Process Modeling :

Data Flow Diagrams 6

06066304
Information System Analysis and Design

Asst. Prof. Manop Phankokkruad, Ph.D.
School of Information Technology
King Mongkut's Institute of Technology Ladkrabang

KMITL

Outline

1. Process Models and Data Flow Diagrams
2. Elements of DFDs
   - Syntax
3. Creating DFDs
   - Context Diagram
   - Level 0 Diagrams
   - Level 1 Diagrams
4. Validating the DFD

KMITL

System Modeling

System Modeling is the use of models to conceptualize and construct systems in business and IT development.

- ***Behavioral Modeling*** describe the internal dynamic aspects of an information system that supports the business processes.
- ***Structural Modeling*** describes the structure of the objects that supports the business processes.
- ***Data modeling*** is a process used to define and analyze data requirements needed to support the business processes within the systems.
- ***Functional Modeling*** describe business processes and the interaction of an information system with its environment.

KMITL

Process models

Process models have been a part of structured systems analysis and design techniques. A process model is a description of a process at the type level. Since the process model is at the type level, a process is an instantiation of it. The same process model is used repeatedly for the development of many applications and thus, has many instantiations.

A **process model** is to prescribe how things must/should/could be done in contrast to the process itself which is really what happens. A process model is roughly an anticipation of what the process will look like. What the process shall be will be determined during actual system development.

KMITL

Data Flow Diagrams

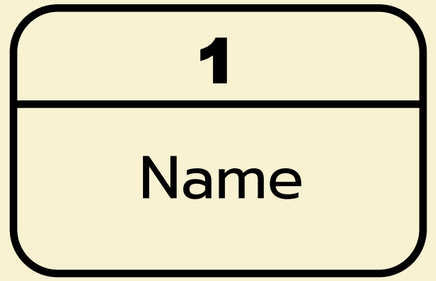
Data Flow Diagram (DFD) is a graphical representation of the flow of data through an information system, modeling its process aspects.

- DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated.
- Although the name DFD implies a focus on data, this is not the case. The focus is mainly on the processes or activities that are performed.
- DFD is a diagram that consists principally of four symbols, namely the external entity, the data flow, the process and the data store.

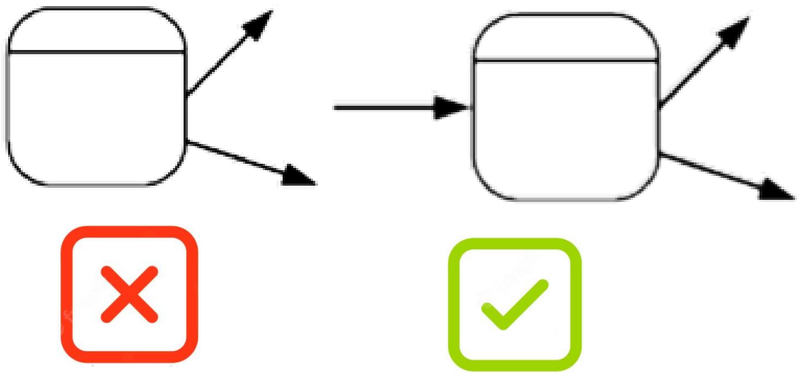
KMITL

Elements of DFDs

- **Process** is an activity or a function that is performed for some specific business reason.
- Processes can be manual or computerized.



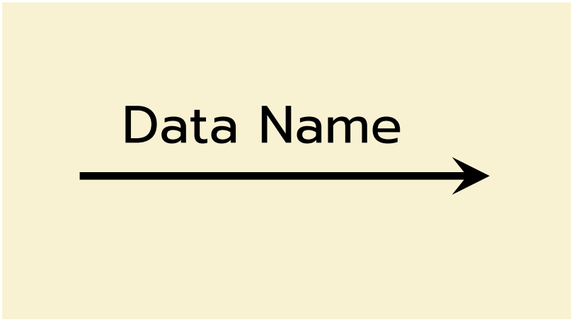
- Every process should be named starting with a *verb and ending with a noun*.
- In addition, every process must have at *least one input data flow and at least one output data flow.*



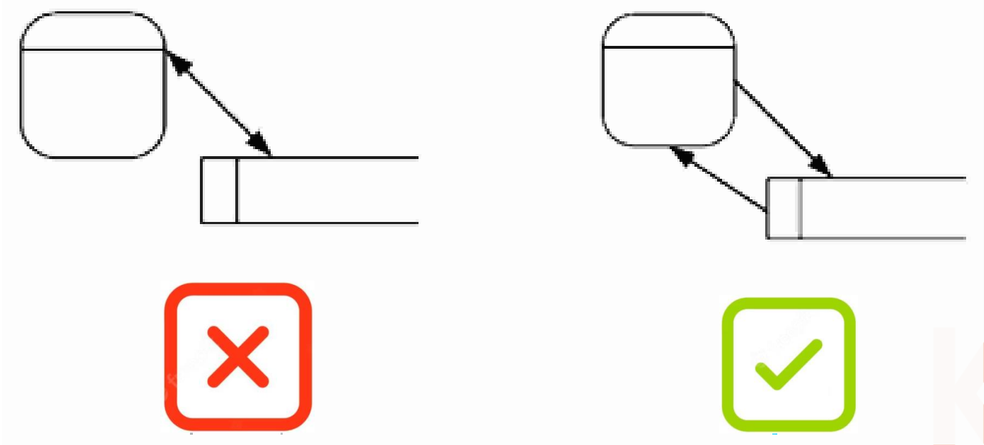
KMITL

Elements of DFDs

- **Data flow** is a single piece of data, or a logical collection of several pieces of information.
- Every data flow should be *named with a noun*.

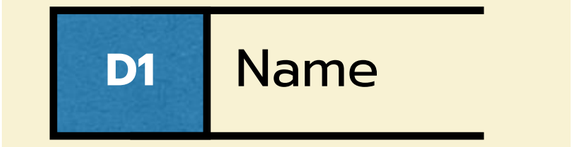


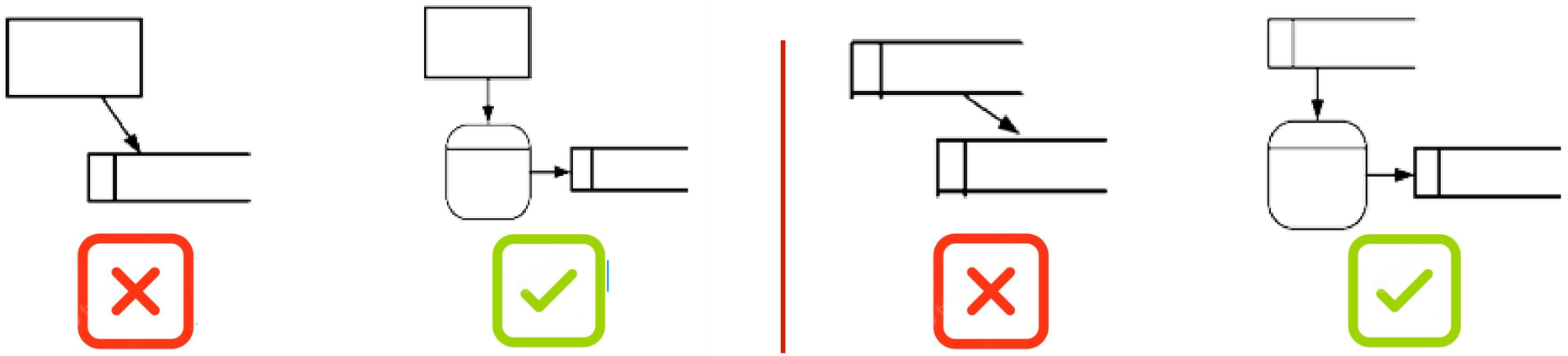
- One end of every data flow will always *come from or go to a process*, with the arrow showing the direction into or out of the process.
- Data flows show what inputs go into each process and what outputs each process produces.



Elements of DFDs

- **Data store** is a collection of data that is stored in some way.
- Every data store is *named with a noun* and is assigned an identification number and a description.
- Data store is *the principal link between the process model and the data model*.

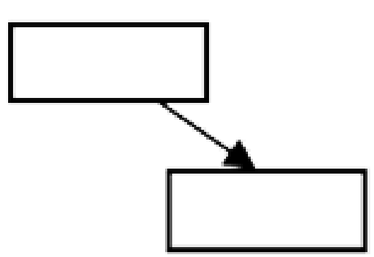


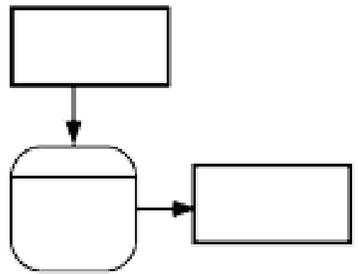


Elements of DFDs

- **External entity** is a person or system that is external to the system but interacts with it.
- The external entity typically corresponds to the primary actor identified in the use case.

- External entities provide data to the system or receive data from the system and serve to establish the system boundaries.
- Every external entity has a name and a description.





