Software Testing Help

Help & Tutorials

- QA Training
- Selenium Tutorials
- QTP Tutorials
- ISTQB Study Guide
- ISTQB Premium Study Guide
- QC Tutorials
- HP LoadRunner Tutorials
- JMeter Tutorials
- JIRA Tutorials
- VBScript Tutorials
- Best Test Management Tools
- Appium Studio Tutorials
- Unix Tutorials
- 101+ Interview Questions

Online Training by STH

Cause and Effect Graph – Dynamic Test Case Writing Technique For Maximum Coverage with Fewer Test Cases

Posted In | Quality Assurance, Testing Concepts, Testing Methodologies | Last Updated: "June 7, 2018"

> **What You Will Learn:** [show]

Dynamic Testing Techniques – Cause and Effect Graph.

Test case writing forms an integral part of testing, whether it is manual or automation. Every project is exclusive and has a <n> number of test conditions that need to be covered.

We should focus on two points whenever we write test cases. These are:
Mitigating the risk
Coverage.

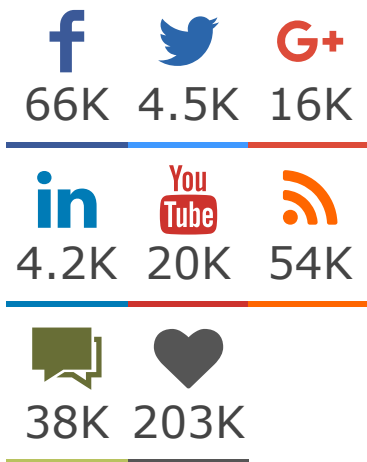This paper revolves around the second point which is "Coverage". To be precise – Requirements Coverage.

Test case writing techniques for dynamic testing

1. Equivalence partitioning
2. Boundary Value Analysis
3. Decision table
4. Cause and effect graph technique
5. State transition diagram
6. Orthogonal array testing(OATS)
7. Error guessing.

We have some really good papers 1st, 2nd and 3rd points (Equivalence portioning, BVA and decision tables) here in STH. I am going to discuss point 4 which is Cause and effect graph.

Introduction

Cause and effect graph is a dynamic test case writing technique. Here causes are the input conditions and effects are the results of those input conditions.

Cause-Effect Graphing is a technique which starts with a set of requirements and determines the minimum possible test cases for maximum test coverage which reduces test execution time and ultimately cost.

The goal is to reduce the total number of test cases still achieving the desired application quality by covering the necessary test cases for maximum coverage.

But at the same time obviously, there are some downsides of using this test case writing technique. It takes time to model all your requirements into this cause-effect graph before writing test cases.

The Cause-Effect graph technique restates the requirements specification in terms of the logical relationship between the input and output conditions. Since it is logical, it is obvious to use Boolean operators like AND, OR and NOT.

Notations we are going to use:

AND – For effect E1 to be true, both the causes C1 and C2 should be true
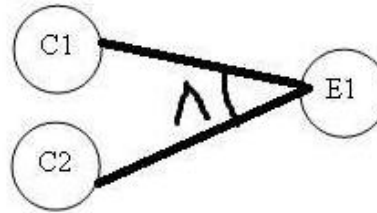
OR – For effect E1 to be true, either of causes C1 OR C2 should be true
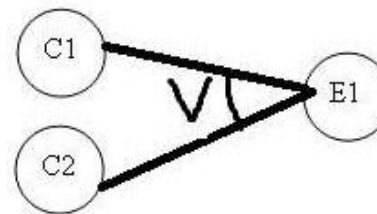
NOT – For Effect E1 to be True, Cause C1 should be false
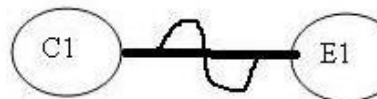
MUTUALLY EXCLUSIVE – When only one of the causes will hold true.

Copyright - SoftwareTestingHelp.com

Now let's try to implement this technique with some example.

1. Draw a cause and effect graph based on a requirement/situation
2. Cause and Effect graph is given, draw a decision table based on it to draw the test case.

Let's see both of them one by one.

Let's draw a cause and effect graph based on a situation

***Situation*:**

The "Print message" is software that read two characters and, depending on their values, messages must be printed.

