

# 2016 / 2017 Model-based Testing User Survey: Results



Anne Kramer  
Bruno Legeard  
Robert V. Binder

## Contents

1	Overview .....	2
1.1	About the 2016 MBT User Survey .....	2
1.2	About this document .....	2
1.3	Result overview .....	2
2	Detailed Results .....	3
	Question 1: Degree of experience .....	3
	Question 2: Expectations .....	4
	Question 3: Does MBT fulfil those expectations? .....	5
	Question 4: Test level(s) .....	6
	Question 5: Type(s) of testing .....	7
	Question 6: Stage of MBT adoption .....	8
	Question 7: Use of models in other development phases .....	9
	Question 8: Type of models in those other development phases .....	10
	Question 9: Degree of reuse .....	11
	Question 10: Testing effort .....	12
	Question 11: MBT model focus .....	13
	Question 12: Degree of detail in MBT model .....	14
	Question 13: Notation type .....	15
	Question 14: Traceability .....	16
	Question 15: Generated artifacts .....	17
	Question 16: Test case generation .....	18
	Question 17: Selection criteria .....	19
	Question 18: Test execution .....	20
	Question 19: Tools integration .....	21
	Question 20: Role the model plays in MBT approach .....	22
	Question 21: How effective do you think MBT has been? .....	23
	Question 22: How likely are you to continue to use MBT? .....	24
	Question 23: Biggest limitation of MBT .....	25
	Question 24: Biggest advantage of MBT .....	26
	Question 25: Best estimate of the effects of MBT .....	27
	Question 26: Hours needed to become a proficient MBT user .....	28
	Question 27: ISTQB Certified Model-Based Tester .....	29
	Question 28: General application domain of the system under test .....	30
	Question 29: MBT project sponsor .....	31
	Question 30: Size of organization .....	32
	Question 31: Overall process approach .....	33
	Question 32: Role of respondents .....	34

## 1 Overview

### 1.1 About the 2016 / 2017 MBT User Survey

This survey is a follow up to the 2014 MBT User Survey (cf. <http://model-based-testing.info/2014/12/09/2014-mbt-user-survey-results/>) and the 2011/2012 MBT User Survey (cf. <http://robertvbinder.com/real-users-of-model-based-testing/>).

It was conducted from May to December 2016. 61 model-based testing practitioners answered, providing a relevant picture of the current state of the practice in the MBT area.

The purpose of the survey was to collect data and experience on the usage of Model-based Testing on a regular basis. We want to learn how MBT users view the efficiency and effectiveness of the approach, what works, what does not work and how MBT evolves.

### 1.2 About this document

This document summarizes the results of the 32 questions. For each question, we first provide a graphical representation of the 2016 result together with some raw data for both 2016 and 2014.

For numerical answers, we provided both the average and the median. The median indicates the middle of a distribution, that is, the point where there are as many answers below as above. It is less sensitive to extreme answers and usually more informative than the average. MBT tools are out of scope of the survey and, thus, not mentioned in this document.

Please, let us know if you have any comments, question or remarks regarding this MBT User Survey 2016.

Anne Kramer, sepp.med gmbh - [anne.kramer@seppmed.de](mailto:anne.kramer@seppmed.de)

Bruno Legeard, Smartesting & University of Franche-Comté – [bruno.legeard@smartesting.com](mailto:bruno.legeard@smartesting.com)

Robert V. Binder - [rvbinder@gmail.com](mailto:rvbinder@gmail.com)

### 1.3 Result overview

In 2016, you have been less to answer the survey, but those who did proved to be more persevering, meaning that fewer questions were skipped.

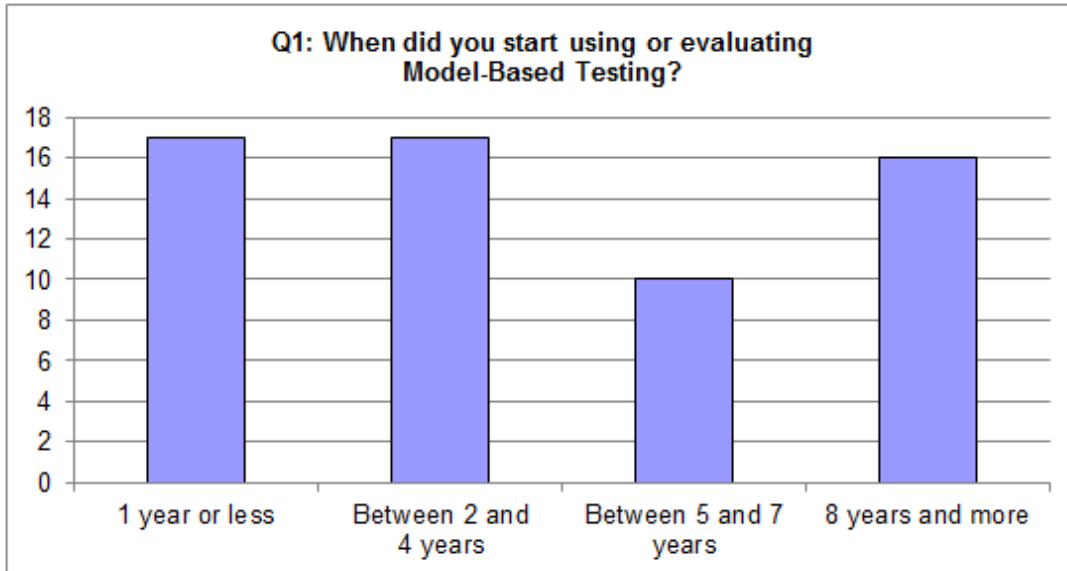
When comparing the results to those from 2014, we observe a shift from towards a more intense use of MBT in system testing instead of integration testing (question 4). Moreover, the MBT models seem to contain fewer details than stated in 2014 (question 9). Finally, but not least, more respondents replied to employ on-the-fly execution of model-based test cases (question 18).

The clearest trend, however, concerns the overall process approach. In 2014, more than 55% of the respondents still relied on phased approaches. Since then, agile approaches have gathered even more adepts. This time, 62,5% of the respondents used MBT in an agile overall process approach (question 31).

A part from those highlights, it is difficult to identify trends, especially if we take the statistical error taken into account. One answer was particularly surprising to us: one fifth of the respondents do not establish traceability between model elements and requirements/ use cases or features/ user stories (question 14).

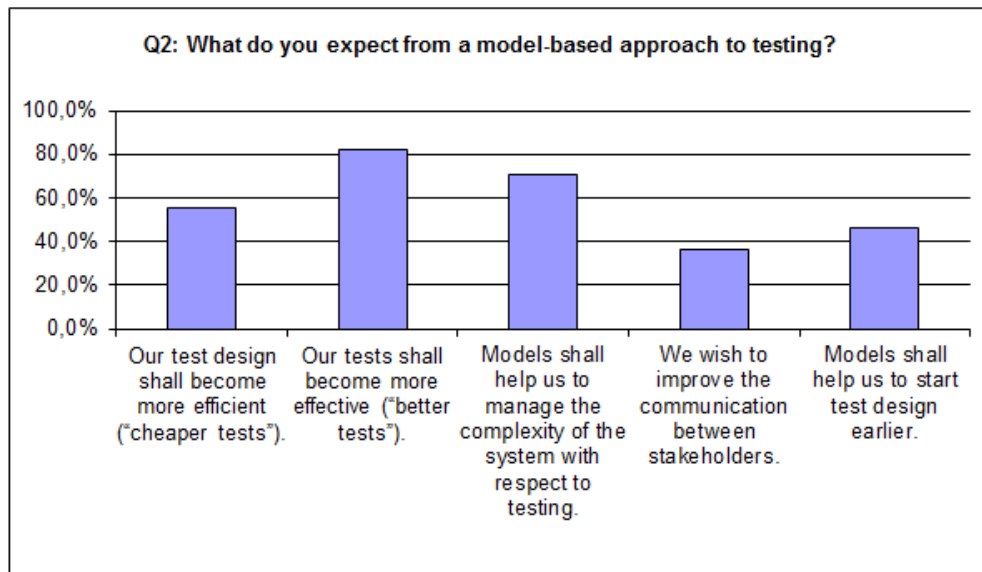
## 2 Detailed Results

### Question 1: Degree of experience



nn years ago	2016	2014
Average	5,9	5,1
<b>Median</b>	<b>4</b>	<b>3</b>
Minimum value	33	34
Maximum value	0	0

Question 2: Expectations



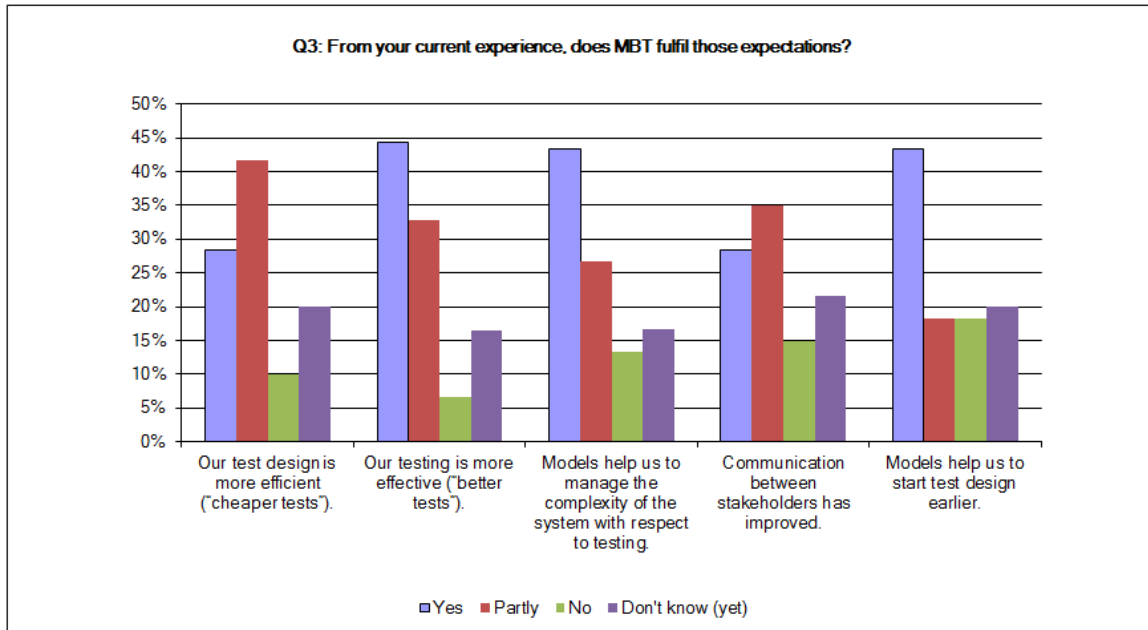
**Answer Options**

	2016	2014
Our test design shall become more efficient ("cheaper tests").	55,7%	73%
Our tests shall become more effective ("better tests").	82,0%	86%
Models shall help us to manage the complexity of the system with respect to testing.	70,5%	79%
We wish to improve the communication between stakeholders.	36,1%	45%
Models shall help us to start test design earlier.	45,9%	60%

**Other expectations:**

- higher coverage
- Easier test maintenance
- Better maintenance
- I will have a reference against which I can compare the on board executable.
- regarding "more effective" tests, I expect a better test coverage and better transparency on the coverage level.
- early start to automation, automation scripts automatically generated
- Our tests shall become more maintainable.
- Handle computer timing issues
- Manual testing becoming obsolete Testing role shifts to designer role