DFD : Context Diagram

- The first DFD in every business process is the **context diagram**.
- It shows the entire system in context with its environment.
- The context diagram shows the overall business process as just one process and shows the data flows to and from external entities.

DFD : Level 0 diagrams

- The **level 0 diagram** (or level 0 DFD) shows all the major high-level processes of the system and how they are interrelated.
- The **level 0 diagram** shows all the processes at the first level the numbering, the data stores, external entities, and data flows among them.
- A process model has one and only one level 0 DFD.

KMITL

DFD : Level 1 diagrams

- Each process on the level 0 DFD can be decomposed into a more explicit DFD called **level 1 diagram** (or level 1 DFD).
- The set of children and the parent are identical; they are simply different ways of looking at the same thing.
- It is important to ensure that level 0 and level 1 DFDs are balanced.
- All process models have as many level 1 diagrams as there are processes on the level 0 diagram.
- The parent process and the children processes are numbered consistently.

KMITL

DFD : Level 2 diagrams

- The next level of decomposition: a **level 2 diagram**, or level 2 DFD.
- A level 2 DFD shows all processes, data flows, and data stores that comprise a single process on the level 1 diagram.
- It is important to ensure that level 1 and level 2 DFDs are balanced.

KMITL

Creating DFDs

- DFDs start with the information in the *use cases* and the requirements definition.
- Generally, the set of DFDs integrates the *individual use cases*.
- The project team takes the use cases and rewrites them as DFDs, following the DFD formal rules about symbols and syntax.
- CASE tools are used to draw process models.

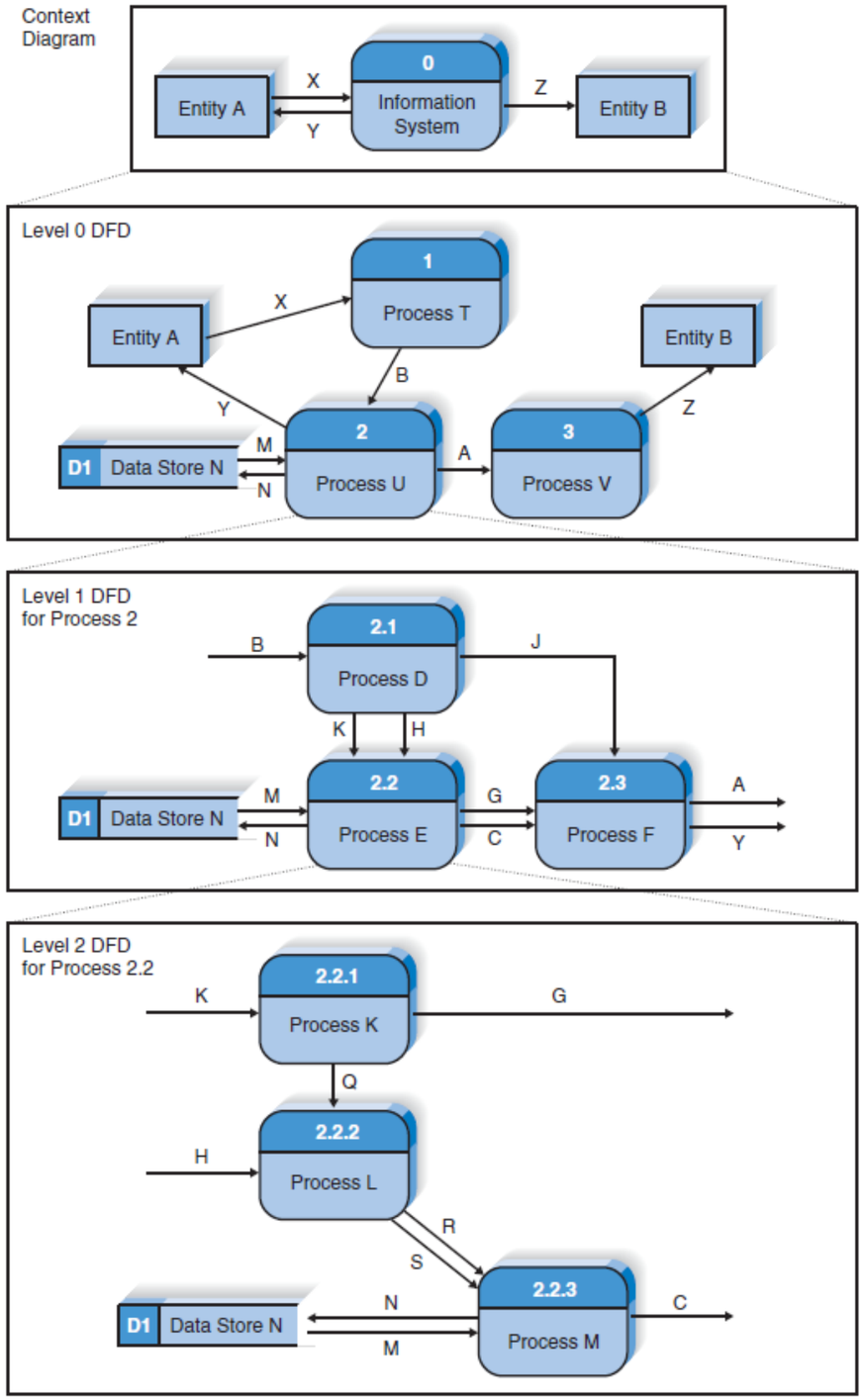
Creating DFDs

1. Make a list of business activities and use it to determine various : *External entities*, *Data flows*, *Processes*, *Data stores*.
2. Build the context diagram.
3. Create DFD fragments for each use case.
4. Organize the DFD fragments into level 0 diagram.
5. Develop level 1 DFDs based on the steps with each use case. In some cases, these level 1 DFDs are further decomposed into level 2 DFDs, level 3 DFDs., and so son.
6. Validate the set of DFDs to make sure that they are complete and correct.

KMITL

Creating DFDs

Figure: Various Level of Data Flow Diagrams



Context diagram

Level 0 diagram

Level 1 diagram

Level 2 diagram

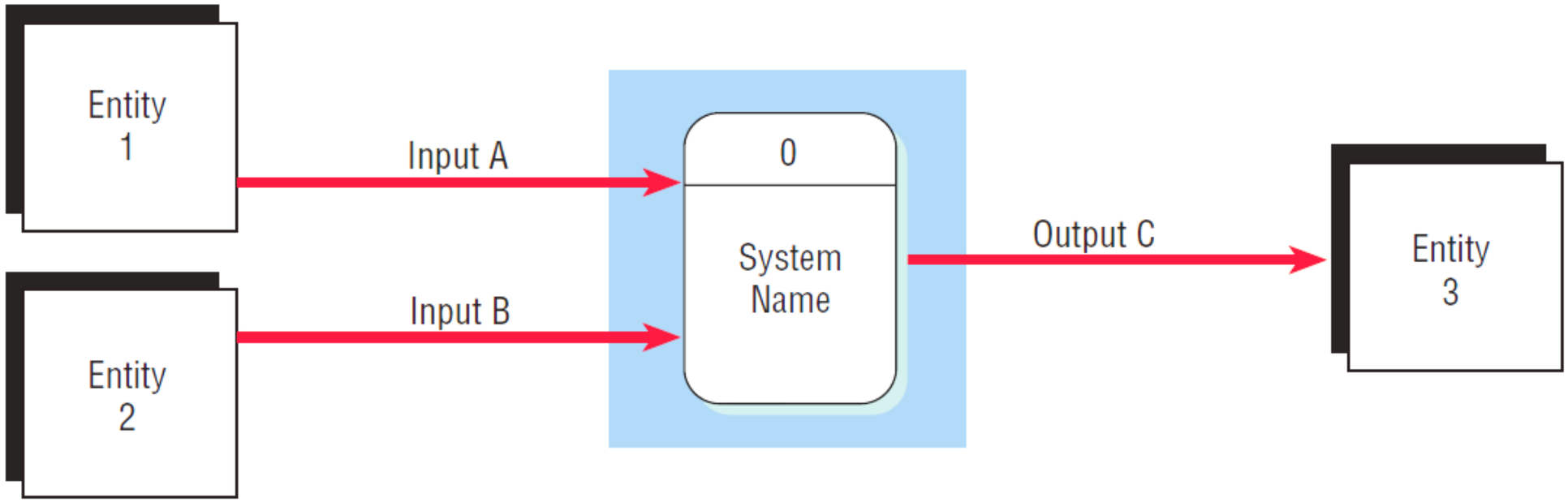
Creating the Context Diagram

- The context diagram defines how the business process or computer system interacts with its environment.
- Draw one process symbol for the business process or system being modeled (numbered 0 and named for the process or system).
- Add all inputs and outputs listed on the form of the use cases as data flows.
- Draw in external entities as the source or destination of the data flows.
- No data stores are included in the context diagram.