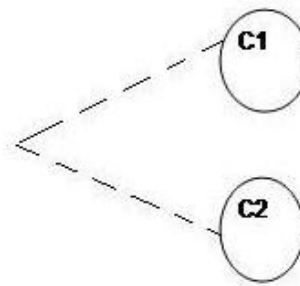- The first character must be an "A" or a "B".
- The second character must be a digit.
- If the first character is an "A" or "B" and the second character is a digit, the file must be updated.
- If the first character is incorrect (not an "A" or "B"), the message X must be printed.
- If the second character is incorrect (not a digit), the message Y must be printed.

***Solution*:**

The causes for this situation are:
C1 – First character is A
C2 – First character is B
C3 – the Second character is a digit

The effects (results) for this situation are
E1 – Update the file
E2 – Print message "X"
E3 – Print message "Y"

LET'S START!!

First, draw the causes and effects as shown below:

Key – Always go from effect to cause (left to right). That means, to get effect "E", what causes should be true.

In this example, let's start with Effect E1.

Effect E1 is to update the file. The file is updated when
– The first character is "A" and the second character is a digit
– The first character is "B" and the second character is a digit
– The first character can either be "A" or "B" and cannot be both.

Now let's put these 3 points in symbolic form:

For E1 to be true – following are the causes:
– C1 and C3 should be true
– C2 and C3 should be true
– C1 and C2 cannot be true together. This means C1 and C2 are mutually exclusive.

Now let's draw this:

So as per the above diagram, for E1 to be true the condition is
(C1   C2)   C3

The circle in the middle is just an interpretation of the middle point to make the graph less messy.
There is a third condition where C1 and C2 are mutually exclusive. So the final graph for effect E1 to be true is shown below:

Let's move to Effect E2:

E2 states to print message "X". Message X will be printed when the
First character is neither A nor B.
Which means Effect E2 will hold true when either C1 OR C2 is invalid.
So the graph for Effect E2 is shown as (In blue line)

For Effect E3.
E3 states to print message "Y". Message Y will be printed when Second character is incorrect.
Which means Effect E3 will hold true when C3 is invalid. So the graph for Effect E3 is shown as (In Green line)

This completes the Cause and Effect graph for the above situation.

Now let's move to draw the **Decision table based on the above graph**.

Writing Decision table based on Cause and Effect graph

First, write down the Causes and Effects in a single column shown below

Key is the same. Go from bottom to top which means traverse from effect to cause.

Start with Effect E1. For E1 to be true, the condition is (C1   C2) C3.
Here we are representing True as **1** and False as **0**

First, put Effect E1 as True in the next column as

Now for E1 to be "1" (true), we have the below two conditions –
C1 AND C3 will be true
C2 AND C3 will be true

For E2 to be True, either C1 or C2 has to be false shown as

For E3 to be true, C3 should be false.

So it's done. Let's complete the graph by adding 0 in the blank column and including the test case identifier.

Writing Test cases from the decision table

I am writing a sample test case for test case 1 (TC1) and Test Case 2 (TC2).

In a similar fashion, you can create other test cases.

(A test case contains many other attributes like preconditions, test data, severity, priority, build, version, release, environment etc. I assume all these attributes to be included when you write the test cases in actual situation)

Conclusion

Summarizing the steps once again:

1. Draw the circles for Causes and Graphs
2. Start from effects and move towards the cause.
3. Look for mutually exclusive causes.
   This finishes the Cause and Effect graph dynamic test case writing technique. We have seen how to draw the graph and how to draw the decision table based on it. The final step of writing test cases based on decision table is comparatively easy.

About the Author: This is a guest article by Shilpa Chatterjee Roy. She is working in software testing field for the past 8.5 years in various domains.

Feel free to discuss your test case writing methods in comments below.

PREV Tutorial | **NEXT Tutorial**

f Share **450**   in Share **17**   G+ +1 **14**   🐦 Tweet   **519** SHARES

← **PREV POST** | **NEXT POST** →

Get Premium eBook For Free

Maximize Your Chances of Getting Freelance QA Testing Work

Freelance Software Testing Job Opportunities

Work from Home Jobs for the Software Testers to Earn
Extra Income!

