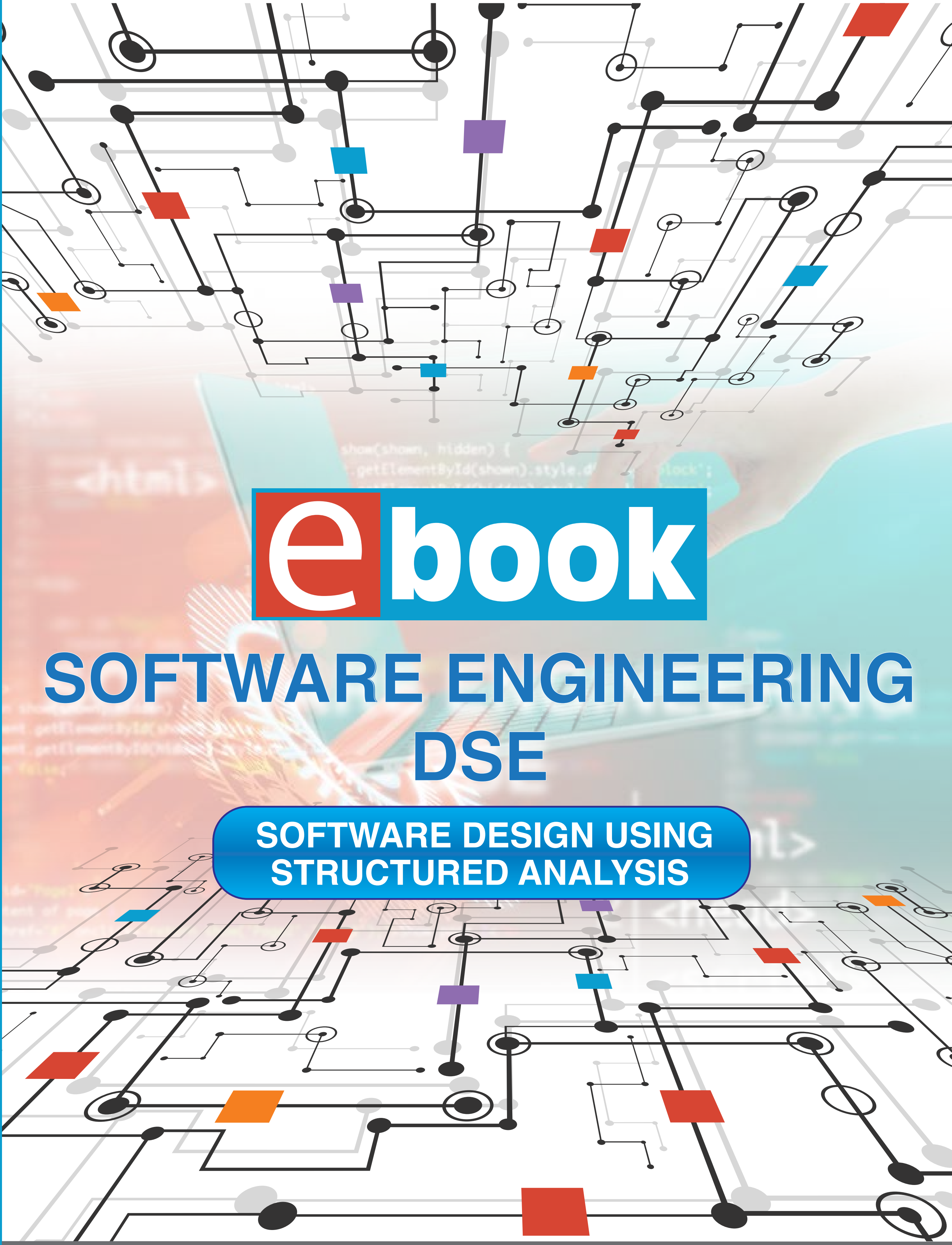
NIBM WORLD WIDE



ebook

SOFTWARE ENGINEERING DSE

SOFTWARE DESIGN USING STRUCTURED ANALYSIS

National Institute of Business Management

E-Book

Lesson 05 – Software Design Using Structured Analysis

Structured Analysis

Structured Analysis is a development method that allows the analyst to understand the system and its activities in a logical way. It is a systematic approach, which uses graphical tools that analyze and refine the objectives of an existing system and develop a new system specification which can be easily understandable by user.

It has following attributes;

- It is graphic which specifies the presentation of application.
- It divides the processes so that it gives a clear picture of system flow.
- It is logical rather than physical
- It is an approach that works from high-level overviews to lower-level details.

Structured Analysis Tools:

