



Lesson Objectives

- Decision Table Test
- The coverage depth of DTT
- Elementary Comparison Test
- The steps involved in ECT
- Variation and Depth analysis of ECT



9.1 Decision Table Test

Introduction

- Thorough technique for the testing of detail functionality. This technique is aimed at the thorough coverage of the conditions and not at combining the functional paths.
- The basic technique used here is:
 - Decision points: multiple condition coverage
- The variations on the decision table test can be created by applying the basic technique:
 - Decision points: condition coverage, decision coverage or condition\decision coverage. With these the decision points can be tested in less depth.
 - Boundary value analysis: with this, the possibilities of a condition can be tested in more depth.
- Quality characteristics
 - Functionality
 - Data controllability

It is a preferred technique for the testing for the testing of functions and/or complex calculations considered to be very important.

9.2 Generic Steps

- Identify test situations
- Specify logical test cases
- Create physical test cases
- Establish the initial data set
- Assemble the test script

9.3 Points of focus

- Taking generic steps as a starting point we will consider this technique step-by-step: the general shape of a decision table after the step 2 i.e. creating logical test cases would be:

Test situations	1	2	...	n
Condition 1	0	0	...	1
Condition 2	0	~	...	0
Condition	0	0
Condition n	0	1	...	0
Results:				
Result 1	x
Result...
Result n
Not possible	x

9.4 Example Situation

- When ordering coffee capsules via the internet, the shipping costs are calculated. These consist of the standard shipping costs, plus a long-distance supplement. The process is:
- Shipping costs calculation:
 - Calculation of standard shipping costs
 - If 200 or more capsules are ordered and if the form of payment is “direct debit”, then no shipping cost is applied. If fewer than 200 capsules are ordered , or if the form of payment is other than “direct debit” then a shipping cost of € 10 is applied.
 - Calculation of long distance supplement
 - If the delivery address for the capsules is within a radius of 50 km of Utrecht, no long distance supplement is applied. If the delivery address is 50 km or more from Utrecht , but still in Netherlands, then a long distance supplement of €5 is applied. If the delivery address is outside the Netherlands, then a long distance supplement of €15 is applied.

9.4 Example

- We will underline the conditions:
- Shipping costs calculation:
 - Calculation of standard shipping costs
If 200 or more capsules are ordered and if the form of payment is "direct debit", then no shipping cost is applied. If fewer than 200 capsules are ordered, or if the form of payment is other than "direct debit" then a shipping cost of € 10 is applied.
 - Calculation of long distance supplement
If the delivery address for the capsules is within a radius of 50 km of Utrecht, no long distance supplement is applied. If the delivery address is 50 km or more from Utrecht, but still in Netherlands, then a long distance supplement of €5 is applied. If the delivery address is outside the Netherlands, then a long distance supplement of €15 is applied.

9.4 Example

Rules while creating a conditions list:

- A condition is singular
- A condition is formulated positively (so avoid “not not” combinations)
- Try to keep the number of conditions per table to five or lower

- **Conditions:**

- **Calculation of standard shipping costs**

- C1 order ≥ 200 capsules
 - C2 form of payment=“direct debit”

- **Calculation of long-distance supplement**

- C3 distance < 50 km from Utrecht
 - C4 country= The Netherlands

Example

Now add the results to the conditions list:

- **Calculation of standard shipping costs**
 - C1 order ≥ 200 capsules
 - C2 form of payment="direct debit"
 - R1 Standard shipping costs:=0
 - R2 Standard shipping costs:=10
- **Calculation of long-distance supplement**
 - C3 distance <50 km from Utrecht
 - C4 country= The Netherlands
 - R3 Long -distance supplement:=0
 - R4 Long -distance supplement:=5
 - R5 Long -distance supplement:=15

9.4 Example

- Since the total number of conditions amounts to four, the tester has decided to include these in one table. The conditions list and the filling in of the tables according to multiple condition coverage deliver table on next slide with test situations for this example:

