

Unit 2

Requirement Analysis and Requirement Determination

Requirement Determination

- Requirement Determination involves studying the current business system to find out how it works and where improvements should be made.

Why Requirement Determination?

- Systems analysts do not work as managers or employees in user departments they do not have same base of facts and details as the managers and users in those areas.

Activities in Requirement Determination

- Requirement Anticipation:
 - Foreseeing (predicting) systems characteristics based on previous experience.
- Requirement Investigation:
 - Using a variety of tools and skills : fact-finding techniques, data flow analysis, decision analysis
- Requirement Specifications
 - This is the description of features for a new system.

Basic Requirements

- Analysts structure their investigation by seeking answers to these four major question.
 - **Understand the Process** - What is the basic business process?
 - **Identify Data Used and Information Produced** - What data are used or produced during that process?
 - **Determine Process Timing and Volume** - What are the limits forced by time and the volume of work?
 - **Identify Controls** - What performance controls are used?

Fact-Finding Techniques

- An analyst is very much like a **detective**. He or she knows that there is a problem to be solved and therefore must look for clues that uncover the solution.
- The second challenge of requirements gathering is choosing the way(s) information is collected.
- The methods used by analysts for collecting data about requirements are called Fact-Finding techniques.
- These include
 - Interview
 - Questionnaire
 - Record inspections
 - Observations

Interview

- An interview is the most commonly used requirements-gathering technique. After all, it is natural—if you need to know something, you usually ask someone.
- In general, interviews are conducted one-on-one (one interviewer and one interviewee), but sometimes, owing to time constraints, several people are interviewed at the same time.
- There are five basic steps to the interview process:
 - Selecting interviewees (employee of organization)
 - Designing interview questions (depends on system requirement)
 - Preparing for the interview (clear background of current system)
 - Conducting the interview
 - Post interview follow-up (prepare document on basis of interview)

Interview Schedule

Name	Position	Purpose of Interview	Meeting
Andria McClellan	Director, Accounting	Strategic vision for new accounting system	Mon., March 1 8:00–10:00 AM
Jennifer Draper	Manager, Accounts Receivable	Current problems with accounts receivable process; future goals	Mon., March 1 2:00–3:15 PM
Mark Goodin	Manager, Accounts Payable	Current problems with accounts payable process; future goals	Mon., March 1 4:00–5:15 PM
Anne Asher	Supervisor, Data Entry	Accounts receivable and payable processes	Wed., March 3 10:00–11:00 AM
Fernando Merce	Data Entry Clerk	Accounts receivable and payable processes	Wed., March 3 1:00–3:00 PM

- Interview can be either structured or unstructured
- **Structured Interview**
 - *Structured interviews*, in which specific sets of questions are developed before the interviews.
 - The former allows respondents to answer in their own words; the latter uses a set of arranged answer.
- **Unstructured Interview**
 - Unstructured Interview are appropriate when analysts want to acquire general information about a system.
 - This format encourages respondents to share their feelings, idea, and beliefs.

Structured Interview

Advantages

- Ensures constant wording of question for all respondents.
- Easy to administer and evaluate.
- More objective evaluation of both respondents and answers to questions
- Results in shorter interviews

Advantages

Unstructured Interview

- Interviewer has greater flexibility in wording question to suit respondent.
- Interviewer can pursue areas that arise unexpectedly during interview.
- May produce information about areas that were overlooked or not thought to be important

Disadvantages

Structured Interview

- Cost of preparation is high
- Respondents may not accept high level of structure questions
- High level of structure may not be suitable for all situations.
- High level of structure reduces respondent freedom and ability of interviewer to follow up on comments of interviewee.

Disadvantages

Unstructured Interview

- May be inefficient use of both respondent and interviewer time.
- Interviews may introduce their bias in questions or reporting results.
- Unimportant information may be gathered.
- Analysis and clarification of results may be lengthy.
- Takes extra time to collect important facts

Questionnaire

- A questionnaire is a set of written questions used to obtain information from individuals.
- Questionnaires are often used when there is a large number of people from whom information and opinions are needed.
- There are two types of interview questions:
 - Closed-ended questions
 - Open-ended questions

Closed-ended questions

- *Closed-ended questions* are those that require a specific answer.
- Closed-ended questions are used when an analyst is looking for specific, precise information.
- They are similar to **multiple-choice**.

Open-ended questions

- *Open-ended questions* are those that leave room for elaboration on the part of the interviewee.
- Open-ended questions are designed to gather rich information and give the interviewee more control over the information that is revealed during the interview.

Record-Review

- Record and report can provide analysts with valuable information about organizations and operations.
- Analysts examine information that has been recorded about the system and users.
- Record inspection can be performed at the beginning of the study as an introduction or later in the study as a basis for comparing actual operations with what the records indicate should be happening.

Observation

- Observation, the act of watching processes being performed.
- It is a powerful tool for gathering information about the as-is system because it enables the analyst to see the reality of a situation, rather than listening to others describe it in interviews.
- Several research studies have shown that many managers really do not remember how they work and how they allocate their time.
- Observation allows analysts to gain information they cannot obtain by any other fact-finding method.