Enter your email address here...

Yes, Send Me the FREE eBook!

37 comments ↓

#1 **Venkata Vandrangi**
This is good article. Also, the tips/tricks/trades being published are very good, informative for testing skills enhancement. I always recomment my team to go through these. Thank you.

#2 **syed**
nice explanation. recently we are engaged in exploratory testing more. but something like this with proven QA methods will certainly help.

#3 **Ibaa Hanbali**
I didn't understand the following step:
For E3 to be true, C3 should be false.
I thought you will fill table with zero values for C3 and "one " value for E3 but instead you filled 1s for C1 and C2 and left C3 blank?

#4 **Jafar**
This look a complex. This needs to be better defined.

#5 **Sandhya**
Hi Vijay,

This article is good enough , This article is helpful for the people who are going to attend ISTQB.

Thanks for this nice article

#6 **Charles**
This is very much helpful. Thank you.

#7 **kiran**
This is really best explanation. I have cleared all doubts . thank you very much…..

#8 **kiran**

This is really best explanation about test cases writing with Cause
Effect graph testing.
Thank you..

#9 **aruna**
nice explanation
can you please explain about test pre conditions…?????

#10 **aruna**
thanks for the detailed explanation
can you please explain about test pre conditions…?????

#11 **Shilpa Chatterjee Roy**
Thank you All!

@Aruna

Test condition are the pre-requisites or assumptions for executing a
test case. To explain with some examples:

1. You want to test a certain section of a web application for a certain
set of user – so in this case the precondition can be- ” take a user
having permission”

2. For validating a student registration, the precondition can be ”
student age is > 18”

In a similar fashion, you can define preconditions for testing negative
scenario also.

Precondition also forms a basis for creating / requesting test data as
well.

#12 **santhosh**
Hi Shilpa,

I had the same confusion what Ibaa Hanbali has posted

Could you please elaborate?

“For E3 to be true, C3 should be false.
I thought you will fill table with zero values for C3 and “one ” value for

E3 but instead you filled 1s for C1 and C2 and left C3 blank?"

#13 **Shilpa Chatterjee Roy**
Hi Santosh / Ibaa Hanbali

Test case ID for E3 to be true are TC5 and TC6
Here E3 is true, so it is 1 for both TC5 and TC6
C3 is false, so it is 0 for both TC5 and TC6.
Now we are left with 2 causes – C1 and C2 for TC5 and TC6
so in TC5, I have kept C1 as true (1) and in TC6, I have kept C2 as
true (1)

#14 **Esraa Eldeeb**
Excellent article , but i still can't understand in case E3 is true , why
put 1 in C1 / TC5 and in C2/TC6

#15 **Veeru**
I didt not understand the table for E2 and E3.
Can you pls explain bit more..

Thanks in advance

#16 **mahesh**
I am fresher, I am searching sample testcases, if any body have,
please send to maheshjutti@gmail.com,

#17 **awesome quote**
HI shilpa i did't understand for E2 to be true for tc3.. c1 should be
false and c3 to be true na.. bt y c3=0 in tc3.. please im bit confused
pls explain me

#18 **Selvakumar V**
Hi i am fresher .I am also searching sample test cases, if you have
anybody, please send to selva.kumarvlss@gmail.com . Eagerly
waiting.

#19 **Joey**
Hi Shilpa,

This looks like a good way of breaking down requirements. Can you
explain what the '\/' and the '/\' mean in the diagrams?

This is the first time I have seen this notation. Does it mean "OR" and "AND" respectively?

Regards,
Joey

#20 **rohit**
Thanks a lot, this was a very helpful piece of info.
Esp. It took no time to understand it… !

#21 **rishabh**
in the decision table in construction the value of c3 taken is 0 which is wrong. It shpuld be 1.

