



**SE-3002**

**SOFTWARE QUALITY ENGINEERING**

**RUBAB JAFFAR**

[RUBAB.JAFFAR@NU.EDU.PK](mailto:RUBAB.JAFFAR@NU.EDU.PK)

# **Part II-Software Testing**

Functional Testing

## **Lecture # 16, 17, 18**

## **04,05,06 Oct**

# TODAY'S OUTLINE

- Decision Table Based Testing
- Cause Effect Testing
- Pairwise Testing

# WHY DO BOTH EP AND BVA?

- If you do boundaries only, you have covered all the partitions as well
  - technically correct and may be OK if everything works correctly!
  - if the test fails, is the whole partition wrong, or is a boundary in the wrong place - have to test mid- partition anyway
  - testing only extremes may not give confidence for typical use scenarios (especially for users)
  - boundaries may be harder (more costly) to set up

# DECISION TABLE BASED TESTING

- In Software Engineering, boundary value and equivalent partition are other similar techniques used to ensure better coverage.
- They are used if the system shows the **same** behavior for a large set of inputs.
- However, in a system where for each set of input values the system behavior is **different**, boundary value and equivalent partitioning technique are not effective in ensuring good test coverage.
- In this case, decision table testing is a good option. This technique can make sure of good coverage, and the representation is simple so that it is easy to interpret and use.
- This table can be used as the reference for the requirement and for the functionality development since it is easy to understand and cover all the combinations.

# DECISION TABLE BASED TESTING

- Decision tables are used in many engineering disciplines to represent complex logical relationships.
- An output may be dependent on many input conditions and decision tables give a pictorial view of various combinations of input conditions.
- There are four portions of the decision table. The decision table provides a set of conditions and their corresponding actions.

		Decision table	
		Stubs	Entries
Condition	$c_1$		
	$c_2$		
	$c_3$		
Action	$a_1$		
	$a_2$		
	$a_3$		
	$a_4$		

## Four Portions

1. Condition Stubs
2. Condition Entries
3. Action Stubs
4. Action Entries

# PARTS OF THE DECISION TABLE

- The four parts of the decision table are given as:
- Condition Stubs: All the conditions are represented in this upper left section of the decision table. These conditions are used to determine a particular action or set of actions.
- Action Stubs: All possible actions are listed in this lower left portion of the decision table.
- Condition Entries: In the condition entries portion of the decision table, we have a number of columns and each column represents a rule. Values entered in this upper right portion of the table are known as inputs.
- Action Entries: Each entry in the action entries portion has some associated action or set of actions in this lower right portion of the table. These values are known as outputs and are dependent upon the functionality of the program.

# TYPICAL STRUCTURE OF DECISION TABLE

Typical structure of a decision table				
Stubs	$R_1$	$R_2$	$R_3$	$R_4$
$c_1$	F	T	T	T
$c_2$	-	F	T	T
$c_3$	-	-	F	T
$a_1$	X	X		X
$a_2$			X	
$a_3$	X			

## DECISION TABLE FOR A LOGIN SCREEN.

- The condition is simple if the user provides correct username and password the user will be redirected to the homepage. If any of the input is wrong, an error message will be displayed.
- Case 1 – Username and password both were wrong. The user is shown an error message.
- Case 2 – Username was correct, but the password was wrong. The user is shown an error message.
- Case 3 – Username was wrong, but the password was correct. The user is shown an error message.
- Case 4 – Username and password both were correct, and the user navigated to homepage





Conditions	Rule 1	Rule 2	Rule 3	Rule 4
Username (T/F)	F	T	F	T
Password (T/F)	F	F	T	T
Output (E/H)	E	E	E	H

## EXAMPLE 2: DECISION TABLE FOR UPLOAD SCREEN

- Develop a decision table for an application that allows a user to upload an image. For uploading an image, application opens a dialogue box which will ask the user to upload photo with certain conditions like –
- You can upload only '.jpg' format image
- file size less than 32kb
- resolution 137\*177.

# DECISION TABLE FOR UPLOAD SCREEN

Conditions	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Case 7	Case 8
Format	.jpg	.jpg	.jpg	.jpg	Not .jpg	Not .jpg	Not .jpg	Not .jpg
Size	Less than 32kb	Less than 32kb	>= 32kb	>= 32kb	Less than 32kb	Less than 32kb	>= 32kb	>= 32kb
resolution	137*177	Not 137*177	137*177	Not 137*177	137*177	Not 137*177	137*177	Not 137*177
Output	Photo uploaded	Error message resolution mismatch	Error message size mismatch	Error message size and resolution mismatch	Error message for format mismatch	Error message format and resolution mismatch	Error message for format and size mismatch	Error message for format, size, and resolution mismatch

# APPLICABILITY

- Decision tables are popular in circumstances where an output is dependent on many conditions and a large number of decisions are required to be taken.
- They may also incorporate complex business rules and use them to design test cases.
- Every column of the decision table generates a test case.
- As the size of the program increases, handling of decision tables becomes difficult and cumbersome.
- In practice, they can be applied easily at unit level only. System testing and integration testing may not find its effective applications.