What should happen

- Standard operating procedures
- Controls and checks for accuracy and completeness
- Properly completed documents
- Efficient and timely, completions of work

What actually occurs

- Delay in doing work
- Information recalled from memory [incorrectly]
- Skipped steps
- Extra photocopy needed
- New controls needed
- Information not in file-telephone calls needed
- Documents not complete as required
- Employees not aware of arranged procedures

Tools for documenting procedures and Decisions

- Tools help analysts to gathered information through the data collection methods.
- Three tools for documenting procedures
 - Decision Tree
 - Decision Table
 - Structured English

Decision Tree in system analysis

- We know that people often have different ways of saying the same thing.
- For example, the **discount conditions** discussed in the following ways

Employee 1 Response	Employee 2 Response	Discount
Greater than 10,000	Not less than 10,000	20%
greater than equal to 5,000 but less than or equal to 10,000	not more than 10,000 but at least 5,000	15%
below 5,000	not 5,000 or more	5%

- Having different ways of saying same thing can create difficulties in communication during systems studies.

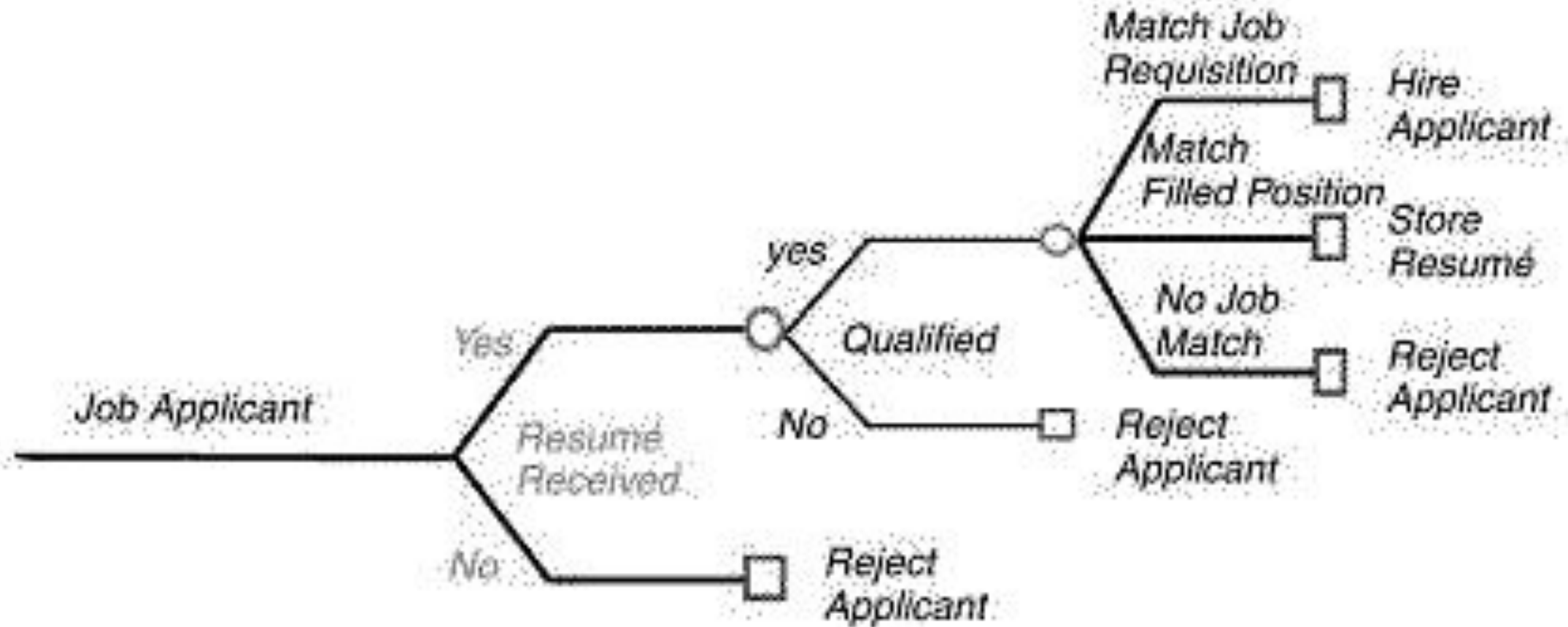
Use of Decision Tree

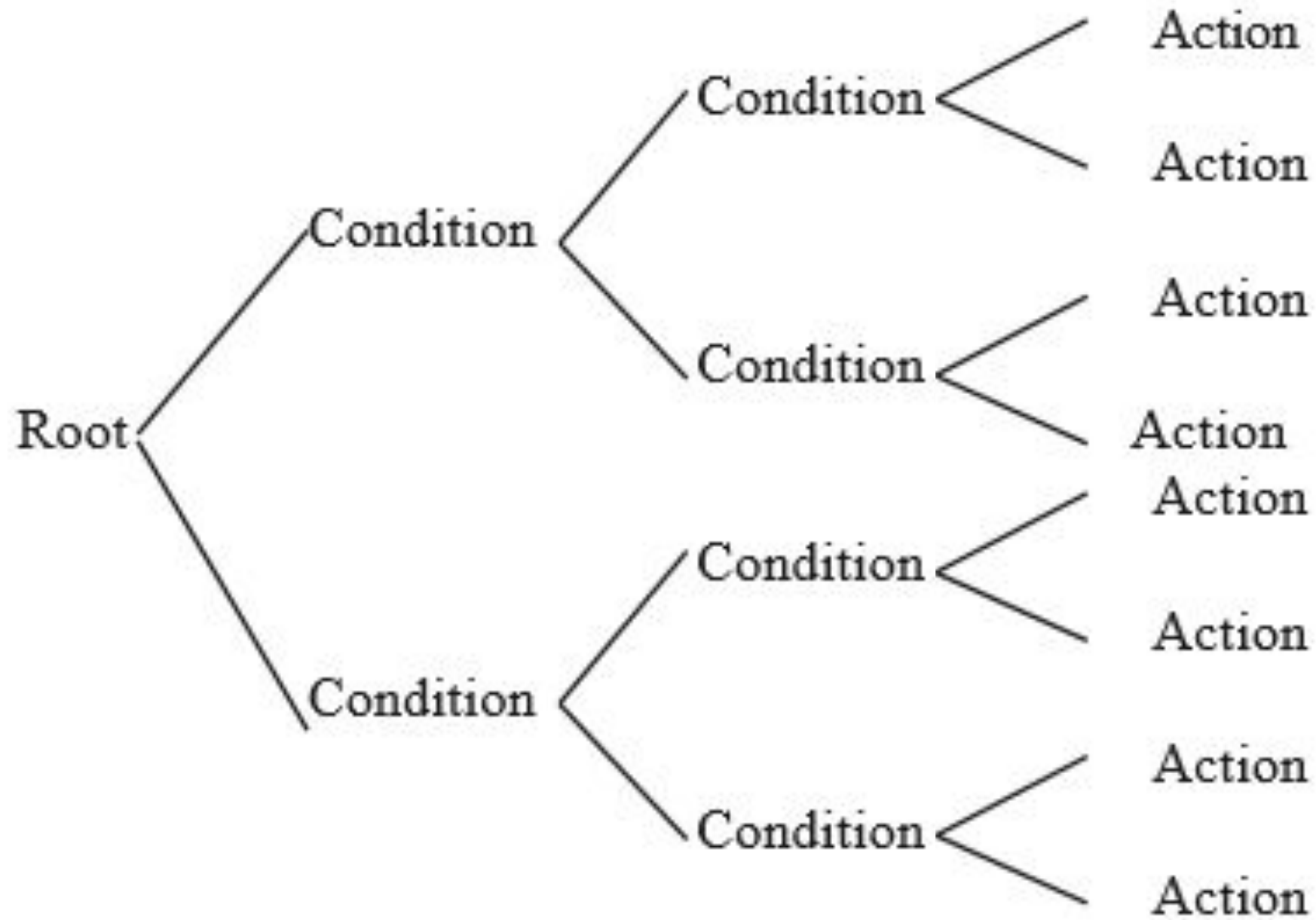
- Developing decision trees is beneficial to analysts in two ways.
 - First of all, the need to describe conditions and action forces analyst identify the actual decision that must be made.
 - Decision trees also force analysts to consider the sequence of decision.

Characteristic of Decision Tree

- A Decision Tree is a diagram that presents
 - Root, Conditions and Actions sequentially
- It shows which conditions to consider first, then second, and so on.
- It's also a method of showing the relationship of each condition and its permission action.

