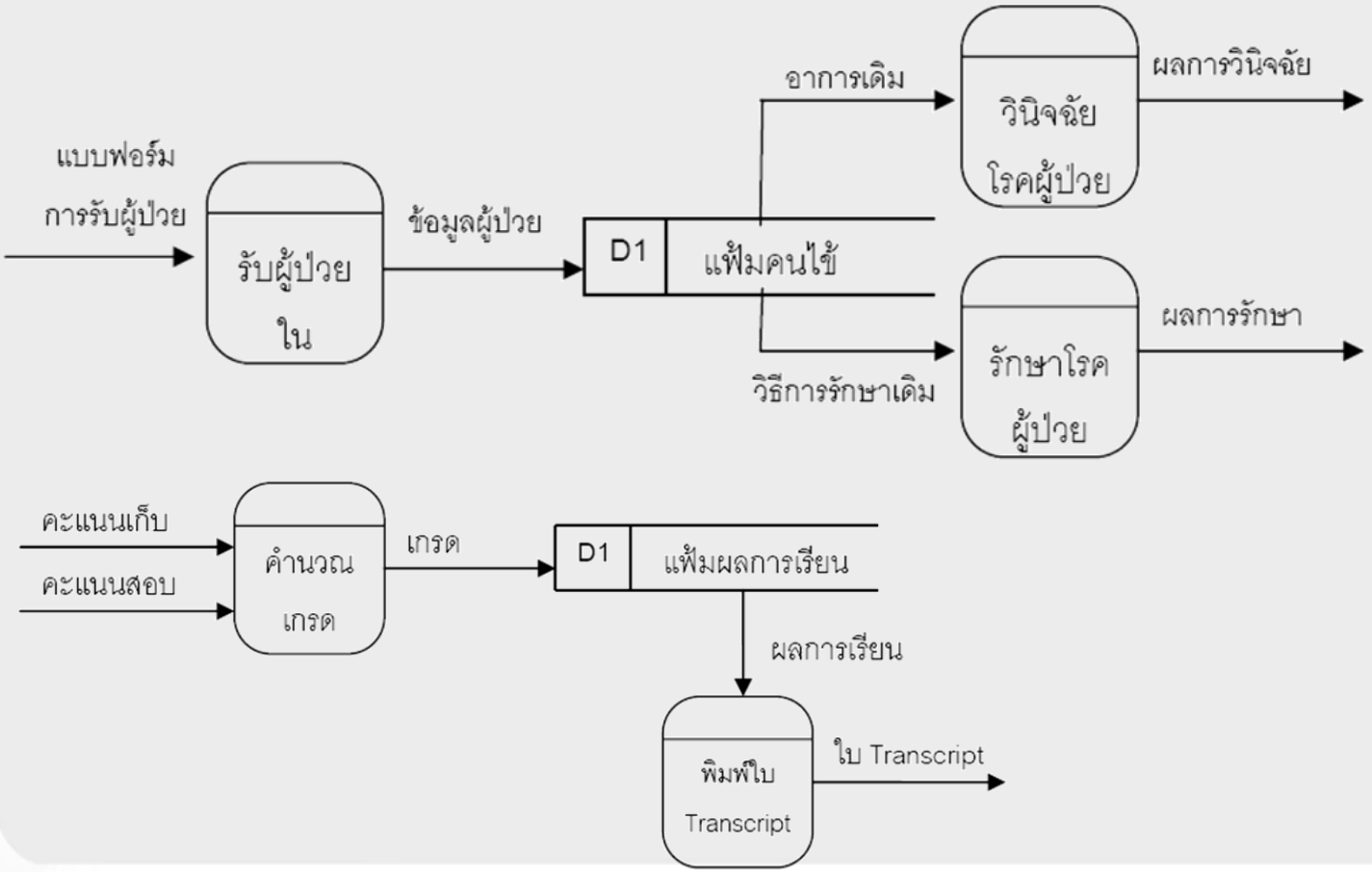
3) เนื่องจาก Data Store ใช้แทนสิ่งที่เก็บข้อมูลเกี่ยวกับคน, สถานที่, หรือสิ่งของ

ดังนั้นควรเขียนชื่อกำกับด้วยคำนาม เช่น “แฟ้มข้อมูลสินค้า”, “แฟ้มเวลาทำงานของพนักงาน” เป็นต้น

สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์แหล่งเก็บข้อมูล (Data Store Symbol)

ตัวอย่างการใช้สัญลักษณ์แหล่งที่เก็บข้อมูล (Data Store)



สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์แหล่งเก็บข้อมูล (Data Store Symbol)

การใช้สัญลักษณ์แหล่งเก็บข้อมูล

4) ใช้อักษรย่อ D1, D2 เป็นต้น เขียนด้านซ้ายมือของสัญลักษณ์ เพื่อแสดงว่าเป็นแหล่งเก็บข้อมูลอันที่เท่าใด สามารถเขียนซ้ำในระดับต่างๆ ของแผนภาพกระแสข้อมูลได้
5) Data Store ใช้แทนสิ่งที่เป็นที่เก็บข้อมูล ซึ่งอาจเป็นการทำด้วยมือ หรือเก็บในรูปแบบคอมพิวเตอร์คือแฟ้มข้อมูลหรือฐานข้อมูล ก็ได้

สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์ สิ่งที่อยู่ภายนอก (External Entity Symbol)

เป็นส่วนที่ใช้แทนคน แผนกภายในองค์กร และแผนกภายนอกองค์กร หรือระบบสารสนเทศอื่นที่เป็นส่วนที่จะให้ข้อมูลหรือรับข้อมูล

สิ่งที่อยู่นอกระบบนี้ใช้แสดงถึงขอบเขตของระบบสารสนเทศ และแสดงถึงว่าระบบที่ศึกษาอยู่นี้จะติดต่อกับสิ่งที่อยู่ภายนอกด้วยวิธีใด (นำข้อมูลเข้ามา หรือได้ข้อมูลออกไป)

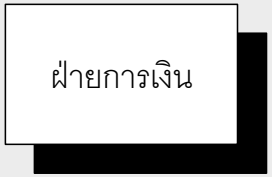
สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์ สิ่งที่อยู่ภายนอก (External Entity Symbol)

<u>ตัวอย่างสิ่งที่อยู่ภายนอก ได้แก่</u>

1) นักศึกษา
2) สินค้าคงคลัง เป็นต้น
3) สมาชิก

เป็นต้น



สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์ สิ่งที่อยู่ภายนอก (External Entity Symbol)