KMITL

Creating the Context Diagram

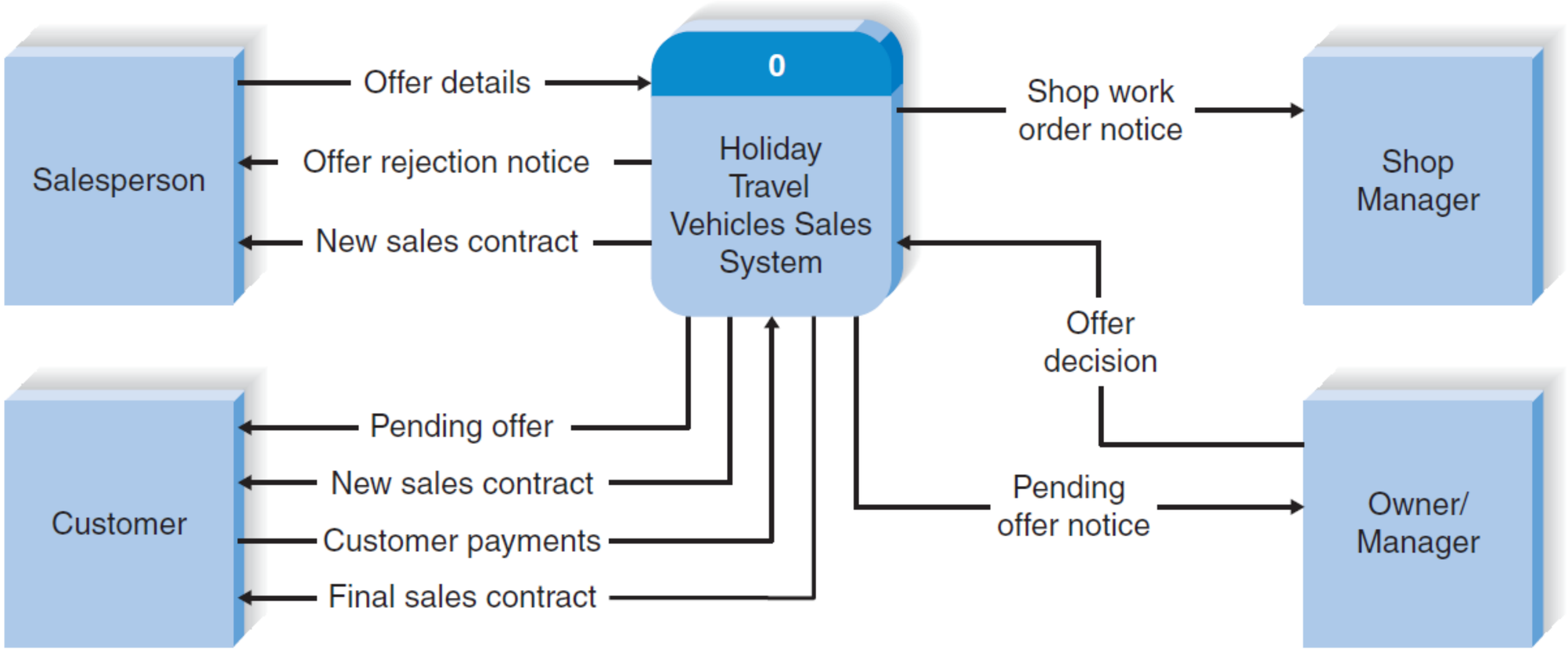
Figure: Example of Context Diagram #1



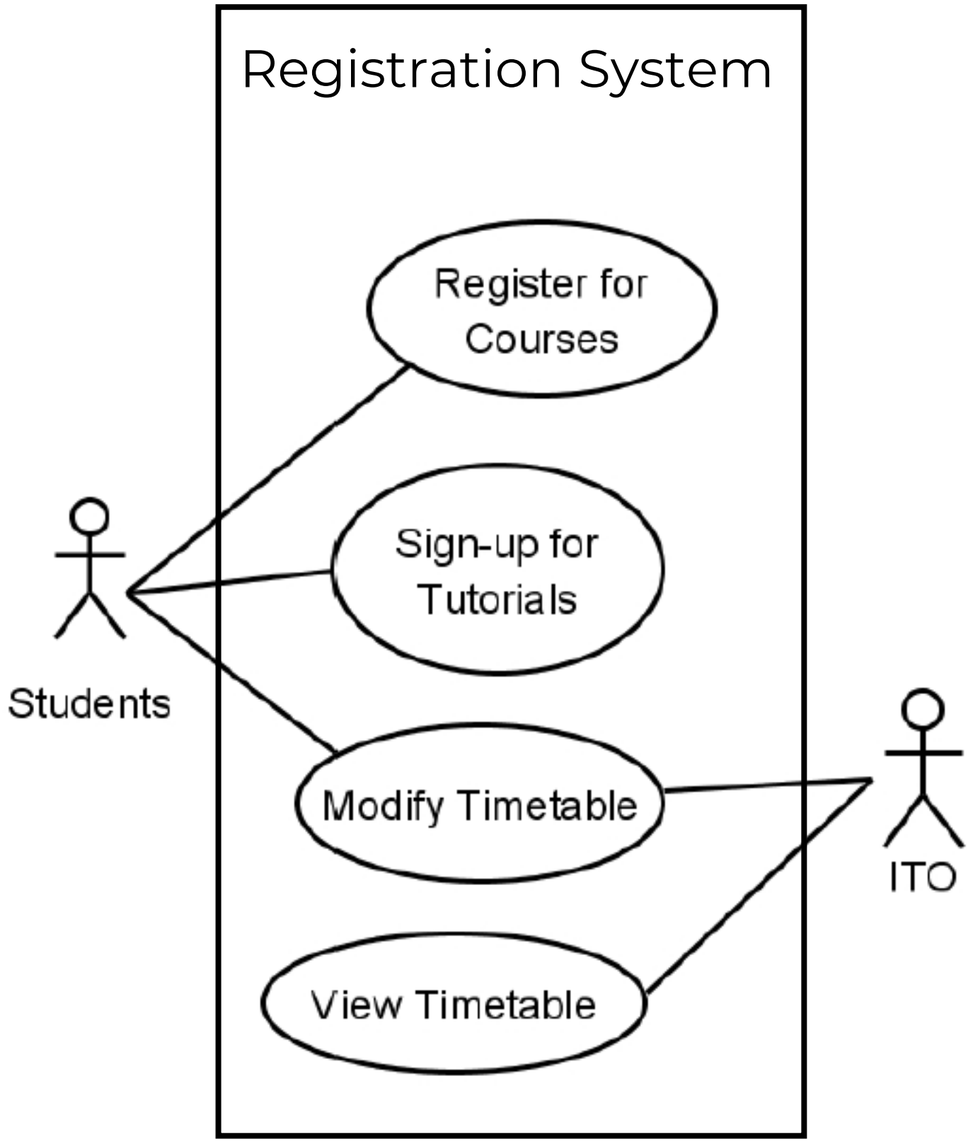
KMITL

Creating the Context Diagram

Figure: Example of Context Diagram



Creating the Context Diagram



Creating the Level 0 Diagram

- Combine the set of DFD fragments into one diagram – the level 0 DFD.
- There are not formal layout rules. Generally,
  a) to put the process that is first chronologically in the upper-left corner and work the way from top to bottom, left to right;
  b) to reduce the number of crossed data flow lines.
- Iteration is the cornerstone of good DFD design.