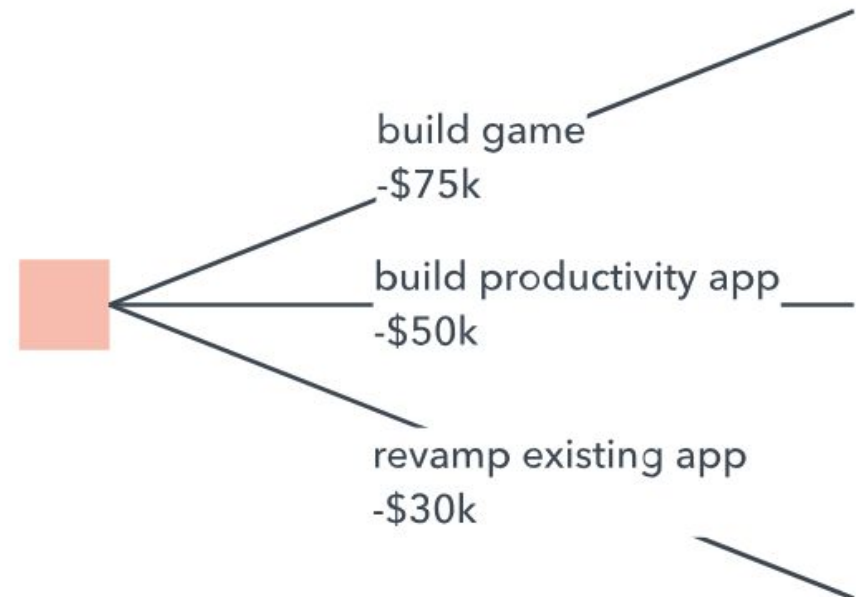
Example of Decision Tree



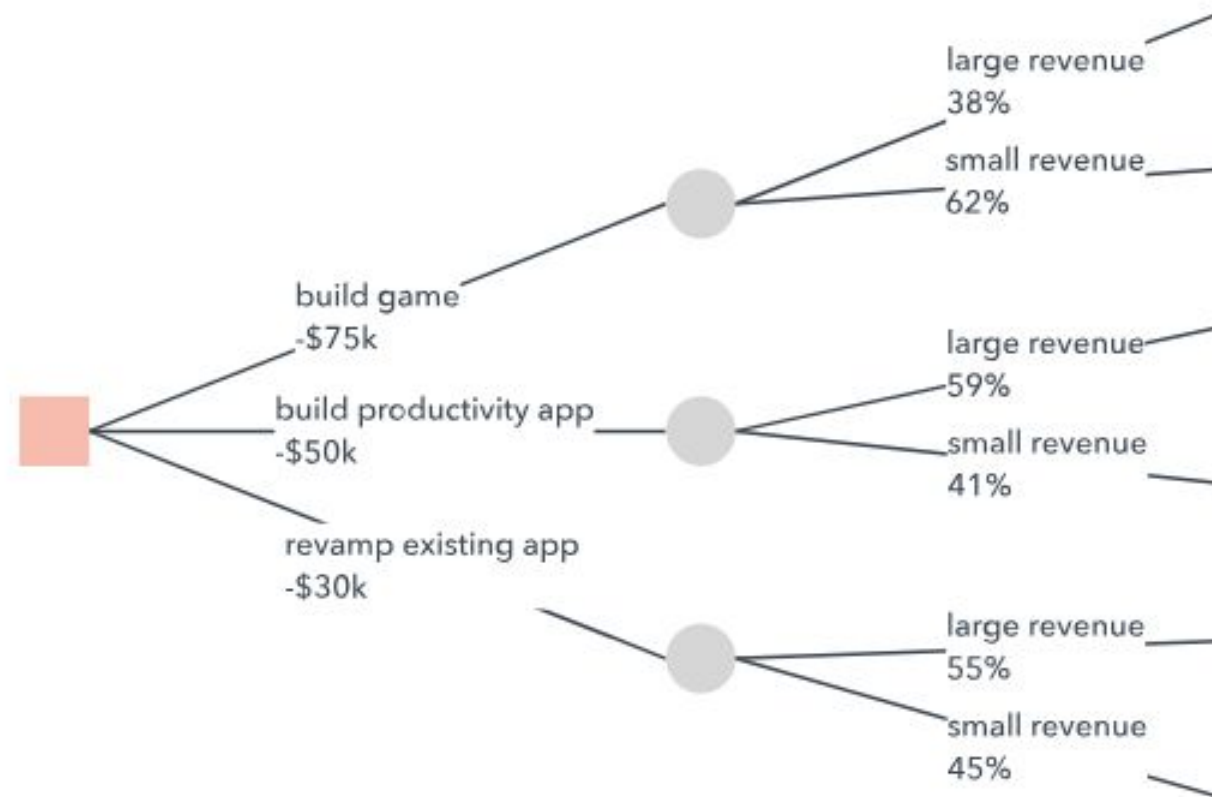


How to draw a decision tree

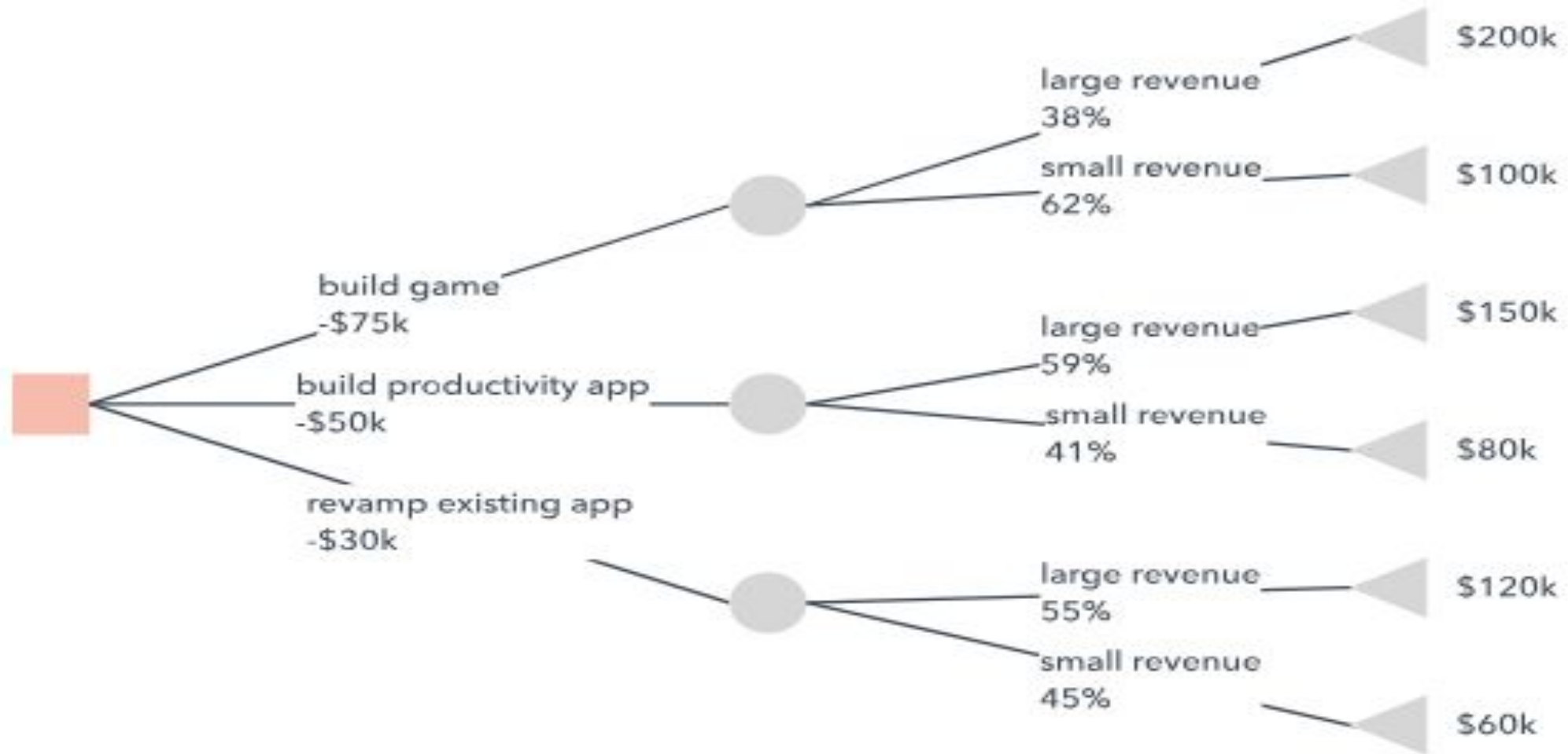
- **Start with the main decision.**
- Draw a small box to represent this point, then draw a line from the box to the right for each possible solution or action. Label them accordingly.



- **Add chance and decision nodes** to expand the tree



Continue to expand until every line reaches an endpoint



Example 1

- If payment with in 10 days and size of order is Over Rs. 10,000 than 3 % discount, Rs.5,000 to Rs. 10,000 than 2 % discount and Below Rs. 5,000 than no discount

Example 2

<u>Condition</u>	<u>Action (Grade)</u>
▫ Less than 35%	F
▫ 35% to 50%	C
▫ 50% to 60%	B
▫ 60% to 70%	A
▫ Greater than 70%	A+

1. If transaction is on credit and customer's record is good the order will be accepted but do not give any discount.
2. If transaction is on credit and customer's record is not good do not accept any order.
3. If transaction is on cash and sells amount is more than 100 rupees discount will be given 20%.
4. If transaction is on cash and sells amount is between 50 & 100 rupees than discount will be given 10%.
5. If transaction is on cash and sells amount is less than 50 rupees, order is accepted but no discount will be made.

Disadvantage Decision Tree

- Decision Tree may not always be the best tool for decision analysis.
 - A Decision Tree for a complex system with many sequences of steps and combination of condition will be confusing.
 - A large number of branches with many paths will confuse rather than help in analysis.



DECISION TABLE

Jalpa Poriya

Decision Table

- In Decision Table the condition and actions are represented in terms of rows and columns.