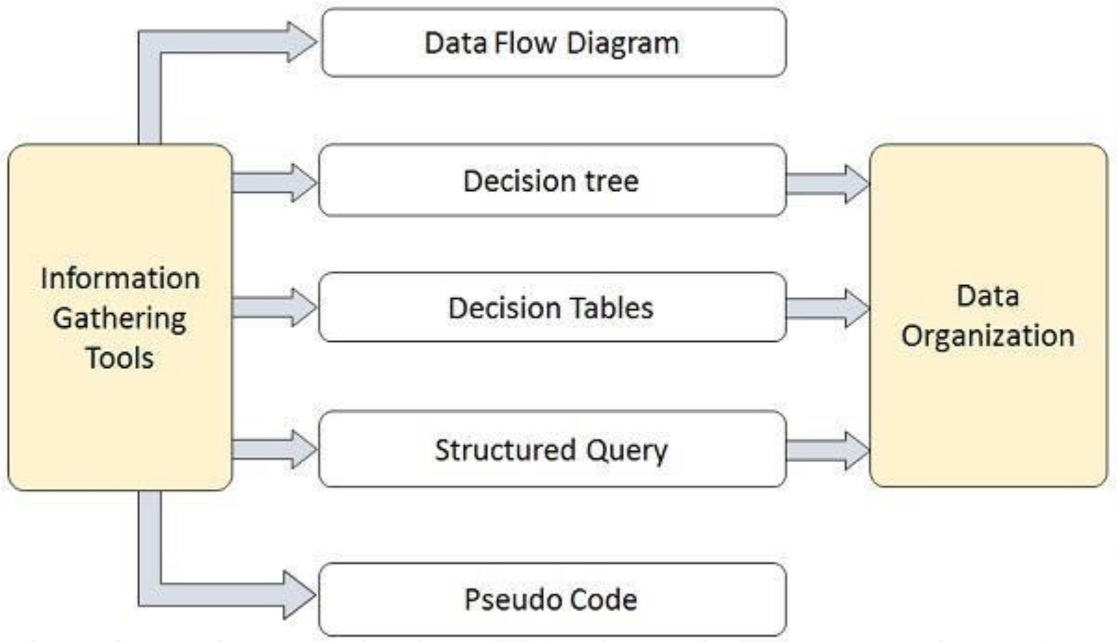


Figure 5.0.1 Structured Analysis

Basically the approach of Structured Analysis is based on the Data Flow Diagram.

Data Flow Diagram (DFD)

- DFD graphically representing the functions, or processes, which capture, manipulate, store, and distribute data between a system and its environment and between components of a system.

- Structure of DFD allows starting from a broad overview and expand it to a hierarchy of detailed diagrams.

Data Flow Diagram Levels

- DFD s are traditionally drawn at three levels.

- The top level DFD is **context diagram** consisting of only one process, representing the entire system.

- Immediately beneath the context diagram is the system level diagram (Level 0). This is a highest level view of the major functions within the system. Further levels can be decomposed to any depth by expanding individual processes

- DFD Levels,
  - Context Diagram
  - Level 0 DFD
  - Level 1 DFD
  - Level 2 DFD

Data Flow Diagram Elements

1. External Entity
2. Processes
3. Data Store
4. Data Flows

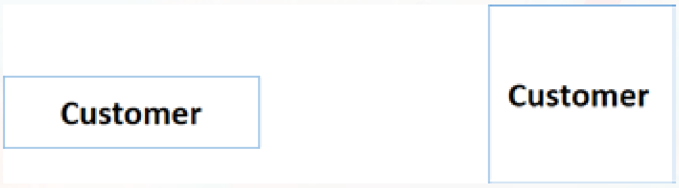
1. External Entity

- An external entity may actually be outside of the business (Such as government agents, Customers, Suppliers or Bank)

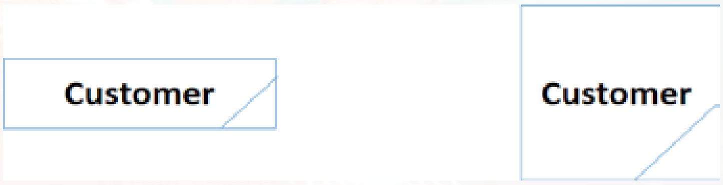
OR

- It may be inside the business but outside of the system scope (Other Departments, Other internal information systems)
- External entities that supply data into the system are called **Sources**.
- External entities that use system data are called **Sinks or Destination**.

Symbol:



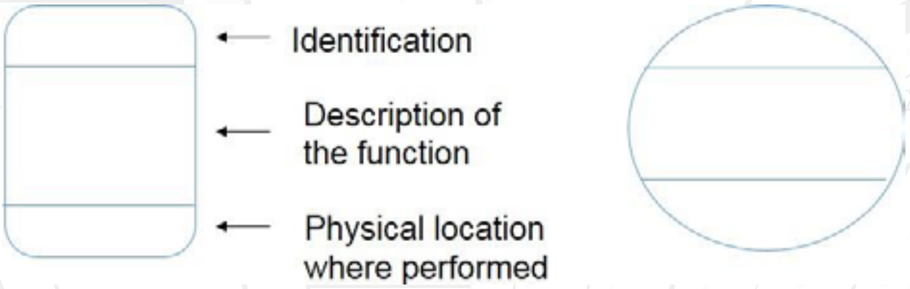
To avoid crossing data flow lines the same entity can be drawn more than once on the same diagram.



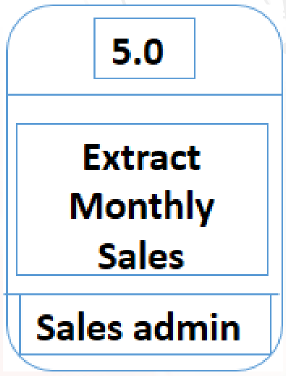
2. Processes

- Any process that changes the data, producing an output. It might perform computations, or sort data based on logic, or direct the data flow based on business rules.

Symbol:


Identification
Description of the function
Physical location where performed

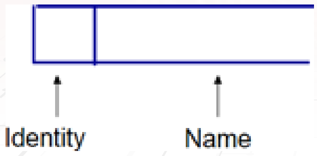
Example:



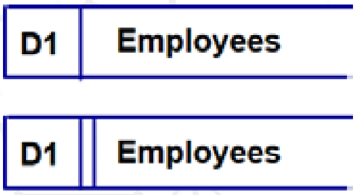
3. Data Store / File

- Files or repositories that hold information for later use, such as a database table or a membership form. Process can enter data in to a data store or retrieve data from a data store. Each data store has a unique name and identity.