### Question 3: Does MBT fulfil those expectations?



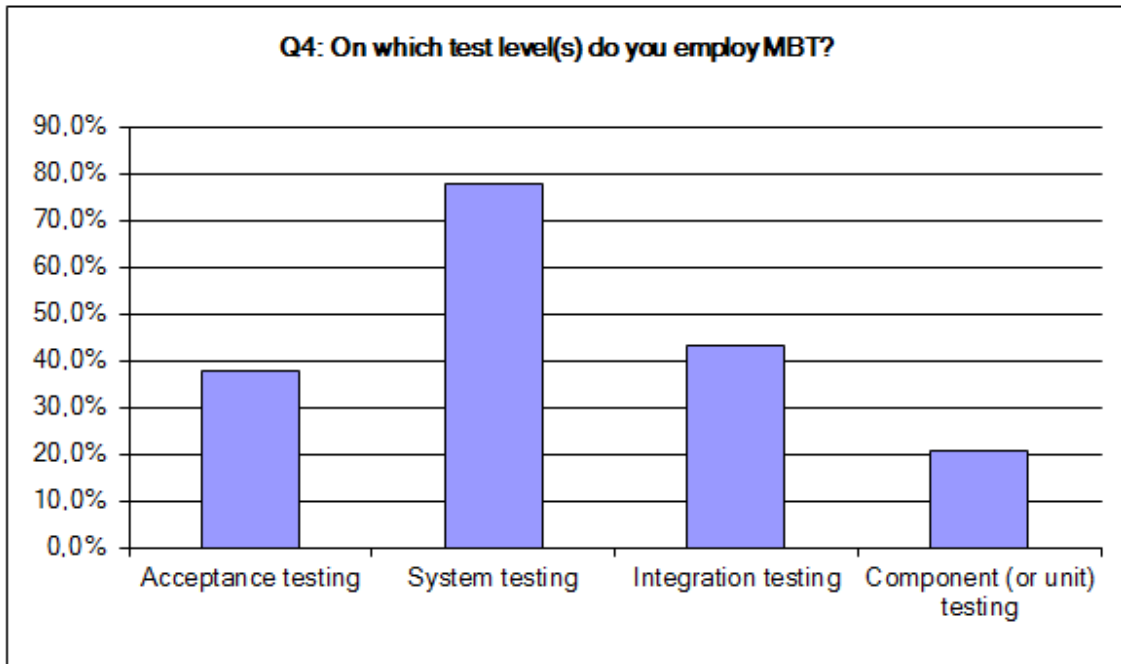
Answer Options	Yes	Partly	No	Don't know (yet)
Our test design is more efficient ("cheaper tests").	28%	42%	10%	20%
Our testing is more effective ("better tests").	44%	33%	7%	16%
Models help us to manage the complexity of the system with respect to testing.	43%	27%	13%	17%
Communication between stakeholders has improved.	28%	35%	15%	22%
Models help us to start test design earlier.	43%	18%	18%	20%

	Yes or Partly		No	
	2016	2014	2016	2014
Our test design is more efficient ("cheaper tests").	70%	68%	10%	15%
Our testing is more effective ("better tests").	77%	77%	7%	8%
Models help us to manage the complexity of the system with respect to testing.	70%	81%	13%	5%
Communication between stakeholders has improved.	63%	59%	15%	15%
Models help us to start test design earlier.	62%	67%	18%	10%

#### Other answers:

- MBT needs to be fully automated
- MBT should only be used in rare cases I think, such as when other tests already validated that the SUT actually works. In these cases and if you still have resources to spend at it, MBT can be one alternative to find alternative "roads" through the system and maybe find some alternative use case bug
- Complexity: it does not ease managing the complexity, you have to carefully restrict where do you apply MBT (really low or high level), not all desired functionality will be automated. Communication between stakeholders: I see no improvements there, models itself even simple or complex ones could not be used as they are in the communication
- The challenge is that the MBT design, that potentially can create a lot of tests, usually expose a lot of flakiness. Which is a problem in continuous delivery. It's a contradiction yes. But the wish of having stable 'tests' sometimes is hard to align with the effectiveness of MBT.
- Telco Complexity seemed to be too much for modeling. During adding functionality to the model it needed to be refactored continuously to keep test generation time low (below 12 minutes)
- Test automation (from abstract to concrete tests) remains a laborious activity that hinders a wider use of MBT.
- It's practicality for large systems where to start
- In most projects models have not been available.
- The non determinism and parallelism in our system makes our tooling explode in states and therefore not usable
- Proof of concept not completed
- MBT has not become grown up. Traditional testing is still favorite according to client

Question 4: Test level(s)



**Answer Options**

- Acceptance testing
- System testing
- Integration testing
- Component (or unit) testing

**2016**

- 37,9%
- 77,6%
- 43,1%
- 20,7%

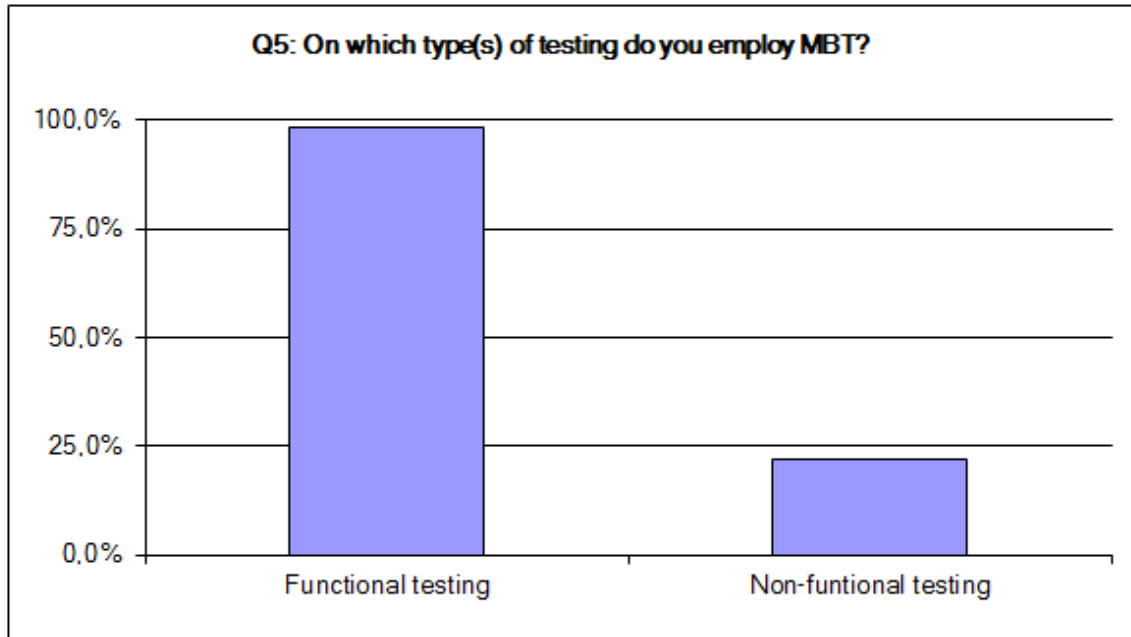
**2014**

- 31,2%
- 49,5%
- 77,4%
- 40,9%

**Others:**

- Not at all, anymore (client does not want to)
- Platform testing

Question 5: Type(s) of testing



**Answer Options**

	2016	2014
Functional testing	98,3%	98,8%
Non-funtional testing	22,0%	*)

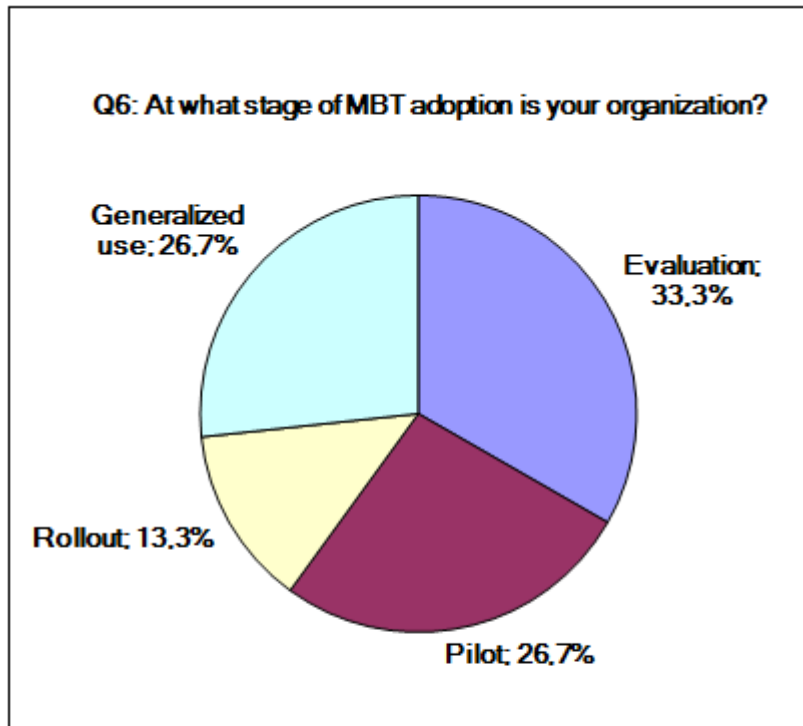
**Details on non-functional testing:**

- security testing
- performance (memory, time)
- GUI testing
- performance and reliability testing
- Security testing: fuzzing techniques.
- Performance is resource efficiency cpu,mem io
- performance testing workflow testing

\*) Answer option is different to 2014 MBT User Survey

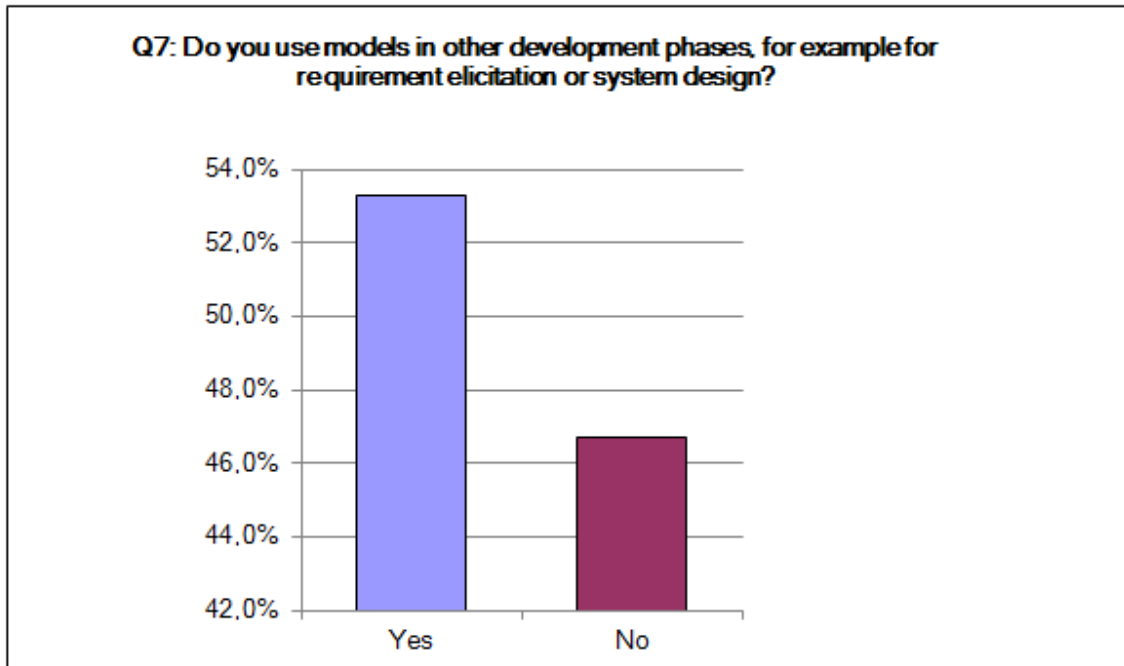


Question 6: Stage of MBT adoption



Answer Options	2016	2014
Evaluation	33,3%	26,3%
Pilot	26,7%	26,3%
Rollout	13,3%	16,8%
Generalized use	26,7%	30,5%

