



DATA FLOW DIAGRAMS

Lecture Notes Prepared by
Melody Angelique C. Rivera
Faculty, College of Computer
Studies, Silliman University

OVERVIEW

A Data Flow Diagram (DFD) shows how data is processed within a system based on inputs and outputs

Visual symbols are used to represent

- the flow of information
- data sources and destinations
- where data is stored

DFDs are often used as a first step toward redesigning a system

They provide a graphical representation of a system at any level of detail, creating an easy-to-understand picture of what the system does

They can be used to depict the flow of data in a current system (manual or automated)

They can also be used to illustrate the flow of data in a proposed system

LEVELS

Context DF Diagram (Level 0)

- the most basic data flow diagrams
- provide a broad view that is easily digestible but offers little detail
- show a single process node and its connections to external entities

Top-Level DF Diagram (Level 1)

- still a general overview, but has more detail than a context diagram
- single process node from the context diagram is broken down into subprocesses
- as subprocesses are added, the diagram will need additional data flows and data stores to link them together

Level of Explosion (Level 2+)

- simply break processes down into more detailed subprocesses
- in theory, DFDs could go beyond level 3, but they rarely do
- Level 3 data flow diagrams are detailed enough that it doesn't usually make sense to break them down further

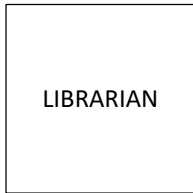
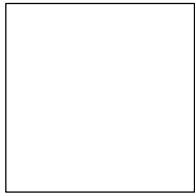


AUTHORS

Gane and Sarson

Yourdon and Coad/DeMarco

SYMBOLS (GANE AND SARSON) (1)

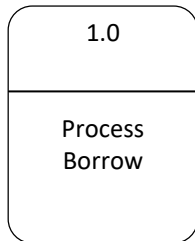
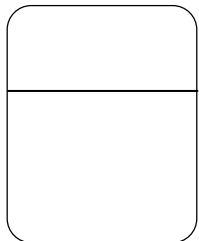


External Entity

- can be a title of a person (e.g., STUDENT) or a department/unit (e.g., CATALOGUING SECTION)
- also known as actors, sources or sinks, and terminators
- typically placed at the boundaries of the diagram
- can represent another system or indicate a subsystem.

Source – where/from whom data comes from

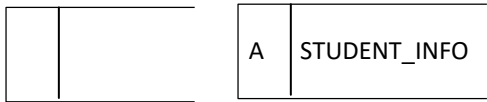
Sink – where/to whom the data is given



Process

- an activity that changes or transforms data flows
- they transform incoming data to outgoing data
- all processes must have inputs and outputs on a DFD
- Example: “Ship Order”
- Processes are typically oriented from top to bottom and left to right on a data flow diagram

SYMBOLS (GANE AND SARSON) (2)

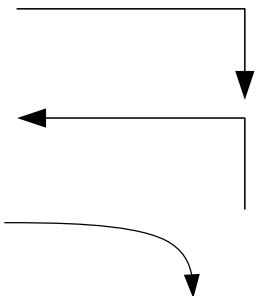


Data Store

- A data store does not generate any operations but simply holds data for later access
- Data stores could consist of files held long term or a batch of documents stored briefly while they wait to be processed
- Input flows to a data store include information or operations that change the stored data
- Output flows would be data retrieved from the store

←Updated Book Data—

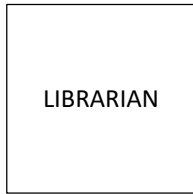
—Book Data→



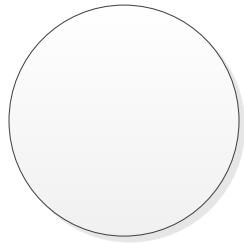
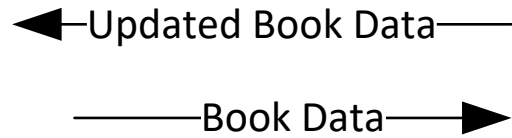
Data Flow