Symbol:



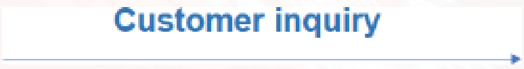
Example:



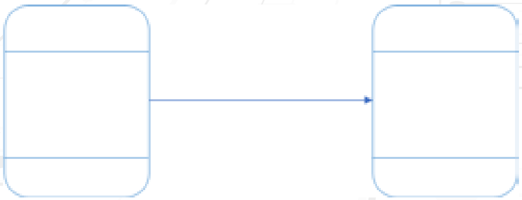
4. Data Flow

- The route that data takes between the external entities, processes and data stores. The direction of the flow is indicated by an arrow and the line is labeled by the name of the data flow.
- Flow of data in system can take place,
    - Between two processes
    - From a data store to a process
    - From a process to data store
    - From source to a process
    - From process to a sink

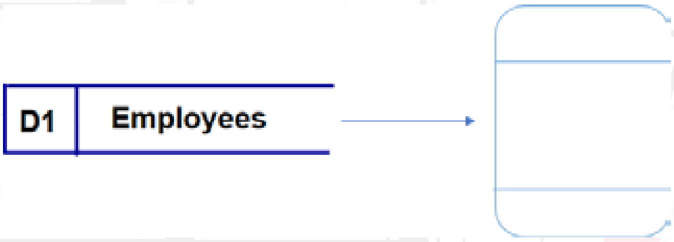
Symbol:



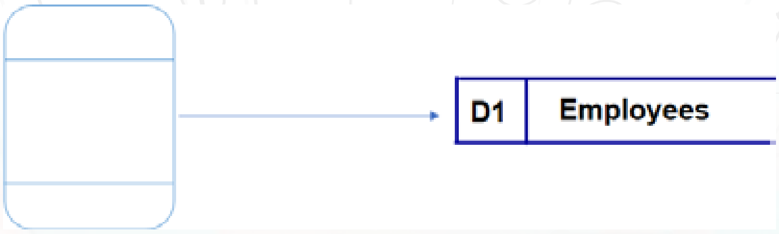
- **Between two processes;**



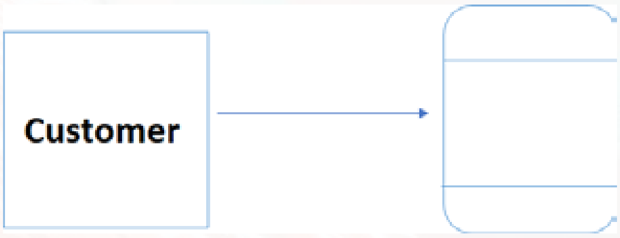
- **From a data store to a process;**



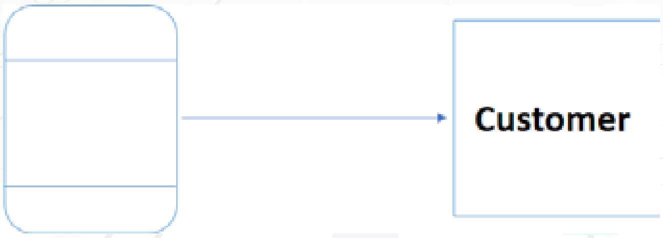
- **From a process to data store;**



- **From source to a process;**



- **From process to a sink;**



Data Flow Diagram RULES

Process

- No process can have only outputs (a miracle)
- No process can have only inputs (black hole)
- A process has a verb phrase label (E.g.: Receive Customer Order)

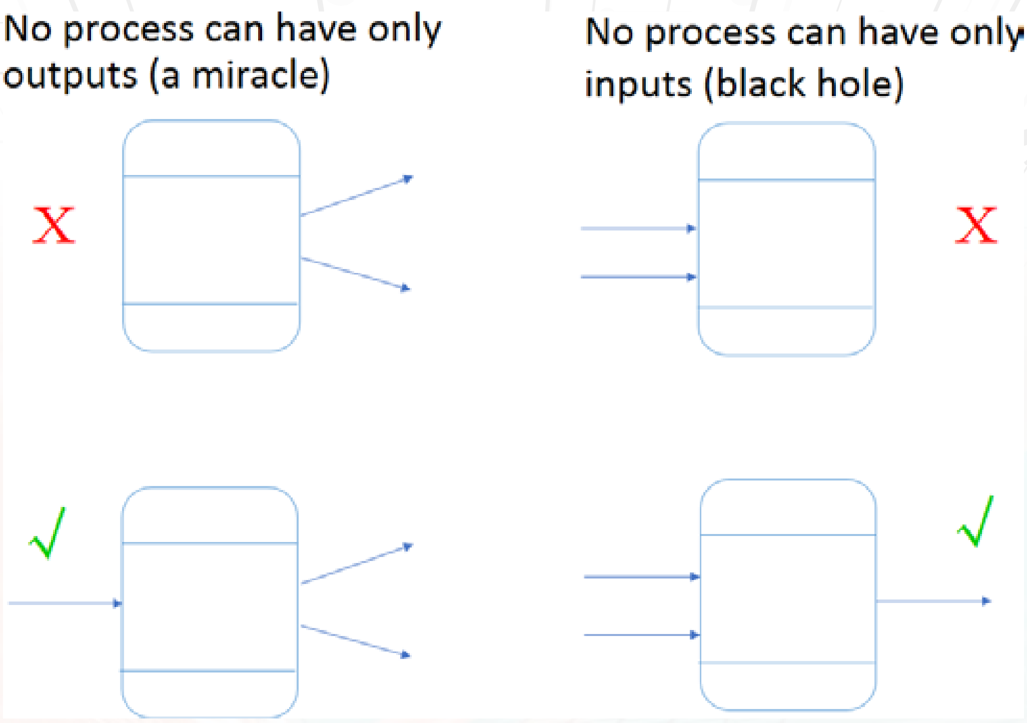

No process can have only outputs (a miracle)
No process can have only inputs (black hole)
X
X
✓
✓

Figure 5.0.2 Data Flow Diagram RULES - Process

External Entity

- Data cannot move directly from one entity to another.
- Data cannot move directly from an entity to data store.
- An entity has a noun phrase label. (E.g.: Customer)
- An entity can be another department, information system, or an organization, that sends or receives data from the system.