Question 7: Use of models in other development phases



**Answer Options**

Yes  
No

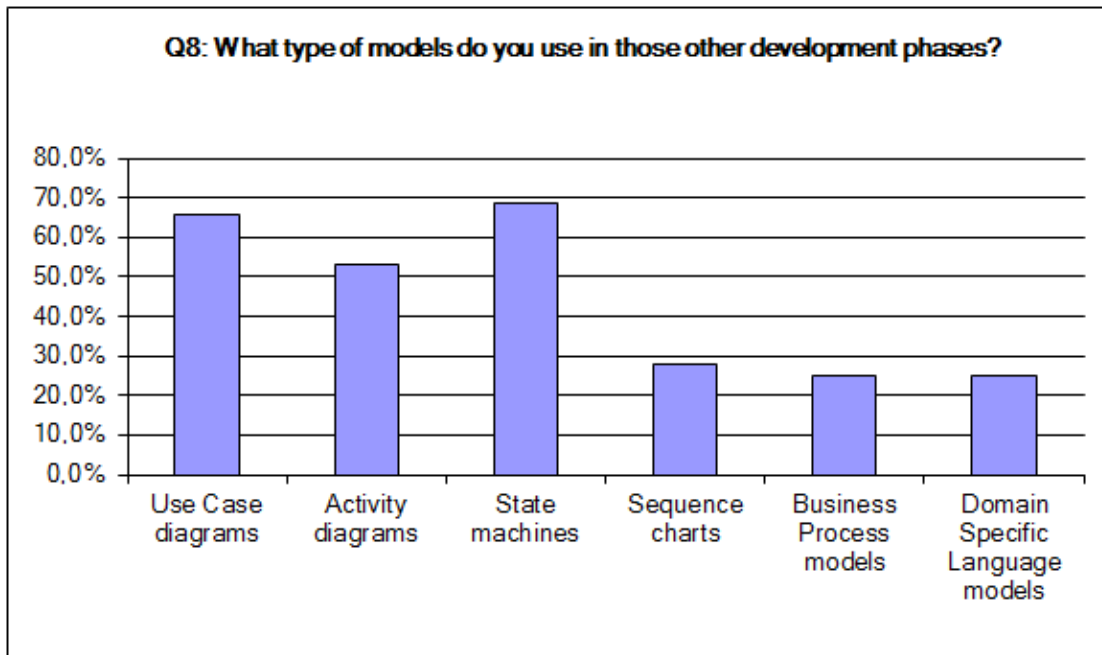
**2016**

53,3%  
46,7%

**2014**

58,9%  
41,1%

Question 8: Type of models in those other development phases



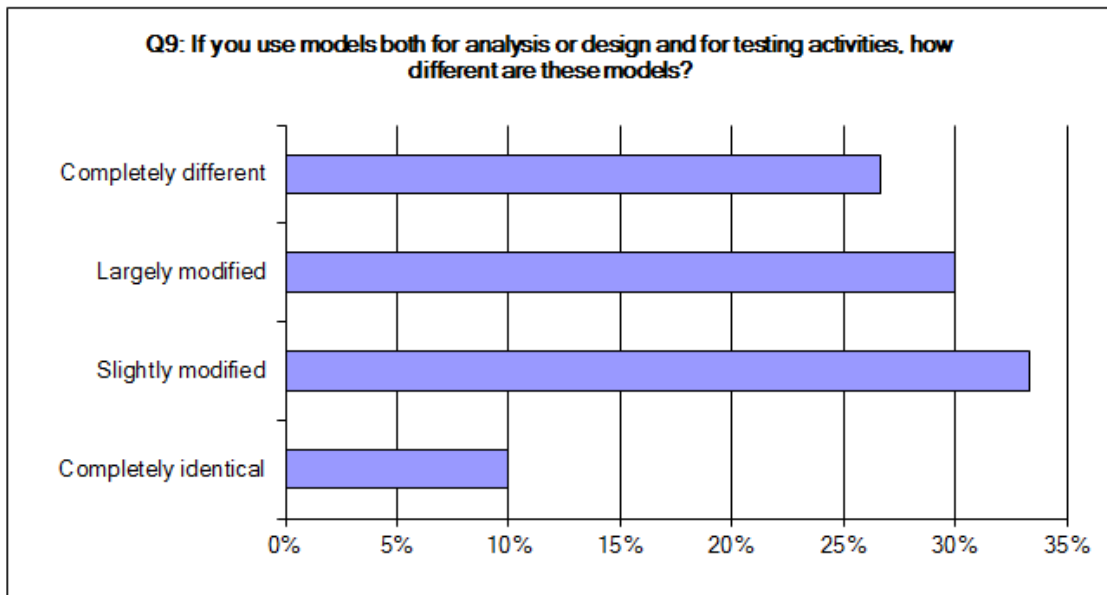
**Answer Options**

	<b>2016</b>	<b>2014</b>
Use Case diagrams	65,6%	65,4%
Activity diagrams	53,1%	46,2%
State machines	68,8%	69,2%
Sequence charts	28,1%	36,5%
Business Process models	25,0%	17,3%
Domain Specific Language models	25,0%	30,8%

**Other answers:**

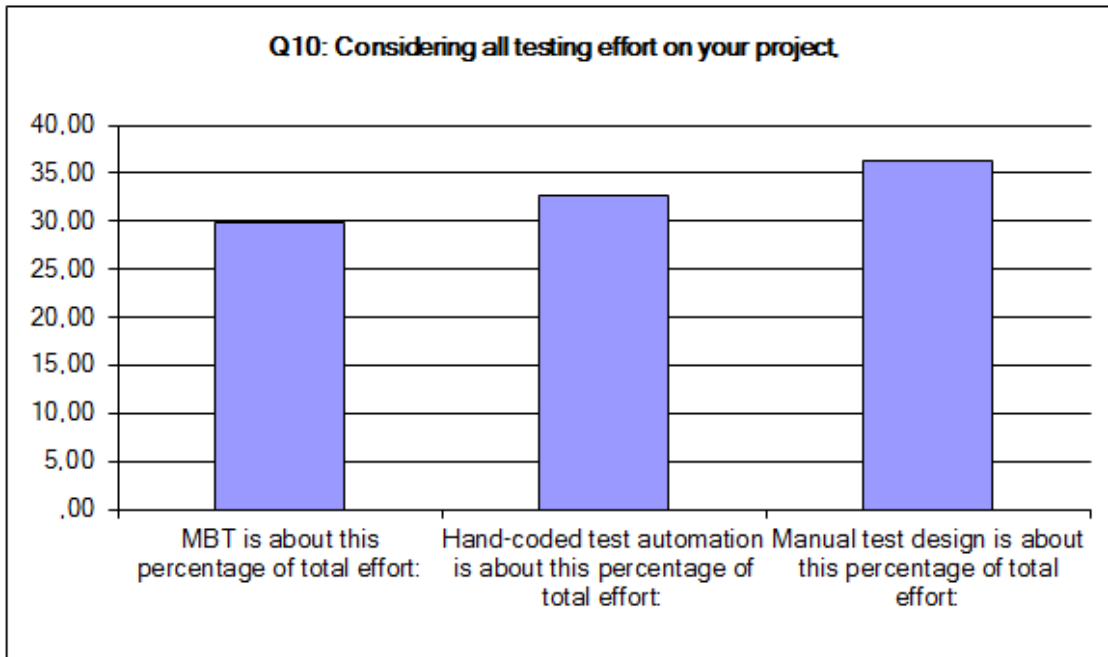
- Signal flow (Simulink)

Question 9: Degree of reuse



Degree of redundancy	2016	2014
Completely identical	10%	15%
Slightly modified	33%	29%
Largely modified	30%	33%
Completely different	27%	23%

