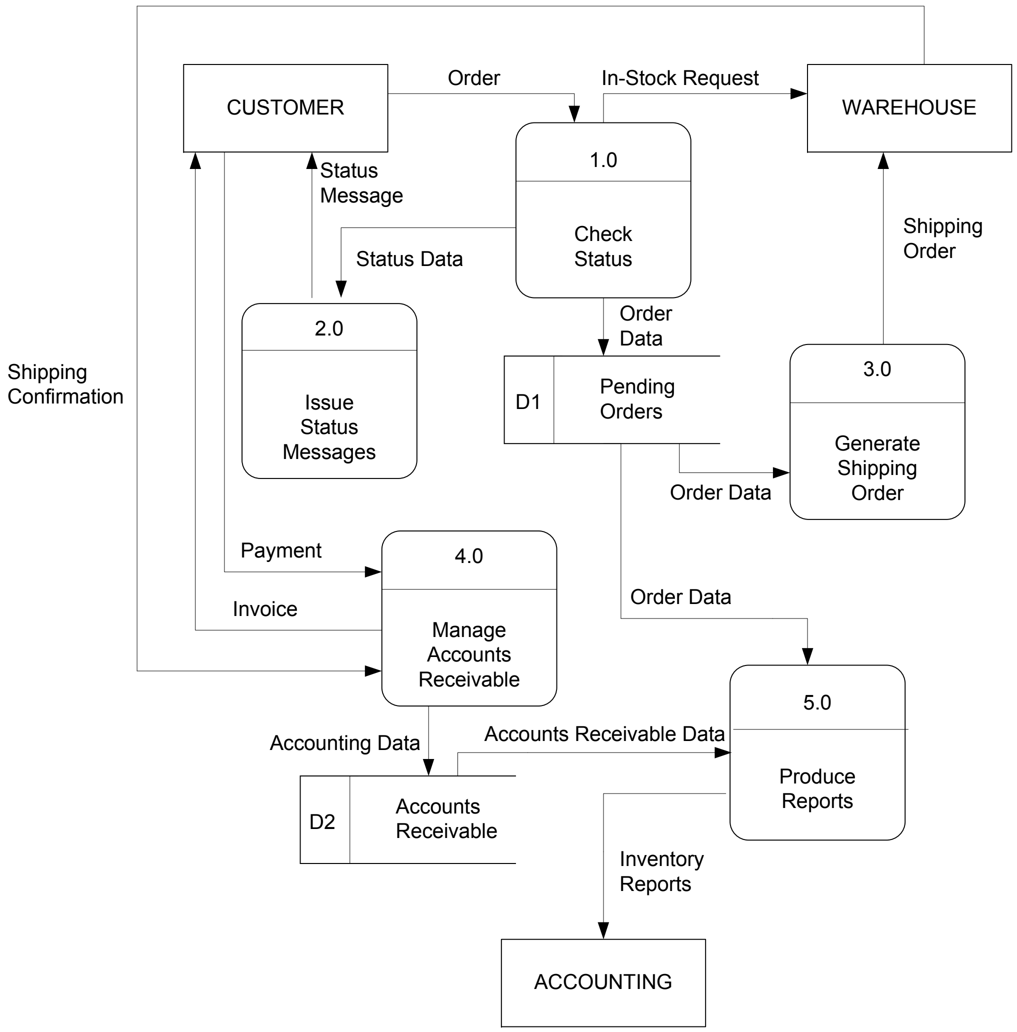
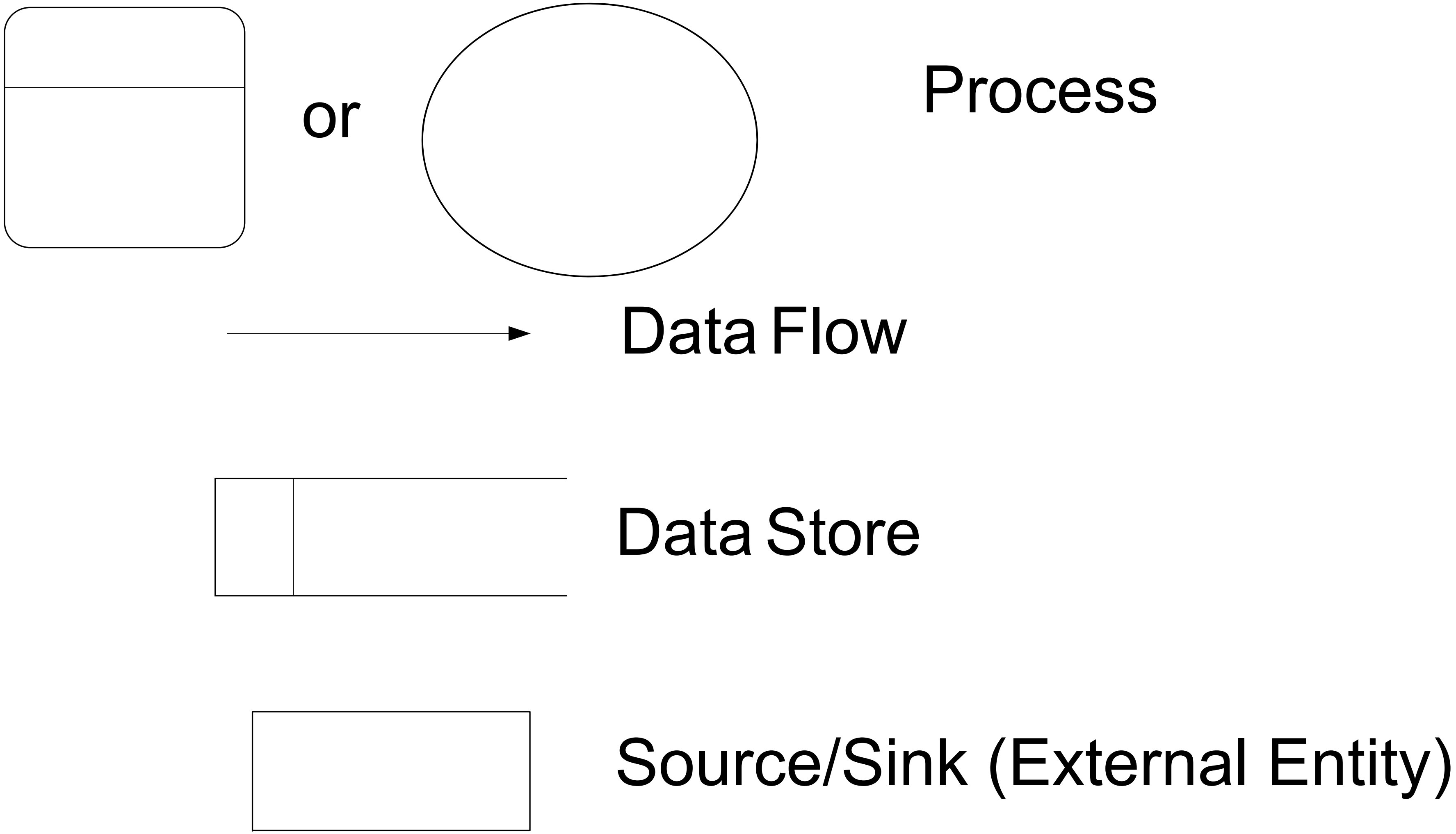
DFDs (Data Flow Diagrams)

Data Flow Diagrams (DFDs)

- Data flow diagram (DFD) is a picture of the movement of data between external entities and the processes and data stores within a system