9.4 Shipping costs calculation (test situations)

Std shipping costs/Long distance supplement	1	2	3	4	5	6	7	8	9	1	1	1	1	1	1	1
	0	1	2	3	4	5	6	7	8	0	1	2	3	4	5	6
C1 order>=200 capsules	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
C2 form of payment = "direct debit"	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
C3 distance <50 km from Utrecht	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
C4 country= The Netherlands	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0
Results:																
R1 Standard shipping costs:=0	x x x x															
R2 Standard shipping costs:=10	x	x	x	x	x	x	x	x					x	x	x	x
R3 Long -distance supplement:=0	x									x	x	x	x	x		
R4 Long -distance supplement:=5	x							x	x			x				
R5 Long -distance supplement:=15	x	x			x	x				x	x		x	x		

9.4.1 Test situation-7 explained

- Reading a table is often thought to be difficult. Test situation 7, for example is read as follows:
- The customer has ordered fewer than 200 capsules AND has selected the “direct debit” payment form AND the delivery address is 50 km or more from Utrecht AND the delivery address is in the Netherlands. The shipping costs amount to € 10 standard cost plus € 5 long-distance supplement, equals € 15.

9.5 Example-filling decision table

- Note the handy way of filling in the “shopping costs calculation” decision table
- This causes only one condition to change in value per column.
- This is helpful for the creation of the physical test cases: copy and paste and change one value.

9.5 Example-filling decision table

- For the filling in, we begin at the bottom row of conditions with one 0 followed by, consecutively two times 1, two times 0 and so on until the last, which is given the value 0. In the row second from the bottom, we begin with two times 0 followed by, consecutively, four times 1, four times 0, and so on until the last two, which are given the value 0. We continue like this with the whole table; in every row the zero and one sets are twice as long as in the previous row.

9.5 Example- creating logical test cases

Std shipping costs/Long distance supplement	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
C1 order>=200 capsules	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
C2 form of payment = "direct debit"	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
C3 distance <50 km from Utrecht	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
C4 country= The Netherlands	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0
Results:																
R1 Standard shipping costs:=0																
R2 Standard shipping costs:=10	x	x	x			x	x	x						x	x	x
R3 Long -distance supplement:=0			x			x					x			x		
R4 Long -distance supplement:=5		x					x			x					x	
R5 Long -distance supplement:=15	x							x	x							x
No possible				x	x								x	x		

9.5 Example- Physical test cases

Std shipping costs/Long distance supplement	1	2	6	7	9	10	11	16
Number of capsules	199	199	199	199	200	200	200	200
Form of payment	cash	cash	Dir dbt	Dir dbt	Dir dbt	Dir dbt	Dir dbt	cash
Distance from Utrecht	178	182	10	182	178	182	10	178
Place of delivery	Brussels	Heerlen	Zeist	Heerlen	Brussels	Heerlen	Zeist	Brussels
Country	B	NL	NL	NL	B	NL	NL	B
Results:								
Standard shipping costs	10	10	10	10	0	0	0	10
Long-distance supplement	15	5	0	5	15	5	0	15
Total shipping cost	25	15	10	15	15	5	0	25

9.6 ECT - Introduction

- Aim
 - Detailed testing of functionality
 - Through coverage of decision points
- Test basis
 - Structured functional requirements, for ex: pseudo code or structured language
- Deriving principle
 - Coverage type: Decision points: Modified Condition/ Decision Coverage
- Quality characteristics
 - Functionality
 - Data controllability

9.7 Steps

- Identify test situations
- Create logical test cases
- Create physical test cases
- Specify initial data set
- Create test script

9.7.1 Step 1:

Identify test situations

- Analyze functional specifications
 - Uniform description of processing needed: if necessary translate specifications into pseudo-code
 - Identify conditions (find the IF statements)
- Give decision points a unique identification
- Apply Modified Condition/ Decision coverage per decision point
- Give the derived test situations a unique identification
- Optional: Specify test situations more in detail

9.7.1 Example Test Basis

Calculation premium health insurance policy

- Someone who is younger than 21 and is a client for less than 3 years, does not qualify for a specific insurance policy; an error message appears
- For those that do qualify for a policy, first age is considered: those younger than 30 or older than 60 pay 1000 Euros towards their premium; others 750 Euros
- Next health, risk category, the claimed amount of the previous year and again age are considered: if health is good and someone is in risk category C or has claimed less than 250 Euro the previous year, and is 50 years or younger, a discount of 10% is deducted from the premium