- Movement of data between external entities, processes and data stores
- Represented with an arrow symbol, which indicates the direction of flow
- This data could be electronic, written or verbal
- Input and output data flows are labeled based on the type of data or its associated process or data store
- The label or data flow name is written alongside the arrow

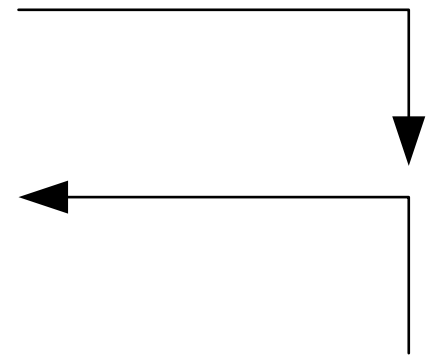
SYMBOLS (YOURDON AND COAD/DEMARCO)



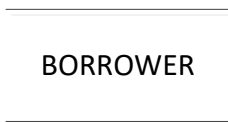
External Entity



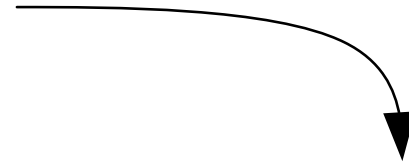
Process



Data Flows



Data Store



HOW TO CREATE A DFD (1)

1. Identify major inputs and outputs in your system

- Nearly every process or system begins with input from an external entity and ends with the output of data to another entity or database/data store
- Identifying the inputs and outputs will give you an overview of your system, showing the broadest tasks the system should achieve
- Knowing them early is important because the rest of your DFD will be built/based on these elements

2. Build a context diagram

- Once the major inputs and outputs have been identified, building a context diagram is simple
- Draw a single process node and connect it to related external entities

HOW TO CREATE A DFD (2)

3. Expand the context diagram into a level 1 DFD

- The single process node of your context diagram doesn't provide much information so you need to break it down into subprocesses
- In Level 1 data flow diagrams, several process nodes, major databases, and all external entities (as seen in the context diagram) should be included
- Walk through the flow of information
 - where does the information start?
 - what needs to happen to it before each data store?

HOW TO CREATE A DFD (3)

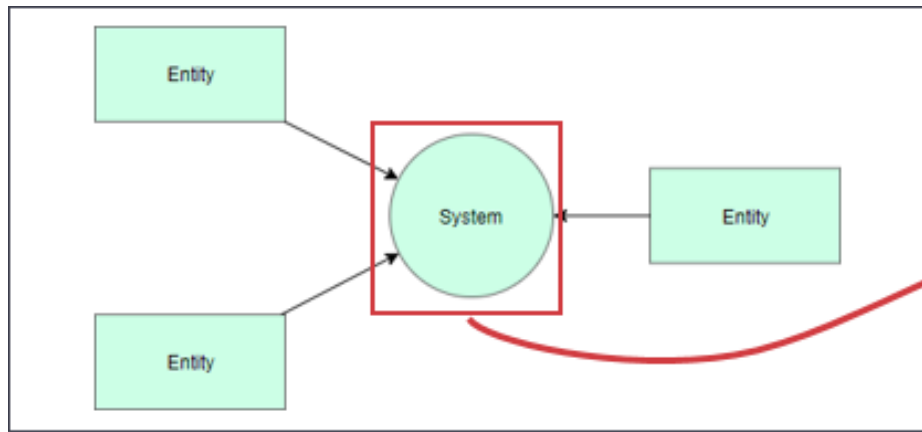
4. Expand to a level 2+ DFD

- To enhance the detail of your data flow diagram, follow the same process as in step 3
- The processes in your level 1 DFD can be broken down into more specific subprocesses
- Ensure that any necessary data stores and flows are added
- To progress beyond a level 2 data flow diagram, simply repeat this process
- Stop once you've reached a satisfactory level of detail