DFD Symbols



or

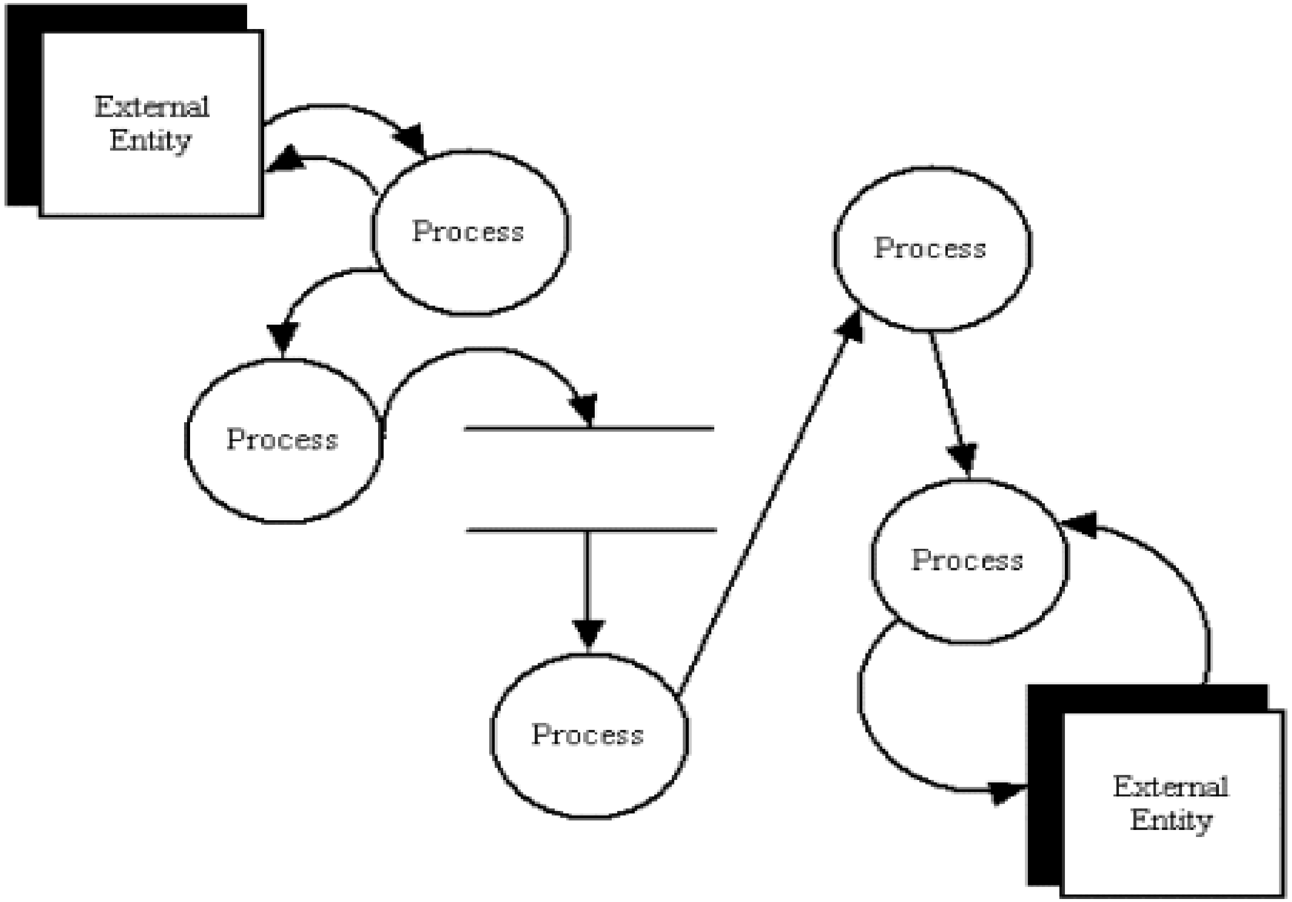
Process

Data Flow

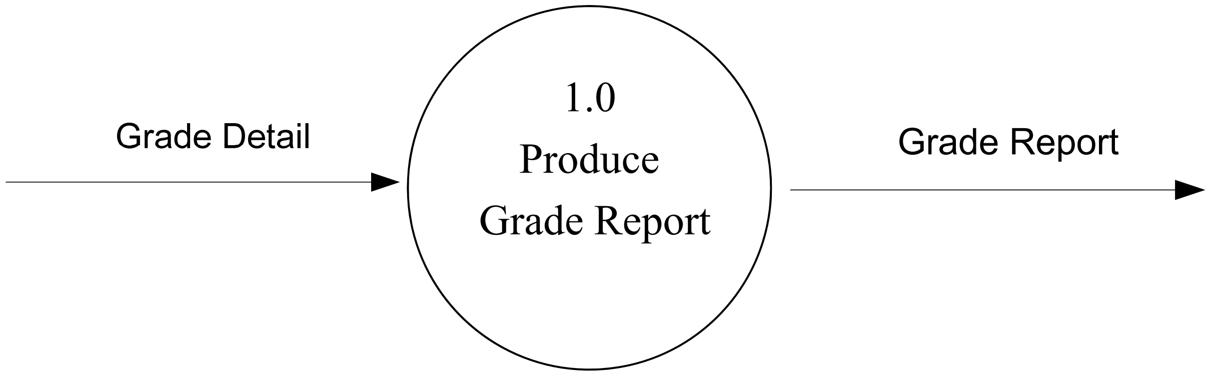
Data Store

Source/Sink (External Entity)

sample


External Entity
Process
Process
Process
Process
Process
External Entity

Process

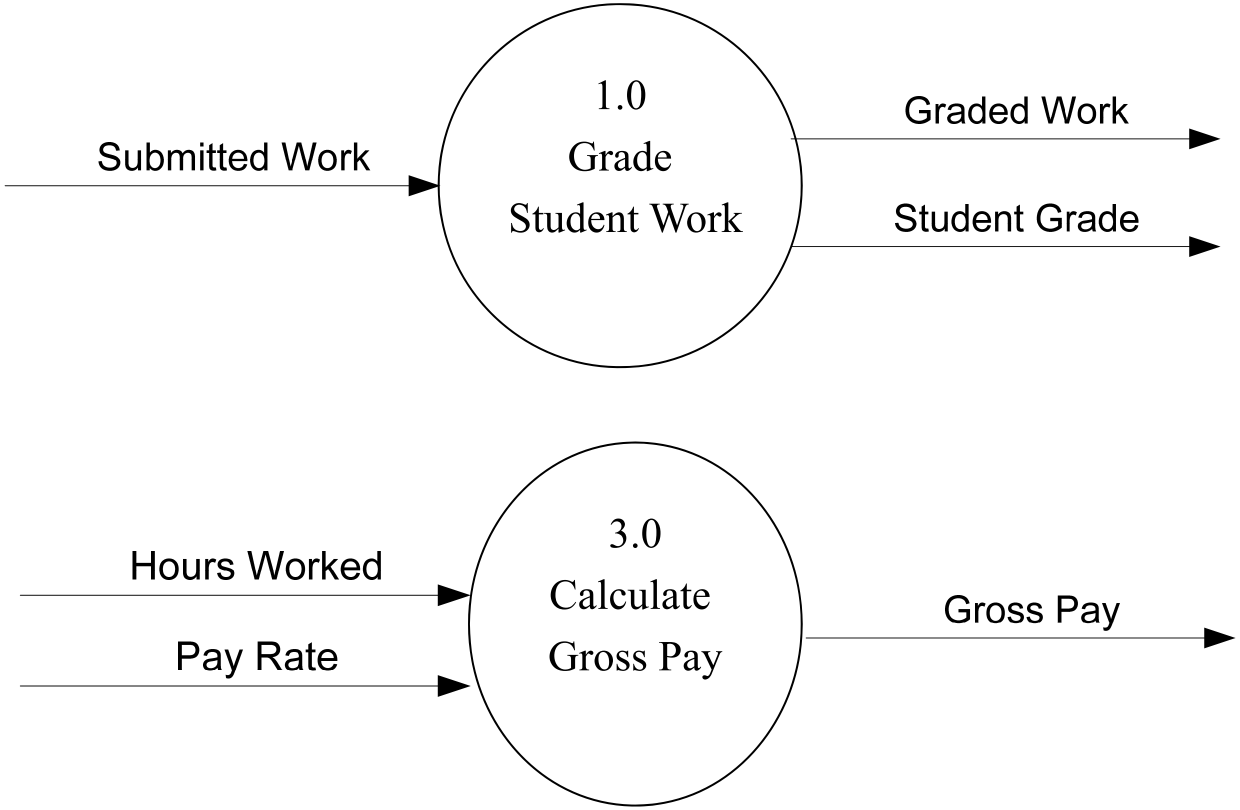


- Work or actions performed on data (inside the system)
- Labels should be verb phrases: active verb + object clause

- Receives input data and produces output

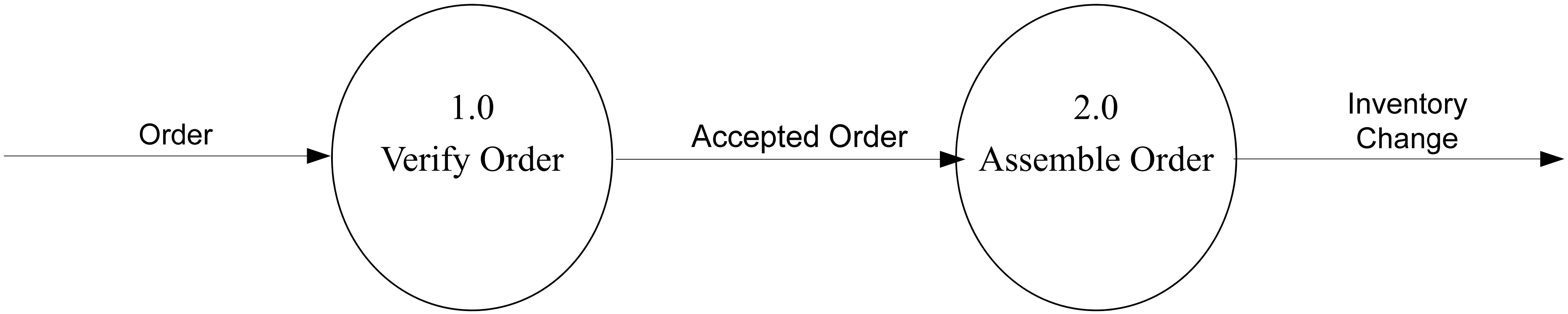
Rule 1: Process

- Can have more than one outgoing data flow or more than one incoming data flow

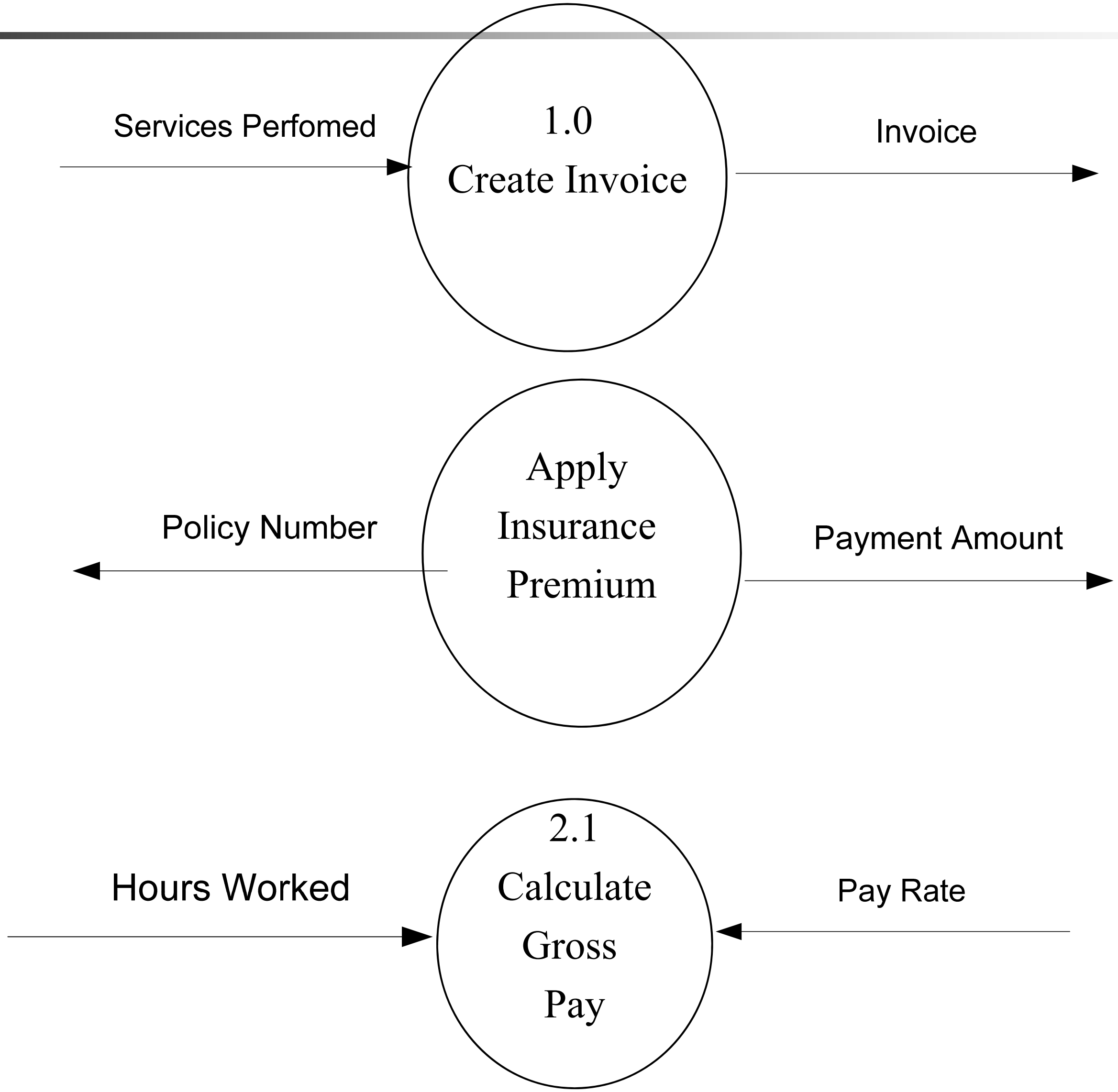


Rule 2: Process

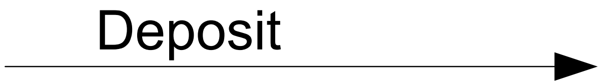
- Can connect to any other symbol (including another process symbol)

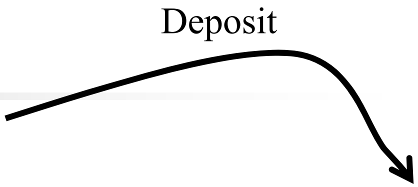


Process: Correct/Incorrect?