9.7.1 Example – Step 1.1

Translate into pseudo-code

Calculation of premium health insurance policy

```
IF          age<21 AND # years client < 3
THEN error message
ELSE       IF          Age<30 OR age>60
           THEN premium = 1000
           ELSE       premium = 750
           ENDIF
           IF          health = good AND
                        (risk category = C OR claimed
                        amount < 250) AND age <= 50
           THEN       premium = premium -/- 10%
           ENDIF
ENDIF
```

9.7.1 Example – Step 1.2

Uniquely identify decision points

Calculation of premium health insurance policy

```
IF (D1)      age<21 AND # years client < 3
THEN error message
ELSE        IF (D2)      Age<30 OR age>60
            THEN premium = 1000
            ELSE        premium = 750
            ENDIF
            IF (D3)      health = good AND
                        (risk category = C OR claimed
                        amount < 250) AND age <= 50
            THEN        premium = premium +/- 10%
            ENDIF
ENDIF
```



Copyright © Capgemini 2015. All Rights Reserved 22

9.7.2 Example – Step 1.3

MCDC per decision point: test situations

D1	1	0
A AND B	(error message)	(calculate premium)
A: age < 21	<u>1</u> 1	<u>0</u> 1
B: # years client < 3	1 <u>1</u>	1 <u>0</u>

D2	1	0
A OR B	(premium = 1000)	(premium = 750)
A: age < 30	<u>1</u> 0	<u>0</u> 0
B: age > 60	0 <u>1</u>	0 <u>0</u>

9.7.3 Example – Step 1.3

MCDC per decision point: test situations

D3 A AND (B OR C) AND D	1 (premium +/- 10%)	0 (premium unchanged)
A: health = good	<u>1</u> 1 0 1 (of 1011)	<u>0</u> 1 0 1 (of 0011 and 0111)
B: riskcategory = C	1 <u>1</u> 0 1	1 <u>0</u> 0 1
C: ClaimedAmount < 250	1 0 <u>1</u> 1	1 0 <u>0</u> 1
D: age <= 50	1 1 0 <u>1</u> (of 1011)	1 0 1 <u>0</u> (of 1100 and 1110)

9.7.4 Example – Step 1.4

Uniquely identify test situations

D1	1	0
A AND B	(error message)	(calculate premium)
A: age < 21	<u>1</u> 1 (1.1)	<u>0</u> 1 (1.2)
B: # years client < 3	1 <u>1</u>	1 <u>0</u> (1.1)

D2	1	0
A OR B	(premium = 1000)	(premium = 750)
A: age < 30	<u>1</u> 0 (2.1)	<u>0</u> 0 (2.3)
B: age > 60	0 <u>1</u> (2.2)	0 <u>0</u>

9.7.5 Example – Step 1.5

Specify test situations

Testsituation D3	D3-1	D3-2	D3-3	D3-4	D3-5
Health	Good	Good	≠ good	Good	good
Riskcategory	C	≠ C	C	≠ C	≠ C
Claimed mopunt	≥ 250	<250	≥ 250	≥ 250	<250
age	≤ 50	≤ 50	≤ 50	≤ 50	>50

Testsituation D2	D2-1	D2-2	D2-3
age	<30	>60	≥30≤60

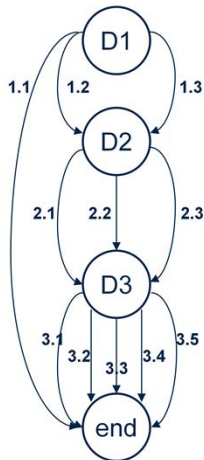
Testsituation D1	D1-1	D1-2	D1-3
age	<21	≥21	<21
# years client	<3	<3	≥3

9.8 Step 2. Specify logical test cases - 1

- Aid
- Create a graphic display of test situations and the relations (the 'graph')
 - Identify excluding test situations
 - Find out which (comparable) parameters are re-occurring in multiple decision points
 - Determine the test situations that exclude each other because of this
 - Combine the test situations to logical test cases
- In such a way that each test situation is represented in at least one logical test case
 - Include expected results to the logical test cases

9.8.1 Example – Step 2.1

Make a graph



Testsituation	Value	Next
D1-1	1	End
D1-2	0	D2
D1-3	0	D2
D2-1	1	D3
D2-2	1	D3
D2-3	0	D3
D3-1	1	End
D3-2	1	End
D3-3	0	End
D3-4	0	End
D3-5	0	End