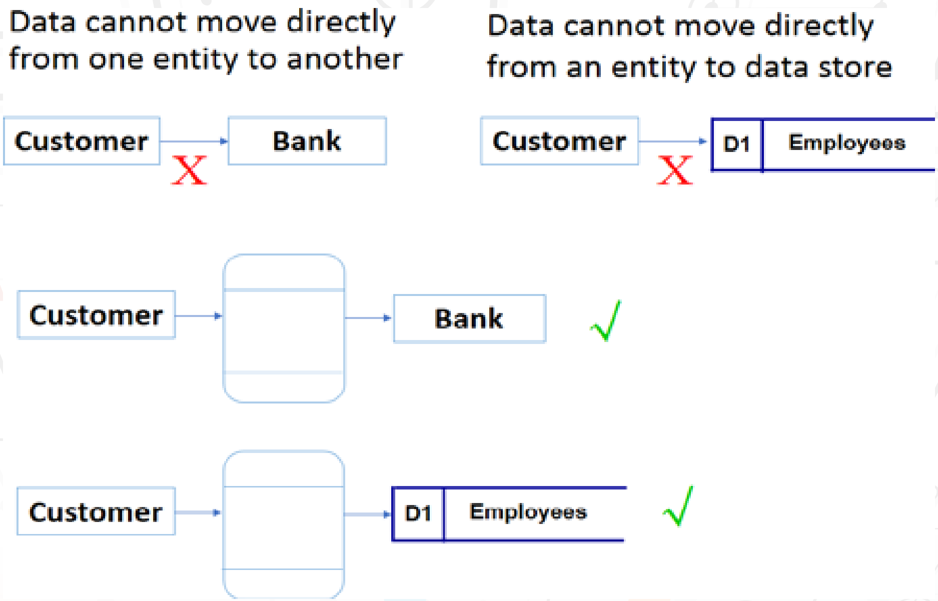

Data cannot move directly from one entity to another
Customer
Bank
X
Data cannot move directly from an entity to data store
Customer
D1
Employees
X
Customer
Bank
✓
Customer
D1
Employees
✓

Figure 5.0.3 Data Flow Diagram RULES – External entity

Data Store

- Data cannot be moved directly from one store to another.
- Data cannot move directly from an outside source to a data store.
- Data cannot move directly from data store to a data sink.
- In order to keep the diagram uncluttered, you can repeat data stores on a diagram.
- Data store has a noun phrase label. (Eg: Employee)

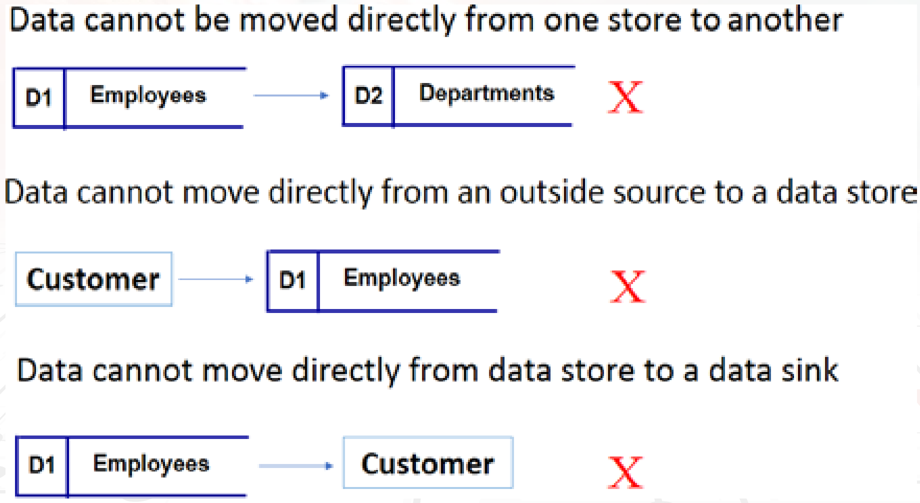

Data cannot be moved directly from one store to another
D1
Employees
D2
Departments
X
Data cannot move directly from an outside source to a data store
Customer
D1
Employees
X
Data cannot move directly from data store to a data sink
D1
Employees
Customer
X