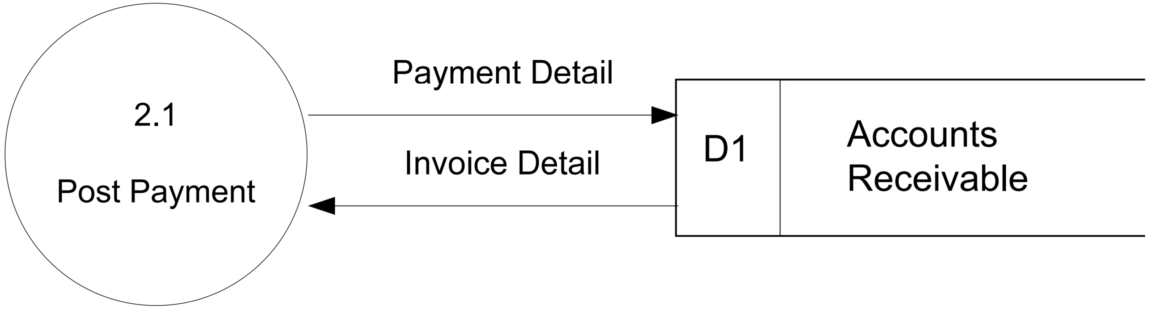


Data Flow

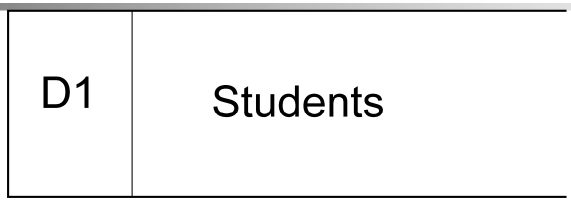




- Is a path for data to move from one part of the IS to another
- Arrows depicting movement of data
- Can represent flow between process and data store by two separate arrows



Data Store

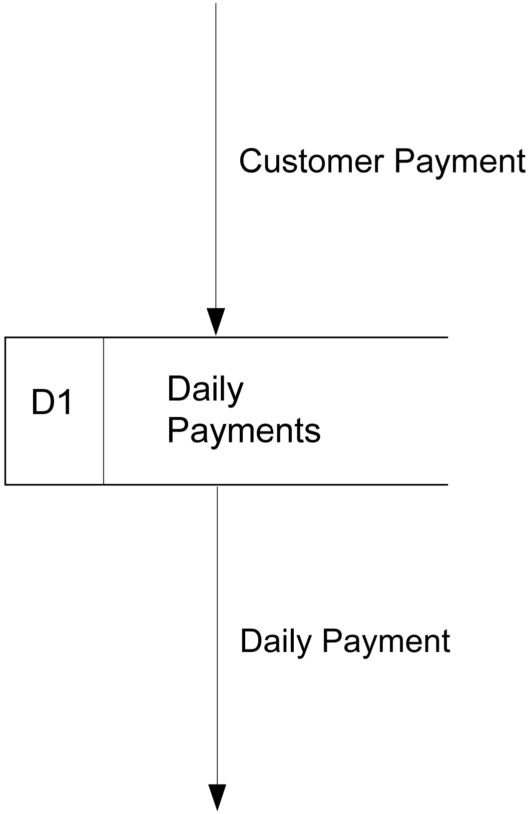


D1 Students

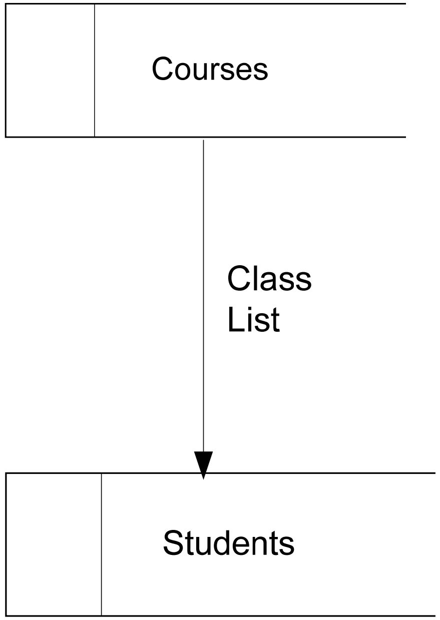
- Is used in a DFD to represent data that the system stores
- Labels should be noun phrases

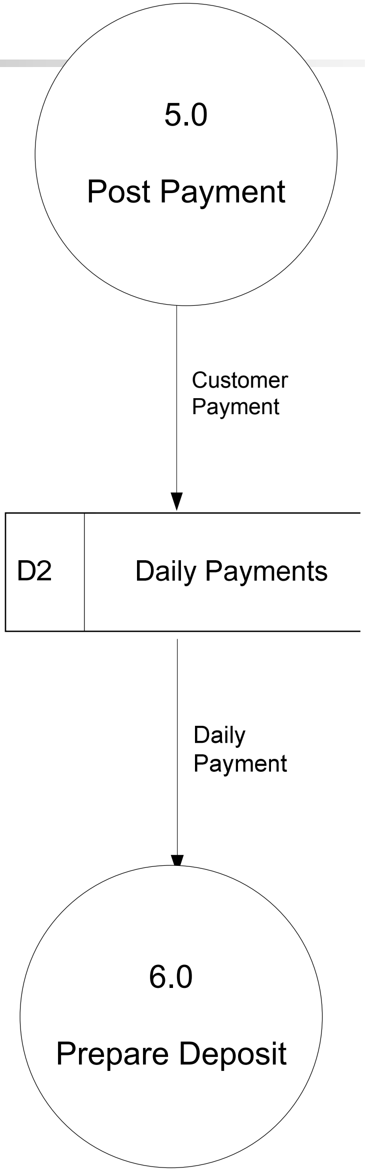
Rule: Data Store

- Must have at least one incoming and one outgoing data flow

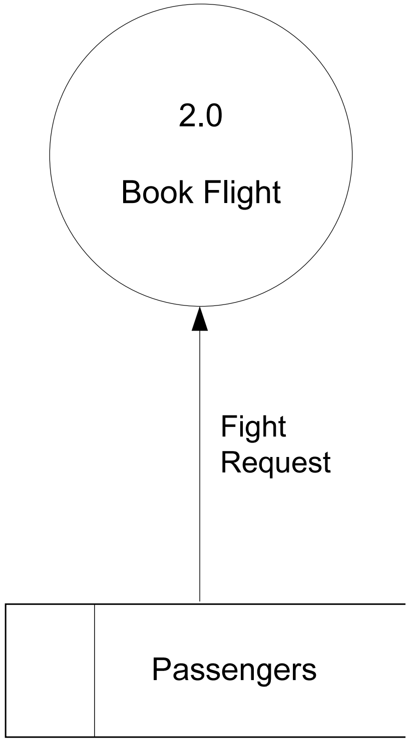


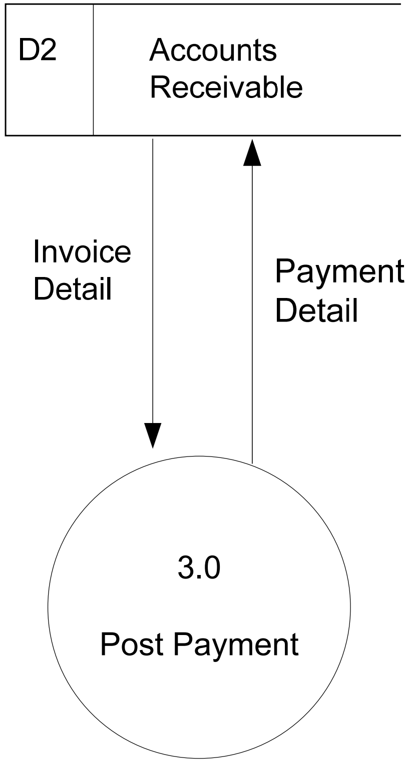
Data Flow: Correct/Incorrect?



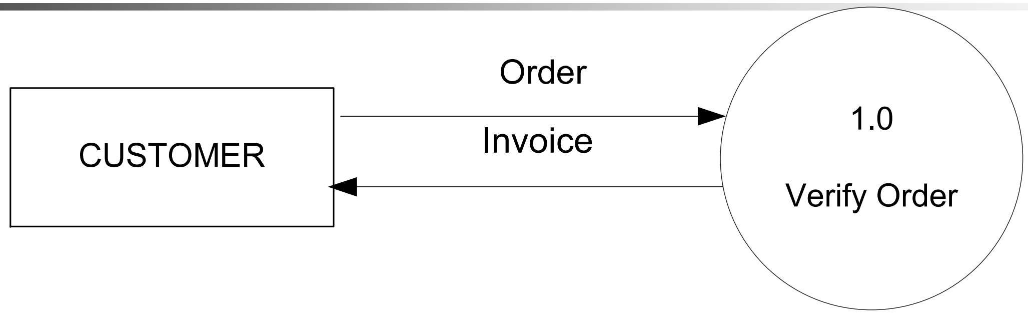


Data Store: Correct/Incorrect?