- A Decision Table is a matrix of rows & columns, rather than a tree that shows condition and actions.


Characteristics of Decision Table

- The Decision Table is made up of four sections, Condition Statement, Condition Entries, Action Statement & Action Entries.

Condition Stub (Condition Statment)	Condition Entries
Action Stub (Action Statment)	Action Entries

Table 1

DECISION TABLE STRUCTURE

	Decision Rule 1	Decision Rule 2	Decision Rule 3	Decision Rule 4	Decision Rule 5	Decision Rule 6																	
if																							
and							Condition Stubs	Condition Entries															
and																							
and																							
then																							
and																							
and																							

- The *condition statements* identify the relevant condition.
- *Condition entries*: which value apply for a particular condition.
- *Action statement* lists the set of all steps that can be taken when a certain condition occurs.
- *Action entries* shows what specify actions in a set to take when selected condition or combinations of condition are true.
- Sometimes notes are added below the table to indicate when to use the table of to distinguish it from other Decision Table.

- The entries in decision table are given by Decision Rules which define the relationships between combinations of conditions and courses of action. In rules section,
 - Y shows the existence of a condition.
 - N represents the condition, which is not satisfied.
 - A blank - against action states it is to be ignored.
 - X (or a check mark will do) against action states it is to be carried out.

Example:

- **Conditions**

- Is E-Mail Empty?

- Is Password Empty?

- Is E-Mail Correct?

- Is Password Correct?