Figure 5.0.4 Data Flow Diagram RULES – Data store – False

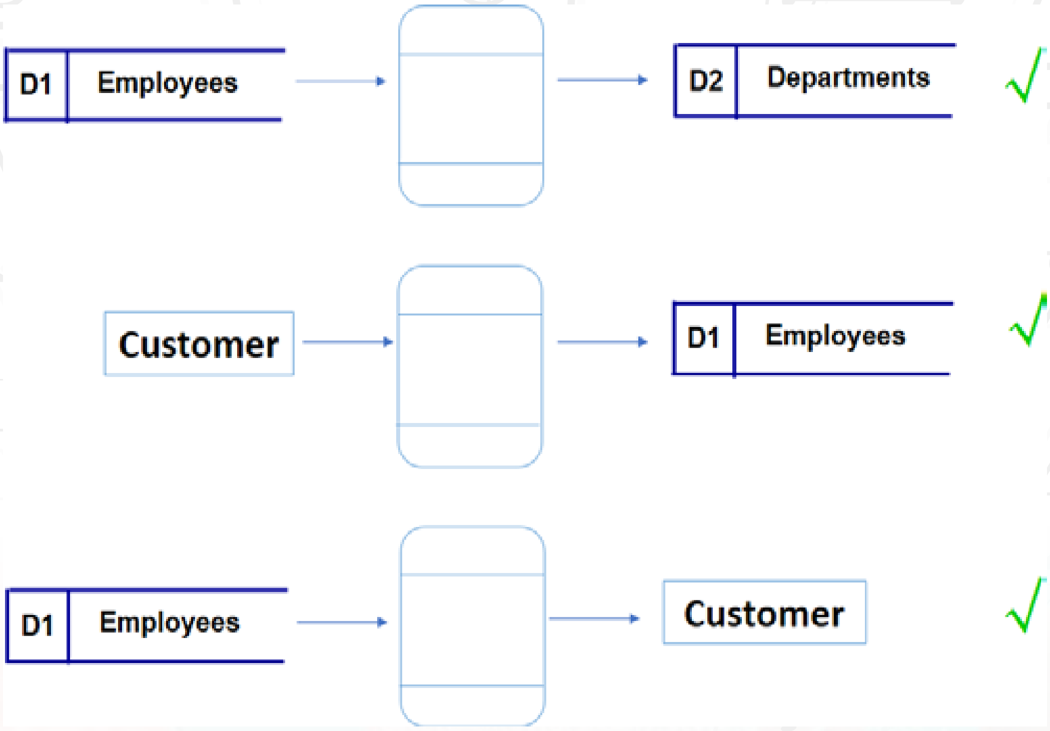


Figure 5.0.5 Data Flow Diagram RULES – Data store – True

- Data flow has only one direction of flow between symbols.
- A fork means that exactly the same data goes from a common location to two or more processes, data stores or sources/sinks.
- A join means that same data comes from any two or more different processes, data stores or entities to a common location.
- A data flow cannot go directly back to the same process it leaves.
- A data flow to a data store means update.
- A data flow from a data store means retrieve or use.
- A data flow has a noun phrase label (Eg: Customer Order)

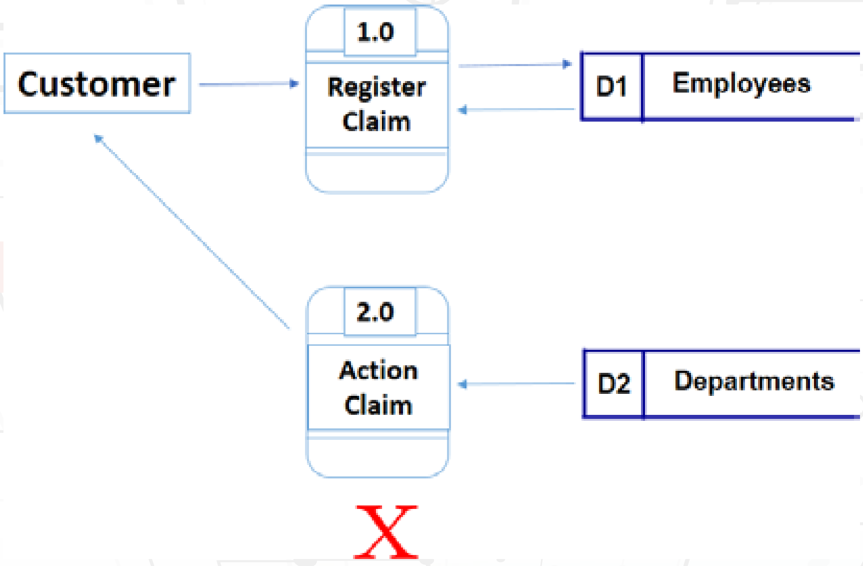


Figure 5.0.6 Data Flow Diagram RULES – Data store – Example

Process 2 must link to Process 1 some way. Either directly with data flow coming from Process 1 to Process 2 or through one data flow going into a data store from Process 1 and another data flow coming out of the data store into Process 2.

Context Diagram

A data flow diagram (DFD) of the scope of an organizational system that shows the system boundaries, external entities that interact with the system and the major information flows between the entities and the system.

- DFD with a highest level of abstraction.
- Represents the whole system as one process (black box).
- Has External Entities.
- Has Data Flows.
- No Data Stores.
- The entire software system is shown as a single process.
- Shows how system interacts with its external entities.

Level 0 Diagram

- Expand the context diagram to show the breakdown of the system.
- Level 0 DFD also mentions basic processes and sources of information.
- It provides a more detailed view of the Context Level Diagram.
- Here, the main functions carried out by the system are highlighted as we break into its sub-processes.

Level 1 Diagram

- Expand the Level 0 diagram to show the breakdown of major processes.
- Level 1 DFD there are a number of data stores, and data-flows between processes and the data stores.
- In Level 1 each process in Level 0 expand to more detail.

DFD Example 1 - Food Ordering System

Hoosier Burger is a restaurant that uses an information system that takes

- Customer orders,
- Send them to the kitchen,
- Bill the customers,
- Monitors goods sold,
- Inventory,
- Generates daily reports (Stock report and Sales Report) to the management.