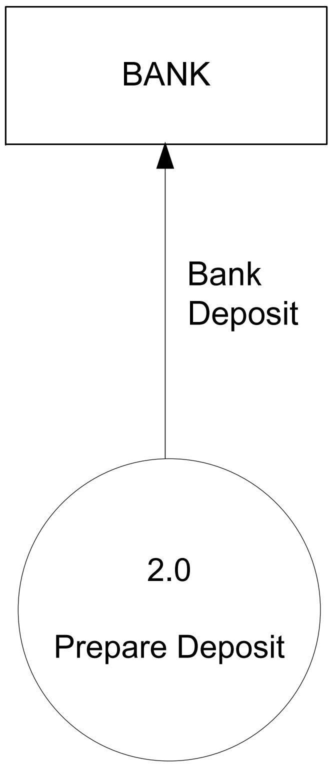
Source/Sink (External Entity)



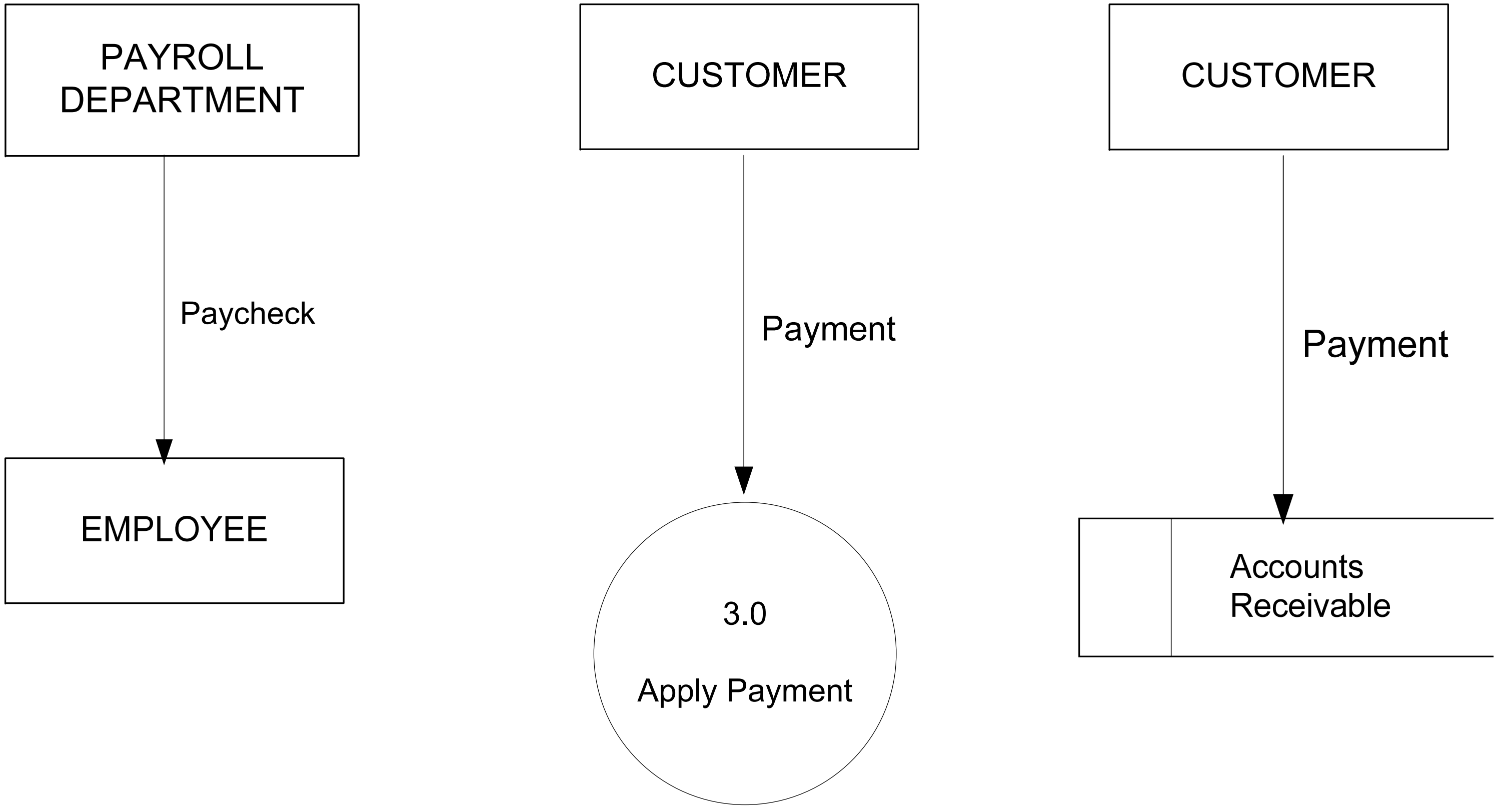
- External entity that is origin or destination of data (outside the system)
- Is the singular form of a department, outside organisation, other IS, or person
- Labels should be noun phrases
- Source – Entity that supplies data to the system
- Sink – Entity that receives data from the system

Rule: External Entity

- Must be connected to a process by a data flow



External Entity: Correct/Incorrect?

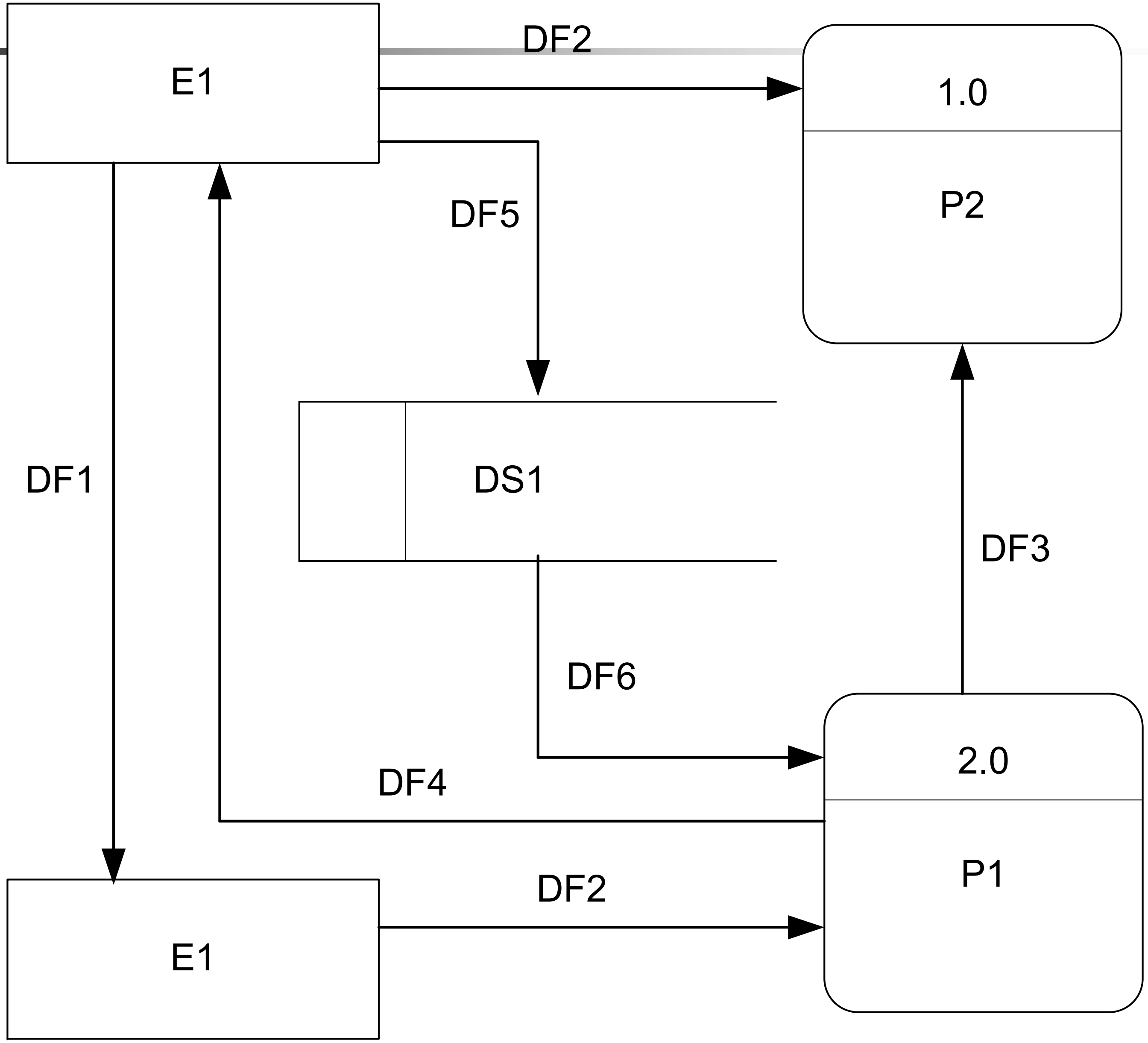


Rules for Using DFD Symbols

- Data Flow That Connects

	YES	NO
A process to another process	✓	
A process to an external entity	✓	
A process to a data store	✓	
An external entity to another external entity		✓
An external entity to a data store		✓
A data store to another data store		✓

List the errors of this DFD



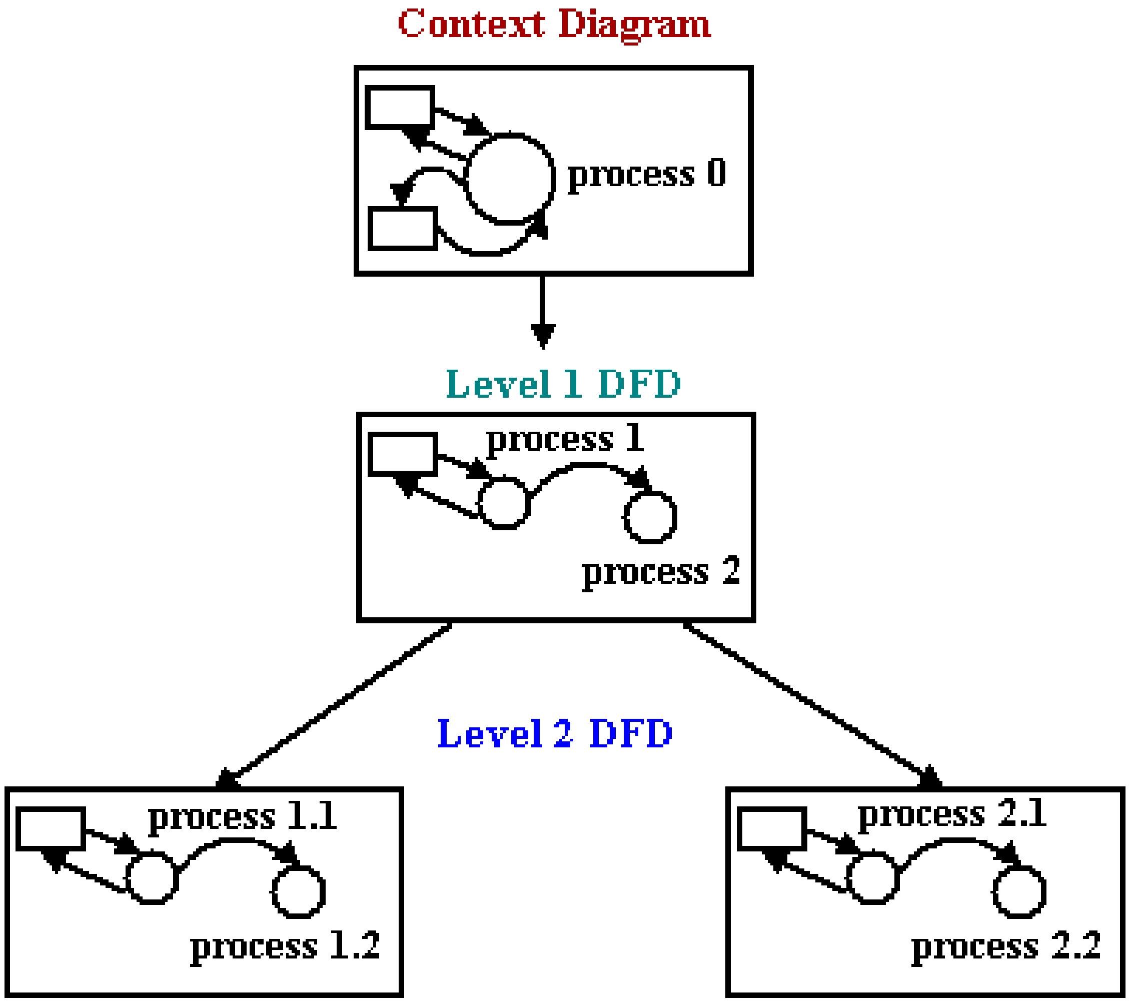
Guidelines for drawing DFDs

- Identify all external entities
- Identify all inputs and outputs
- Work your way through from Inputs to outputs
- Label all data flows and data stores descriptively
- Follow the rules