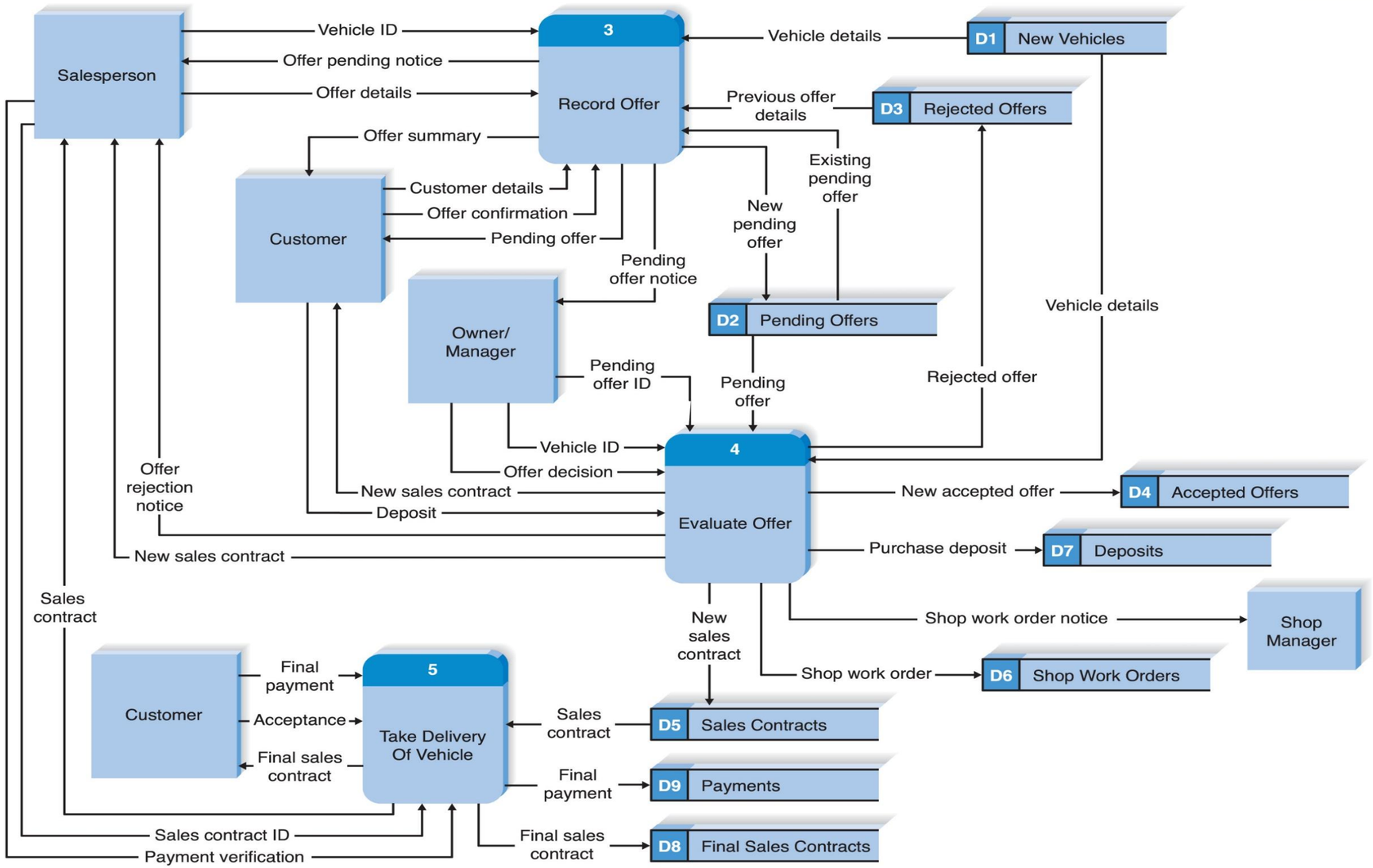
MP.ISAD.IT.KMITL

KMITL

Creating the Level 0 Diagram

MP.ISAD.IT.KMITL

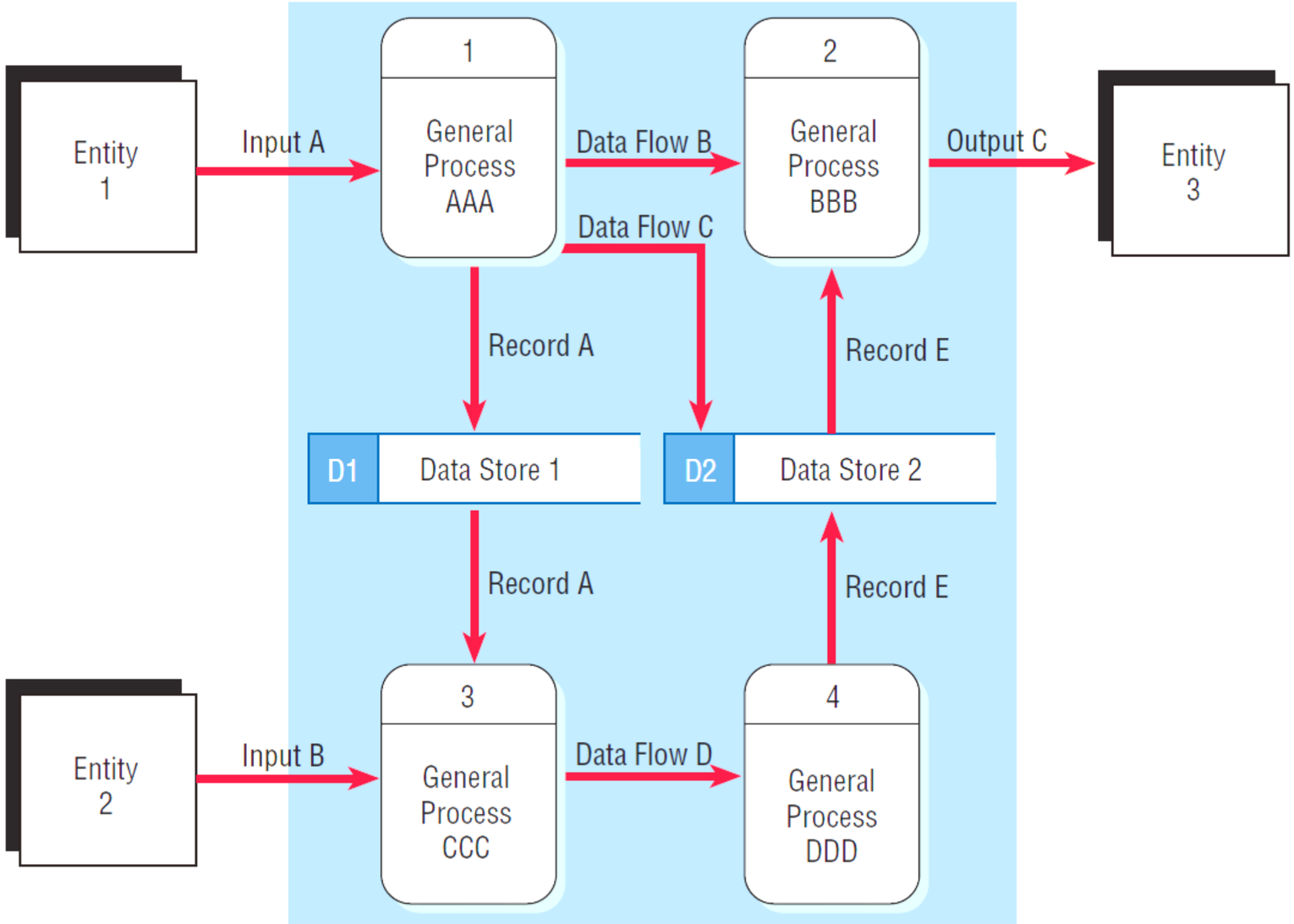
Figure:
Example of Level 0 diagram # 1



Creating the Level 0 Diagram

MP.ISAD.IT.KMITL

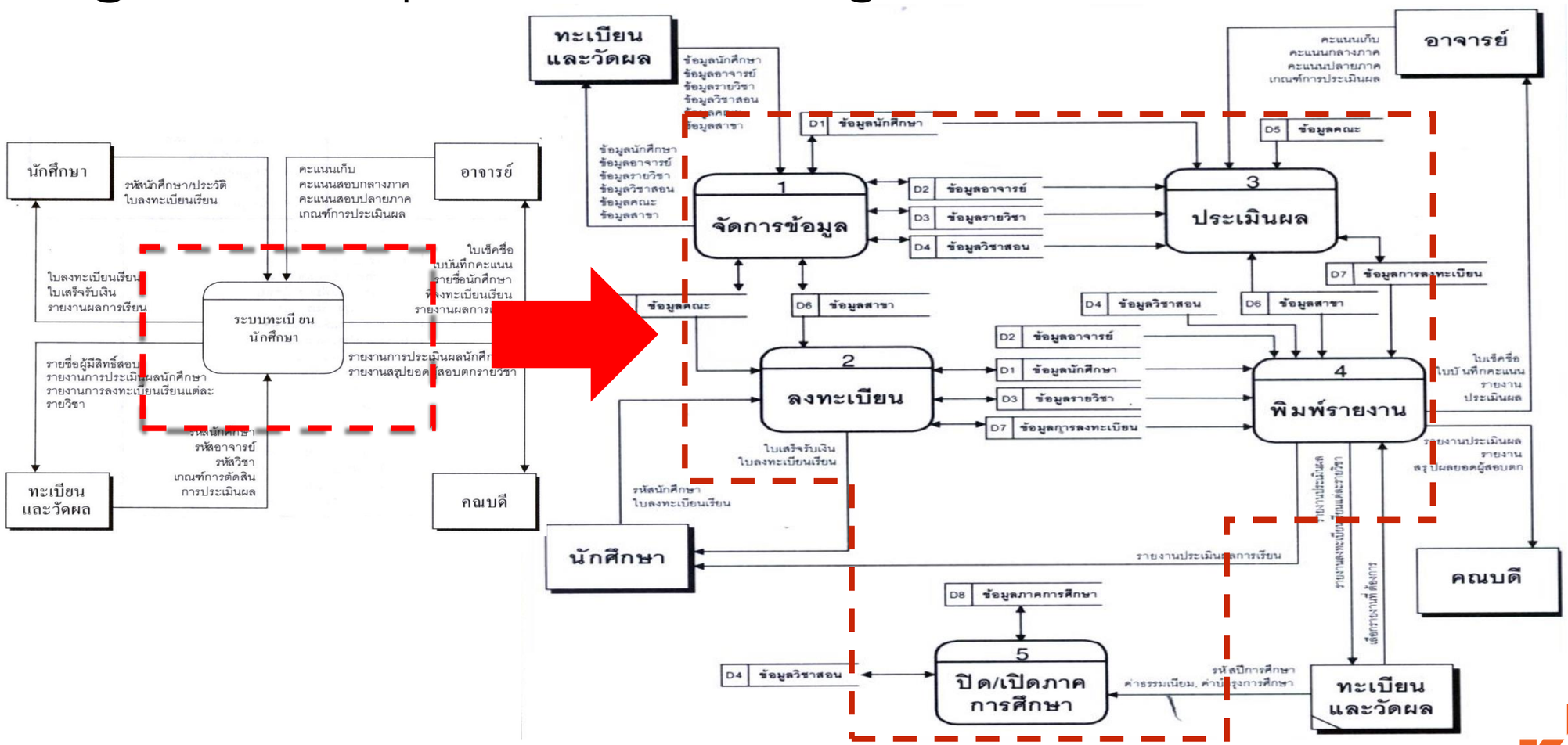
Figure: Example of Level 0 diagram # 2



Creating the Level 0 Diagram

MP.ISAD.IT.KMITL

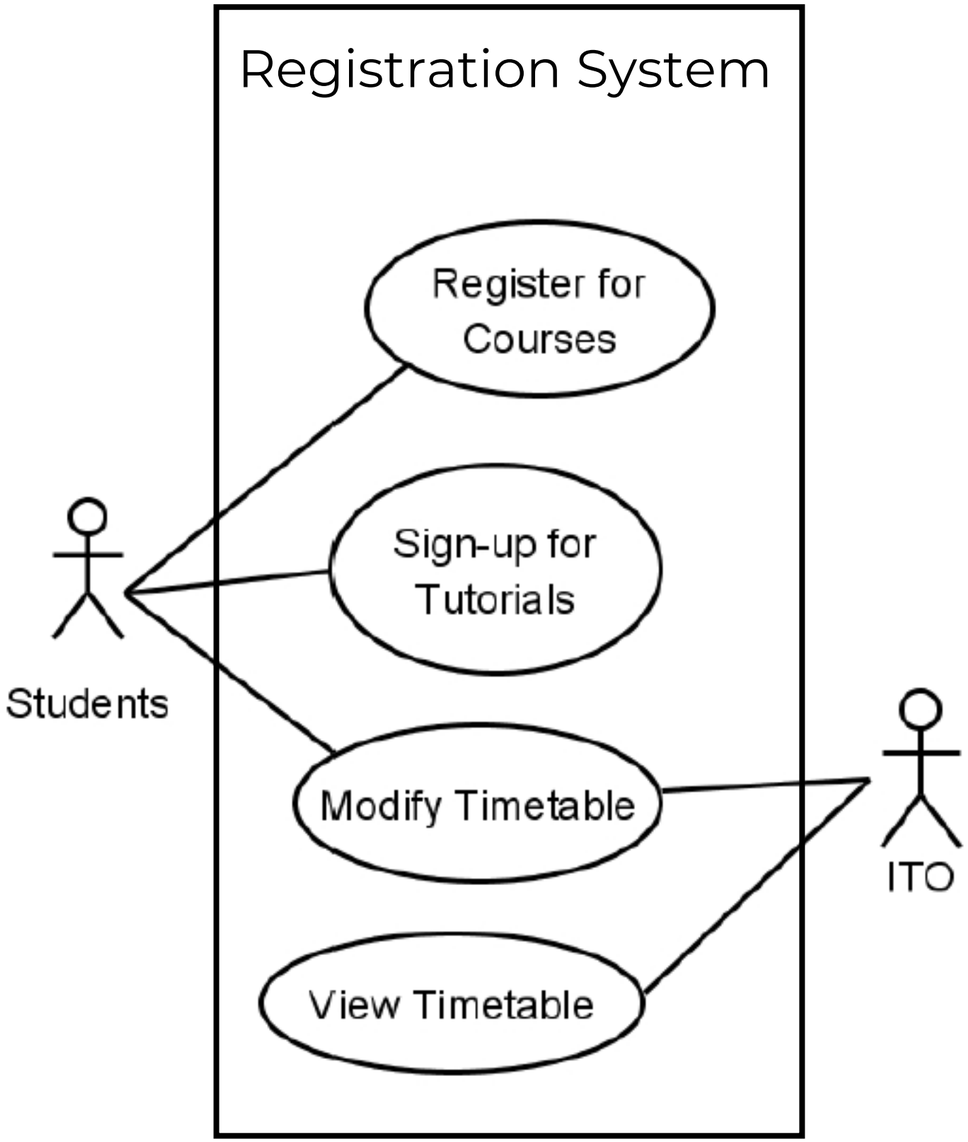
Figure: Example of Level 0 diagram #3



DFD Context Diagram

DFD Level 0 Diagram

Creating the Level 0 Diagram



Creating the Level 0 Diagram

Creating Level 1 diagrams and below

- **Level 1 DFD** – lower-level DFDs for each process in the level 0 DFD.
- Each one of the use cases is turned into its own DFD
- Each major step in the use case becomes a process on the level 1 DFD, with the inputs and outputs becoming the input and output data flows.