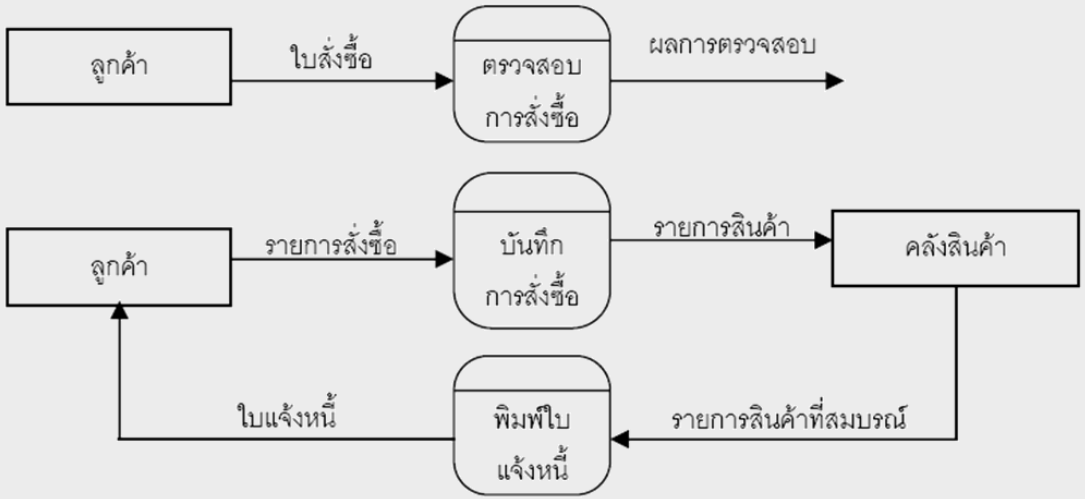
การใช้สัญลักษณ์สิ่งที่อยู่ภายนอก

1) ใช้สัญลักษณ์สิ่งที่อยู่ภายนอกคู่กับสัญลักษณ์กระแสข้อมูลเสมอ

โดยที่

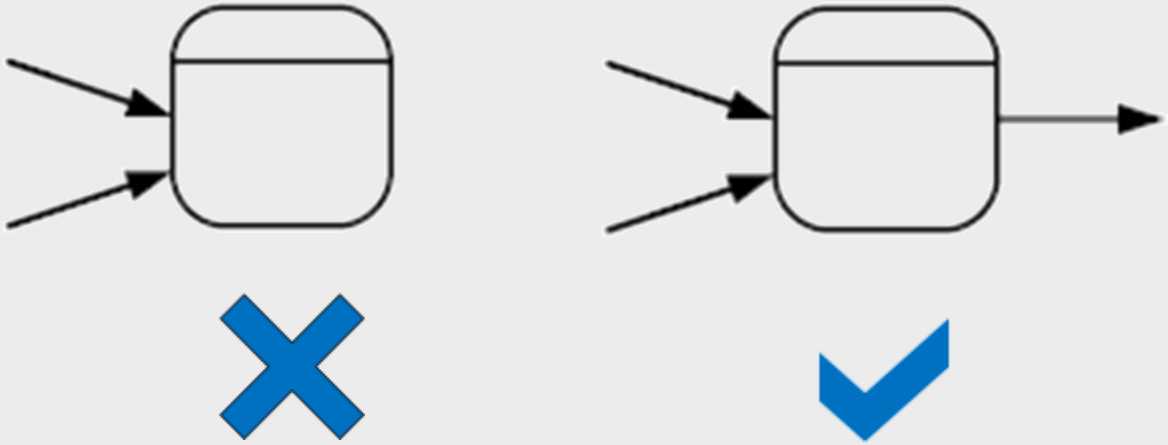
ลูกศรชี้เข้า หมายถึง เป็นการนำข้อมูลจากหน่วยงานภายนอกเข้าสู่ระบบ

ลูกศรชี้ออก หมายถึง ส่งข้อมูลจากระบบไปให้หน่วยงานภายนอก



กฎที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

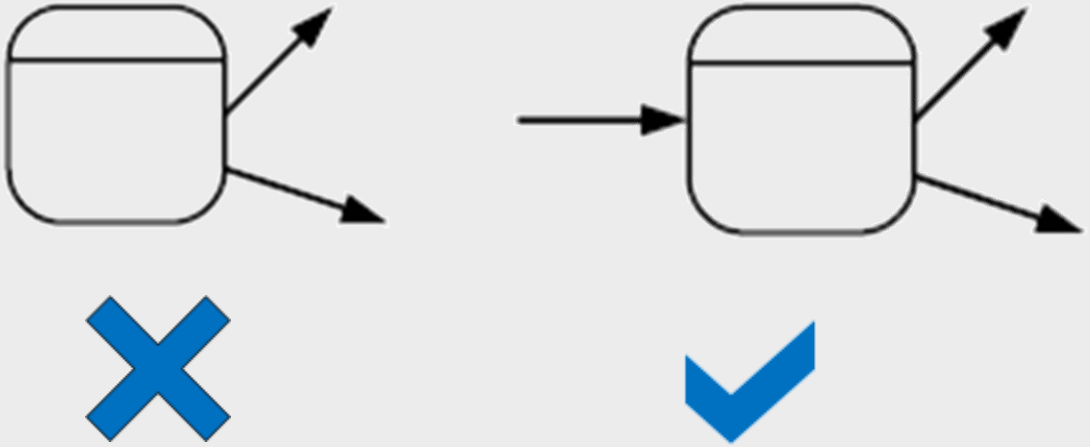
การประมวลผล(Process)



ในการประมวลผลจะมีข้อมูลเข้าอย่างเดียวไม่ได้

กฎที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

การประมวลผล(Process)



ในการประมวลผลจะมีข้อมูลออกอย่างเดียวไม่ได้

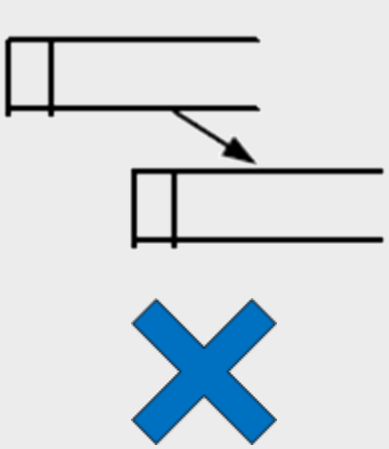
กฎที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

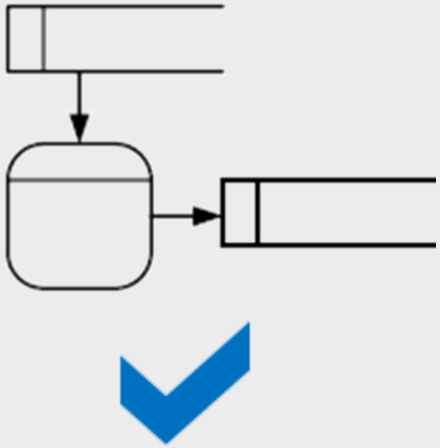
การประมวลผล(Process)

ข้อความที่บรรจุอยู่ในการประมวลผลจะต้องเป็นคำกริยา หรือคำนามที่แสดงถึงกริยาเท่านั้น

กฎที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

แหล่งเก็บข้อมูล(Data Store)