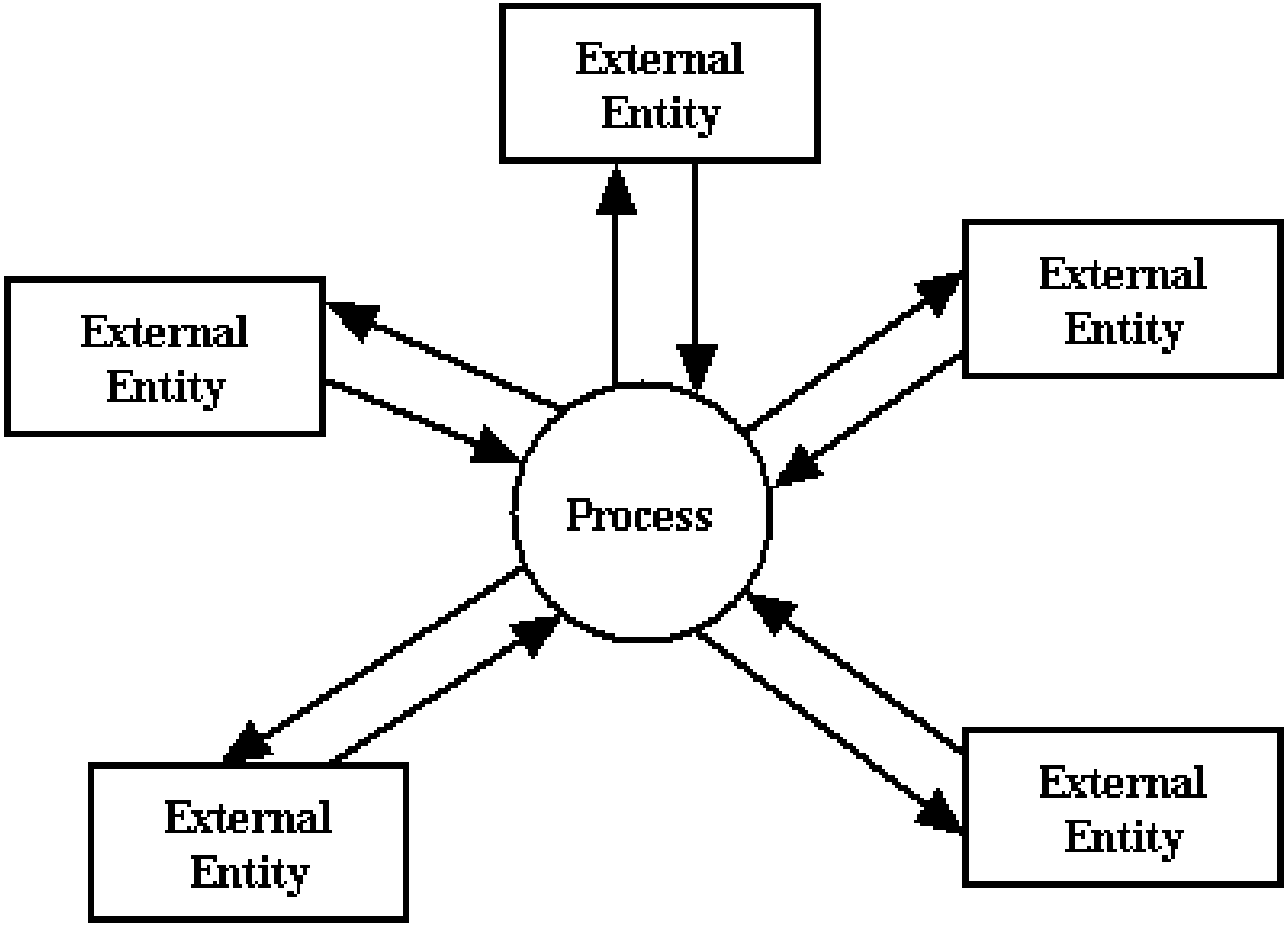
Levelling

- Typically DFD are traditionally drawn at three levels
  - The top-level DFD is ***context diagram (level 0)*** consisting of only one process, representing the entire system; It shows the interfaces between the system and the external entities.
  - Immediately beneath the context diagram is the ***system level diagram (level 1)*** i.e. highest level view of the major functions within the system, as well as the major interfaces between those functions.
  - Further levels can be decomposed to any depth by expanding individual processes