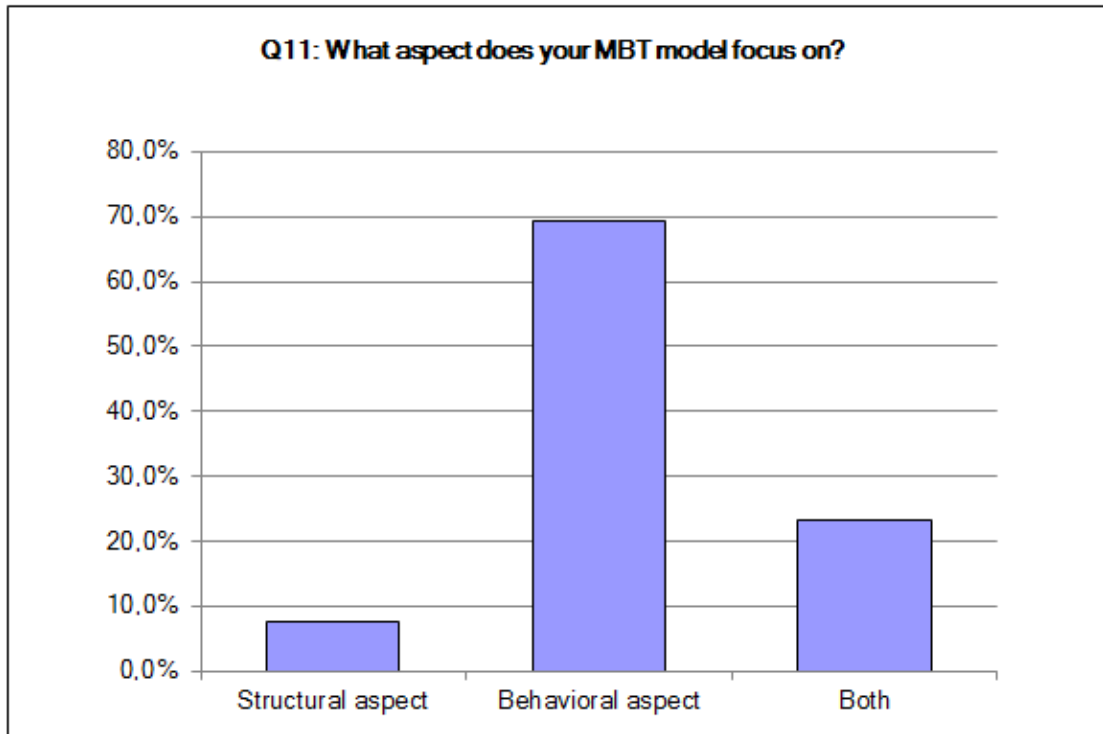
Question 10: Testing effort



**Answer Options**

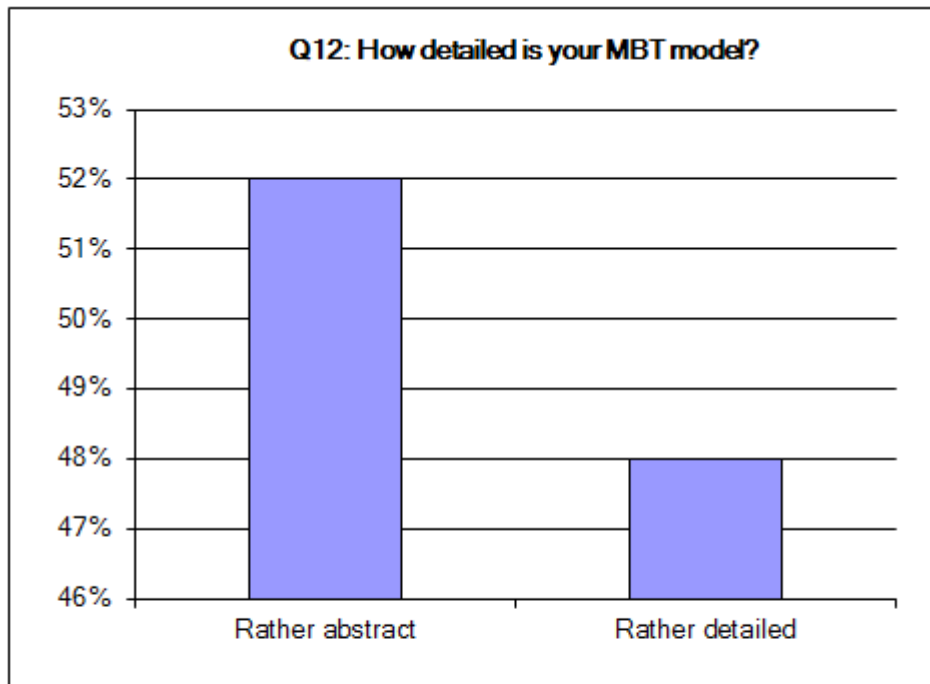
	<b>2016</b>	<b>2014</b>
MBT is about this percentage of total effort:	29,77	28,95
Hand-coded test automation is about this percentage	32,74	34,62
Manual test design is about this percentage of total	36,32	36,89

Question 11: MBT model focus



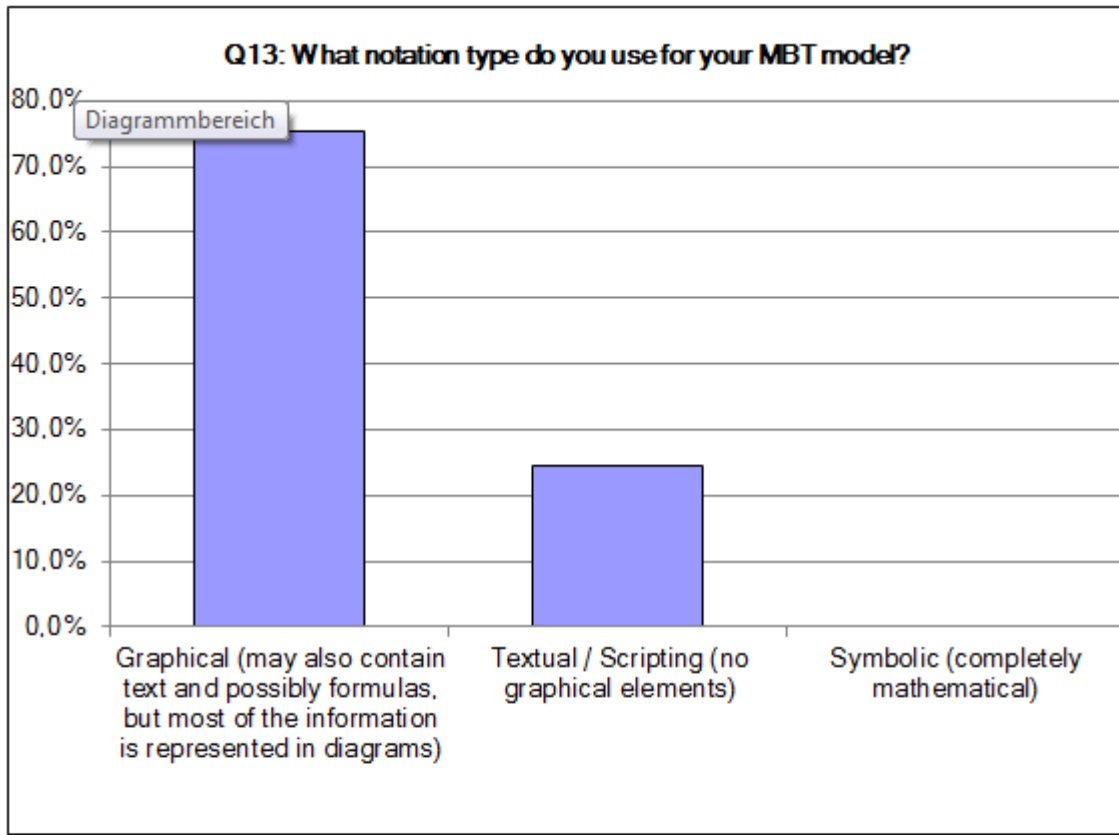
Answer Options	2016	2014
Structural aspect	7,7%	6,3%
Behavioral aspect	69,2%	58,8%
Both	23,1%	35,0%

Question 12: Degree of detail in MBT model



Degree of abstraction	2016	2014
Rather abstract	52%	43%
Rather detailed	48%	57%

Question 13: Notation type



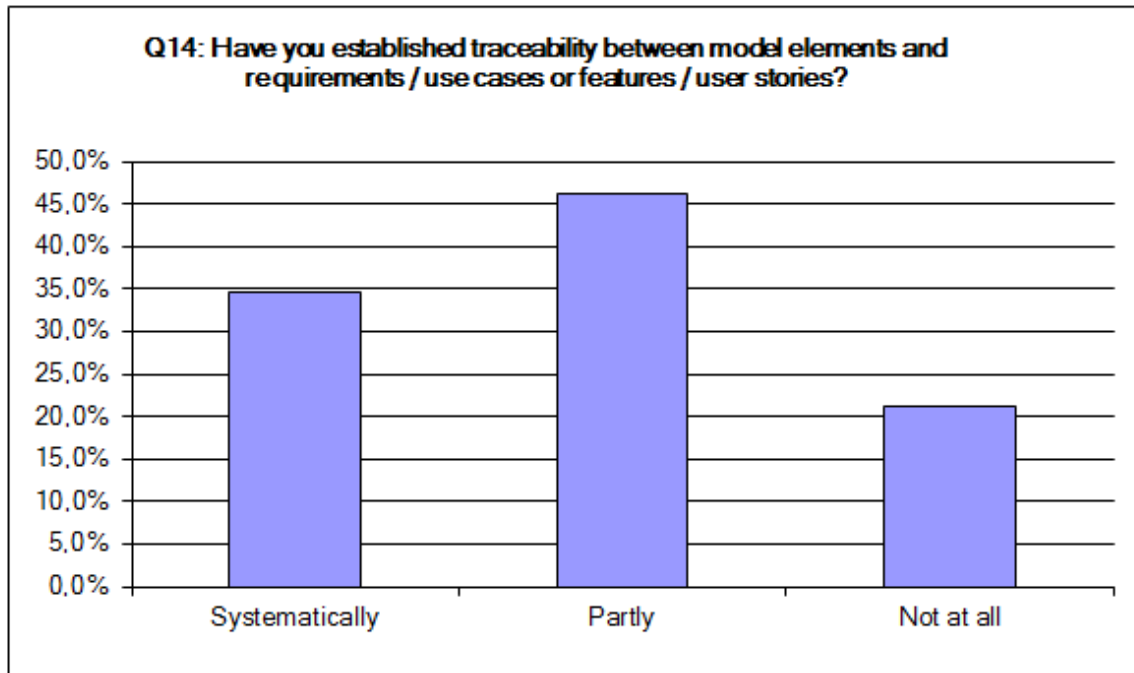
Answer Options	2016	2014
Graphical	75,5%	81,0%
Textual / Scripting	24,5%	13,9%
Symbolic	0,0%	5,1%

**Other answers:**

- based on Abstract State Machines
- We use simulink
- I use a combination of Graphical and Textual



Question 14: Traceability



**Answer Options**

Systematically

Partly

Not at all

**2016**

34,6%

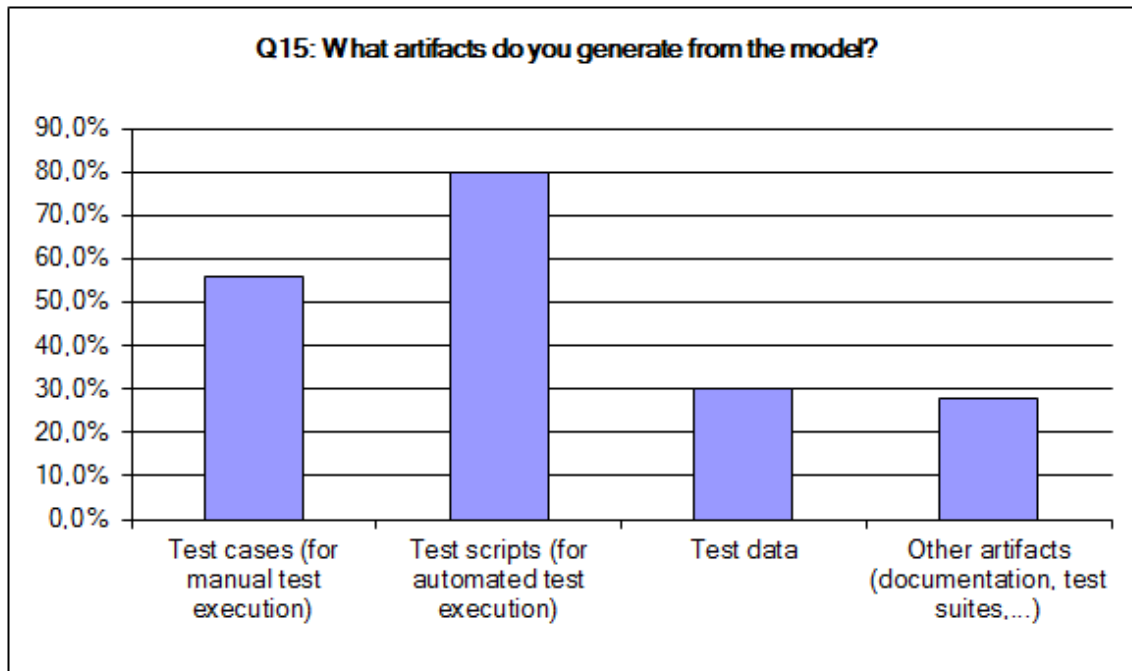
46,2%

21,2%

\*)