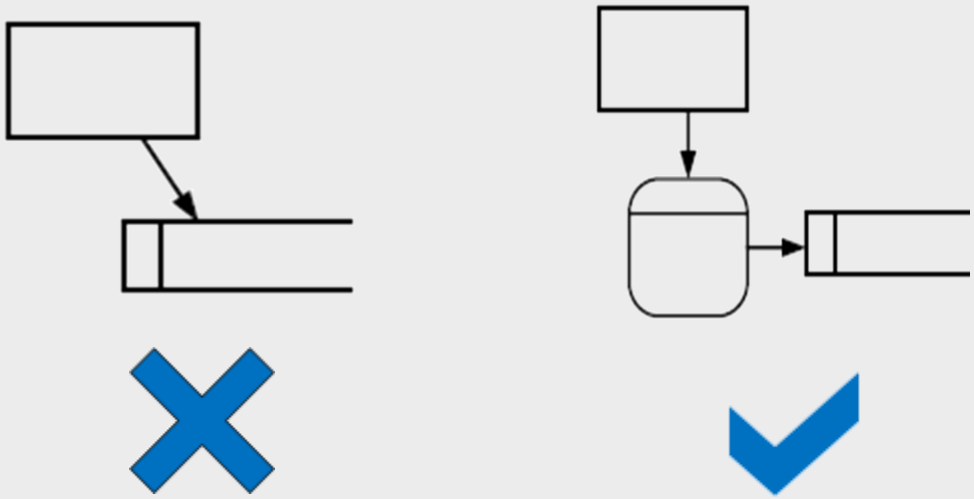




ไม่สามารถส่งข้อมูลโดยตรงจาก แหล่งเก็บข้อมูลอันหนึ่งไปยังแหล่งเก็บข้อมูลอีกอันได้โดยตรง ข้อมูลต้องผ่านการจากประมวลผลจึงจะไปที่ แหล่งเก็บข้อมูลได้

กฎที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

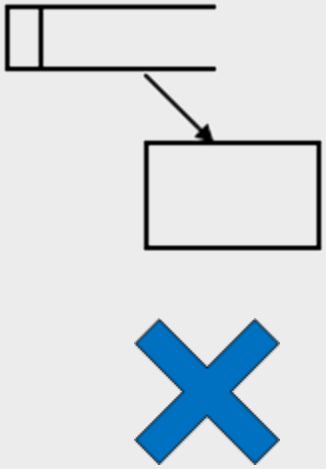
แหล่งเก็บข้อมูล(Data Store)

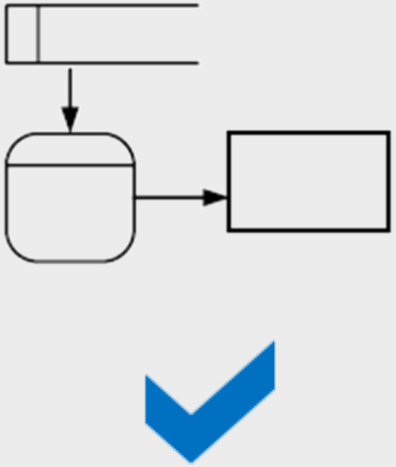


ไม่สามารถส่งข้อมูลจากสิ่งที่อยู่ภายนอก(External Entity)ไปยังแหล่งเก็บข้อมูลได้โดยตรง ข้อมูลต้องผ่านจากการประมวลผล ที่ได้รับข้อมูลจากแหล่งข้อมูลภายนอกแล้วจึงส่งต่อให้ แหล่งเก็บข้อมูล

กฎที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

แหล่งเก็บข้อมูล(Data Store)





ไม่สามารถส่งข้อมูลจากแหล่งเก็บข้อมูลไปยังสิ่งที่อยู่ภายนอกระบบปลายทางได้โดยตรง ต้องส่งผ่าน การประมวลผลไปยังสิ่งที่อยู่ภายนอกระบบ

กฎที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

แหล่งเก็บข้อมูล(Data Store)

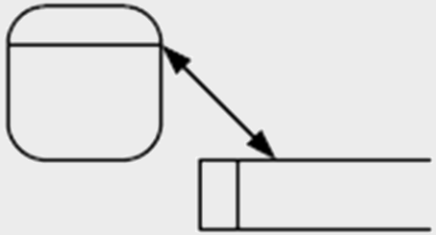
D1	พิมพ์ใบเสร็จ

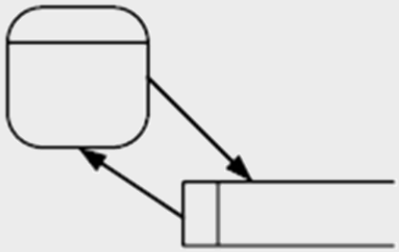
D1	แพ้มใบเสร็จ

ข้อความที่บรรจุอยู่ในแหล่งเก็บข้อมูลต้องเป็นคำนาม

กฎที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

กระแสข้อมูล(Data Flow)

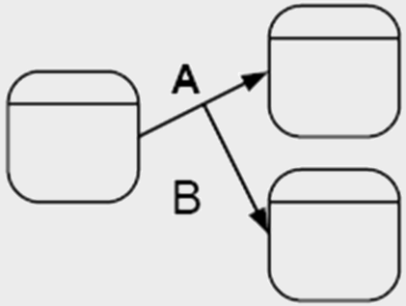


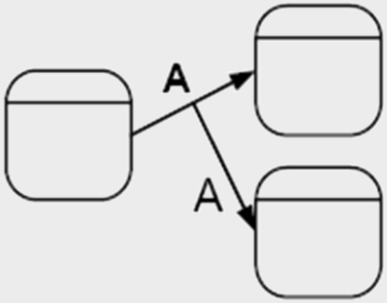


กระแสข้อมูล จะมีทิศทางการไหลของข้อมูลเพียงทิศเดียวใน 1 กระแสข้อมูล เนื่องจากไม่มีการทำงานใดที่เกิดขึ้นพร้อมกัน

กฎที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

กระแสข้อมูล(Data Flow)

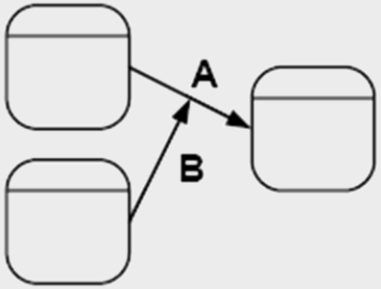


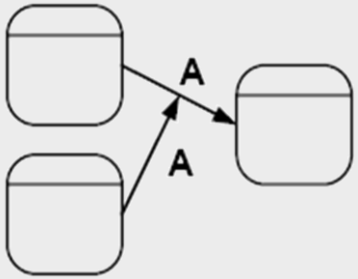


การแยกของข้อมูลออกจาก เส้นกระแสข้อมูล ไปสู่ การประมวลผล อื่น ๆ พร้อมกัน ต้องเป็นข้อมูลเดียวกัน

กฎที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

กระแสข้อมูล(Data Flow)

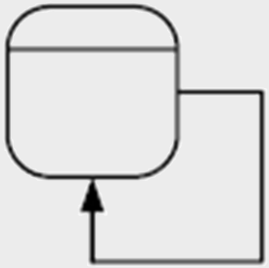


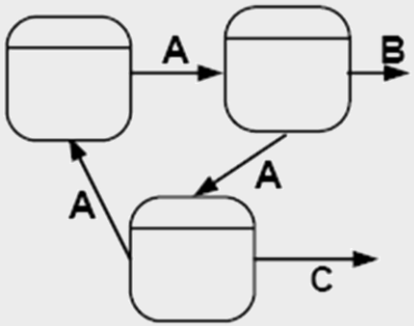


การเชื่อมข้อมูลจากข้อมูลมากกว่า 1 แหล่งเข้าสู่ เส้นกระแสข้อมูลเดียวกันต้องเป็นข้อมูลเดียวกัน

กฎที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

กระแสข้อมูล(Data Flow)