- Level 1 DFDs include the sources and destinations of data flows for data stores and data flows to processes.
- Including external entities in level 1 and lower DFDs can simplify the readability of DFDs.

KMITL

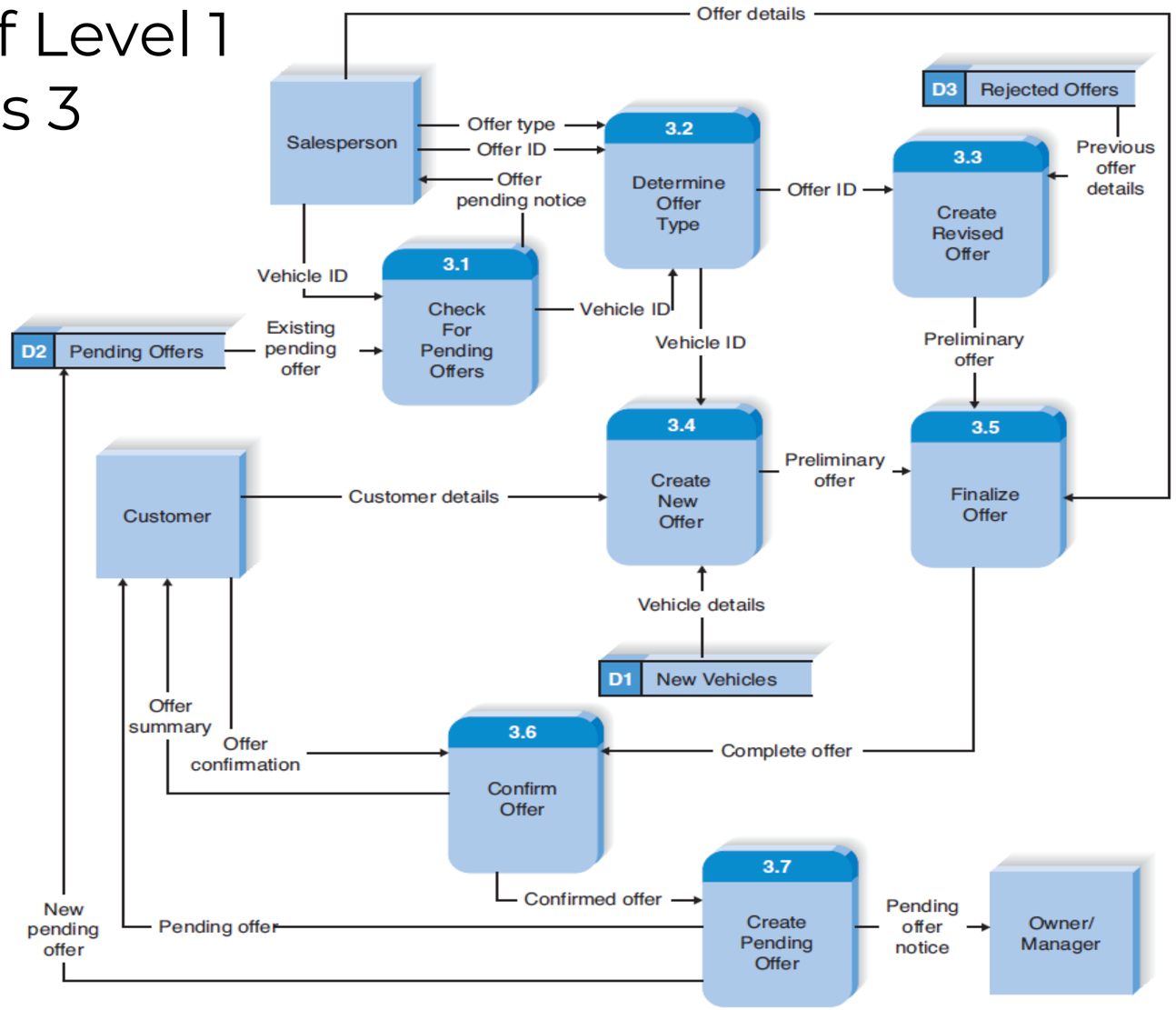
Creating Level 1 diagrams and below

- There is no simple answer to the “ideal” level of decomposition, because it depends on the complexity of the system or business process being modeled.
- In general, you decompose a process into a lower-level DFD whenever the process is sufficiently complex that additional decomposition can help explain the process.
- Rules of thumb:
  a) There should be at least 3, and no more than 7-9, processes on every DFD.
  b) Decompose until you can provide a detailed description of the process in no more than 1 page of process descriptions.