# APPLICABILITY

- It's a tabular representation of input conditions and resulting actions. Additionally, it shows the causes and effects. Therefore, this technique is also called a ***cause-effect table***.
- Testing combinations can be a challenge, especially if the number of combinations is enormous. Moreover, testing all combinations is not practically feasible as it's not cost and time effective. Therefore, we have to be satisfied with testing just a small subset of combinations. That is to say, the success of this technique depends on our choice of combinations

Dated: September 20, 2022

Subject: Policy # ST-037 Selteq Solutions (SMC-Private) Limited

**Selteq Targets Evaluation Fines Policy**

We, Selteq Solutions, are always keen to keep committed employees happy and satisfied. There are target bonuses, performance bonuses, Extra Working Day and Salary Booster Policy to appreciate the hard work. Along with appreciation, it is very important to identify the weakest link of Team and boost his/her performance. Therefore, we have introduced target and performance fines. Aim of this policy is to meet the time lines and increase productivity along with efficiency and accuracy. Please find the conditions and fines below:

- Developers shall use separate branch to different task, breach of this clause shall cause fine of Rs. 10,000 for not using the different branch. If these tasks are not completed on time, it will further charge Rs. 10,000 fines.
- Daily task shall be completed on assigned time (pre-agreed deadlines). If tasks are delayed, it will cause Rs. 10,000 fine every day.
- For Quality Assurance department specifically, they shall test any module completely before approving.
- If there are any minor bugs found in the module after approval from QA Department, it will cause fine of Rs. 1,000 each minor bug.
- If there are any major bugs/issues found in the module after approval from QA Department, it will cause fine of Rs. 10,000 per major issue.

The above-mentioned policy has been duly approved, agreed and signed by the Manager of Operations and Director of the Company.