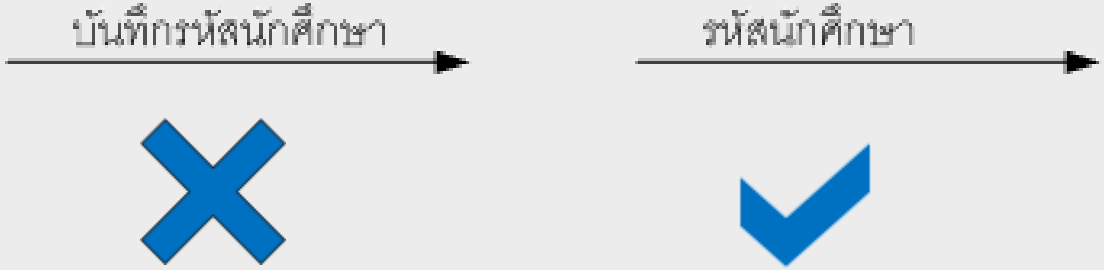




กระแสข้อมูลไม่สามารถไหลกลับเข้าสู่การประมวลเดิมได้โดยตรง จะต้องมีการประมวลอย่างน้อยหนึ่งอย่างมาจัดการกับข้อมูลดังกล่าวก่อน สร้างกระแสข้อมูลอื่น แล้วจึงคืนค่ากระแสข้อมูลเดิมกลับมายังการประมวลผลเริ่มต้น

กฎที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

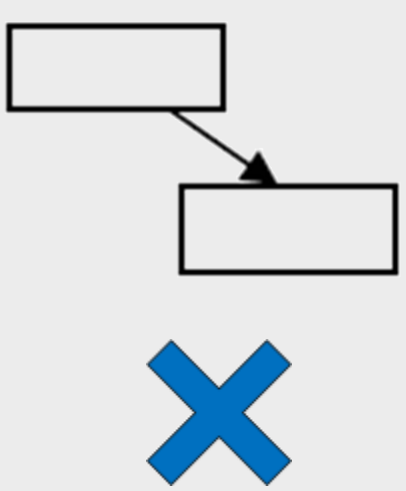
กระแสข้อมูล(Data Flow)

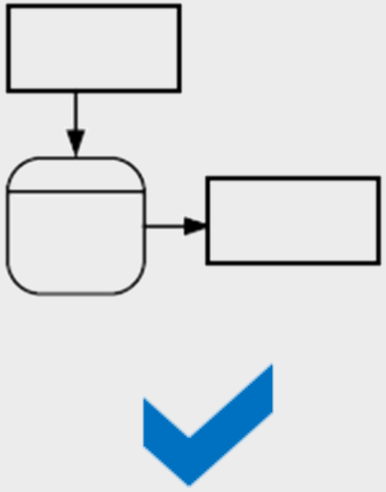


ข้อความที่อยู่บน Data Flow จะเป็นคำนาม

กฎที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สิ่งที่อยู่ภายนอก(External Entity)

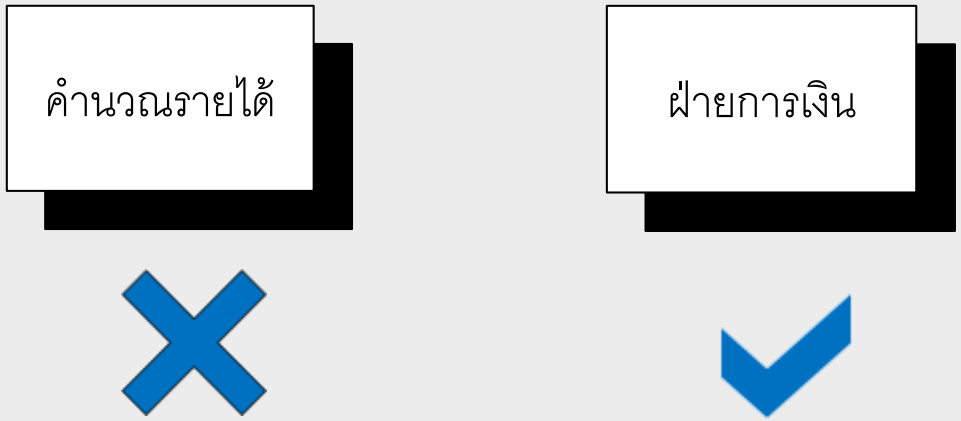




ไม่สามารถส่งข้อมูลโดยตรงจาก สิ่งที่อยู่ภายนอกระบบอันหนึ่งไปยังสิ่งที่อยู่ภายนอกระบบอีกอันได้โดยตรง ข้อมูลต้องผ่านการจากประมวลผลภายในระบบก่อนจึงส่งไปยังสิ่งที่อยู่ภายนอกระบบภายปลายทางได้

กฎที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สิ่งที่อยู่ภายนอก(External Entity)



ข้อความที่บรรจุอยู่ในสิ่งที่อยู่ภายนอกระบบต้องเป็นคำนาม

ประเภทของแผนภาพกระแสข้อมูล

การเขียน DFD นั้นสามารถเขียนได้ 2 แบบ คือ

1) แบบตรรกะ (Logical Data Flow Diagram)

แผนภาพนี้จะเป็นการเน้นในส่วนของธุรกิจ ว่าธุรกิจมีการทำงานอย่างไร มีเหตุการณ์อะไรบ้างที่เกิดขึ้น, ข้อมูลที่ต้องการมีอะไรบ้าง และได้ข้อมูลอะไรจากเหตุการณ์นั้นๆ แต่ไม่ได้บอกว่าระบบจะถูกสร้างอย่างไร

2) แบบกายภาพ (Physical Data Flow Diagram)

แผนภาพนี้จะบอกได้ว่าจะสร้างระบบอย่างไร ประกอบด้วยฮาร์ดแวร์, ซอฟต์แวร์, แฟ้มข้อมูล และบุคคลใดที่เกี่ยวข้องบ้างในระบบ

ประเภทของแผนภาพกระแสข้อมูล

ข้อแตกต่างระหว่างแผนภาพกระแสข้อมูลแบบตรรกะ กับแบบกายภาพ

- ขั้นตอนการประมวลผล

Logical DFD -> แสดงถึง ธุรกิจสามารถที่จะดำเนินการได้อย่างไร

Physical DFD -> แสดงถึง ระบบสามารถที่จะทำงานได้อย่างไร

Logical DFD -> เป็นกิจกรรมทางธุรกิจ

Physical DFD -> เป็นโปรแกรม หรือโมดูล หรือขั้นตอนการทำงานที่ทำด้วยมือ

ประเภทของแผนภาพกระแสข้อมูล

ข้อแตกต่างระหว่างแผนภาพกระแสข้อมูลแบบตรรกะ กับแบบกายภาพ

- แหล่งเก็บข้อมูล

Logical DFD -> เป็นแหล่งที่เก็บรวบรวมข้อมูลลักษณะใดก็ได้

Physical DFD -> เป็น Physical File, Database และ Manual File

- ชนิดของแหล่งเก็บข้อมูล

Logical DFD -> เป็นการเก็บข้อมูลแบบถาวร

Physical DFD -> เป็นการเก็บข้อมูลที่แยกเป็นแฟ้มหลัก และแฟ้มรายการ ต้องการประมวลผลในเวลาที่แตกต่างกัน สามารถเชื่อมต่อกันได้โดยใช้แหล่งเก็บข้อมูล