- **Action**

- Error Message

- Welcome Page

Structured English

- Structured English is an additional method to overcome problems of ambiguous language in stating conditions and actions in decisions and procedures.
- This method does not use trees or tables but use narrative statements to describe procedure.
- No special symbols or formats are used, a feature that some dislike about decision trees and tables.

Developing Structured Statements

- Three basic types
 - Sequence Structures
 - Decision Structures
 - Iteration Structures

Sequence Structures

- A sequence structure is a single step or action include in a process.
- Typically several sequence instructions are used together to describe a process
- For example, to **purchase** a book in a **bookstore**
 1. Pick out a **required book**
 2. Take the book to the **checkout counter**
 3. **Pay** for the book
 4. Take a **receipt**
 5. **Leave** the store

Decision Structures

- Decision structure occur when two or more actions can be taken, depending on the value for a specific condition.
- For Example, to purchase a book in a bookstore

IF a required book is **found**, **THEN**

Take the book to the checkout counter

Pay for the book

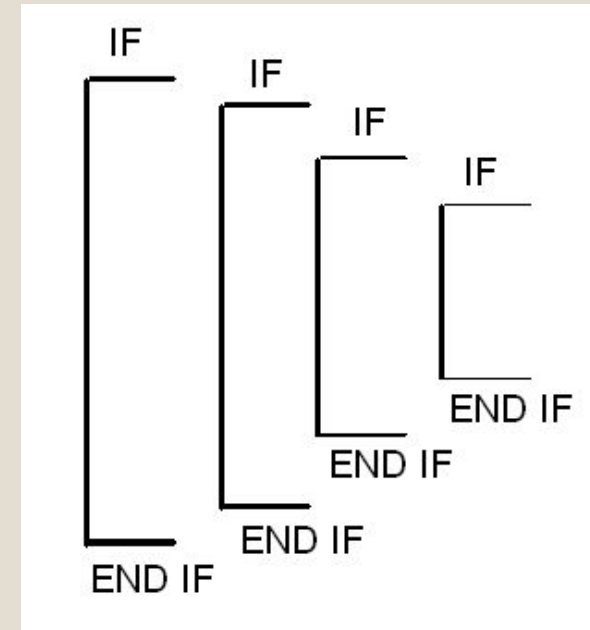
Take a receipt

Leave the store

OTHERWISE

Leave the store.

END IF



Iteration Structures

- In routine operating activities, it is common to find that certain activities are repeated while a certain condition exists or until a condition occurs.

DO WHILE still examining more books:

Read title of the book.

IF the title sound interesting THEN

pick up the book

Look at the price

IF you decide you want the book THEN

put it in the required book stack

ELSE

put it back on the shelf

ENDIF

END IF

ENDDO

DATA FLOW DIAGRAM

Structure analysis

Tools for Data Flow Strategy

- Data flow strategy shows the use of data in the system pictorially.
- Data flow analysis makes use of the following tools:
 - Data Flow Diagram
 - Data Dictionary
 - Data Structure Diagram
 - Structure Chart

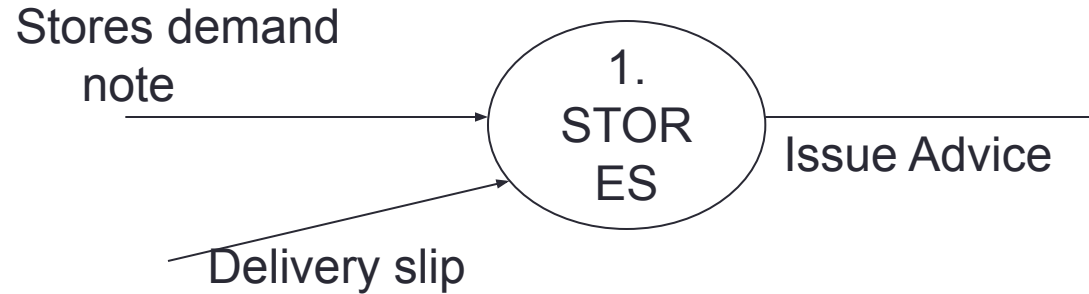
DATA FLOW DIAGRAMS

DFDs models the system by representing:

- External entities from which the data flows and where results terminate.
- Processes which transform data flows.
- Data stores from which the data are read or into which data are written by the processes.

SYMBOLS USED IN DFD

Process



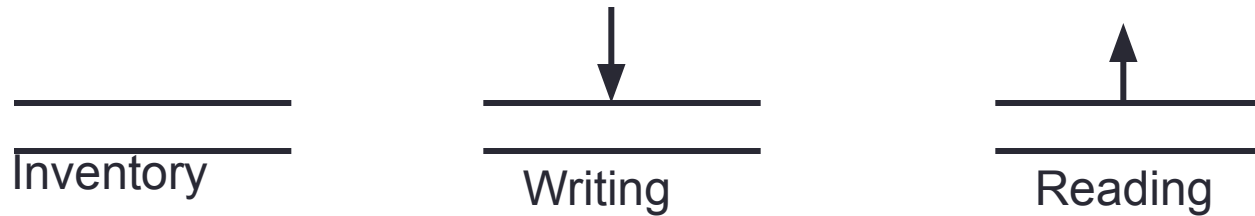
- A circle represents a process
- Straight lines with incoming arrows are input data flows
- Straight lines with outgoing arrows are output data flows
- Processes are given serial numbers for easy reference
- Labels are assigned to Data flow.
- Process should provide at least one input and one output.