Context Diagram for Hoosier Burger Food Ordering System

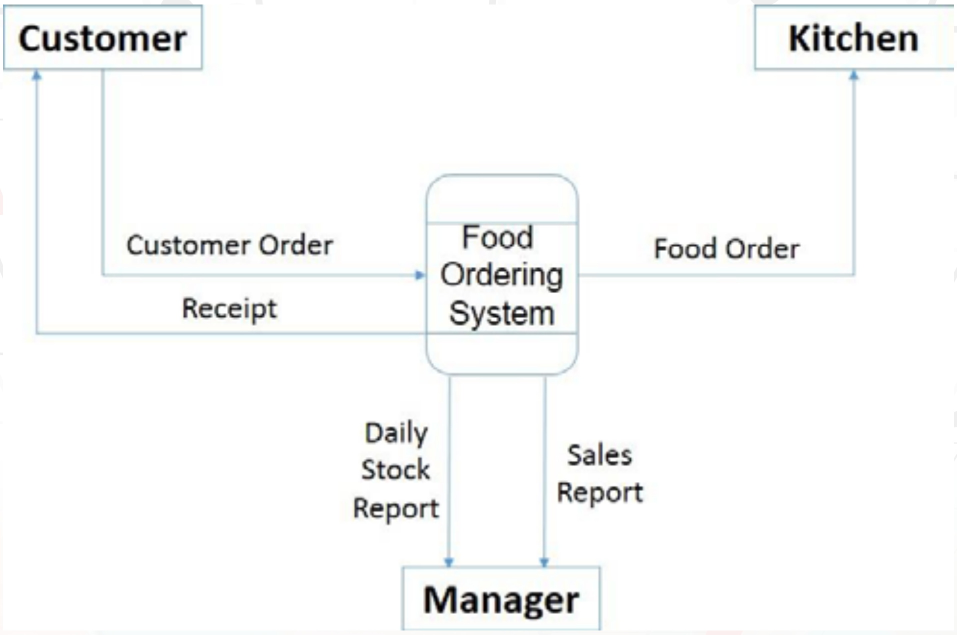


Figure 5.0.7 Context Diagram - Hoosier Burger Food Ordering System

Level 0 for Hoosier Burger Food Ordering System

Expand the context diagram to show the breakdown of major processes.

- **Identify the main possesses of the system**
  - Receive Customer orders
  - Update Inventory file
  - Update Goods sold file
  - Generate Reports
- **Identify the data stores**
  - Generate Reports
  - Inventory file
  - Sold Goods File
  - Customer Orders File
- **Identify the data flows between elements**