ประเภทของแผนภาพกระแสข้อมูล

ข้อแตกต่างระหว่างแผนภาพกระแสข้อมูลแบบตรรกะ กับแบบกายภาพ

- การควบคุมระบบ

Logical DFD -> เป็นการควบคุมที่เกี่ยวข้องกับธุรกิจ

Physical DFD -> แสดงถึง การควบคุมสำหรับการป้อนข้อมูลที่ถูกต้อง

ประเภทของแผนภาพกระแสข้อมูล

ประโยชน์ของการสร้างแผนภาพกระแสข้อมูลแบบตรรกะ (Logical DFD)

1) ช่วยให้การสื่อสารระหว่างผู้ใช้งานให้ดีขึ้น
2) ช่วยให้ระบบมีความมั่นคงมากขึ้น
3) ช่วยให้นักวิเคราะห์ระบบมีความเข้าใจกับการดำเนินงานของระบบได้ชัดเจน
4) ช่วยในการบำรุงรักษาและมีความยืดหยุ่นมากขึ้น
5) ช่วยลดความซับซ้อน และง่ายต่อการสร้างแผนภาพกระแสข้อมูลแบบกายภาพต่อไป

ประเภทของแผนภาพกระแสข้อมูล

ประโยชน์ของการสร้างแผนภาพกระแสข้อมูลแบบกายภาพ (Physical DFD)

1) ขั้นตอนการประมวลผลนั้นเป็นการทำงานด้วยระบบมือ หรือระบบอัตโนมัติ
2) แผนภาพแบบกายภาพจะอธิบายรายละเอียดขั้นตอนการประมวลผลได้ละเอียด
3) มีการแสดงถึงลำดับขั้นตอนการทำงาน

การพัฒนาแผนภาพกระแสข้อมูล

1) กำหนดรายการกิจกรรมต่างๆ ของธุรกิจ และแยกออกมาว่าอยู่ในรูปแบบใด (External Entities หรือ Data Flows หรือ Processes หรือ Data Stores)

2) สร้างแผนภาพระดับสูงสุด (Context Diagram) ซึ่งแสดงถึง External Entities และข้อมูลที่ไหลเข้าและออกจากระบบหลัก โดยไม่สนใจแหล่งเก็บข้อมูล

3) เขียน Diagram ระดับถัดไป คือแผนภาพระดับล่าง (เรียกว่า Diagram 0 หรือ Parent Diagram) ซึ่งแสดงถึง Process ต่างๆ ที่มี แต่เขียนให้อยู่ในรูปแบบทั่วไป พร้อมกันนั้นให้แสดง Data Store ในระดับนี้ด้วย

การพัฒนาแผนภาพกระแสข้อมูล

4) สร้างแผนภาพระดับลูกของแต่ละ Process ใน Diagram 0 เรียกแผนภาพในระดับนี้ว่า แผนภาพกระแสข้อมูลระดับ 1 (Level-1 diagram) ถ้าหากมีรายละเอียดของการทำงานย่อยจากระดับนี้อีก ก็ให้แตกรายละเอียด ลงไปจนกระทั่งสิ้นสุด ส่วนชื่อของระดับก็จะเป็น Level-2 diagram , Level-3 diagram ไปเรื่อยๆ จนกระทั่งหมด

5) ตรวจสอบหาข้อผิดพลาด และดูว่าคำกำกับบนเส้น Data Flow แต่ละเส้น รวมถึง Process แต่ละอันนั้น สื่อความหมายหรือไม่

6) หลังจากเขียนแผนภาพจนครบทุกการทำงานแล้ว ให้ทำการตรวจสอบสมดุลระหว่างข้อมูลเข้าและข้อมูลออกของแผนภาพ DFD กับ Context diagram

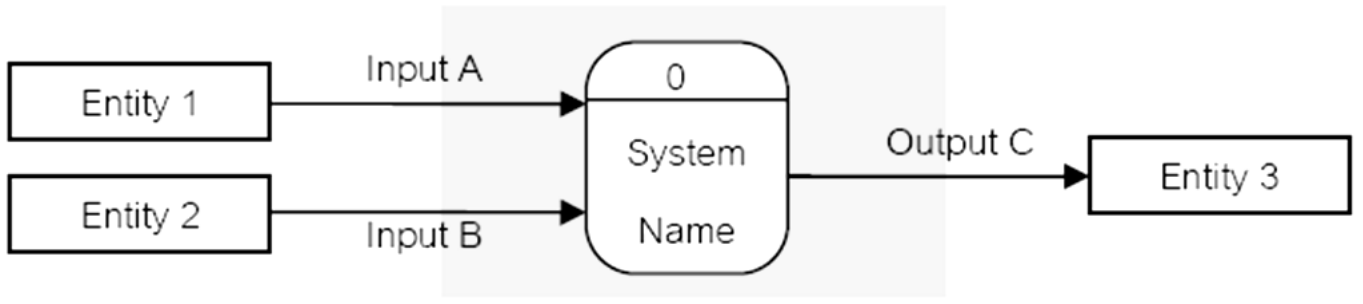
การพัฒนาแผนภาพกระแสข้อมูล

7) พัฒนารูปแบบใหม่ จาก Logical Data Flow Diagram ให้ไปอยู่ในรูป Physical Data Flow Diagram เพื่อแยกระหว่างระบบที่ทำด้วยมือ กับระบบที่ทำงานอัตโนมัติ

8) แบ่งส่วนของ Physical Data Flow Diagram โดยการแยกหรือแบ่งกลุ่มของ Diagram ออก เพื่อให้สามารถนำไปเขียนโปรแกรม หรือเพื่อการดำเนินการระบบได้

การพัฒนาแผนภาพกระแสข้อมูล

แผนภาพกระแสข้อมูลระดับสูงสุด (Context Diagram)



แผนภาพกระแสข้อมูลระดับสูงสุด เป็นแผนภาพที่แสดงถึงขอบเขตของสารสนเทศนั้น โดยจะเป็นมุมมองระดับสูงว่ามีหน่วยงานใดเกี่ยวข้องบ้าง ติดต่อกับระบบโดยมีการรับและส่งข้อมูลใดกับระบบ ซึ่งแผนภาพระดับนี้จะยังไม่กล่าวถึง สัญลักษณ์การเก็บข้อมูล (Data Store Symbol)

การพัฒนาแผนภาพกระแสข้อมูล

แผนภาพกระแสข้อมูลระดับสูงสุด (Context Diagram)