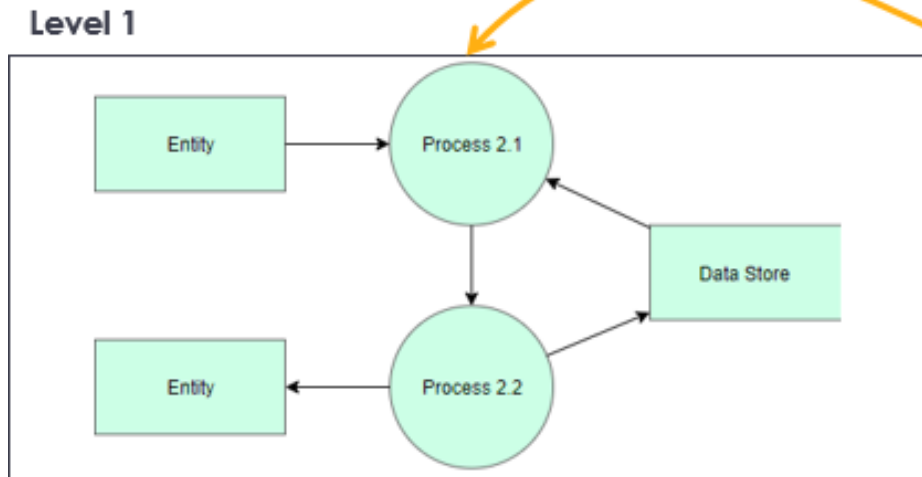
5. Confirm the accuracy of your final diagram

- When your diagram is completely drawn, walk through it
- Pay close attention to the flow of information from an entity to a process to a data store
- By looking at your final diagram, other readers should be able to understand how your system functions
- Before presenting your final diagram, check with the project team to ensure that your diagram is understandable

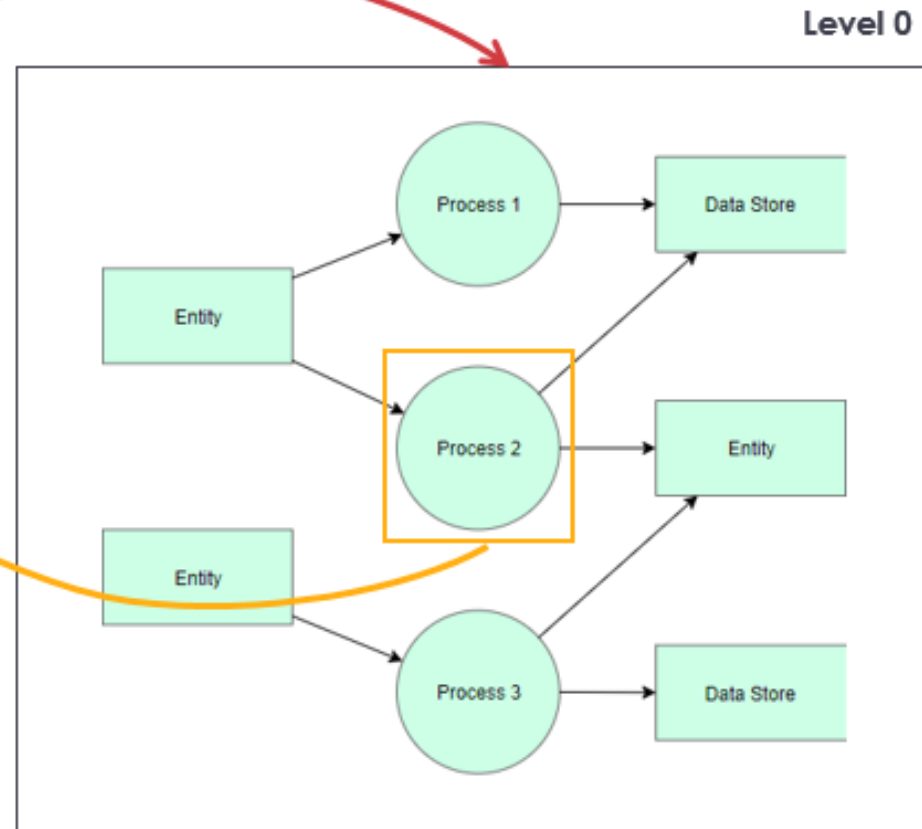
Top-Down Decomposition Technique in DFDs