KMITL

Creating Level 1 diagrams and below

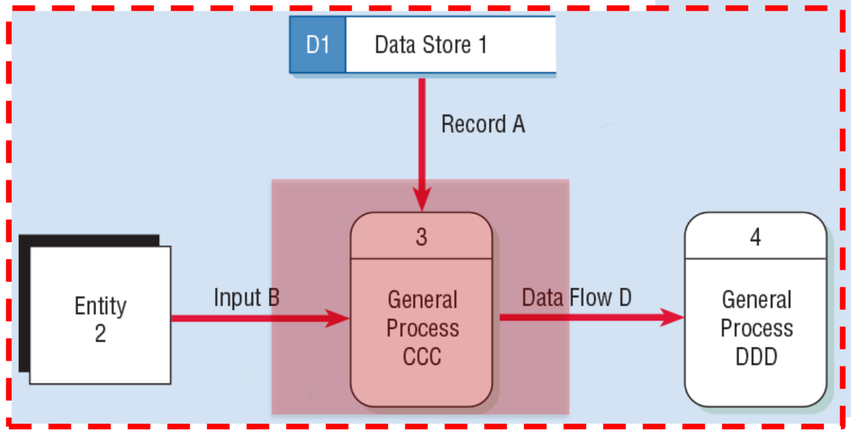
Figure : Example of Level 1 diagram for process 3

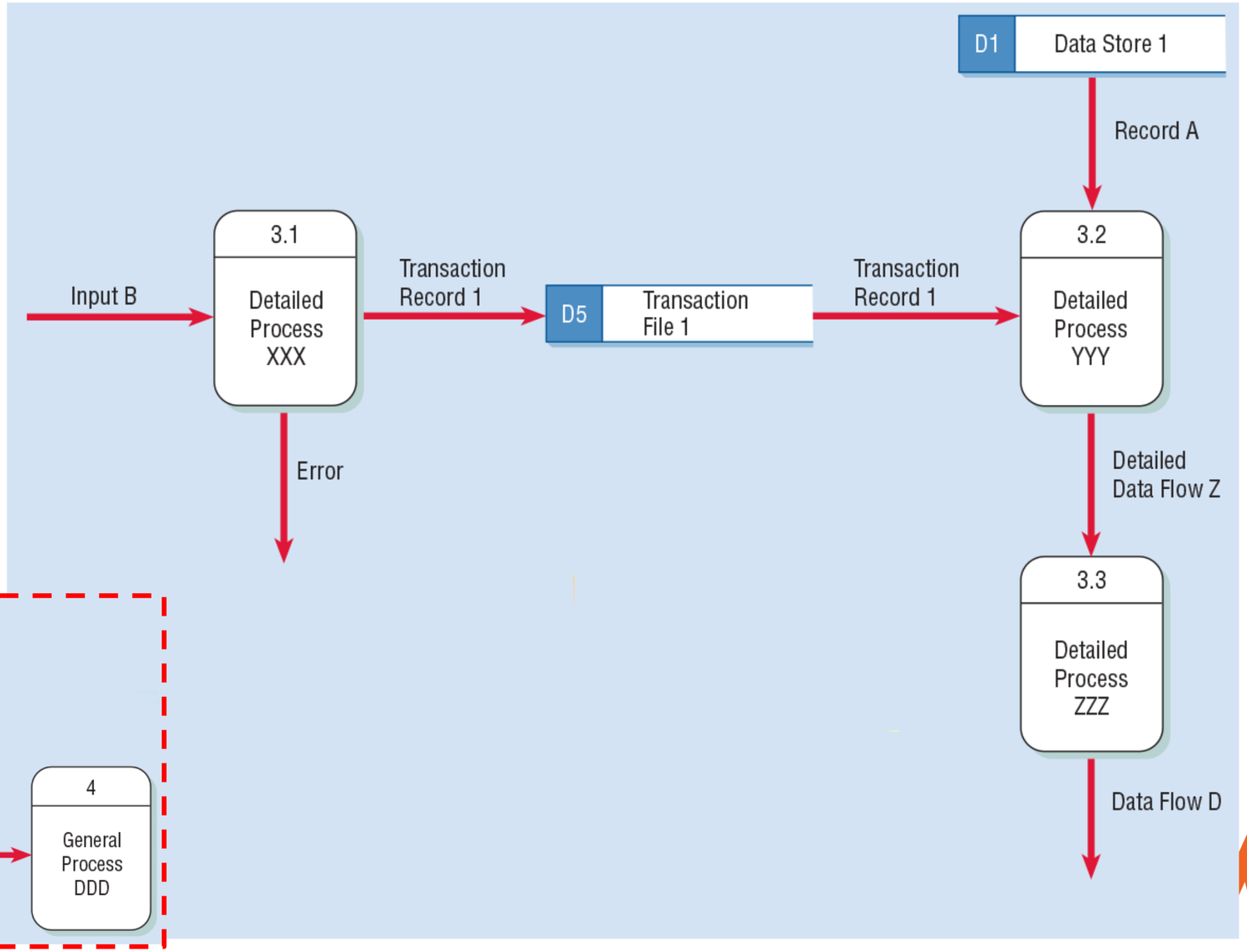


MP.ISAD.IT.KMITL

Creating Level 1 diagrams and below

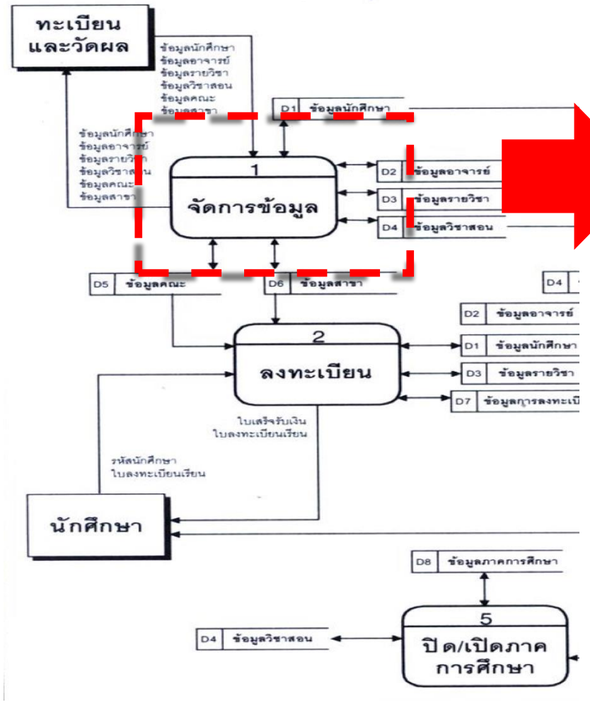
Figure : Example of Level 1 diagram #2 for process 3