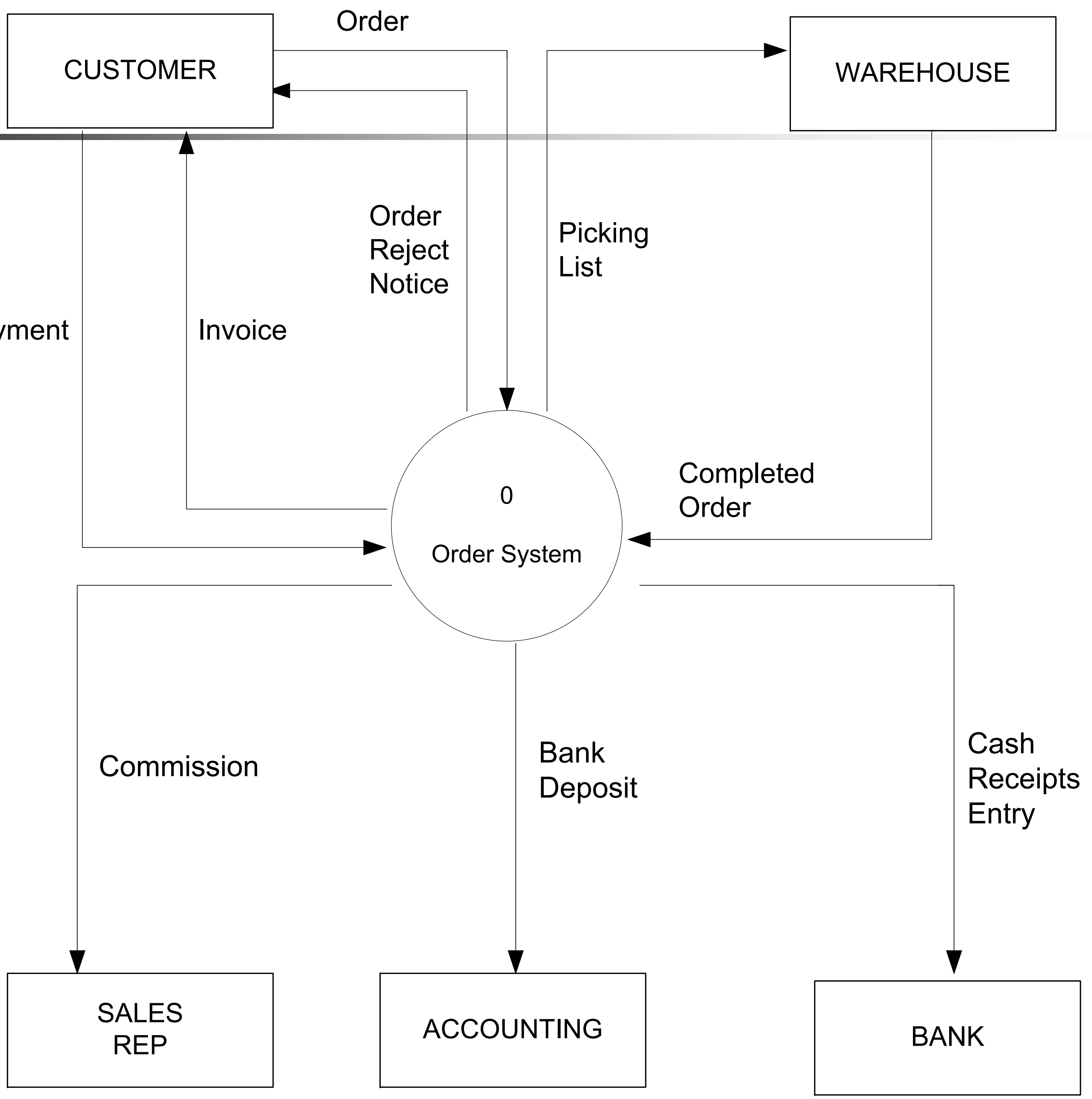
Levelled Diagram



Context Level Diagram

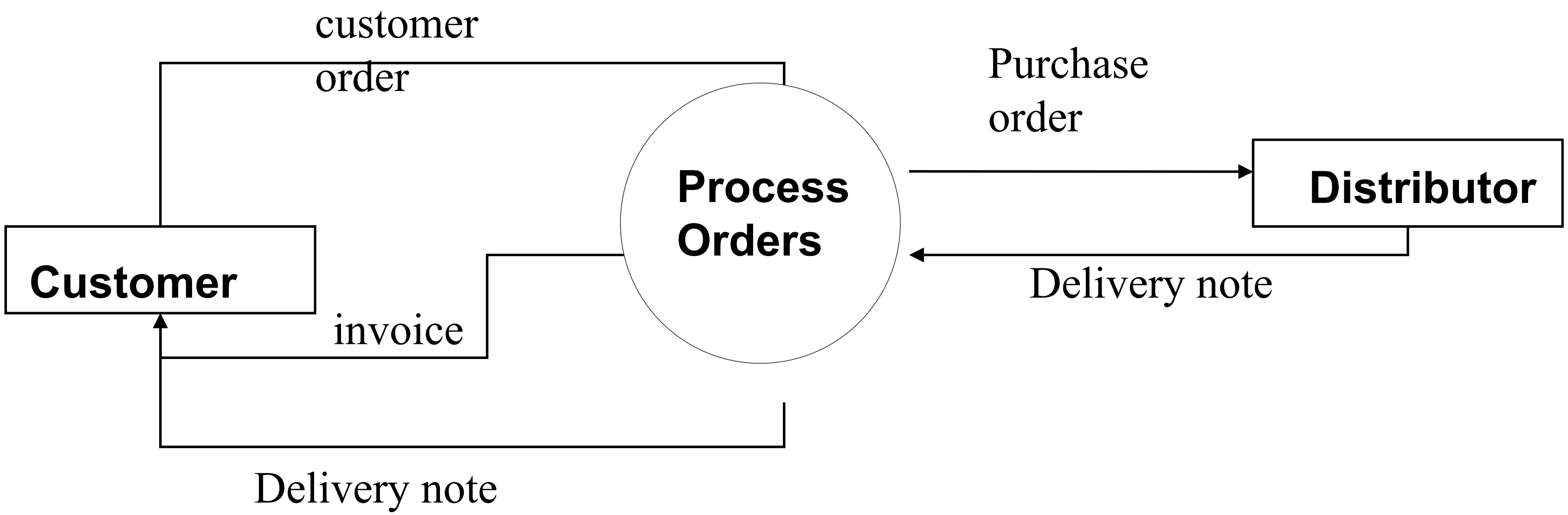


Context Diagram of Order System

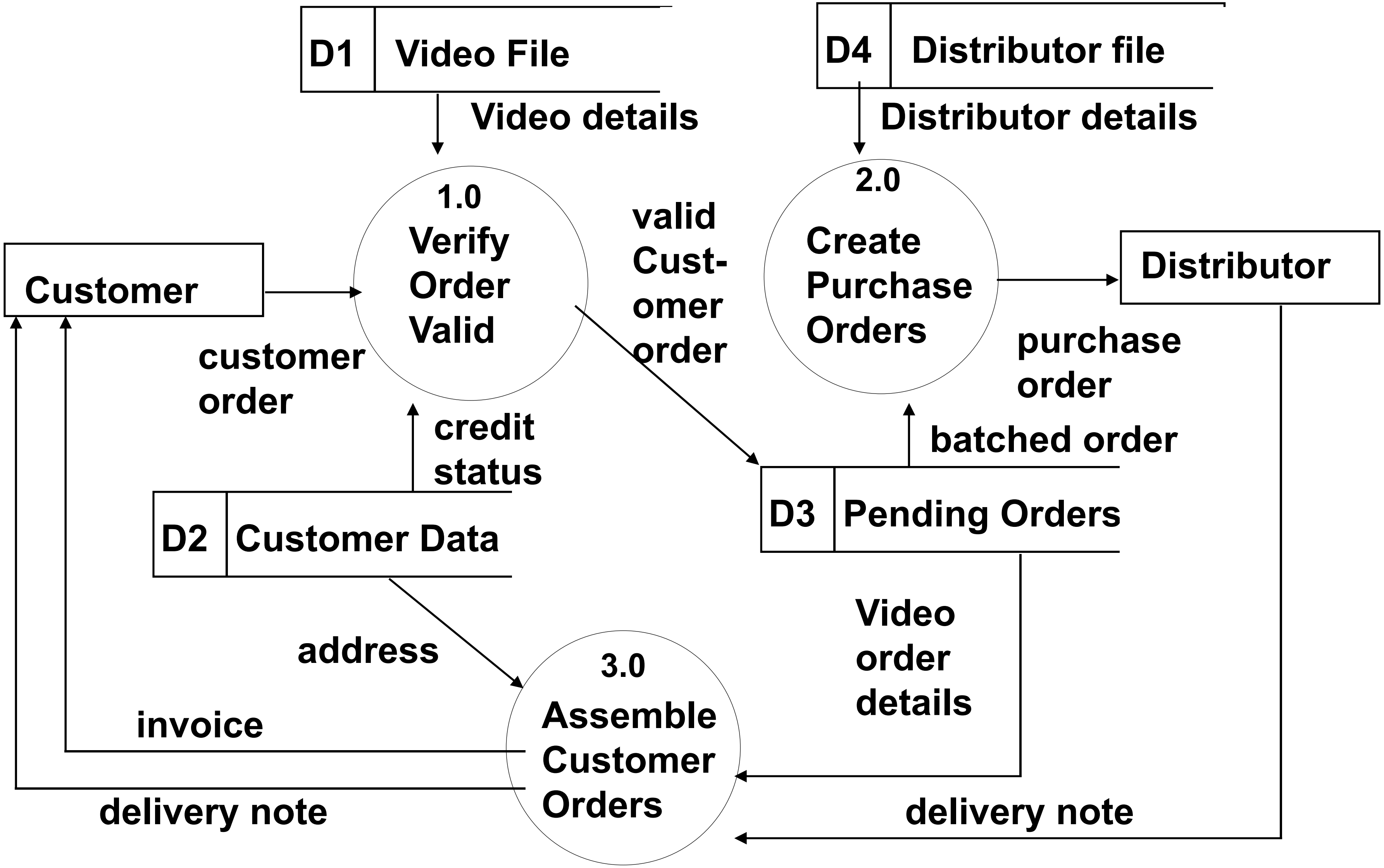


A video shop (1)

- Expanding this process (context level)

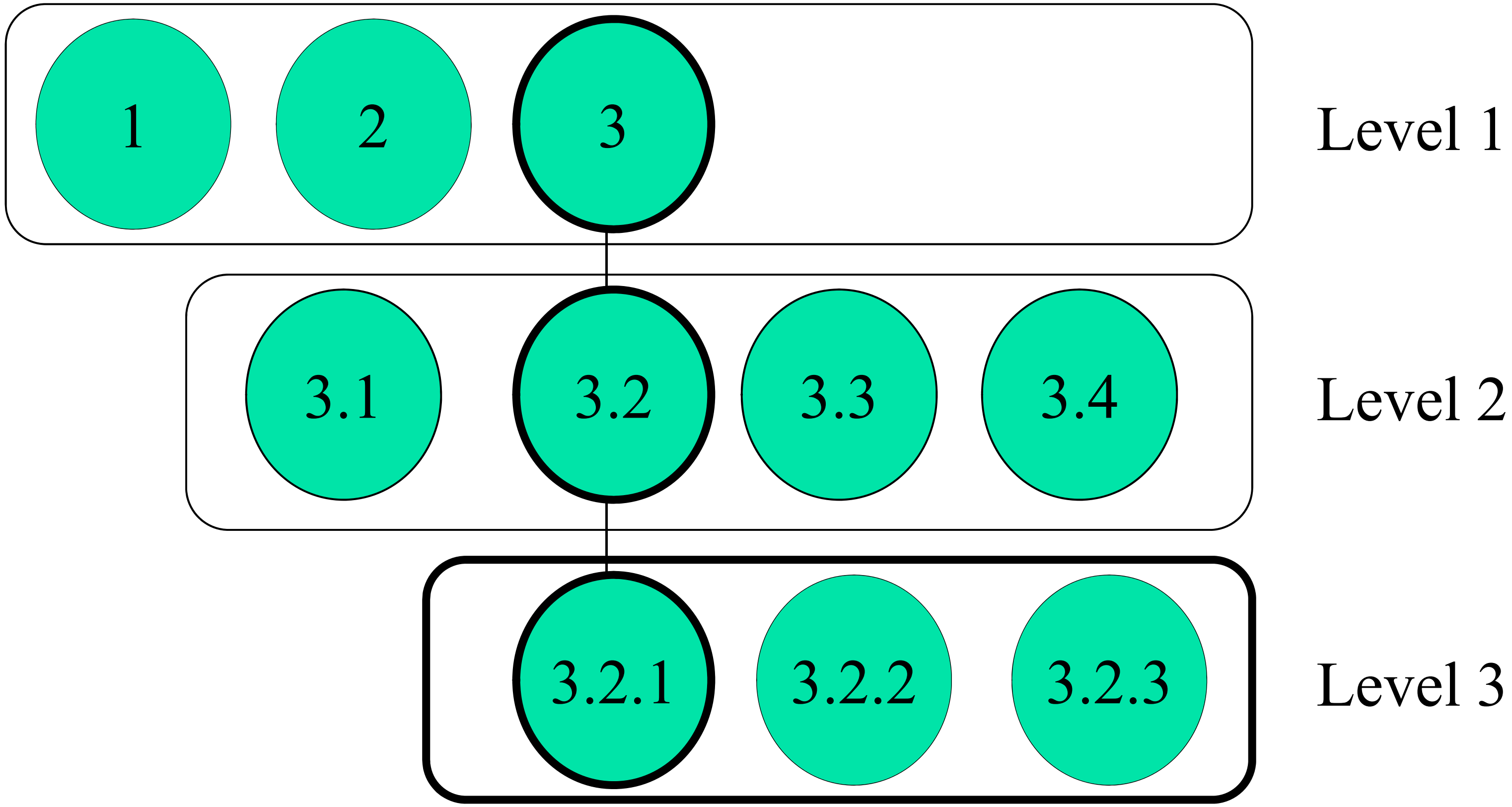


A video shop (2)



DFD Process Numbering Rules

The process boxes on the level 1 diagram should be numbered arbitrarily, so that no priority is implied. Even where data from one process flows directly into another process, this does not necessarily mean that the first one has to finish before the second one can begin.