EXTERNAL ENTITIES



- A Rectangle represents an external entity
- They either supply data or receive data.
- They do not process data.

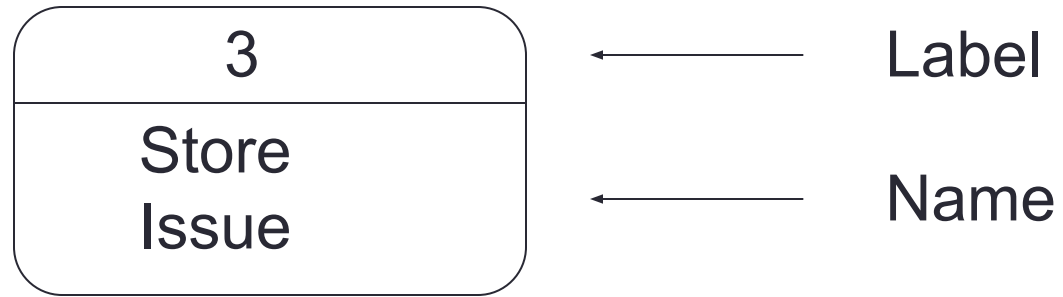
DATA STORES



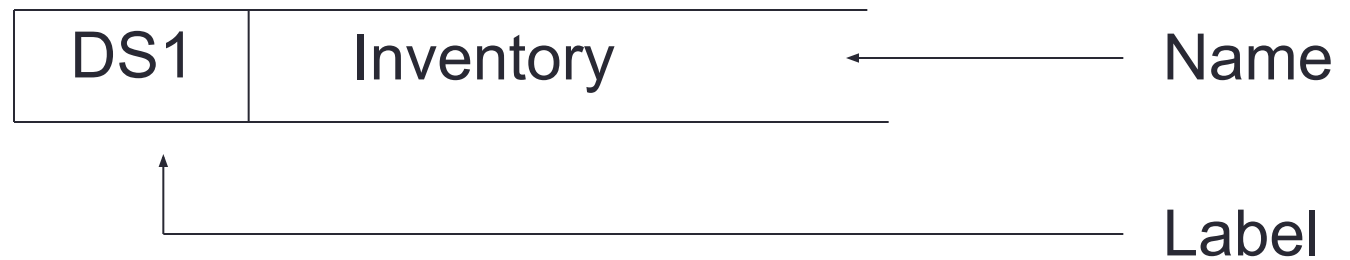
- A Data Store is a repository of data
- Data can be written into the data store. This is represented by an incoming arrow.
- Data can be read from a data store This is represented by an outgoing arrow.
- Two data stores cannot be connected by a data flow.

An alternate notation is often used

A Process



A Data store



Levels of DFD

- **Level 0 (context level DFDs)**
 - It's a basic overview of the system
- **Level 1**
 - The Level 0 DFD is broken down into more specific, Level 1 DFD.
- **Level 2**
 - At this level, DFD shows how data flows inside the modules mentioned in Level 1.

DFD Level 0 (Context Diagram)

- It's a basic overview of the whole system or process being analyzed or modeled.
- It's designed to be an **at-a-glance view**, showing the system as a **single high-level process**, with its relationship to external entities.
- **Why Context Diagram?**
- It should be easily understood by a wide audience, including stakeholders, business analysts, data analysts and developers.

DFD Level 1

- Level 1 DFD depicts **basic modules** in the system and **flow of data among various modules**.
- Level 1 DFD also mentions **basic processes** and **sources of information**.
- Some important points are:
- Level 1 DFD must balance with the context diagram it describes.
- Input going into a process are different from outputs leaving the process.
- **Data stores are first shown at this level.**

DFD Level 2

- Some important points are:
- Level 2 DFD must balance with the Level 1 it describes.
- Input going into a process are different from outputs leaving the process.
- Continue to show data stores.

Example : Hostel Mess (Canteen) System

- **Student**

- Student can update record
- Student mention that he is attending a lunch/dinner or not.
- The student can view menu and notices
- Get Bill by manager
- Payment done by students
- Feedback of mess

- **Mess manager**

- Manage student - The admin can manage student. In this module student addition, deletion, updating etc.
- Menu and price - The administrator provides options of menu and price of each meal
- Notices - Administrator can generates notices about occasion, events or in the case of mess closed.
- Calculate Bill – by student attendance

- **Mess secretary**

- Overdue bills - The mess secretary checks bills if there are bills which are not paid by student then he can take an appropriate action.
- Bill confirmation - Administrator is responsible for providing the all reports.

Modules

- Student Info
- Meal Management
- Payment Management
- Attendance

Data Store : Tables list

- Stud_Info
- Bill_Info
- Attendance
- Meal_Menu

Disadvantages

- Modification to a data layout in DFDs may cause the entire layout to be changed.
- The number of units in a DFD in a large application is high. Therefore, maintenance is harder, more costly.