\*) Question is new

Question 15: Generated artifacts



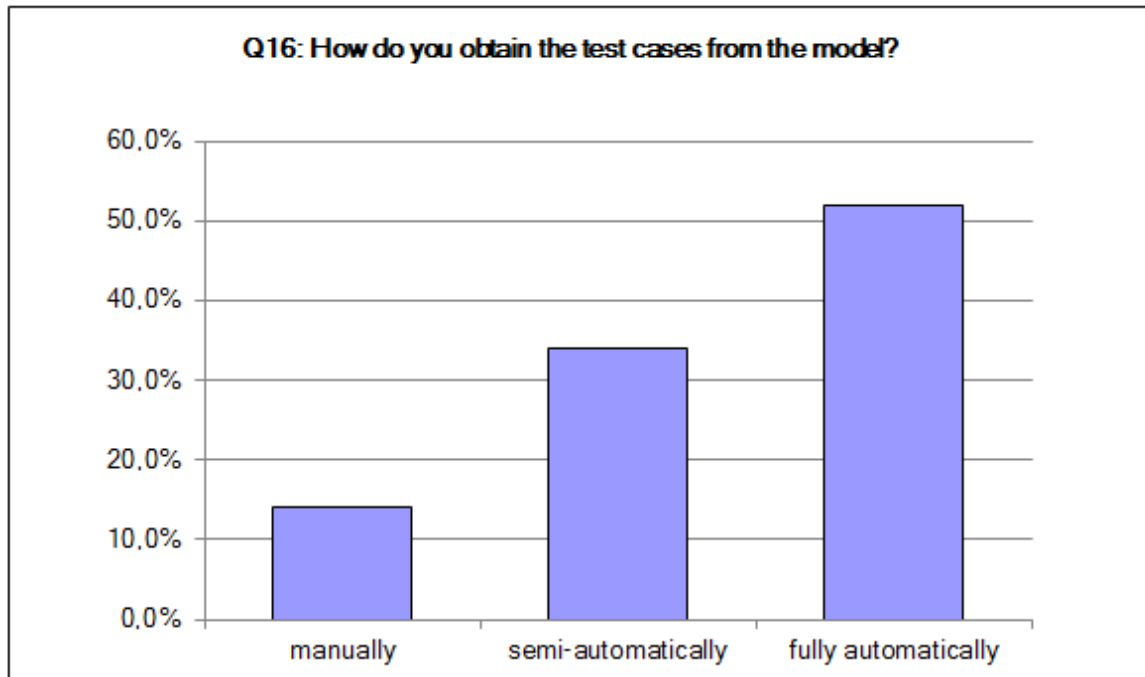
**Answer Options**

	2016	2014
Test cases (for manual test execution)	56,0%	56,6%
Test scripts (for automated test execution)	80,0%	84,2%
Test data	30,0%	39,5%
Other artifacts (documentation, test suites,...)	28,0%	28,9%

**Other answers:**

- Test Documentation to be uploaded directly to Quality Center Test Macros to be executed in existing Test Automation frame
- requirements-links
- Test suites
- Stubs of the system
- diagramms for documentation
- Detailed test steps (in Excel) for review of the tests by QA
- test plan

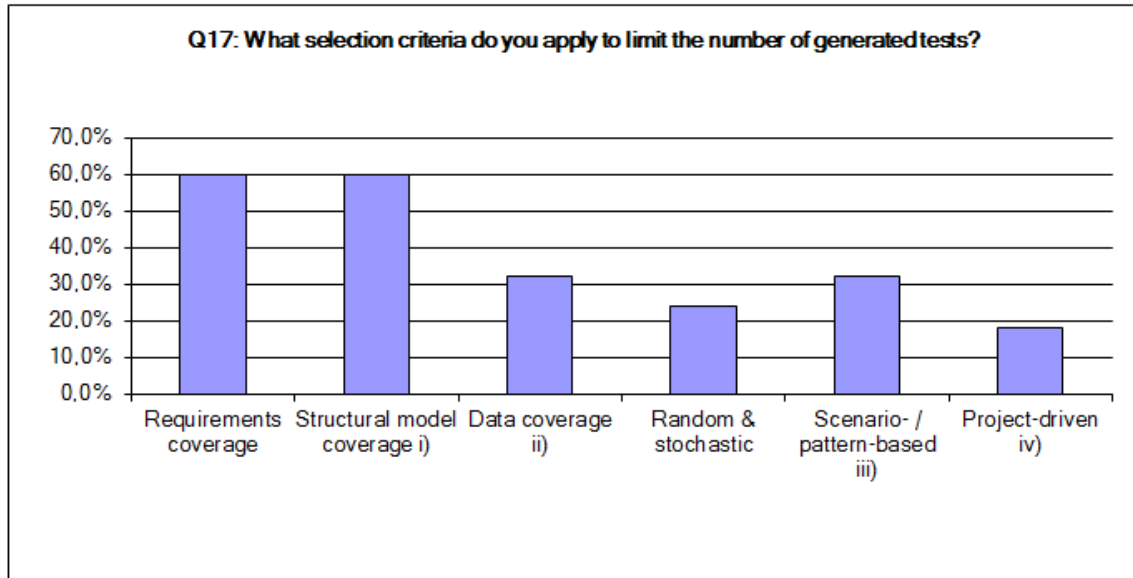
Question 16: Test case generation



Answer Options	2016	2014
manually	14.0%	11.8%
semi-automatically	34.0%	35.5%
fully automatically	52.0%	52.6%

This question aimed at the test artifact generation process. For tool integration, see question 19.

Question 17: Selection criteria



**Answer Options**

- Requirements coverage
- Structural model coverage i)
- Data coverage ii)
- Random & stochastic
- Scenario- / pattern-based iii)
- Project-driven iv)

**2016**

- 60,0%
- 60,0%
- 32,0%
- 24,0%
- 32,0%
- 18,0%

**2014**

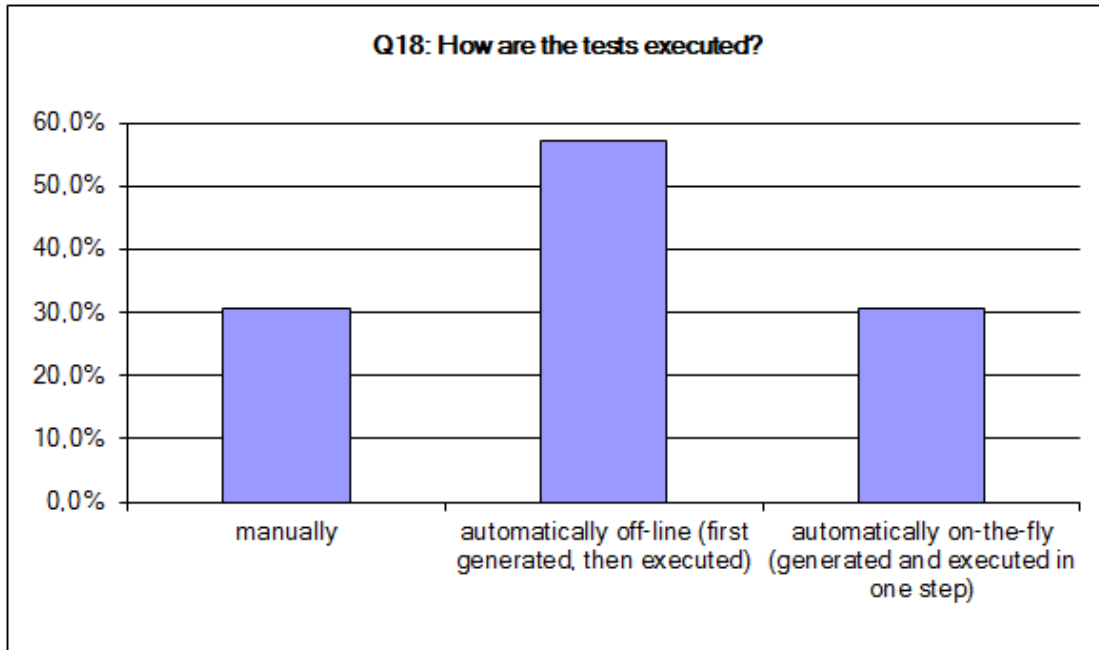
- 69,3%
- 58,7%
- 28,0%
- 29,3%
- 49,3%
- 25,3%

**Other answers:**

- For the low level MBTs we used time and test count based

- i) Structural model coverage is a generic term for a variety of coverage criteria that are based on the internal structure of the model (e.g. transition coverage).
- ii) Data coverage focuses on the equivalence partitions of input and output data, possibly combined with a boundary value analysis.
- iii) Scenarios or patterns are specific paths through the model that correspond to frequently used functionality or fault-based scenarios.
- iv) Project-driven test case selection criteria uses specific additional information added to the model (e.g. risk, priorities...) to drive test generation to achieve specific test objectives for the project.

Question 18: Test execution



**Answer Options**

manually

**2016**

30,6%

**2014**

34,7%

automatically off-line (first generated, then executed)

57,1%

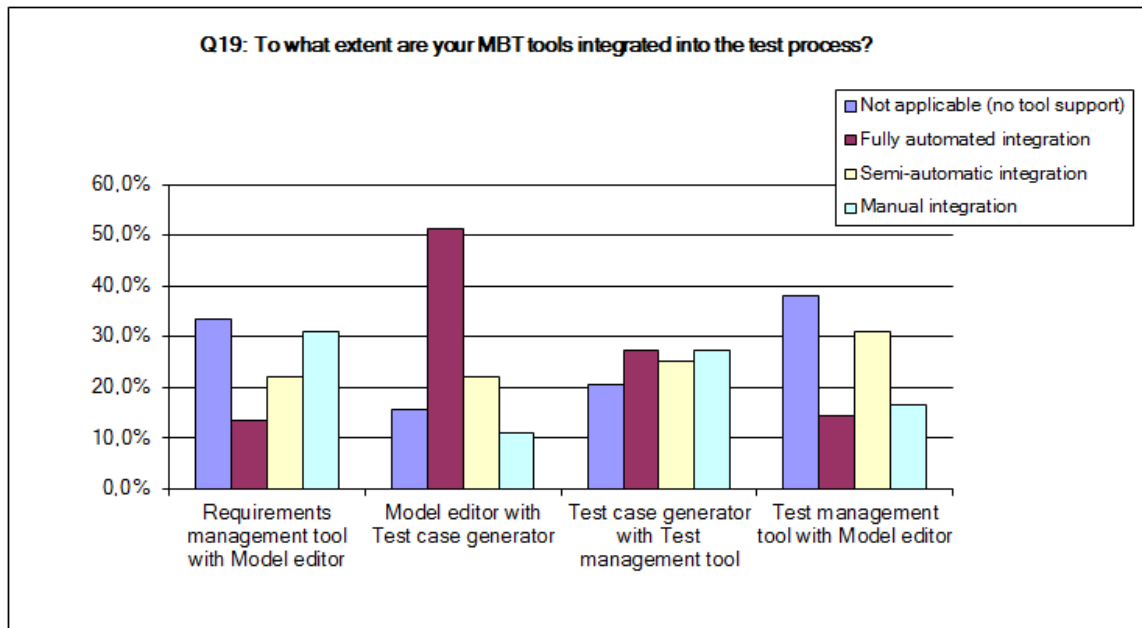
69,3%

automatically on-the-fly (generated and executed in one step)