#22 **colleen**
Hi, I try to this please comment:
I put – in decision for data that does not affect the effect (result)
tc1 tc2 tc3 tc4 tc5 tc6
c1 1 – 0 – 1 –
c2 – 1 – 0 – 1
c3 1 1 0 1 0 0
e1 1 1 – – – –
e2 – – 1 1 – –
e3 – – – – 1 1

and the test case for TC3 to TC6:

TC3
TC3_Print_Msg_X

validate that system print msg X when first character is neither A nor B

1. Open the application
2. Enter 1st character as J
3. Enter 2nd character as h

print msg X

-=-=-=-=-=-=
TC4

TC4_Print_Msg_X

validate that system print msg X when first character is neither A nor
B

1. Open the application
2. Enter 1st character as J
3. Enter 2nd character as 8

print msg X

-=-=-=-=-=-=
TC5
TC5_Print_Msg_Y
validate that system print msg Y when 2nd character is a not digit

1. Open the application
2. Enter 1st character as A
3. Enter 2nd character as w

print msg Y

-=-=-=-=-=-=
TC6
TC6_Print_Msg_Y
validate that system print msg Y when 2nd character is a not digit

1. Open the application
2. Enter 1st character as B
3. Enter 2nd character as w

print msg Y

#23 **leena**
Very clear & detailed explanation.Thank you so much for your best
explanation.

#24 **Dr M. Prasanna**
Nice Illustration. Very useful for the students to understand the cause
and effect technique in a lucid way.

#25 **SANI**
Explanation Better to understand as compared to other examples and material over internet.

#26 **Pallavi**
Thank you…

#27 **sharon**
nicely explained but 1 thing that is not clear is that in the 3rd n 4th test cases u have given c3=0 whereas in d que its written for msg X to b print 1st char is not a A or B so test case should be c1=0 c3=1 => e2=1; c2=0 c3=1 => e2=1.
and in 5th n 6th Message Y will be printed when Second character is incorrect i.e c3=0 and here u have considered c1 and c2.
whereas in 3rd n 4th TC u have not considered c3..
pls explain

#28 **Avi**
Thanks for the informative papers. However, I could not help but discover several typos and mistakes in most papers on this website. I understand that all the papers are written by several different people. But it should not be very hard to review the work before publishing. This is after all about quality assurance.

For instance half way through the paper, we were advised to work from left to right going from effects to causes . However, no where in the entire paper are effects mentioned on the left side. I do not want to come off as nitpicking minor details. But if we are going to talk about quality assurance, I believe we should start with some good quality standards.

Having said that in the above paper I did not understand why anyone has to go through so much work to develop a decision table. It appears that a lot of time could have been saved by going straight to the decision table using the conditions listed right at the beginning. May be it is a useful technique when working with complex applications. But I am unable to see the value of this technique for the given example. No explanation is given for the original premise for using this technique – that we would be able to cover more requirements with lesser number of test cases by using this technique.

#29 **Cocoa**

I think there is something wrong on decision table,if C1 C2 C3 0 0 0
,then the result should be 0 1 1 ,

#30 Ravella Baji
It really help full one
Thank you sir !?????

#31 Ekansh Gautam
The article is quite informative.Can you please tell me any software
that can be used to generate cause effect flow graph?

#32 Amrita
Thank You for this superb example.. explanation Better to understand
as compared to other examples and i followed in my
presentation..everyone understood easily. Thank You so much.. :)

#33 v
sd

#34 d
hdhdfhfdyhfhbhc

#35 bdfhd
jrtutfmkuty

#36 Anders
I agree with Cocoa. Test Case 3 is wrong, E3 should also be 1 there –
Thus the program should write both message X and message Y.

Test Case 3
IF INPUT
C1 = 0
C2 = 0
C3 = 0
THEN OUTPUT
E1 = (C1 OR C2) AND C3 = (0 OR 0) AND 0 = 0 AND 0 = 0
E2 = NOT(C1 OR C2) = NOT(0 OR 0) = NOT(0) = 1
E3 = NOT(C3) = NOT(0) = 1
SO
E1 = 0
E2 = 1
E3 = 1

#37 **Muhammad Usman**
The table is wrong.

Leave a Comment

Name*

Email*

Website (Optional)

Submit

Search

About us | Contact us | Advertise | Testing Services

All articles are copyrighted and can not be reproduced without permission.

© 2006 - 2018 Software Testing Help — Read our Copyright Policy | Privacy Policy | Terms | Cookie Policy | Link to Us

Thanks For Reading! 50 Million Visits and Counting!