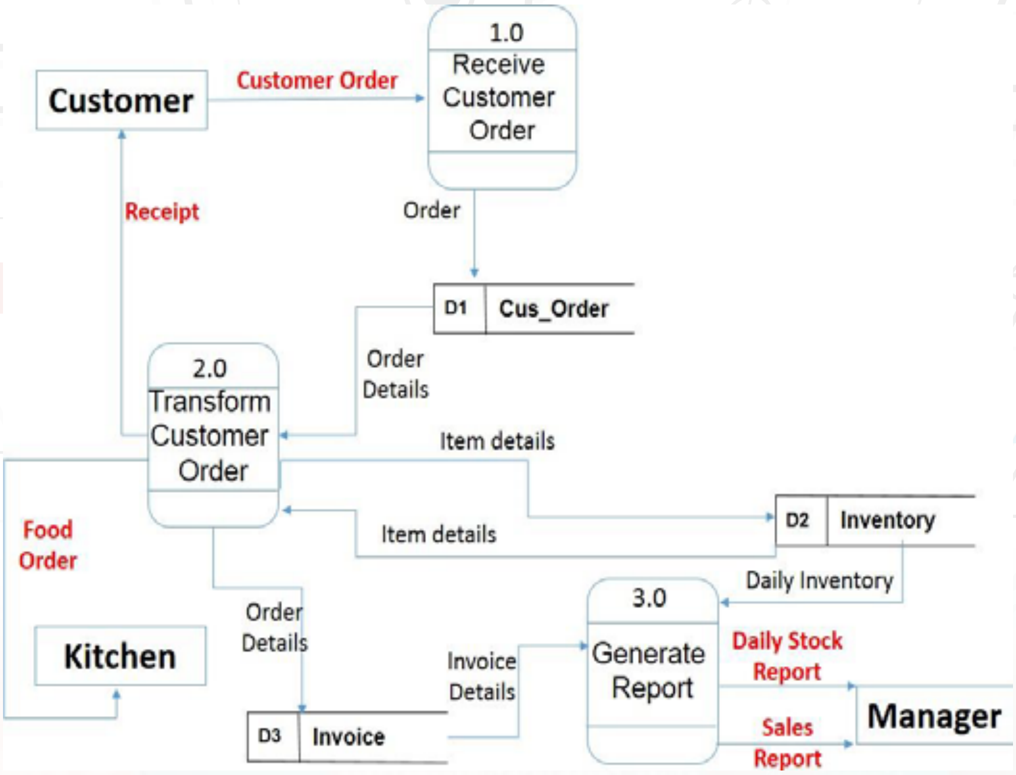


Figure 5.0.8 Level 0 for Hoosier Burger Food Ordering System

In detail way of Level 0 for Hoosier Burger Food Ordering System

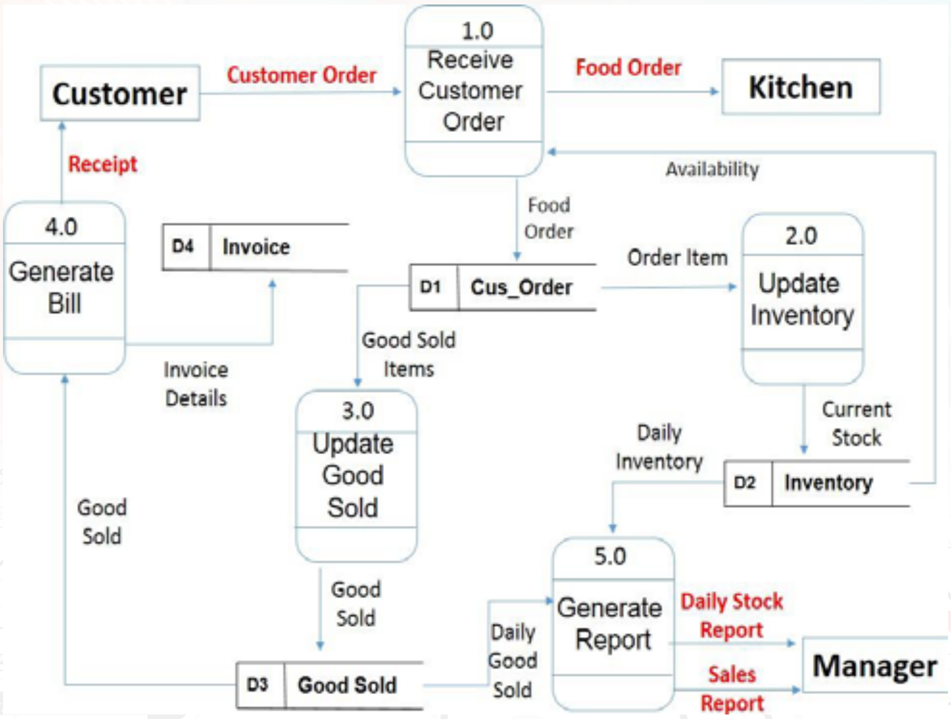


Figure 5.0.9 In detail way of Level 0 for Hoosier Burger Food Ordering System

Level 1 for Hoosier Burger Food Ordering System

Here Generate Reports (5.0) major process we can further sub divide into 2 processes as follow.

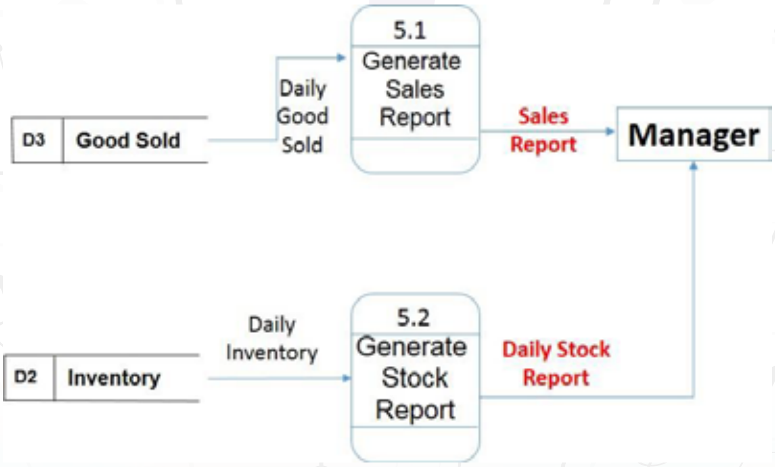


Figure 5.0.10 Level 1 for Hoosier Burger Food Ordering System

Remember you must balance the Context Diagram, Level 0 and Level 1. Also remember you can create processes and data stores depend on the requirements. Above diagrams are not the only corrected diagrams.

As an example in Context Diagram Customer Entity has one input and one output. In Level 0 Customer Entity has only one input and one output. You cannot add new data flow into the Level 0 if the data flow is not available in the Context Diagram.

In the Level 0 Diagram Cus_Order (D1) Data store has only two inputs and two outputs to the Receive Customer Order (1.0) process. Also in the Level 1 we must maintain the number of inputs and number of outputs of Cus_Order (D1) Data store.

DFD Example 2 - Video Ordering System

- A Customer can order a video tape.
- The Video shop will deliver the Video with a delivery note and Invoice.
- System also can send purchase order to the Supplier if the Item is not available.
- Supplier will deliver the goods with a delivery note.

- Also System need to save the Supplier details as well.