Context diagram

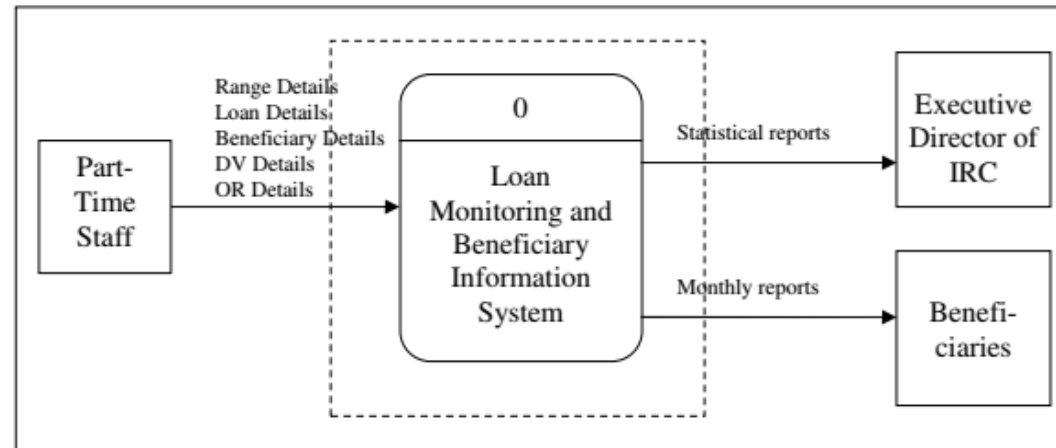


Level 1

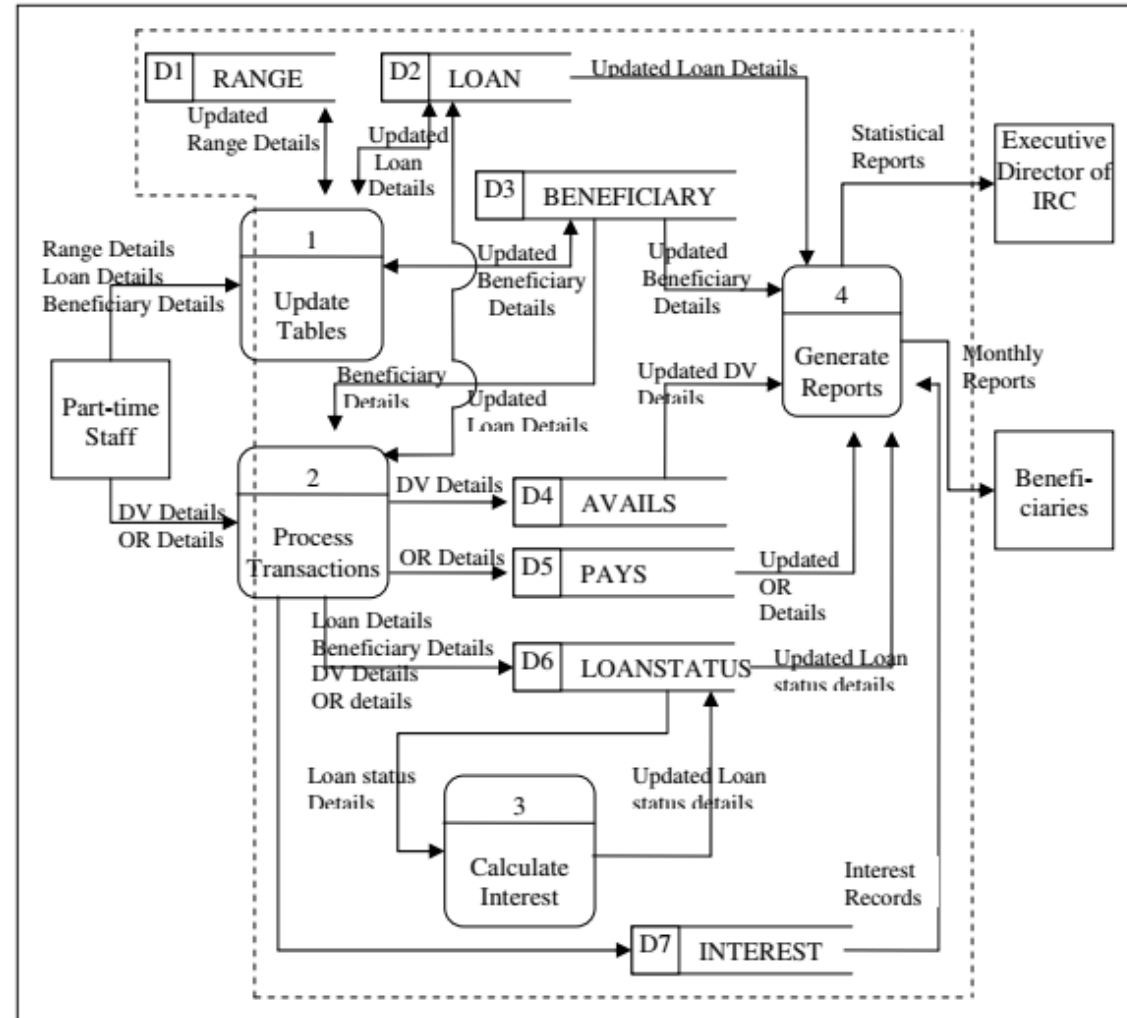


Level 0