**Raheem Dad**  
CEO

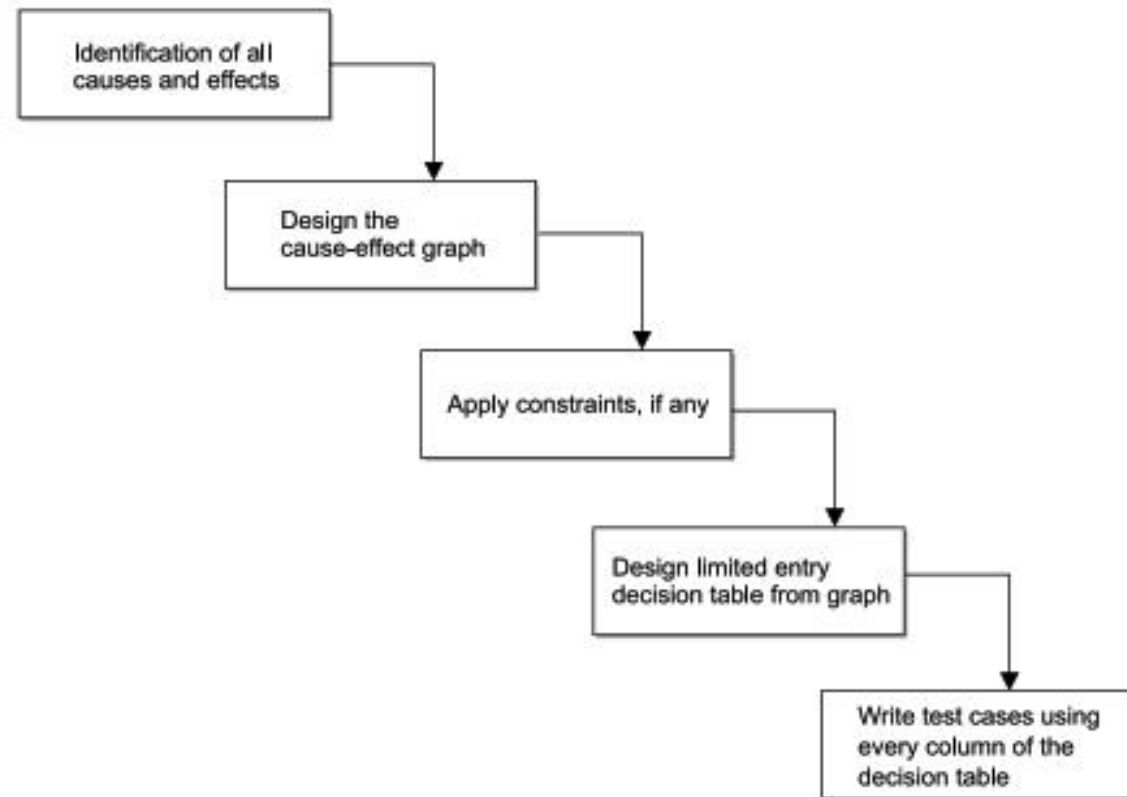
Name: Mr. Raheem Dad  
Designation: CEO

**Selteq Solutions**  
(SMC-Private) Ltd  
Evacuee Trust Complex  
F-5/1, Islamabad

Name: Mr. Saad Bin Masood  
Designation: Manager Operations

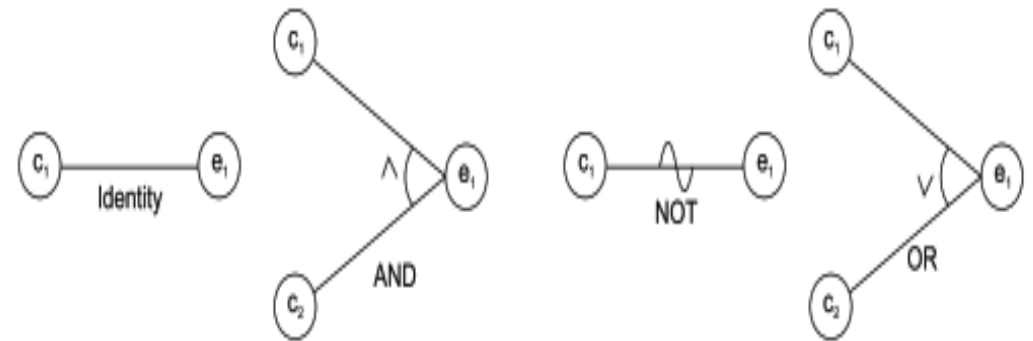
# CAUSE-EFFECT GRAPHING TECHNIQUE

- Popular technique for small programs and considers the combinations of various inputs.
- Two terms: **Causes and Effects**, which are nothing but inputs and outputs respectively.
- SRS document is used for the identification of causes and effects.
- A list is prepared for all causes and effects.



# DESIGN OF CAUSE-EFFECT GRAPH

- Each node represents either true or false state and may be assigned 1 and 0 value respectively.



Basic notations used in cause-effect graph

- (a) **Identity:** This function states that if  $c_1$  is 1, then  $e_1$  is 1; else  $e_1$  is 0.
- (b) **NOT:** This function states that if  $c_1$  is 1, then  $e_1$  is 0; else  $e_1$  is 1.
- (c) **AND:** This function states that if both  $c_1$  and  $c_2$  are 1, then  $e_1$  is 1; else  $e_1$  is 0.
- (d) **OR:** This function states that if either  $c_1$  or  $c_2$  is 1, then  $e_1$  is 1; else  $e_1$  is 0.

The AND and OR functions are allowed to have any number of inputs.



# USE OF CONSTRAINTS IN CAUSE-EFFECT GRAPH

(a) **Exclusive**

The Exclusive (E) constraint states that at most one of  $c_1$  or  $c_2$  can be 1 ( $c_1$  or  $c_2$  cannot be 1 simultaneously). However, both  $c_1$  and  $c_2$  can be 0 simultaneously.

(b) **Inclusive**

The Inclusive (I) constraint states that at least one of  $c_1$  or  $c_2$  must always be 1. Hence, both cannot be 0 simultaneously. However, both can be 1.

(c) **One and Only One**

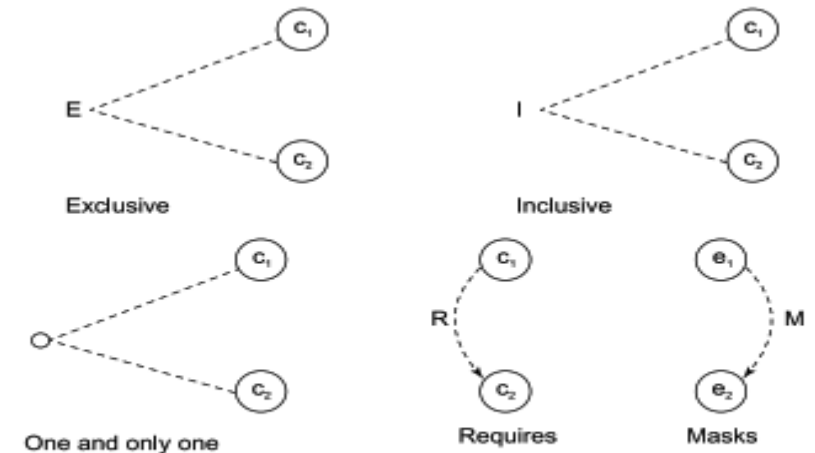
The one and only one (O) constraint states that one and only one of  $c_1$  and  $c_2$  must be 1.