หลักการสร้าง Context Diagram ที่ดี

1) ควรจะเขียนให้ครอบคลุมระบบ ให้อยู่ใน 1 หน้ากระดาษ

2) ชื่อของ Process ควรเป็นชื่อของระบบสารสนเทศ นั้น

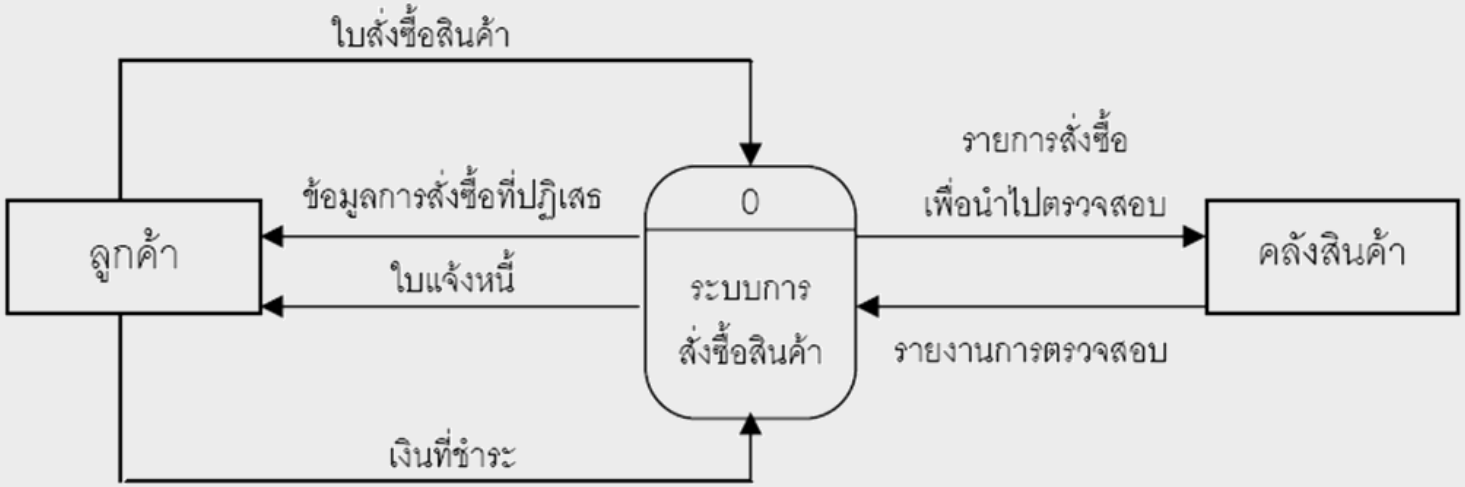
3) ชื่อที่เขียนกำกับสัญลักษณ์ต่างๆ จะต้องเขียนโดยไม่ให้ชื่อซ้ำกัน ถ้าสัญลักษณ์นั้นแทนสิ่งที่แตกต่างกัน

4) ควรหลีกเลี่ยงเส้นที่จะต้องเขียนคร่อมเส้นกัน

5) ในการเขียนชื่อย่อ จะต้องเขียนโดยบันทึกความหมายของตัวย่อนั้น ในพจนานุกรมข้อมูล (Data Dictionary)

การพัฒนาแผนภาพกระแสข้อมูล

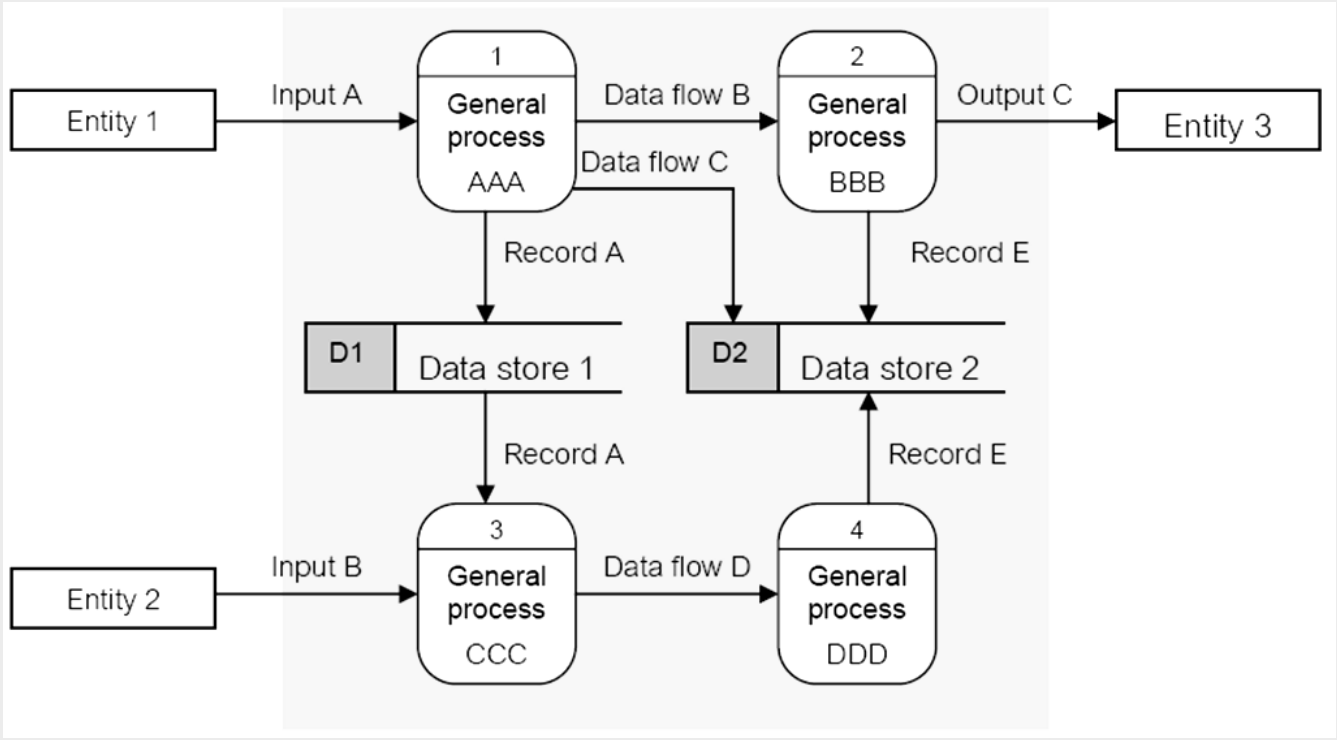
แผนภาพกระแสข้อมูลระดับสูงสุด (Context Diagram)



ตัวอย่างการเขียน Context Diagram ของระบบการสั่งซื้อสินค้

การพัฒนาแผนภาพกระแสข้อมูล

แผนภาพกระแสข้อมูลระดับล่าง (Diagram 0 หรือ Parent Diagram)



แผนภาพกระแสข้อมูลระดับล่าง เป็นแผนภาพที่ให้รายละเอียดเพิ่มเติมจากแผนภาพระดับสูงสุด ทำให้เห็นภาพรวมของแผนภาพกระแสข้อมูล ซึ่งจะมีรายละเอียดมากกว่าแผนภาพกระแสข้อมูลระดับสูงสุด

การพัฒนาแผนภาพกระแสข้อมูล

แผนภาพกระแสข้อมูลระดับล่าง (Diagram 0 หรือ Parent Diagram)