30,6%

24,0%

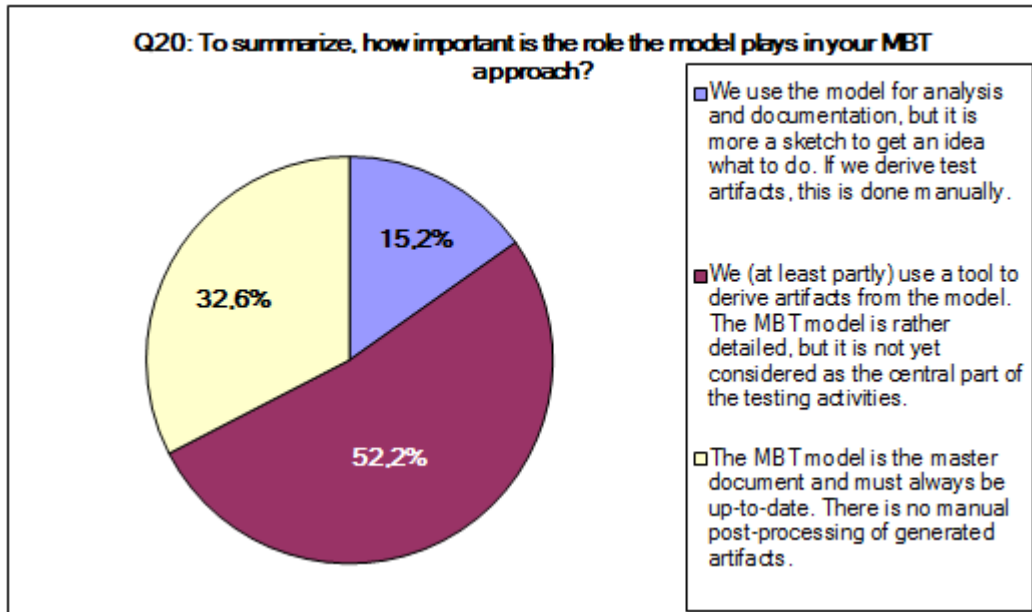
Question 19: Tools integration



Answer Options	Manual integration	Semi-automatic integration	Fully automated integration	Not applicable (no tool support)
Requirements management tool with Model editor	31,1%	22,2%	13,3%	33,3%
Model editor with Test case generator	11,1%	22,2%	51,1%	15,6%
Test case generator with Test management tool	27,3%	25,0%	27,3%	20,5%
Test management tool with Model editor	16,7%	31,0%	14,3%	38,1%

2014	Manual integration	Semi-automatic integration	Fully automated integration	Not applicable (no tool support)
Requirements management tool with Model editor	31,9%	24,6%	13,0%	30,4%
Model editor with Test case generator	18,6%	22,9%	41,4%	17,1%
Test case generator with Test management tool	21,4%	30,0%	18,6%	30,0%
Test management tool with Model editor	21,7%	15,9%	14,5%	47,8%

Question 20: Role the model plays in MBT approach



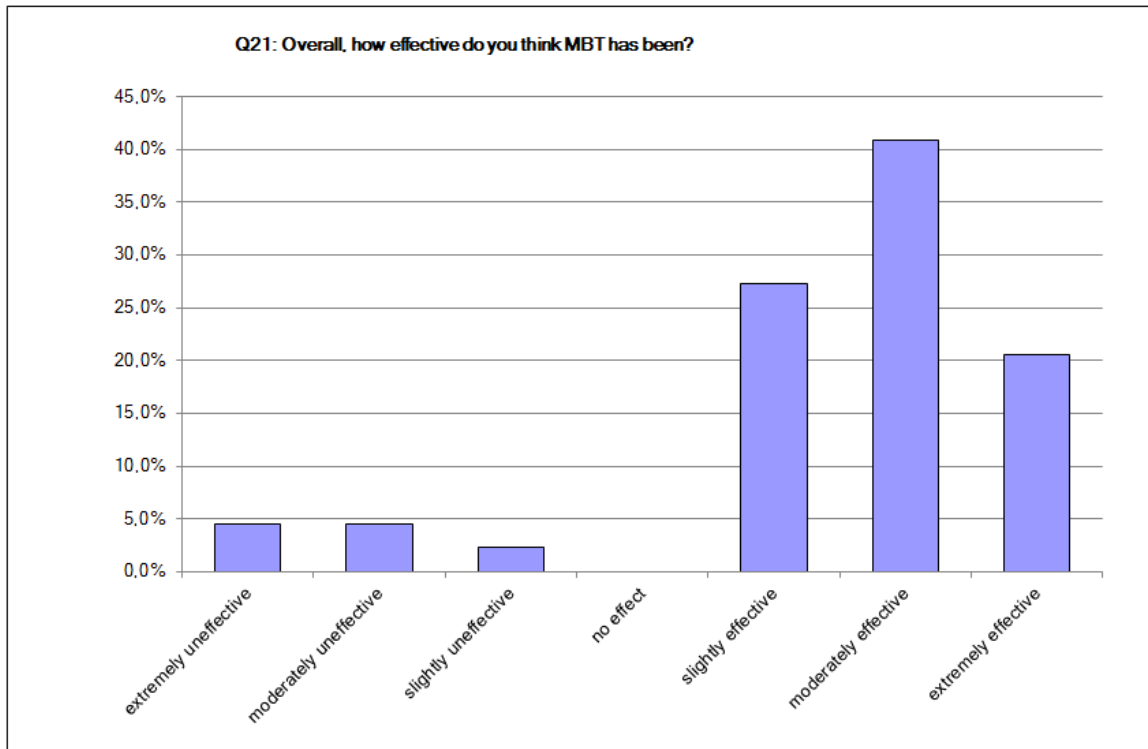
**Answer Options**

	2016	2014
We use the model for analysis and documentation,	15,2%	15,2%
We (at least partly) use a tool to derive artifacts from	52,2%	47,0%
The MBT model is the master document and must	32,6%	37,9%

**Other answers:**

- The role of the MBT was to have extra attention on the low level interactions with the database in the core module, on the other hand to have a "user-like" system test what randomly navigates through the web-page. So we used MBT as extra tools to have better coverage in certain areas they were really important but not in the main position.
- For us, the test model must be executable, i.e. compilable to code.
- We use simulink models to describe the software application layer. The tests are generated automatically or manually. Expected results are from models. These are used to test the code on target
- Model will act like a heart for the MBT , which helps us to replicate the system behavior in a simpler form so that every one can understand easily.
- Models will get a bigger role through out 2016 and 2017

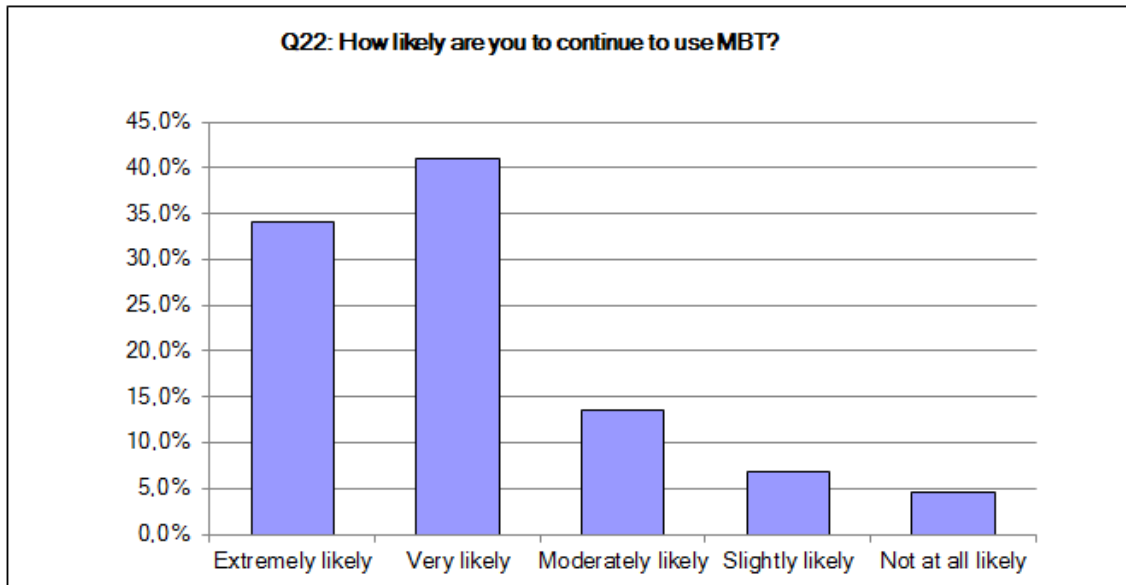
Question 21: How effective do you think MBT has been?



Answer Options	2016	2014
extremely ineffective	4,5%	2,8%
moderately ineffective	4,5%	2,8%
slightly ineffective	2,3%	1,4%
no effect	0,0%	5,6%
slightly effective	27,3%	23,6%
moderately effective	40,9%	40,3%
extremely effective	20,5%	23,6%



Question 22: How likely are you to continue to use MBT?



**Answer Options**

Extremely likely  
Very likely  
Moderately likely  
Slightly likely  
Not at all likely

**2016**

34,1%  
40,9%  
13,6%  
6,8%  
4,5%

**2014**

35,6%  
37,0%  
21,9%  
4,1%  
1,4%

### Question 23: Biggest limitation of MBT

**Q23: I see the biggest limitation of MBT as (regarding the approach itself and/or its introduction and use):**

**Answers:**

- if not fully automated it does not scale
- \*MBT is not the best way to produce, execute and maintain test cases when the purpose is to verify if the system works as expected.
- The biggest limitation I think is that you cannot put all your automation into MBT (specific use-cases, bug verifications) without increasing the complexity of the model too much. It has to be side-by-side with other automation approaches.
- Integrating it into Continuous Delivery/Deployment. The challenge is to reuse parts of model(s) for the happy paths, that is to be used in CD.
- Full scale deployment have been stopped after one year based on ROI recalculation. License fee shall be on 10% of current one to make Business Case fly. However this was calculated barely on effort gain on the Development side without considering the gain from better quality
- Effectiveness of generated abstract tests is limited because most off-line test generation approaches are trivial when based on simple structural model coverage.
- models may introduce a new point of failure
- Reference approach and tool
- Complete end to end software testing life cycle tool chain integration
- Changing Functional Requirements, almost everyday Dependency on the model creator
- Available tools and real life examples
- large amount of generated tests
- usability of modeling tools
- Efforts to produce the models
- Not used from the design phase
- Models creation, manual or automatic?
- I have the idea that the tooling is not mature enough to be already widespread used
- Model complexity
- none
- that a modeler need to have skills (and motivation) both in:
  - domain expertise - modeling/programming/abstraction.
- still looking for interface objects for backend testing like DB , Putty -unix Box, Webservice call with SOAP Ui etc ...need to enhance custom action where we can get data coverage as well.
- Few limitations in usage of MBT where there are multiple test data which is dynamic in nature