(d) **Requires**

The requires (R) constraint states that for  $c_1$  to be 1,  $c_2$  must be 1; it is impossible for  $c_1$  to be 1 if  $c_2$  is 0.

(e) **Mask**

This constraint is applicable at the effect side of the cause-effect graph. This states that if effect  $c_1$  is 1, effect  $c_2$  is forced to be 0.



Constraint symbols for any cause-effect graph

# EXAMPLE

- Consider the example of keeping the record of marital status and number of children of a citizen. The value of marital status must be 'U' or 'M'. The value of the number of children must be digit or null in case a citizen is unmarried. If the information entered by the user is correct then an update is made. If the value of marital status of the citizen is incorrect, then the error message 1 is issued. Similarly, if the value of number of children is incorrect, then the error message 2 is issued.

causes are:

$c_1$ : marital status is 'U'

$c_2$ : marital status is 'M'

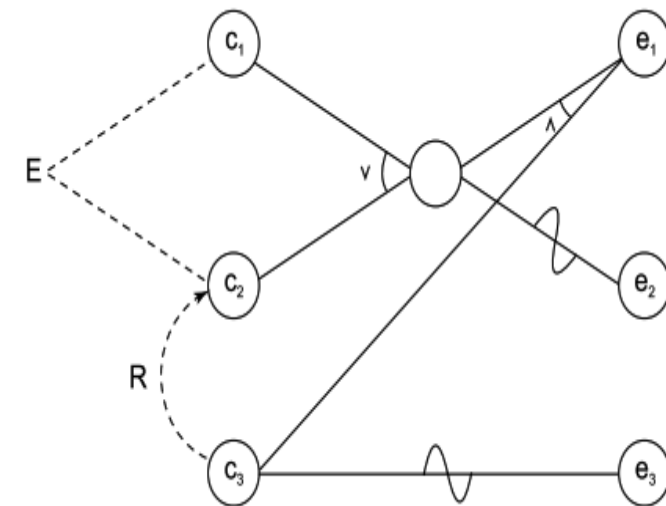
$c_3$ : number of children is a digit

effects are:

$e_1$ : updation made

$e_2$ : error message 1 is issued

$e_3$ : error message 2 is issued



# SUMMARY OF THE STEPS

- Draw the circles for effects and Causes.
- Start from effect and then pick up what is the cause of this effect.
- Draw mutually exclusive causes (exclusive causes which are directly connected via one effect and one cause) at last.
- Use logic gates to draw dynamic test cases.

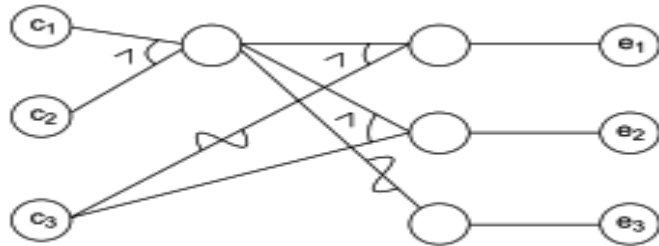
# APPLICABILITY

- Cause-effect graphing considers dependency of inputs using some constraints.
- Effective only for small programs because, as the size of the program increases, the number of causes and effects also increases and thus complexity of the cause-effect graph increases.
- For large-sized programs, a tool may help us to design the cause-effect graph with the minimum possible complexity.
- Limited applications in unit testing and hardly any application in integration testing and system testing.

## EXAMPLE

- A tourist of age greater than 21 years and having a clean driving record is supplied a rental car. A premium amount is also charged if the tourist is on business, otherwise it is not charged. If the tourist is less than 21 year old, or does not have a clean driving record, the system will display the following message: “Car cannot be supplied”
- Draw the cause-effect graph and generate test cases.

# SOLUTION



**Figure 2.15.** Cause-effect graph of rental car problem

**Table 2.45.** Decision table of rental car problem

Conditions	1	2	3	4
$c_1$ : Over 21 ?	F	T	T	T
$c_2$ : Driving record clean ?	-	F	T	T
$c_3$ : On Business ?	-	-	F	T
$e_1$ : Supply a rental car without premium charge			X	
$e_2$ : Supply a rental car with premium charge				X
$e_3$ : Car cannot be supplied	X	X		

**Table 2.46.** Test cases of the given decision table

Test Case	Age	Driving_record_clean	On_business	Expected Output
1.	20	Yes	Yes	Car cannot be supplied
2.	26	No	Yes	Car cannot be supplied
3.	62	Yes	No	Supply a rental car without premium charge
4.	62	Yes	Yes	Supply a rental car with premium charge.



That is all