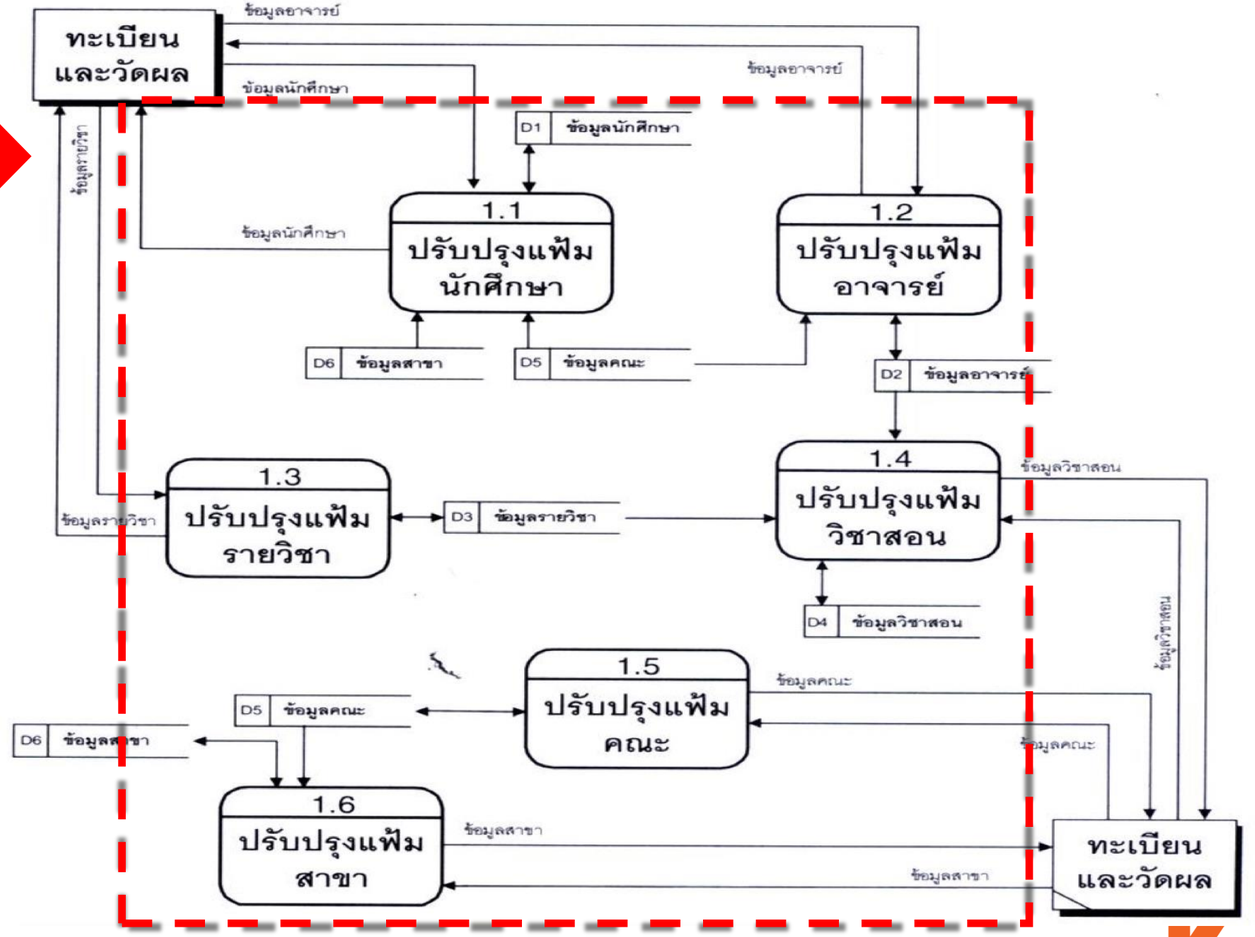


Creating Level 1 diagrams and below

Figure: Example of Level 1 diagram #3 for process 1



Level 0 Diagram



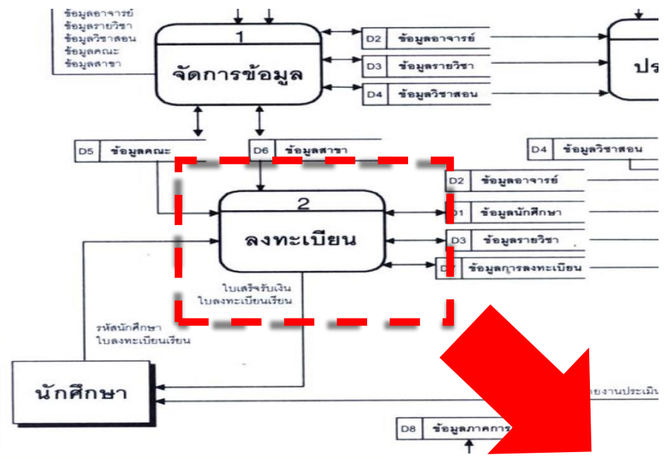
Level 1 Diagram

MP.ISAD.IT.KMITL

Creating Level 1 Diagrams and below

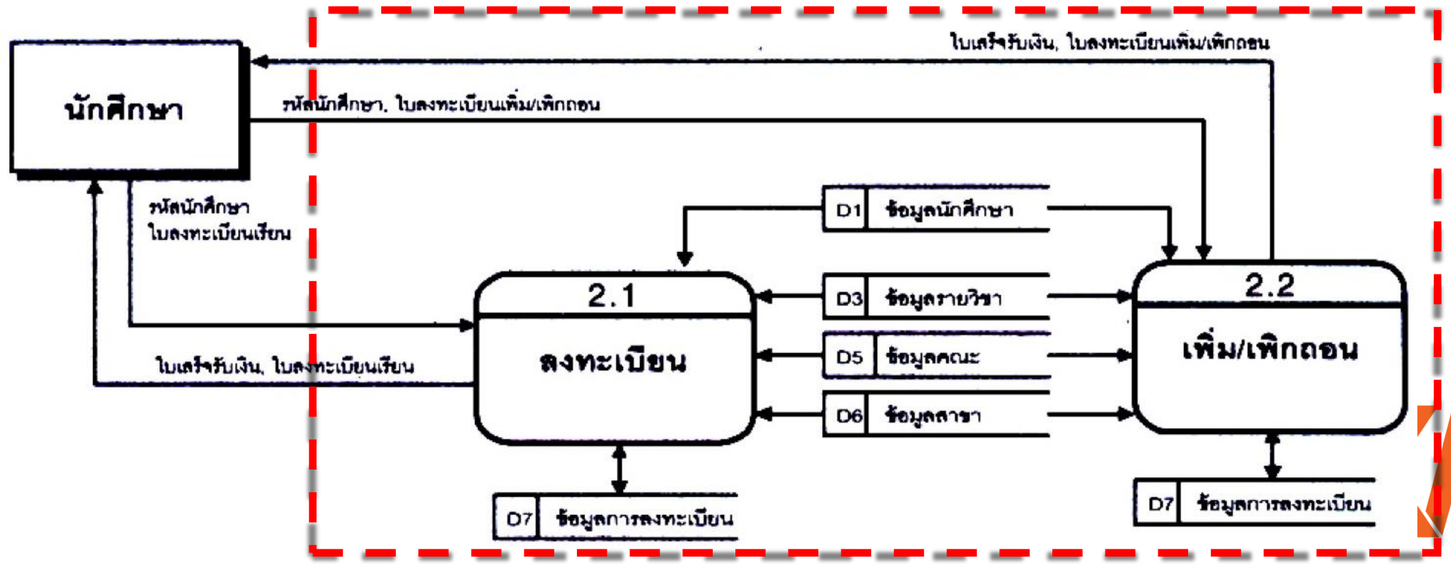
MP.ISAD.IT.KMITL

Figure: Example of Level 1 diagram #4 for process 2



Level 0 Diagram

Level 1 Diagram



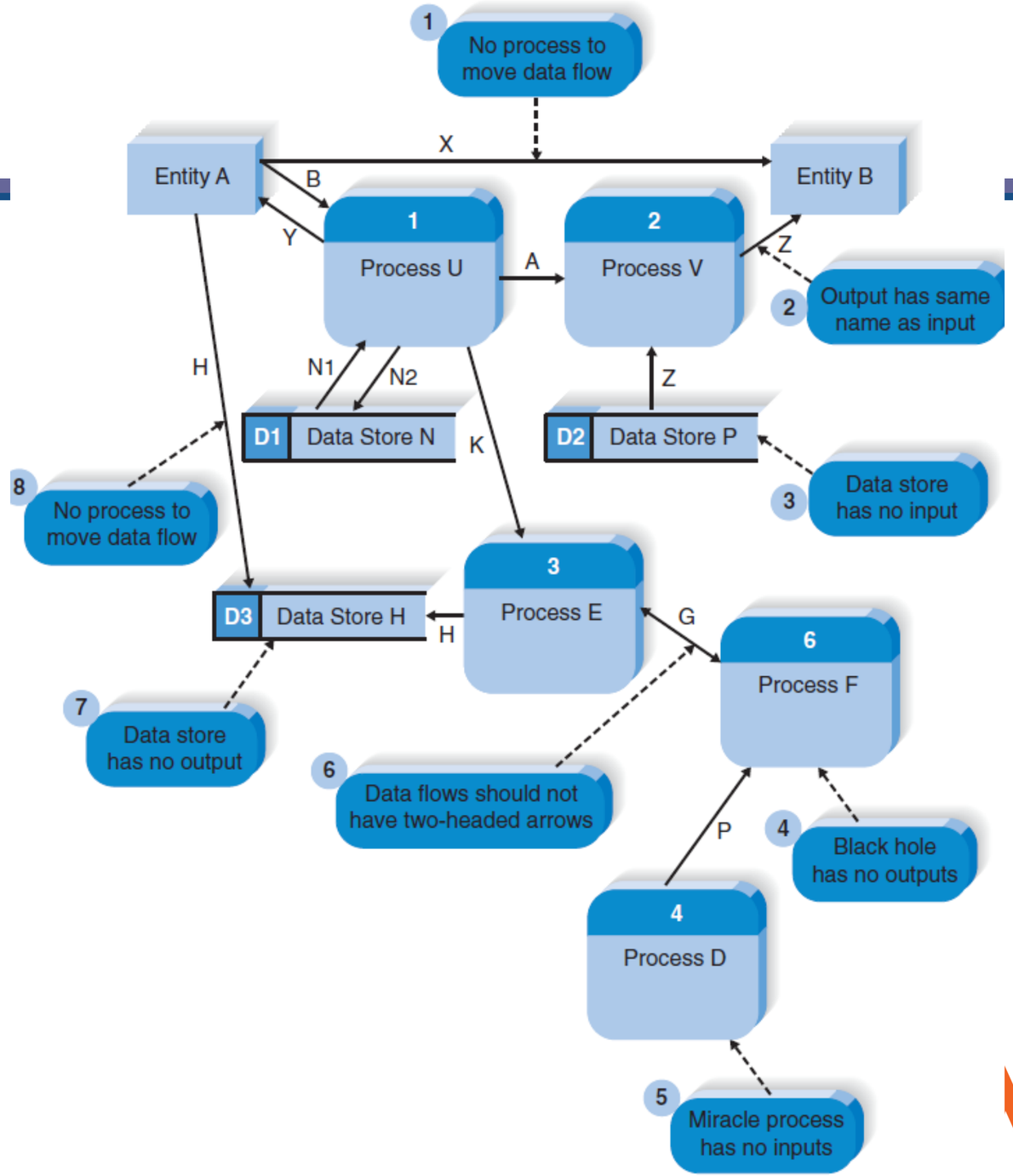
Validating the DFD

There two fundamental types of errors in DFDs:

1. **Syntax errors** – can be thought of as grammatical errors that violate the rules of the DFD language.
   - Syntax errors are easier to find and fix than are semantics errors because there are clear rules that can be used to identify them.
   - Most CASE tools have syntax checkers that will detect syntax errors.