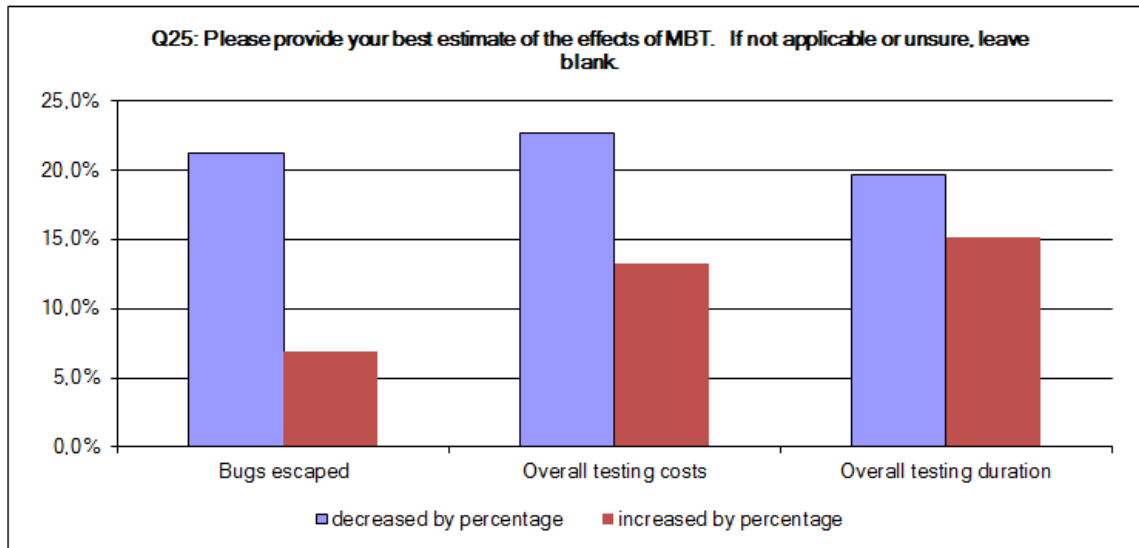
Question 24: Biggest advantage of MBT

**Q24: I see the biggest advantage of MBT as:**

**Answers:**

- Forcing a graphical representation into the development team, giving everyone the same picture of what is going on. If made correctly, this picture can be used and maintained to do some of the optional tests.
- Extremely effective on simple models (CRUD) and on high level important/critical user paths. It adds a new dimension to testing with combining random actions together. It is also really good it can allow you to limit the sequence of actions to the smallest set what can reproduce the error (like proper or quickcheck)
- Having transparent test design, to show stakeholders and team members. Maintenance.
- Communication Agile Development
- Use of computer power to generate tests that are beyond comprehension of most humans.
- getting better grasp of requirements (and possible missing areas / contradictions)
- Long term maintainability of tests, better requirements coverage and higher ROI for test automation
- Complete Automation
- Logical, rational and seems the way to go
- tool supporting
- Reusability of test code
- managing complexity
- Quality and coverage improvements, good maintainability.
- If done properly a great tool to break up system into components, behaviour description, and to generate tests and formal proofs.
- Generating test cases automatically when the requirements are changed.
- It saves time but generates at the same time more test
- Automation
- none
- in a confidence in product quality brought by systematic test design
- It best for GUI testing .
- Test coverage, handling changes
- Automation and using more test techniques driven by tooling
- The possibilities of test coverage and proof.
- limited standards in language and processes
- The biggest limit of MBT are the ones doing the models. Models must fit a specific purpose, and only used for that.

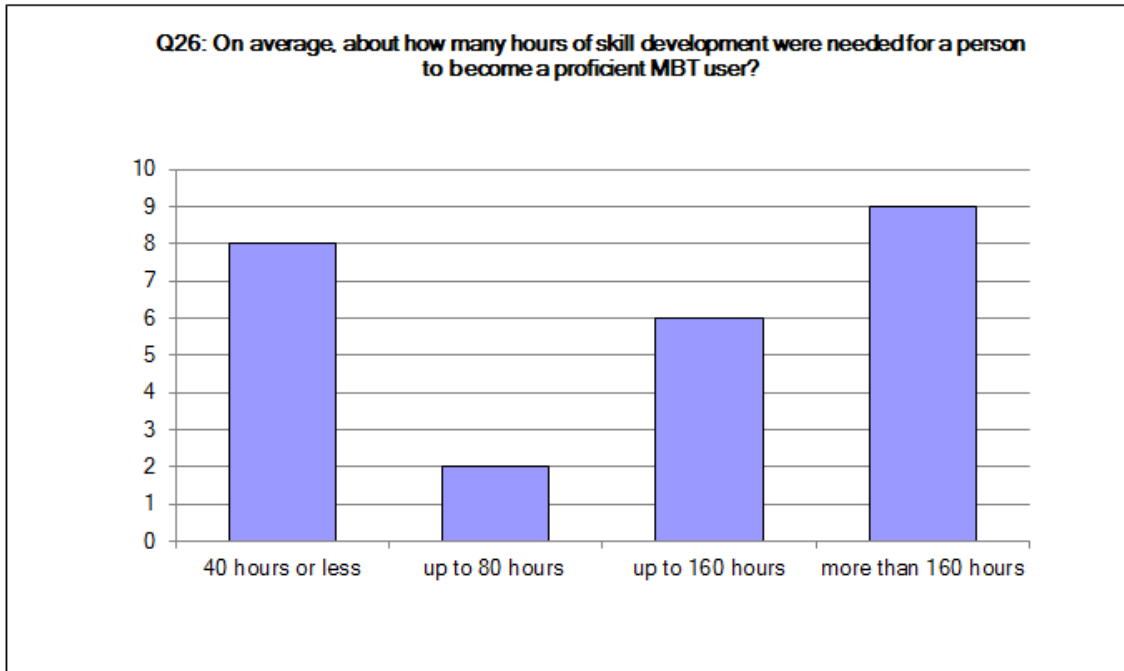
Question 25: Best estimate of the effects of MBT



**Answer Options**

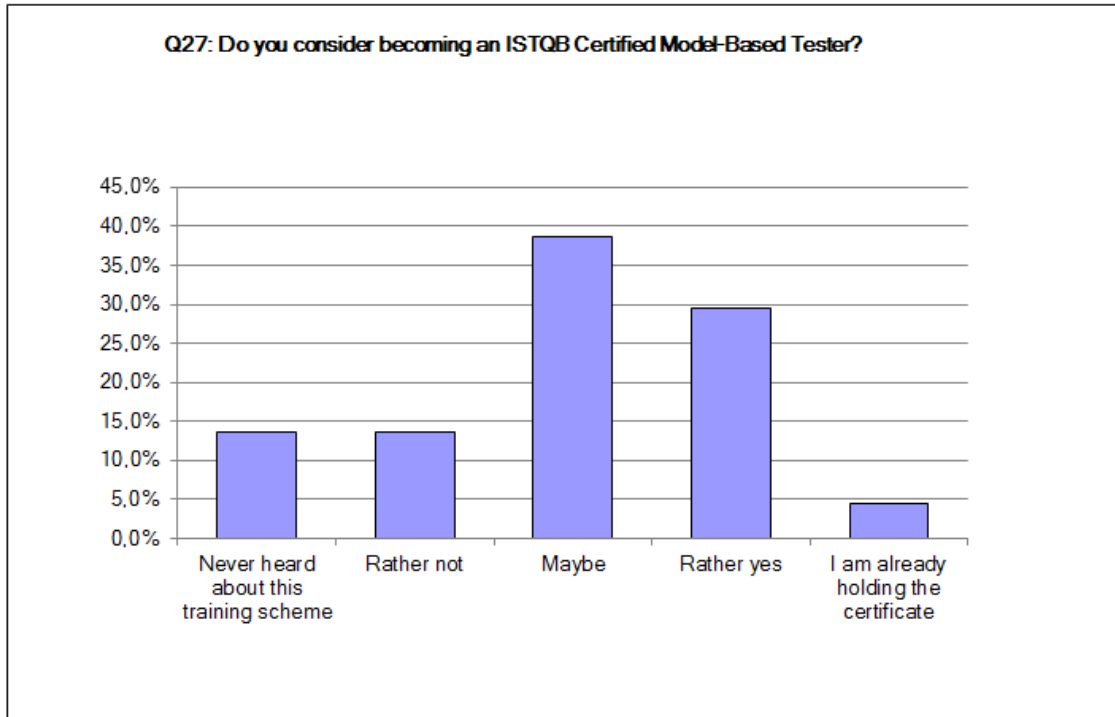
	2016	2014
Bugs escaped decreased by percentage:	21,23	27,50
Bugs escaped increased by percentage:	6,89	13,75
Overall testing costs decreased by percentage:	22,69	34,89
Overall testing costs increased by percentage:	13,22	6,25
Overall testing duration decreased by percentage:	19,62	30,93
Overall testing duration increased by percentage:	15,11	6,25

Question 26: Hours needed to become a proficient MBT user



Average hours needed to become proficient	2016	2014
Average	185.12	196
<b>Median</b>	<b>100</b>	80
Minimum value	8	0
Maximum value	700	2000

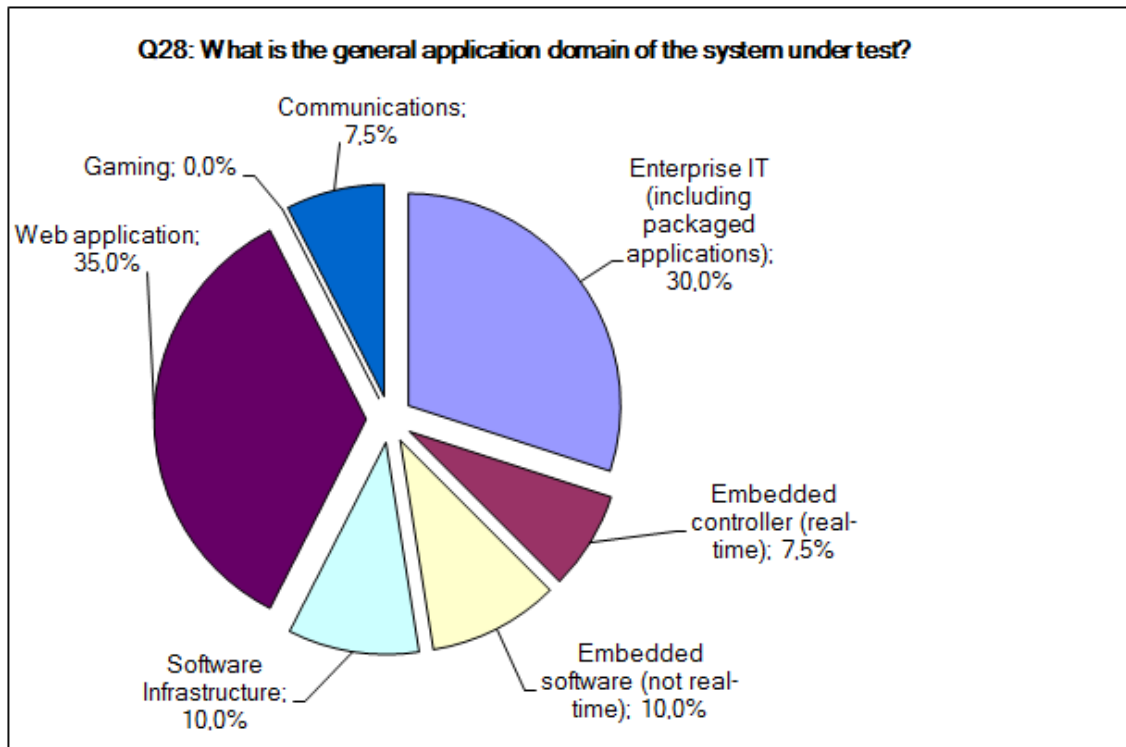
Question 27: ISTQB Certified Model-Based Tester



Answer Options	2016	*
Never heard about this training scheme	13,6%	
Rather not	13,6%	
Maybe	38,6%	
Rather yes	29,5%	
I am already holding the certificate	4,5%	