1 2 3

Level 1

3.1 3.2 3.3 3.4

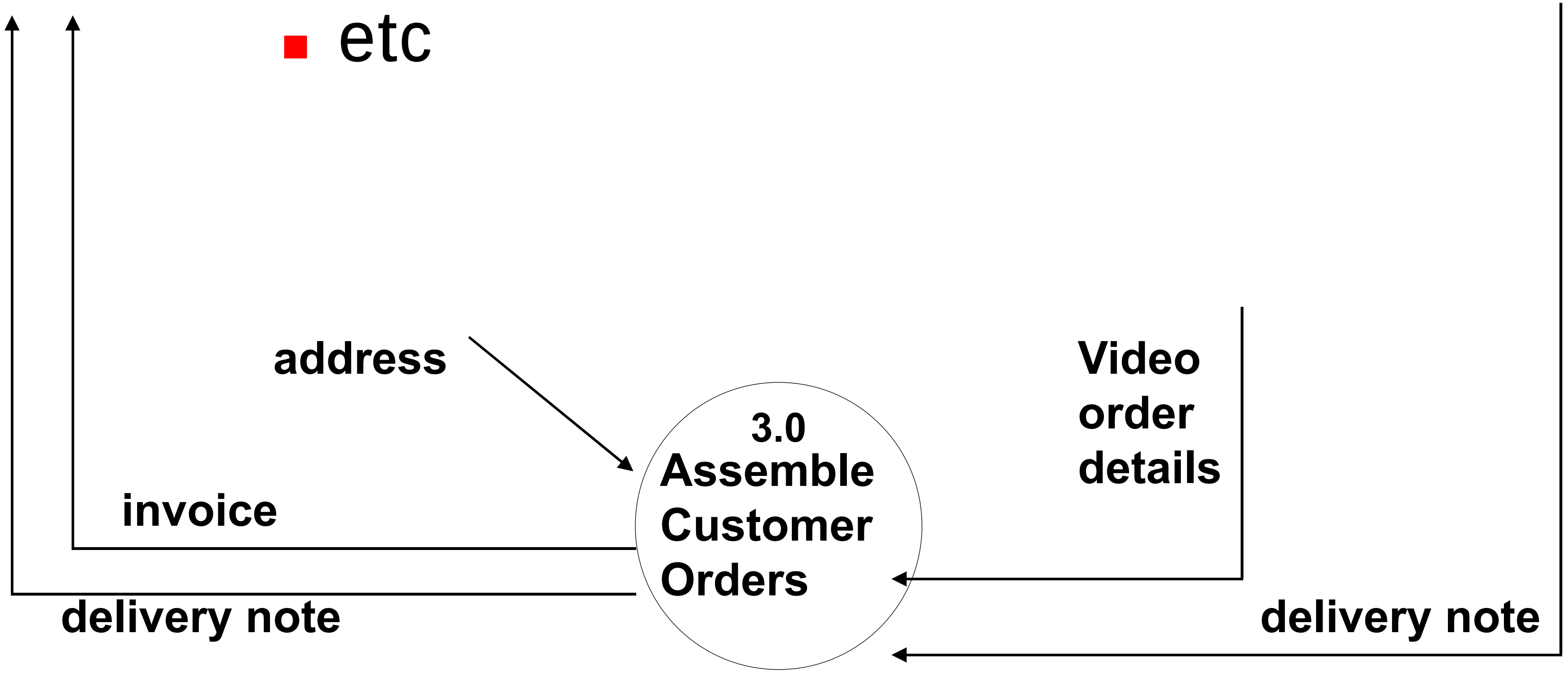
Level 2

3.2.1 3.2.2 3.2.3

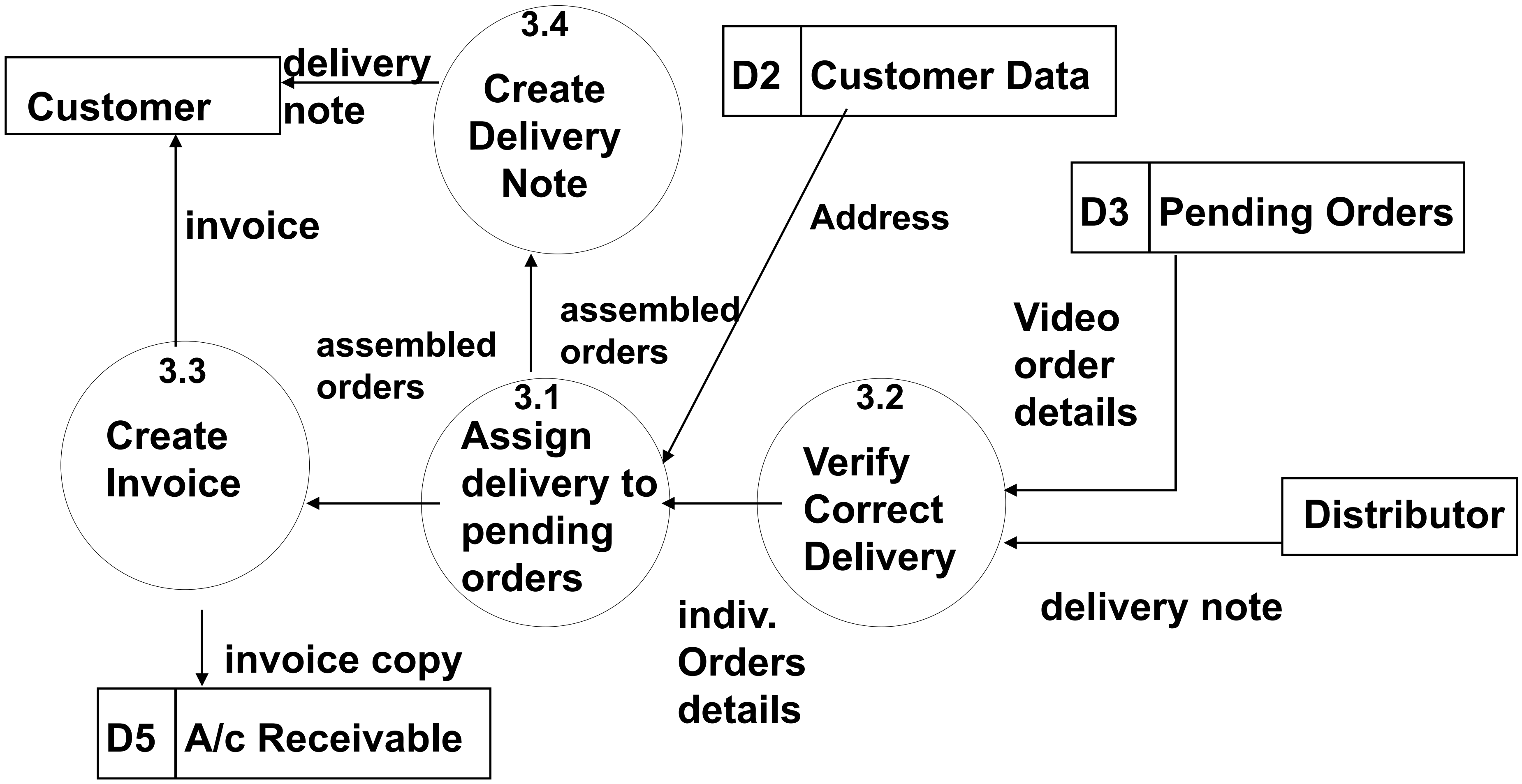
Level 3

Numbering

- Expand **Assemble Customer Orders**
  - New process 3.1
  - New process 3.2
  - etc



Expanded


Customer
delivery note
3.4
Create Delivery Note
D2
Customer Data
Address
D3
Pending Orders
invoice
assembled orders
assembled orders
Video order details
3.3
Create Invoice
3.1
Assign delivery to pending orders
3.2
Verify Correct Delivery
Distributor
delivery note
indiv. Orders details
invoice copy
D5
A/c Receivable

Some Principles (1)

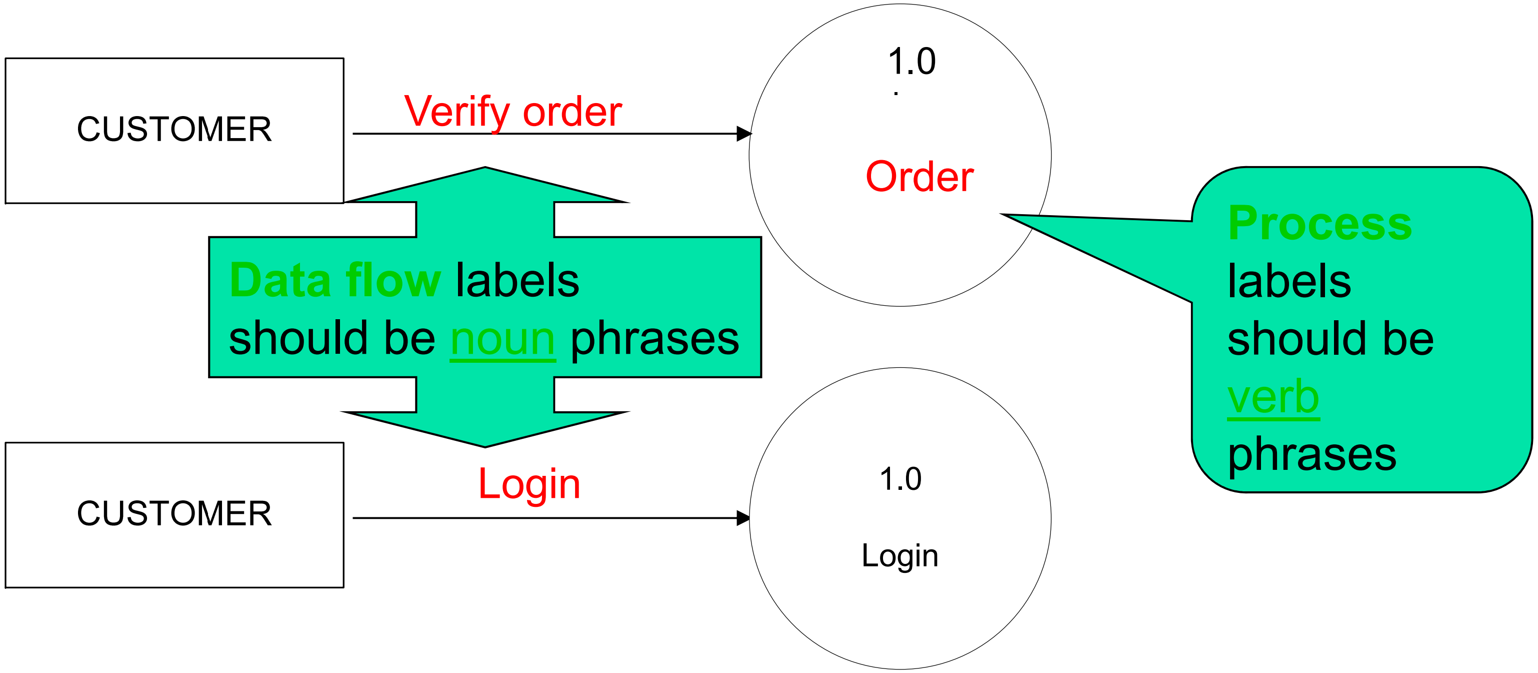
- Each DFD figure should have no more than 8 or 9 processes and related stores.
- The number of level and partition should be based on the above principle:
  - The number of levels is dependent on system complexity.
  - Complexity of each level may not be the same.
  - Some can be more complex than other levels.