Validating the DFD

Figure: Common Syntax Errors



MP.ISAD.IT.KMITL

Validating the DFD

There two fundamental types of errors in DFDs:

2. **Semantics errors** – can be thought of as misunderstandings by the analyst in collecting, analyzing, and reporting information about the system.
   - Semantics errors cause the most problems in system development.
   - Three useful checks to help ensure that models are semantically correct:

KMITL

Logical and Physical DFD

Data flow diagrams are categorized as either logical or physical.

- A **logical data flow diagram** focuses on the business and how the business operates. It is not concerned with how the system will be constructed.
- A **physical data flow diagram** shows how the system will be implemented, including the hardware, software, files, and people involved in the system.

KMITL

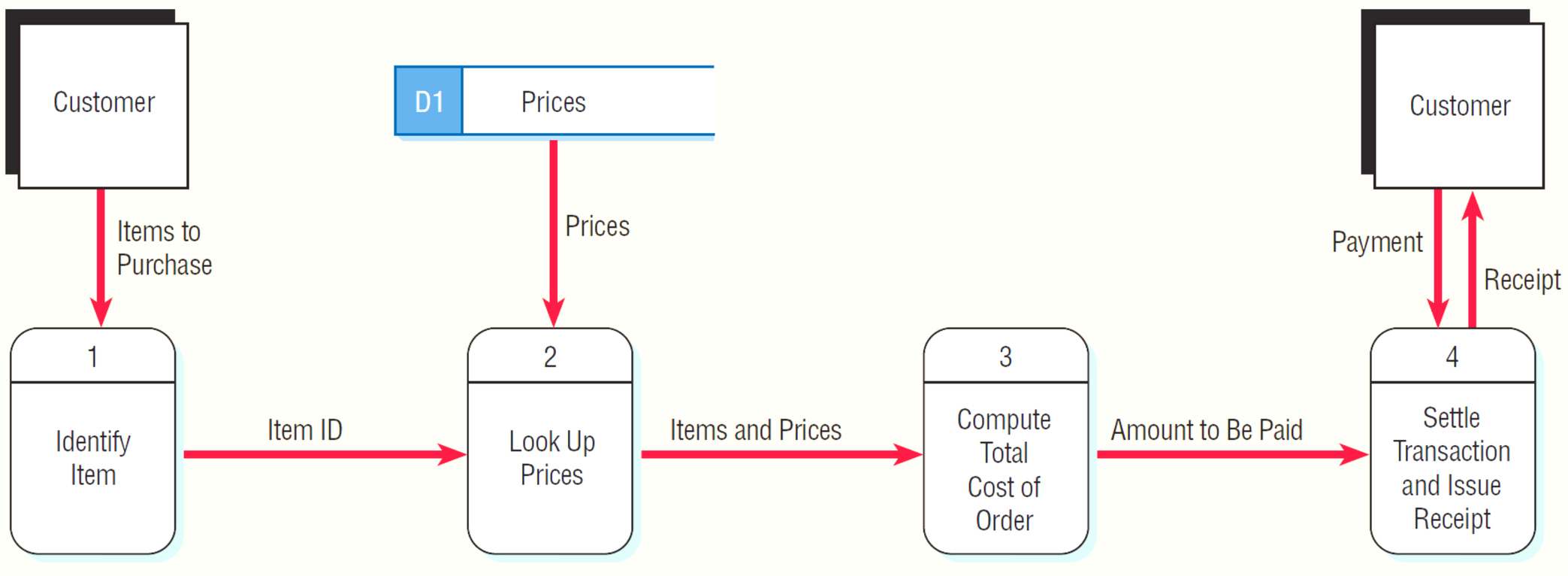
Logical and Physical DFD

Figure : Features common to both logical and physical data flow diagrams.

Design Feature	Logical	Physical
What the model depicts	How the business operates.	How the system will be implemented (or how the current system operates).
What the processes represent	Business activities.	Programs, program modules, and manual procedures.
What the data stores represent	Collections of data regardless of how the data are stored.	Physical files and databases, manual files.
Type of data stores	Show data stores representing permanent data collections.	Master files, transition files. Any processes that operate at two different times must be connected by a data store.
System controls	Show business controls.	Show controls for validating input data, for obtaining a record (record found status), for ensuring successful completion of a process, and for system security (example: journal records).

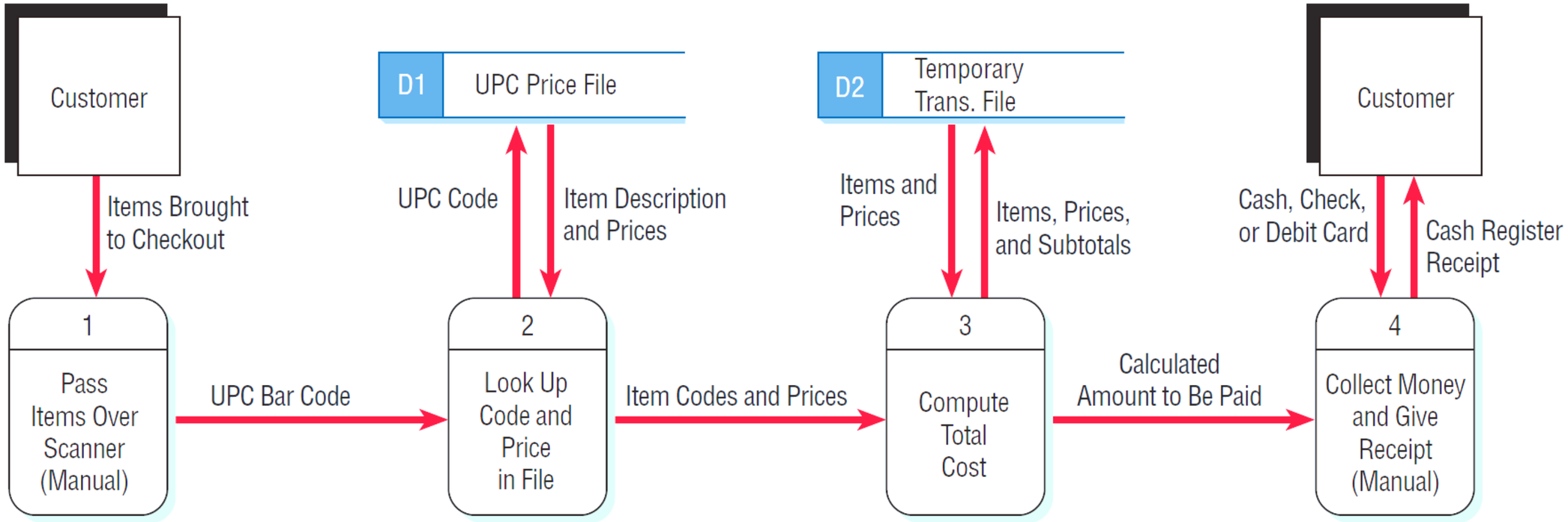
Logical DFD - example

Figure: The logical data flow diagram.



Physical DFD - example

Figure: The physical data flow diagram



SUMMARY

- Data Flow Diagram Syntax – four symbols are used on data flow diagrams (processes, data flows, data stores, and external entities).
- Creating Data Flow Diagrams
  a) The DFDs are created from use cases.
  b) Every set of DFDs starts with a context diagram.
  c) DFDs segments are created for each use case and are then organized into a level 0 DFD.
  d) Level 1 DFDs are developed on the basis of the steps within each use case.
  e) The set of DFDs are validated to make sure that they are complete and correct and contain no syntax or semantics errors.

KMITL

Assignment

- ให้เขียน Data Flow Diagram ของระบบโครงงานดังนี้

1. Context diagram
2. Level 0 diagram
3. Level 1 diagram ให้เขียนอย่างน้อย ทุก process ของ entity หลักของโครงงาน (ให้เลือก entity หลักเอง เช่น ลูกค้า)

ลำดับเอกสาร

1. ปกหน้า (หมายเลขกลุ่ม ชื่อโครงงาน สมาชิก)
2. สารบัญ
3. System Request
4. Work plan
5. Gantt Chart
6. Use Case diagram
7. Use Case Description
8. Activity diagram
9. Data Flow diagram
10. ภาคผนวก

- ระบบต้นแบบ
- เอกสารประกอบการเก็บ Requirements
- เอกสารประกอบอื่นๆ