หลักการสร้างแผนภาพกระแสข้อมูลระดับล่าง

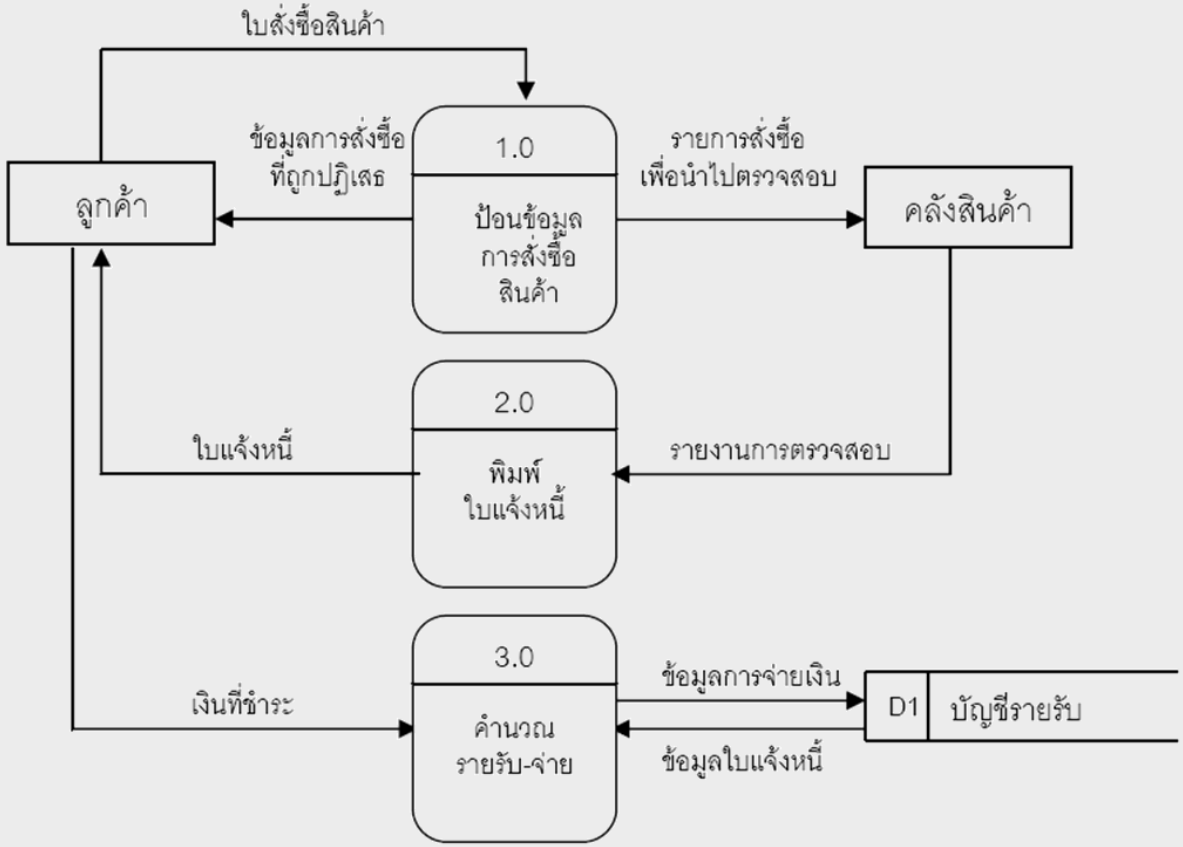
1) การเขียนแผนภาพกระแสข้อมูลควรเขียนให้ดูง่ายและให้แสดงอยู่ในกระดาษแผ่นเดียว

2) ลูกศรไม่ควรทับหรือข้ามกัน

3) แต่ละ Process จะต้องมีหมายเลขกำกับเป็นเลขจำนวนเต็ม โดยการลำดับหมายเลข Process ไม่ได้ หมายถึงการทำงานต้องทำงานตามลำดับของ Process และ Process ไม่สามารถทำการซ้ำได้

การพัฒนาแผนภาพกระแสข้อมูล

แผนภาพกระแสข้อมูลระดับล่าง (Diagram 0 หรือ Parent Diagram)



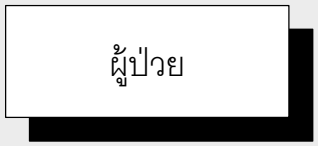
ตัวอย่างการเขียน Diagram 0 ของระบบการสั่งซื้อสินค้า

การพัฒนาแผนภาพกระแสข้อมูล

แผนภาพกระแสข้อมูลระดับล่าง (Diagram 0 หรือ Parent Diagram)

หลักการสร้างแผนภาพกระแสข้อมูลระดับล่าง

4) External Entity ทั้งหมดของระบบ ที่เขียนใน Context Diagram จะต้องแสดงอยู่ใน Diagram 0 ด้วย โดยที่สามารถทำการซ้ำได้ ด้วยเครื่องหมาย \ ตรงมุมล่างซ้ายของสัญลักษณ์ External Entity



สิ่งที่อยู่นอกระบบและการทำซ้ำสิ่งที่อยู่นอกระบบ

5) จำนวน Process ไม่ควรมีมากเกินไป หรือน้อยเกินไป จำนวน Process ที่มากเกินไปจะทำให้แผนภาพกระแสข้อมูลอ่านยาก และมีความซับซ้อนยิ่งขึ้น

การพัฒนาแผนภาพกระแสข้อมูล

แผนภาพกระแสข้อมูลระดับต่ำ (Lower Level Data Flow Diagram)

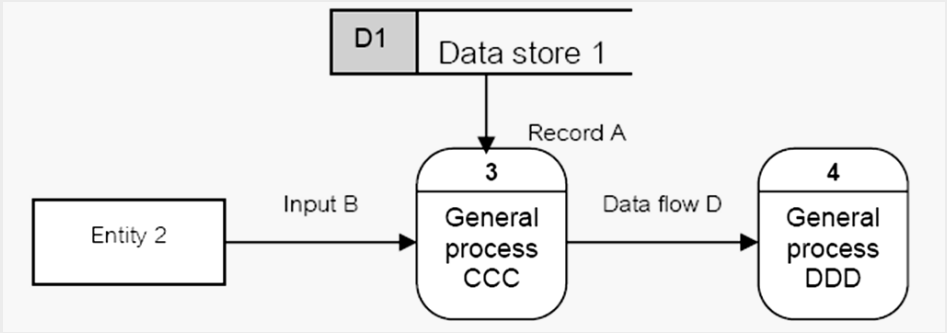
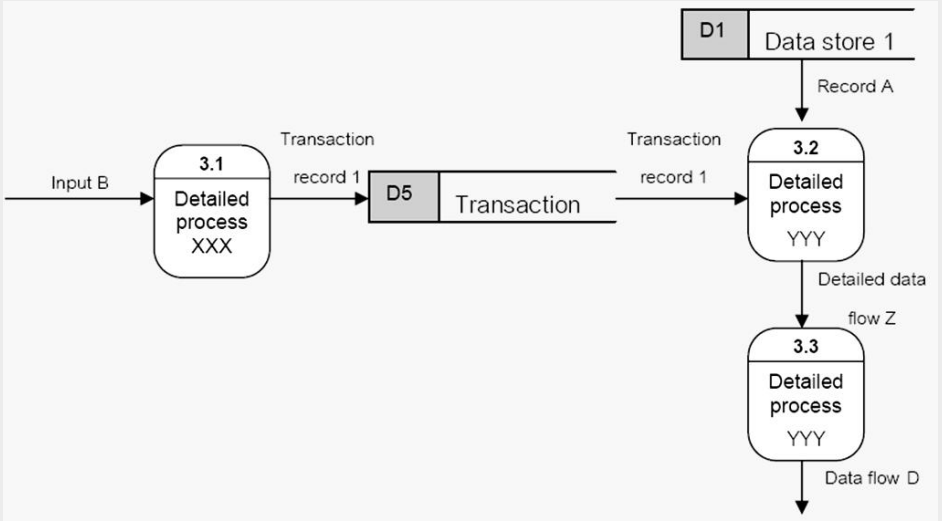