Some Principles (1)

- Make sure that the levels of DFDs are consistent with each other: the data flow coming into and going out of a process at one level must correspond to the data flows coming into and going out of an entire figure at the next lower level which describe that process.
- Show a store at the highest level where it first serves as an interface between two more processes; then show it again in EVERY lower level

DFD
COMMON
MISTAKES

Mistake 1 : Verb vs. Noun

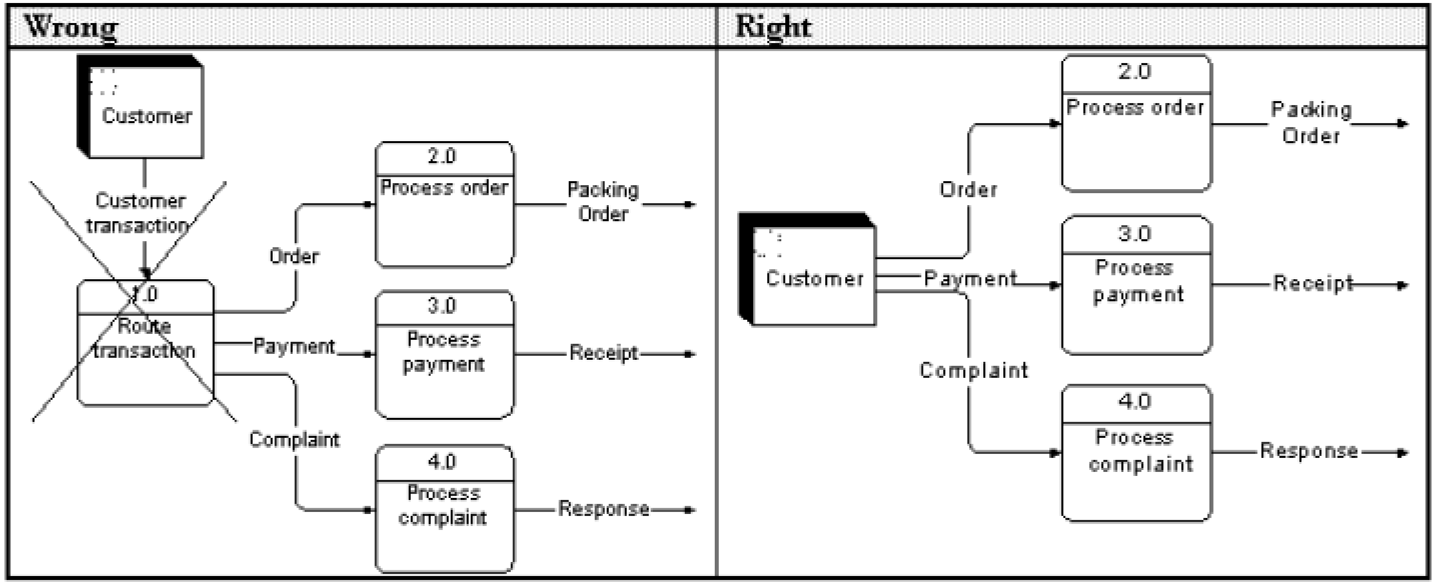

CUSTOMER
Verify order
1.0
Order
Data flow labels should be noun phrases
Process labels should be verb phrases
CUSTOMER
Login
1.0
Login

Incorrectly labeling processes or data flow

What DFD’s do not do (1)

- **DFDs do not handle sequence**
  - There is no notion of time in a DFD. It cannot be inferred that process 2 always follows process 1.
  - In many instances DFDs are drawn by considering a sequence of actions (which is a very sensible way of going about things) but that still does not mean that we can infer sequentially from a DFD.
- **DFDs do not handle priorities**
  - In a situation where two processes want to read from the same file at the same time, which one wins? DFDs do not address the problem.

Mistake 2 : DFD = Flowchart

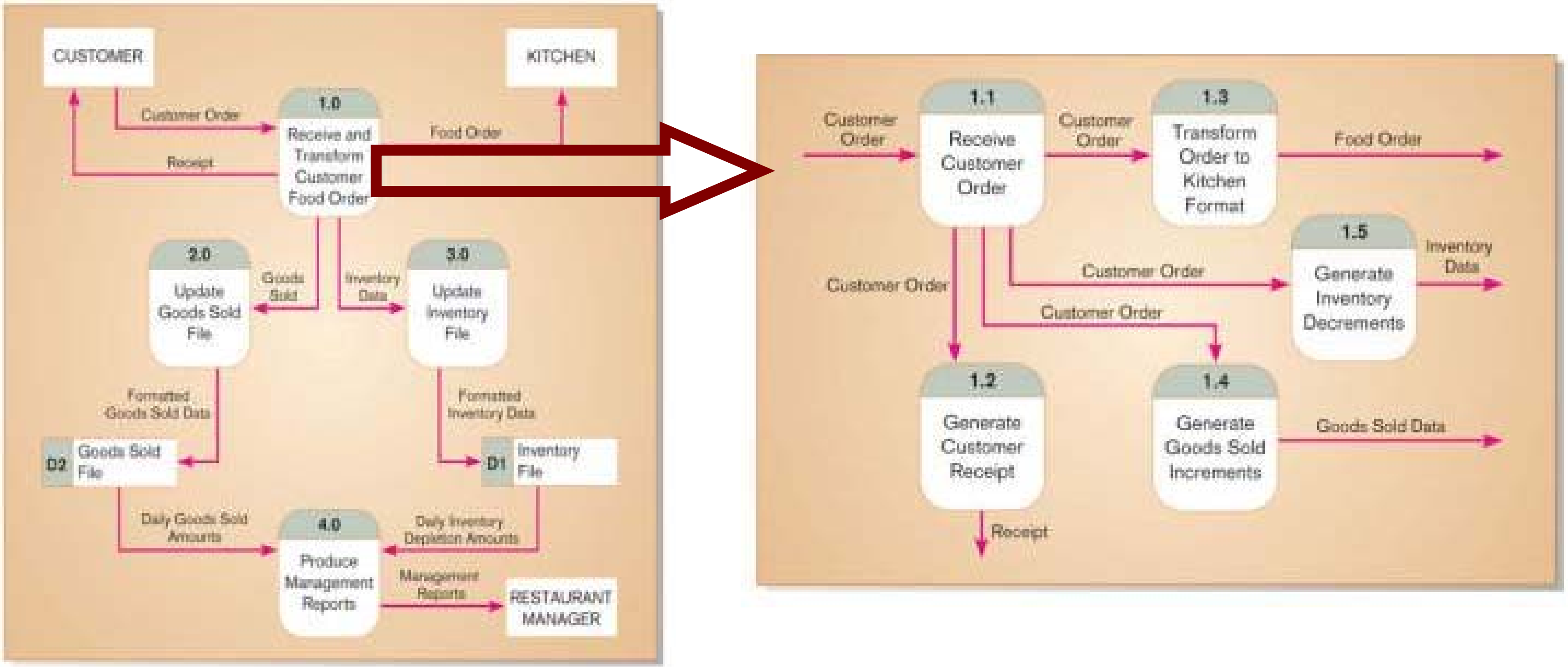


Source: Adapted from Figure 9.6, p. 357 in Whitten, J. L.; Bentley, L. D.; Barlow, V. M. (1994). Systems Analysis and Design Methods (Third Edition). Burr Ridge, IL: Irwin.

DFDs are not Flowcharts

Mistake 3 : Complicated

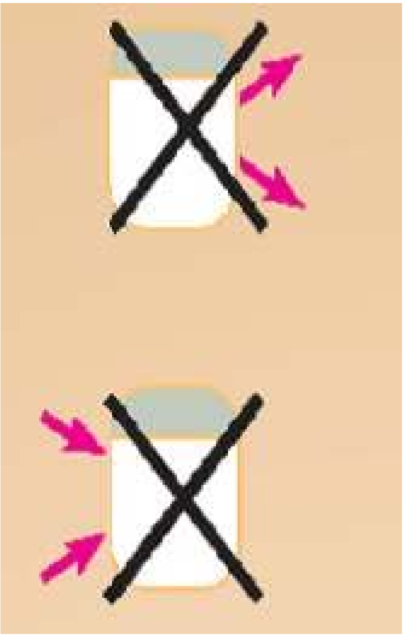
- Including more than nine processes on a DFD
  - Decomposition is needed
- The purpose of data flow diagrams is to provide a semantic bridge between <u>USERS</u> and systems developers.
  - User understanding and reviewing

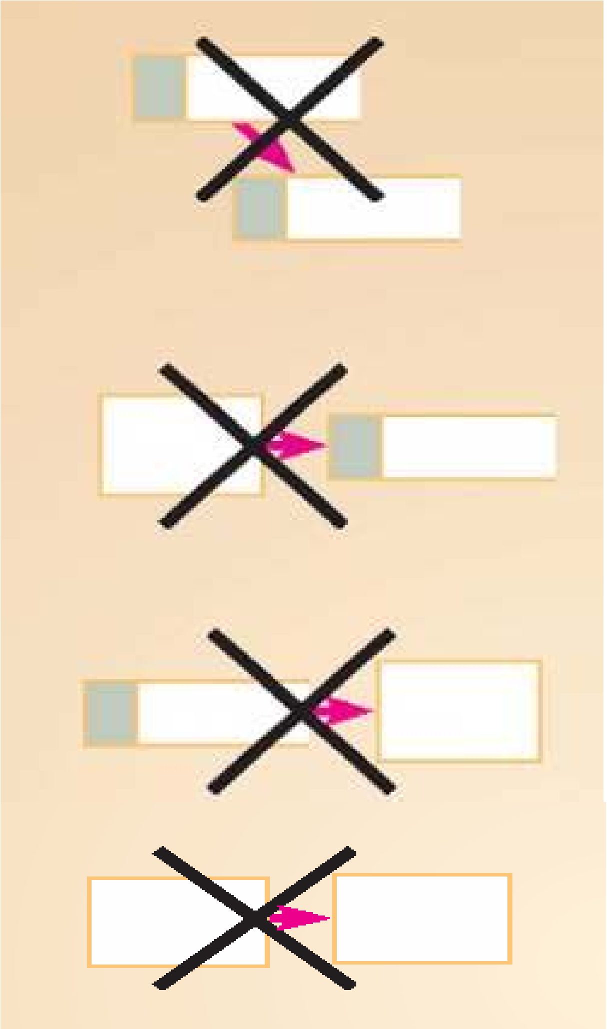


Mistake 4: Diagramming rules

Miracles

Black holes





What DFD’s do not do (2)

- **DFDs do not define the structure of the data**
  - The structure of the data in the data stores is glossed over in DFDs and any structure of the data in data flows is similarly ignored in DFDs. The structure of the data is left to the data view of the system.

Summary

- Know 4 main symbols
- Naming is very important
- Know Context level, systems level and lower levels
- Know how to decompose processes to lower levels

The end...Thank you!