Data Dictionary

- The data dictionary provides additional information about the system.
- In a data dictionary you will find a list of all the elements composing the data flowing through a system.
- Major elements are
 - Data flows
 - Data store
 - Processes

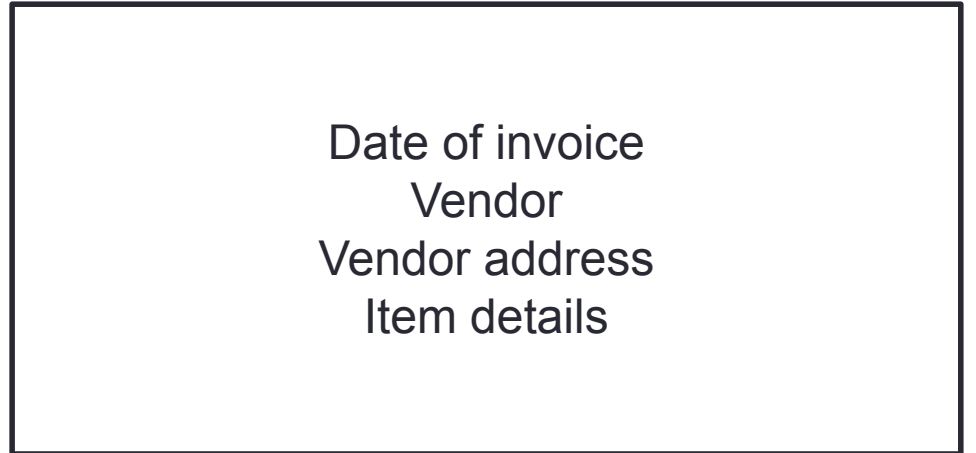
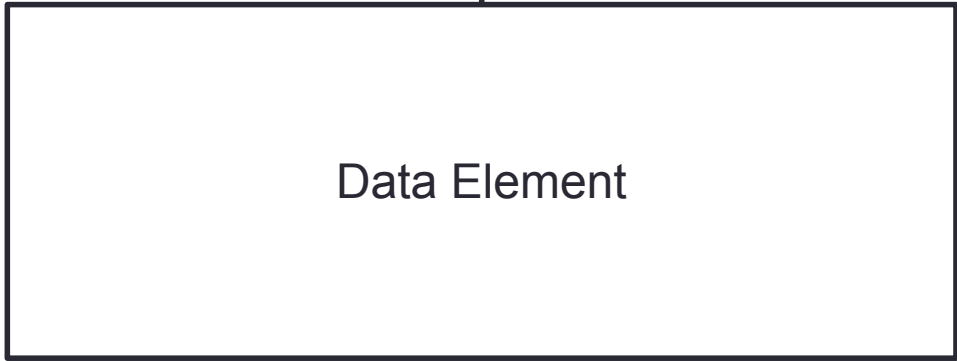
What does a data dictionary records?

- **Data element**

- The most fundamental data level is the data elements (fields, data item or elementary item).

- **Data structure**

- The data structure is a set of data item that are related to one another and that collectively describe a component in the system.



Describing Data Elements

- **Data Names :**
 - To understand items of data from another, analysts assign meaningful names.
- **Data Description :**
 - The data description briefly states what the data item represents in the system.
- **Aliases :**
 - The additional names are called aliases.
- **Length :**
 - Length identifies the number spaces (for letters, numbers, or symbols)
- **Data Value :**
 - In some processes, only specific data values are permissible.

Example

client

Field Name	Data type	Field Length	Constrain	Description
Client_id	Number	10	Primary key	Client id,Auto generated
Client_name	Varchar	20	Not null	Name of client
Password	Varchar 2	30	Not null	Login Password for client
Client_Company	Varchar	20		Name of client company
Activation_date	Date		Not null	Starting date
Contact_no	Number	15	Not null	Landline or mobile number
Email_id	Varchar2	30	Not null	Any email id
Max_Users	Number	10	Not null	Maximum number of users
Current_users	Number	10	Not null	Currently present user

Describing Data Structures

1. Sequence Relationship

- Example: FULL_NAME
 - FIRST NAME
 - MIDDLE INITIAL
 - LAST NAME

2. Selection (Either/Or) Relationship

- NUMBER

And one of the following

- MOBILE NUMBER
- LANDLINE NUMBER

3. Iteration (Repetitive) Relationship
Selection of SUBJECTS in University
4. Optional Relationship
MIDDLE_NAME in FULL_NAME

Notation used to show structure relationships in data

- **Symbol Explanation**

- = Alias
- + Concatenation
- [] Selection either/or
- { } Iteration
- () Define optional relationship that occurs only 0 or 1 times

Example

- STUDENT DATA =
FIRST NAME + (MIDDLE INITIAL) + LAST NAME
+ STREET ADDRESS + CITY + STATE
+ [MOBILE NUMBER | LENDLINE NUMBER]
+ { COURSE NUMBER + COURSE NAME + TIME + DAY +
INSTRUCTOR } + TERM + YEAR