Diagram 0

(DFD Level 0)

Lower Level

(DFD Level 1)



การพัฒนาแผนภาพกระแสข้อมูล

แผนภาพกระแสข้อมูลระดับต่ำ (Lower Level Data Flow Diagram)

หลักการสร้างแผนภาพกระแสข้อมูลระดับต่ำ

1) แต่ละ Process บน Diagram 0 (หรือ Parent Process) สามารถนำมาสร้าง Diagram ย่อย (เรียกว่า Child Diagram) ได้

2) ในการสร้าง Child Diagram นั้น ข้อมูลที่เข้ามาและออกจาก Child Diagram จะต้องเป็นข้อมูลที่เข้ามาและออกจาก Parent Process

3) เมื่อมีการแยก Process ออกเป็น Child Diagram จะต้องมีการกำหนดหมายเลขลำดับให้กับ Child Diagram นั้นโดยจะตั้งตาม Parent Process (เช่น Process 3 แยกเป็น Diagram Level-1 ของ Process ที่ 3)

การพัฒนาแผนภาพกระแสข้อมูล

แผนภาพกระแสข้อมูลระดับต่ำ (Lower Level Data Flow Diagram)

หลักการสร้างแผนภาพกระแสข้อมูลระดับต่ำ

4) หมายเลขกำกับแต่ละ Process ใน Child Diagram ก็จะใช้หมายเลขขึ้นต้นเหมือน Parent Process แล้วตามด้วยจุดทศนิยม และหมายเลขย่อยลงไป (เช่น Parent Process เป็น Process 3 ดังนั้นใน Diagram 3 ก็จะมี Process 3.1,3.2,3.3 ไปเรื่อยๆจนกว่าจะหมด)

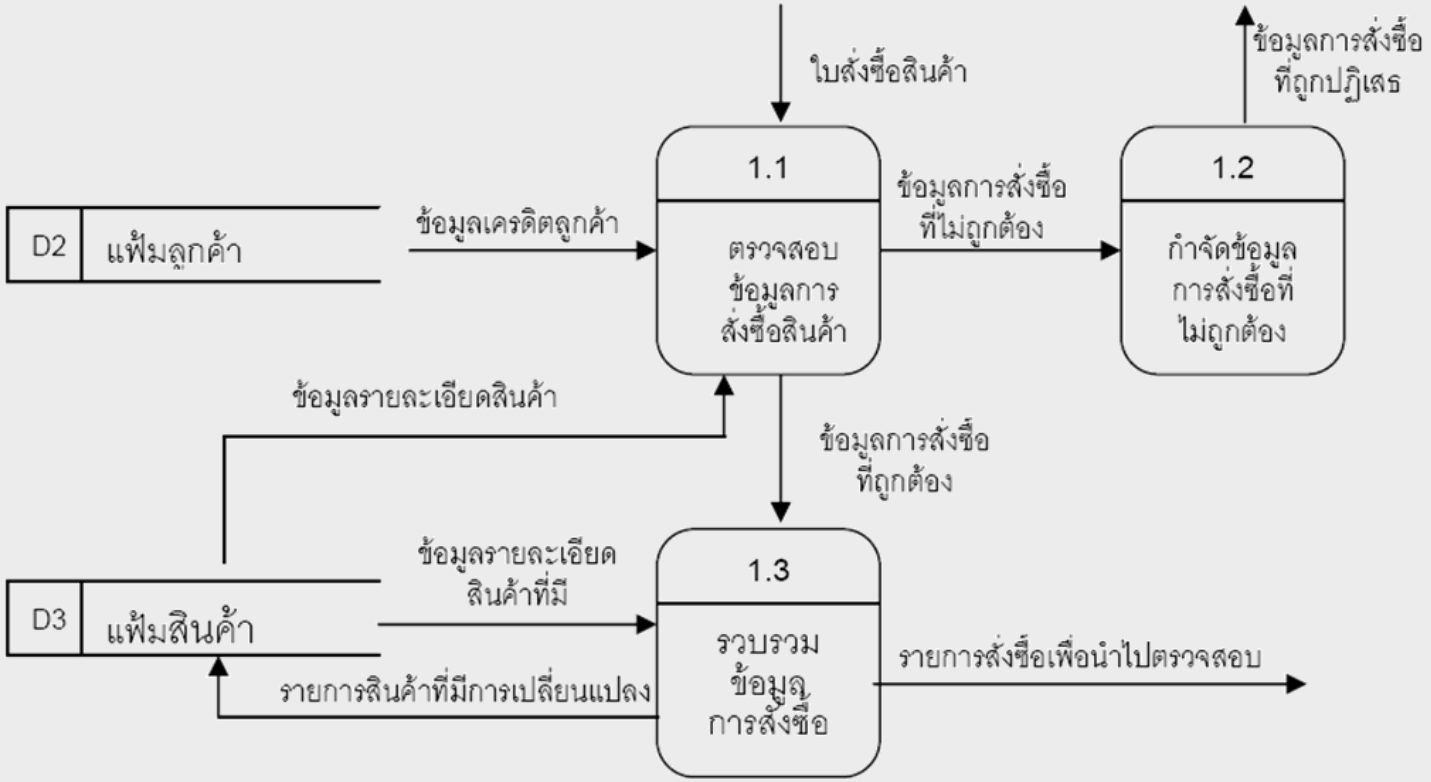
5) โดยปกติแล้ว External Entity มักจะไม่แสดงบน Child Diagram ที่ต่ำกว่า Diagram 0

6) บน Child Diagram อาจมี Data Store ปรากฏขึ้นมา โดยที่ไม่มีใน Parent Process ได้

7) Process ต่างๆ อาจมีการแตกหรือไม่แตกเป็น Child Diagram ก็ได้ ขึ้นอยู่กับระดับความซับซ้อนของ Process นั้นๆ

การพัฒนาแผนภาพกระแสข้อมูล

แผนภาพกระแสข้อมูลระดับต่ำ (Lower Level Data Flow Diagram)



การพัฒนาแผนภาพกระแสข้อมูล

การตรวจสอบแผนภาพกระแสข้อมูลเพื่อหาข้อผิดพลาด

1) ตรวจสอบว่า มีการใช้สัญลักษณ์ต่างๆ ถูกต้องตามกฎการเขียนแผนภาพกระแสข้อมูล หรือไม่

2) ตรวจสอบหาข้อผิดพลาด และดูว่าคำกำกับบนเส้น Data Flow แต่ละเส้น รวมถึง Process แต่ละอันนั้น สื่อความหมายหรือไม่

3) ตรวจสอบสมดุลระหว่างข้อมูลเข้าและข้อมูลออกของแผนภาพ นั่นคือ ตรวจสอบว่าแผนภาพระดับสูงสุด (Context Diagram) มีความสมดุลระหว่างข้อมูลเข้าและข้อมูลออก กับแผนภาพระดับล่าง (Diagram 0) หรือไม่ และตรวจสอบว่าแผนภาพระดับล่าง มีความสมดุลระหว่างข้อมูลเข้าและข้อมูลออกกับแผนภาพระดับต่ำ (Child Diagram) หรือไม่