Context Diagram for Video Shop System

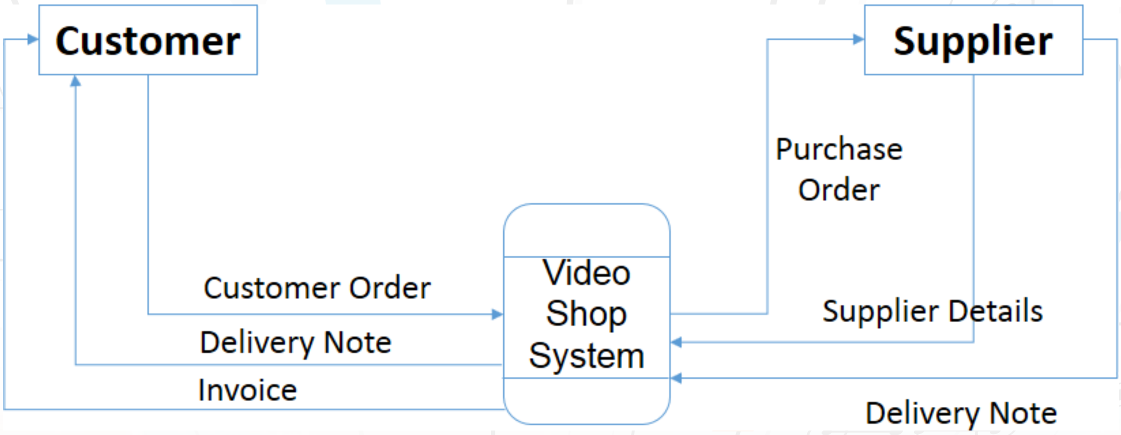


Figure 5.0.11 Context Diagram for Video Shop System

Level 0 for Video Shop System

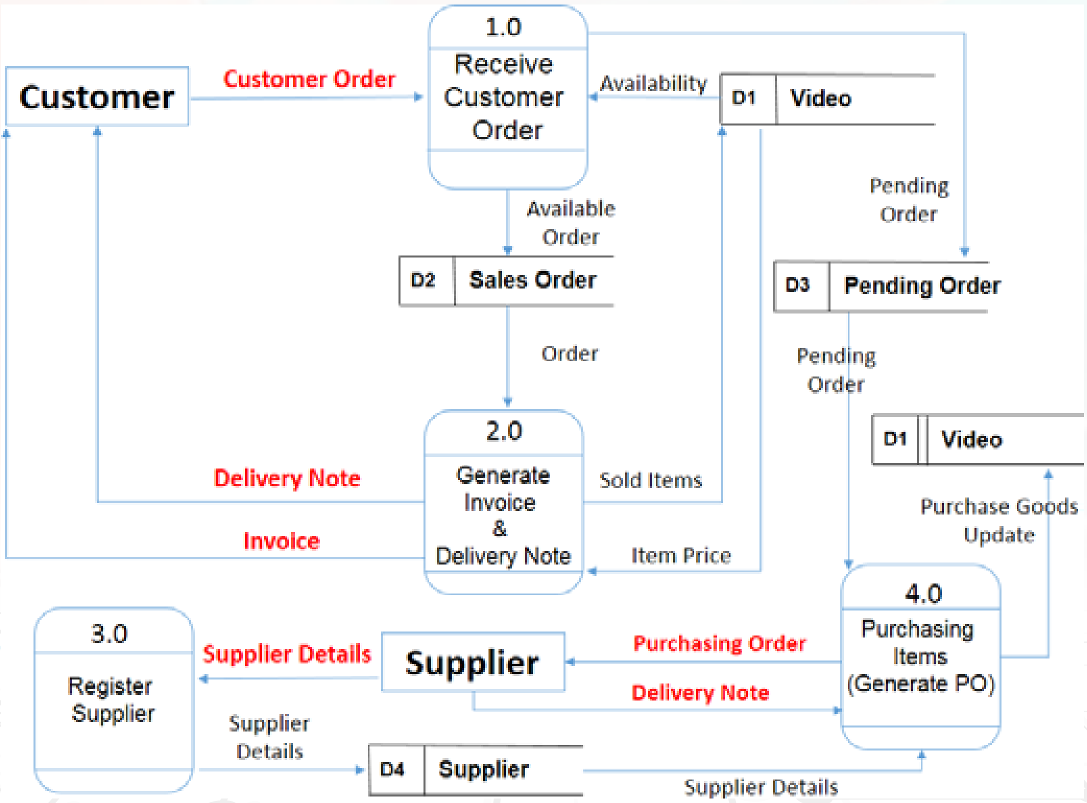


Figure 5.0.12 Level 0 for Video Shop System

Level 1 for Generate Purchase Order Process

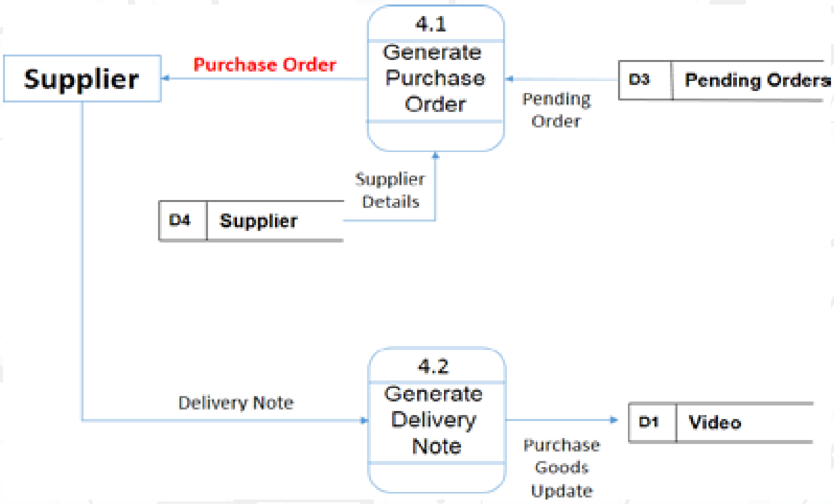


Figure 5.0.13 Level 1 for Generate Purchase Order Process

Please remember balancing the DFD is most important rule in DFD.

You should always check the Data Flow (Input, Output) in Context Diagram and Level 0 Diagram. Then Level 0 and Level 1 Diagram.

DFD Example 3 - Student Management System

- The Staff design courses and provide the course details.
- The Students can enroll to the course through an enrollment request.
- Before starting each course registered students are provided with a course schedule.
- On the starting date of the course, registered students list should be sent to the course director.
- Unavailability of a requested course should also inform to the student.

DFD Example 4 Order Processing System

- Customer can register with the company by sending his/her details to Order Processing system.
- Customer send orders to the Company. And the company sends the product availability. The system accepts orders only if the sufficient inventory exists.
- If the product is available Invoice is generated and send it to the Customer. At the same time Inventory files and Goods Sold files should be updated.
- If the product is not available those orders should save in pending orders File. Then those orders with the item details should send to supplier through a purchase order. Before send the orders to the Supplier, Suppliers should register in the system.
- Once the supplier delivers the product purchasing department get the notification from the system and update the inventory file. At the same time Supplier sends the Delivery notice and Purchasing Department sends the Goods Receive Note (GRN) to the Supplier.
- Each Purchase order and Delivery Note of the Supplier need to save.
- If the products are damaged customer will return the products. Also if the products are damaged company will return the products back to the supplier. Those damage items can be save in Damage Item File.
- Monthly sales report and monthly stock report needs to be send to the Manager.