\* ) Question is new

Question 28: General application domain of the system under test



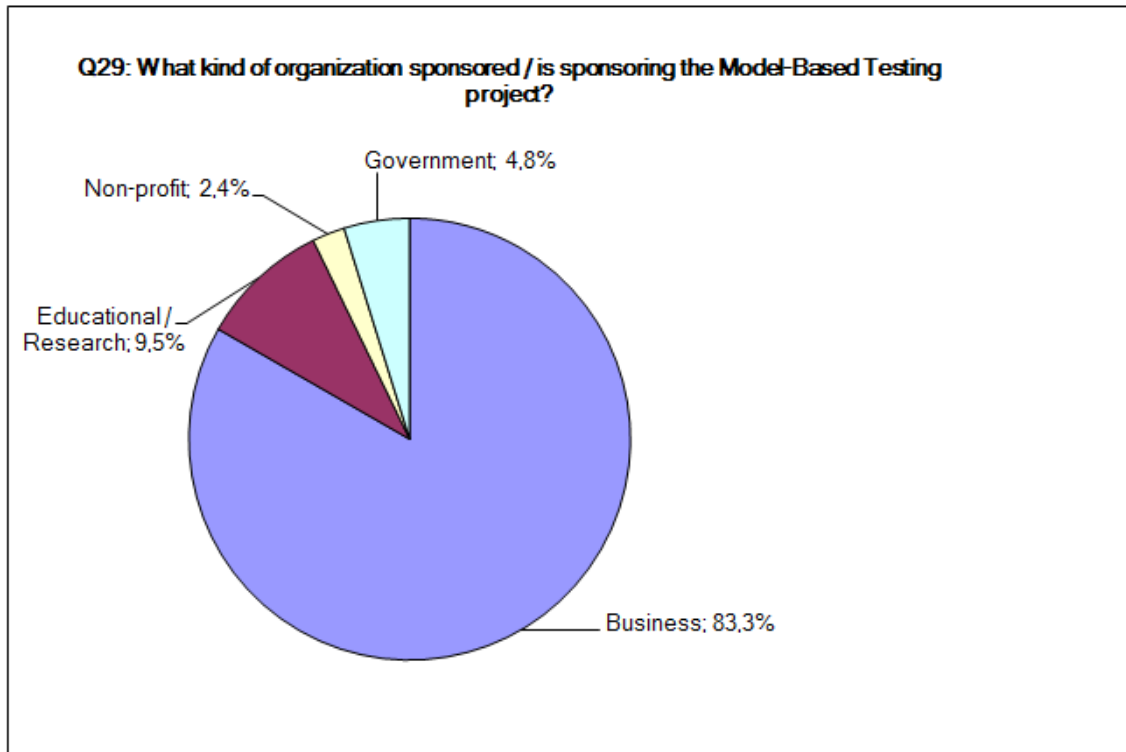
**Answer Options**

	2016	2014
Enterprise IT (including packaged applications)	30,0%	30,0%
Embedded controller (real-time)	7,5%	27,1%
Embedded software (not real-time)	10,0%	11,4%
Software Infrastructure	10,0%	5,7%
Web application	35,0%	18,6%
Gaming	0,0%	2,9%
Communications	7,5%	4,3%

**Other answers:**

- Telecommunications, Mobile Networks
- Aerospace
- Consumer electronics

Question 29: MBT project sponsor



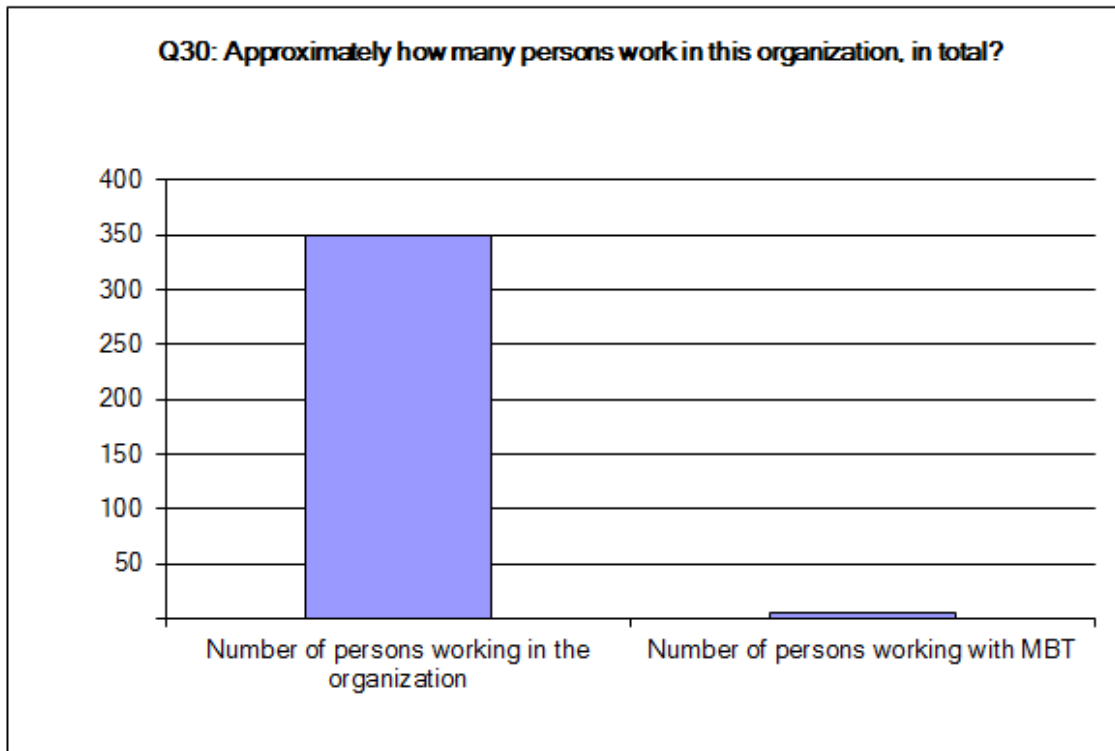
**Answer Options**

Business  
Educational / Research  
Non-profit  
Government

	<b>2016</b>	<b>2014</b>
Business	83,3%	86,2%
Educational / Research	9,5%	9,2%
Non-profit	2,4%	3,1%
Government	4,8%	1,5%

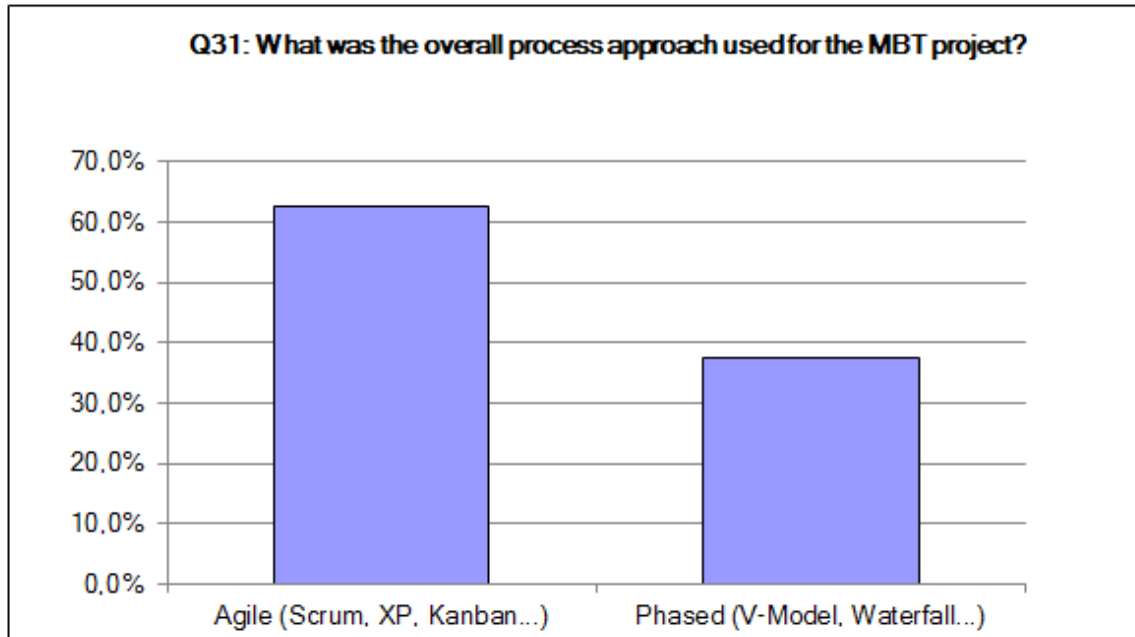


Question 30: Size of organization



	2016	2014
<b>Number of persons working in the organization</b>		
Average	27.010	12.236
<b>Median</b>	350	15
Minimum value	10	3
Maximum value	200.000	3.000.000
<b>Number of persons working with MBT</b>		
Average	120	325
<b>Median</b>	5	5
Minimum value	1	1
Maximum value	2.000	200

Question 31: Overall process approach



**Answer Options**

Agile (Scrum, XP, Kanban...)  
Phased (V-Model, Waterfall...)

**2016**

62,5%  
37,5%

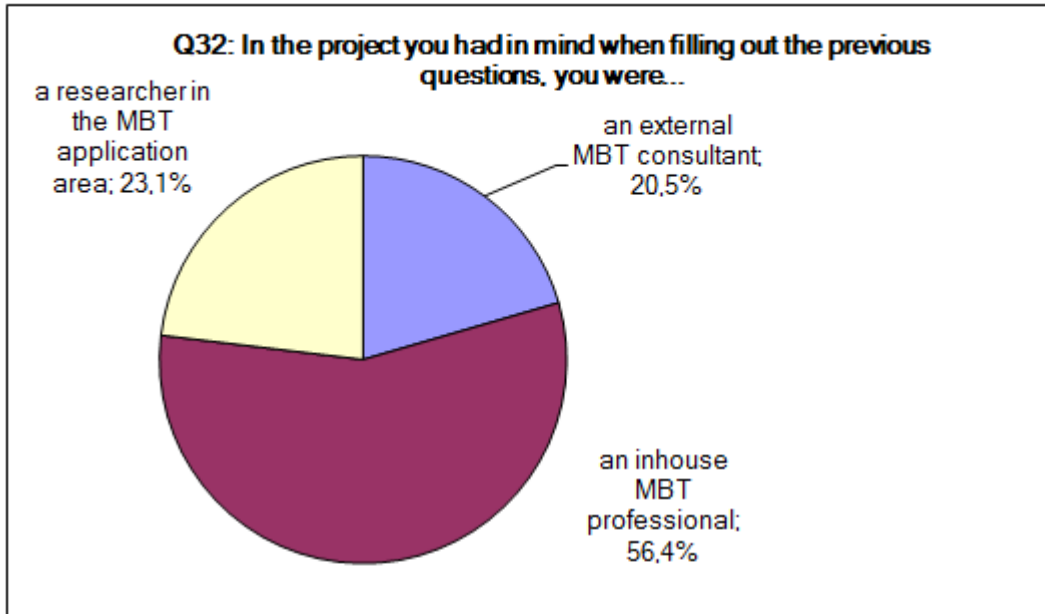
**2014**

44,4%  
55,6%

**Other answers:**

- We use for all the model, waterfall, V and agile

Question 32: Role of respondents



**Answer Options**

- an external MBT consultant
- an inhouse MBT professional
- a researcher in the MBT application area

**2016**

- 20,5%
- 56,4%
- 23,1%

**2014**

- 17,9%
- 59,7%
- 22,4%

**Other answers:**

- test automater
- Automation Manager evaluating MBT