

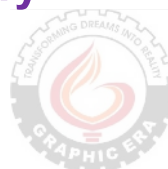
Unit 2 – Requirements Engineering

Tools for Gathering Requirements

Decision Tables

Introduction

- Several methods used to know the user requirements
- **Tools** that are traditionally used to **document** the gathered information
 - ❖ **Decision Table**
 - ❖ **Decision Tree**
 - ❖ **Data Flow Diagrams**
 - ❖ **Data Dictionary**



Decision Tables

- Different **actions** are taken under different **conditions**
- Compact way of **documenting** and presenting them is by using decision tables

Conditions	Condition Entries
Actions	Action Entries

Decision table for library Requisition

Conditions	Decision rules			
	1	2	3	4
Textbook?	Y	Y	N	N
Funds Available?	Y	N	Y	N
Actions				
Buy	X		X	
Waitlist for Next Year.		X		
Return the Reco to the HOD.				X



Condition and Condition Entries

- **Conditions** are usually defined in a manner such that they can be expressed in a **binary** manner -True or False, or Yes or No.
- Examples of conditions are
 - Is the price minimum among all quotations?
 - Is age less than 40?
- **Condition entries** in the above situations are always either Yes (Y) or No (N).

Action

- For a given **set of conditions**, one needs to know the **action** which is usually followed in the system under consideration.
- **Examples of actions are**
 - Recruit the applicant.
 - Admit the student.
 - Place order.
- **Cross marks (X)** are always used for **action entries**. They are placed one in each column.
- A **condition-action** combination defines a **decision rule**. The columns spanning the decision entries and the action entries compartments are the various decision rules.

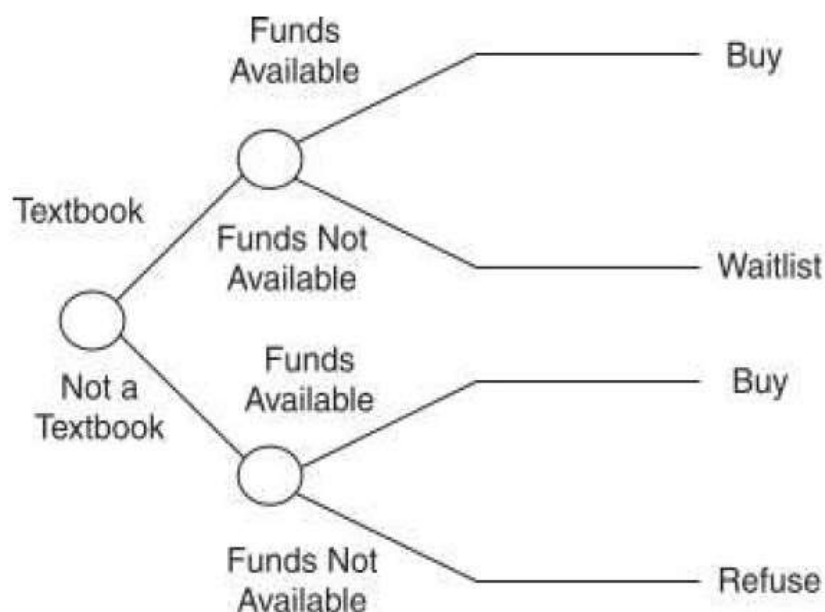


Unit 2 – Requirements Engineering

Tools for Gathering Requirements

Decision Trees

Decision Tree for Library Requisition



Decision Tree

- Decision trees provide a very useful way of showing **combinations of conditions** and resulting **action** for each such combination
- A decision tree starts from a root node, with **branches showing conditions**.
- Each **node** is a **decision point** and a choice has to be made at each node
- Each **branch** has a **corresponding value** to the decision choice

Applications

- Decision Trees are best used for **logic verification** or **moderately complex decisions** which result in a small number of actions.
- They are especially also useful to represent the **order** of decision making in the system.
- Decision Tables may also be used for problems involving **complex combinations**.
- They can represent a actions resulting from a large **number of combinations** of conditions.



Unit 2 – Requirements Engineering

Tools for Gathering Requirements

Data Flow Diagram (DFD)

Data Flow Diagrams (DFD's)

- Graphical way to understand how **data flows** and get **transformed** and **stored** in a organization or system
- DFDs are **easier** to understand than text.
- DFDs are useful tools for **database designers**, system programmers and **analysts**, end users and other members of the team.
- Uses four **symbols**



Data Flow Diagrams Symbols


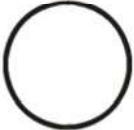


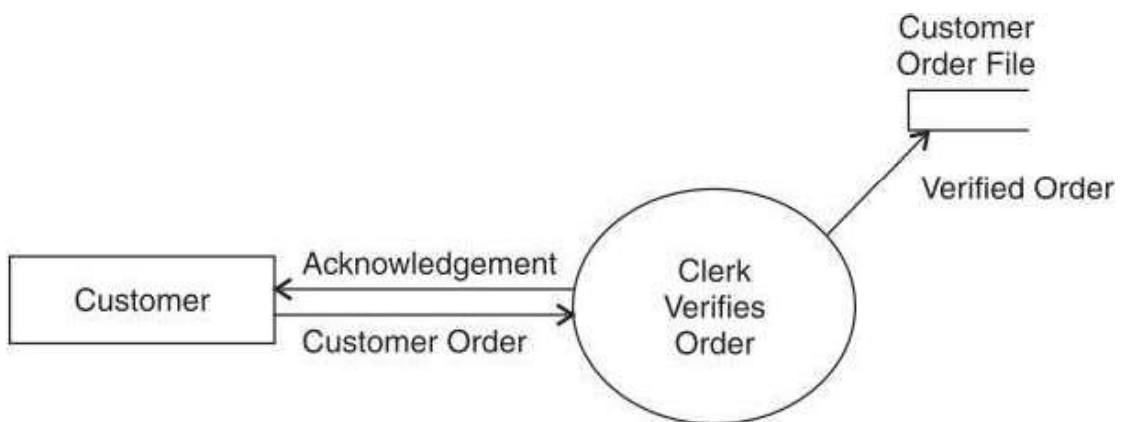
	dataflow	Arrows showing direction of flow
	process	circles
	file	horizontal pair of lines
	data-source, sink	rectangular box

Illustration of DFD – Customer Order Receipt



Meaning of Symbols[1/2]

- A data flow depicts an **input** to or an **output** of a process. The data may be in the form of a document or a record.
- The **arrowhead** of the symbol indicates the direction of flow of the data.
- A **data transform** (or a **process**) receives data as input and transforms it to produce output data.

Meaning of Symbols[2/2]

- A **data store** represents a **repository of data** that is stored for use as input to one or more processes. It can be a computer database or a manually operated file.
- An **external entity** lies outside the boundary of the system under consideration. It may be the **origin** of certain data that flows into the system boundary thus providing an input to the system, or it may be the **destination** of a data that originates within the system boundary.
- Frequently, an external entity may be both an **originator** and a **receiver** of data.



Hierarchical Data Flow Diagrams

- The DFDs express different **levels of abstraction** in a system. DFDs are divided into three levels: **Level-0**, **Level-1** and **Level-2**. These level diagram present information depending on the scope.
- That is **Level-0** DFDs cover the **overview** of the entire project
- **Level-1** DFDs describe some of the **sub-systems** or sub-processes and the data flow.
- The **Level-2** DFDs go deeper and breaks the subsystems in a more **detailed manner**.