9.8.2 Example – Step 2.2

Excluding test situations

- Excluding test situations because of
 - AGE
 - 1.3 and 2.2 / 2.3 / 3.5
 - 2.1 and 3.5
 - 2.2 and 3.1 / 3.2 / 3.3 / 3.4

9.8.3 Example – Step 2.3

Logical test cases

- The test situations are combined to logical test cases. In such a way that each test situation is represented minimum once in a logical test case, considering excluding test cases.
- The graph serves as an aid.
- TIP1: start covering those test situations, that occur with excluding test situations
- TIP2: the minimum number of test cases can directly be derived from the graph i.e. the maximum number of parallel lines/ arrows. Because of excluding test situations this could be more

9.8.4 Example – Step 2.3

Logical test cases

Test situation	Value	Next	TC-1	TC-2	TC-3	TC-4	TC-5	TC-6
D1-1	1	End	X					
D1-2	0	D2			X	X	X	X
D1-3	0	D2		X				
D2-1	1	D3		X				
D2-2	1	D3			X			
D2-3	0	D3				X	X	X
D3-1	1	End		X				
D3-2	1	End				X		
D3-3	0	End					X	
D3-4	0	End						X
D3-5	0	End			X			

9.8.5 Example – Step 2.3

Specify logical test cases

Test case	LTC-1	LTC-2	LTC-3	LTC-4	LTC-5	LTC-6
Consists of testsituations	1.1	1.3	1.2	1.2	1.2	1.2
		2.1	2.2	2.3	2.3	2.3
		3.1	3.5	3.2	3.3	3.4
Age	< 21	< 21	> 60	≥ 30, ≤ 50	≥ 30, ≤ 50	≥ 30, ≤ 50
# years client	<3	≥ 3	< 3	< 3	< 3	< 3
Health	-	Good	Good	Good	≠ Good	Good
Riskcategory	-	C	≠ C	≠ C	C	≠ C
Claimed amount	-	≥ 250	< 250	< 250	≥ 250	≥ 250

9.9 Step 3 Create physical test cases

- Give all parameters concrete values
- Make expected result concrete for each test case

9.9.1 Example – Step 3

Create physical test cases

Testcase	LTC-1	LTC-2	LTC-3	LTC-4	LTC-5	LTC-6
Consists of testsituations	1.1	1.3	1.2	1.2	1.2	1.2
		2.1	2.2	2.3	2.3	2.3
		3.1	3.5	3.2	3.3	3.4
Age	20	19	61	30	50	31
# years client	2	3	1	0	2	2
Health	Good	Good	Good	Good	≠ Good	Good
Riskcategory	C	C	≠ C	≠ C	C	≠ C
Claimed amount	-	≥ 250	< 250	< 250	≥ 250	≥ 250
Result						
Error Message	X					
Premium		900	1000	675	750	750

9.9 Step 4. Establish the initial data set

- The persons with the characteristics as described in the physical test cases, must be present in the database

9.10 Step 5. Assemble the test script

- Form and content are specific to organization

9.11 Variations

More depth level b.m.o the following coverage types:

- Decision points: Multiple Condition Coverage
 - More thorough testing within the (possibly specifically selected) decision points
- Boundary Value Analysis
 - More thorough testing within the (possibly specifically selected) decision points
- Pairwise testing
 - Adding testing of possible combinations of functional paths

Summary

- The concept of DTT was discussed
- An example of DTT was analyzed
- ECT was discussed
- Steps to design using ECT
- An example was analyzed



Add the notes here.

Review Question

- The DTT is based on _____ Principle
 - Condition Coverage
 - MCDC
 - Decision Coverage
 - Multiple Coverage
- The minimum number of test situation that can be applied in ECT is equal to
 - The number of parallel lines in the data flow
 - The number of decision point in the flow
 - The multiple of IN and OUT path of decision point
 - None of the above



Add the notes here.

Additional Reading

- URL
- ECT and DTT
 - https://capgemini.sumtotalsystems.com/sumtotal/app/sys_error.aspx?mode=accessdenied&UserMode=0&ru=/sumtotal/app/management/LMS_ActDetails.aspx%3fUserMode%3d0%26ActivityId%3d151274
- Book
 - Please read TMap Next pg 654-660 on the TTST ECT
 - Please read TMap Next page 639-650 on Test technique Decision table test,
 - Page no. 605-606 on Decision Coverage