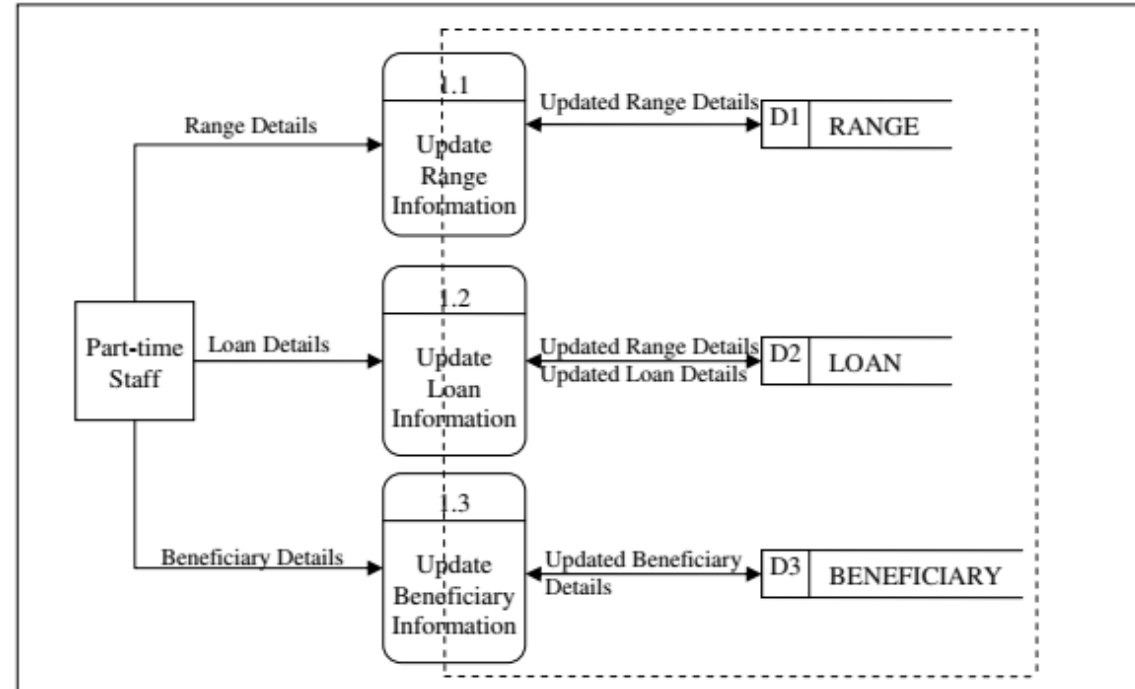
CONTEXT DIAGRAM (LEVEL 0)



TOP LEVEL DIAGRAM (LEVEL 1)



LEVEL OF EXPLOSION





END OF PRESENTATION

