Erik van Veenendaal

# Test Design Techniques to Mitigate Product Risks

Column

In reviewing the contributions to this issue of "Testing Experiences", I noticed that many authors start immediately by explaining how to use and apply a specific test design technique. However, we should not forget why we are doing this, i.e. what the objective is. The objective is never to "just" use a test design technique, the objective is to use to right test design technique that supports in mitigating product risks, either functional or non-functional.

## Risk-Based Testing

In risk-based testing, risk identification, risk analysis, and establishing risk mitigation activities are the foundation for defining the test approach [4]. The level of risk associated with each risk item determines the extent of the testing effort (i.e. mitigation action) associated with each risk. Some safety-related standards prescribe the test techniques to be used and degree of coverage to be achieved based on the level of risk (see below).

With respect to product risks, testing is a way to mitigate those risks. To the extent that defects are found, testers reduce risk by providing awareness of defects and opportunities to deal with the defects before release. To the extent testers do not find defects, testing reduces risk by ensuring that, under certain conditions (e.g. the conditions tested), the system operates correctly.

## Test Design Techniques

One option to mitigate product risks is to use test design techniques. The level and type of risk should be a major parameter in varying the test intensity by using different test design techniques, e.g. using the decision table technique on high risk test items and using "only" equivalence partitioning for lower risk test items, or using full decision tables for high risk test items and collapsed decision tables for low risk test items, etc. Risk (both risk level and risk type) should be a primary driver for choosing a test design technique or a variant within a certain test design technique. A test approach should be risk-based!

The greater the risk, the greater the need for more thorough, intense, and formal testing. For example, choosing to use two boundaries or three boundaries with boundary value analysis should be a risk-based decision. Testing with three boundaries is more thorough (intense), but it takes more effort which can only be justified if it is mitigating a higher level of risk. The commercial risk when releasing a product may be influenced by quality issues (so more formal test design techniques would be appropriate) or by time-to-market issues (so exploratory testing would be a more appropriate choice).

Of course, risk is not the only factor (albeit a very important one) when choosing the test design techniques to be used. The decision will be based on a number of factors, both internal and external such as [2]:

- Internal factors
  - Models used
  - Tester knowledge and experience
  - Type of defects expected
  - Documentation available
  - Life cycle model
  - Life cycle phase, e.g. new development or maintenance
- External factors (in addition to risk level and risk type)
  - Customer/contractual requirements
  - Type of system
  - Regulatory requirements
  - Time and budget

## Risk-Based Test Approach

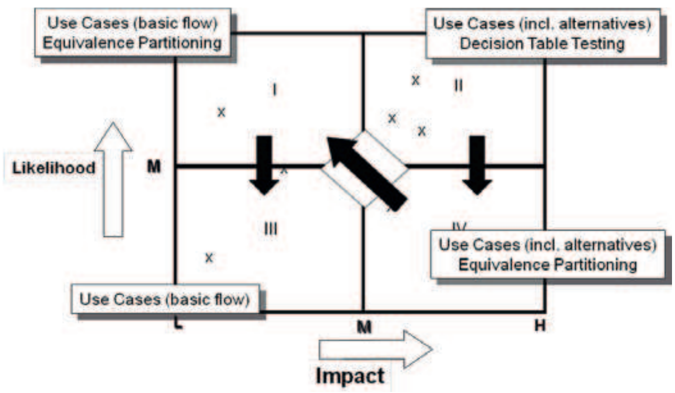


*Figure 1. Example of differentiated risk-based test approach for system testing*

To explain what is meant by a risk-based test approach, the following simplified system testing example is provided. The system testing approach based on a product risk matrix [4] is shown in Figure 1. The example shows that the most critical items, quadrant II, are testing by means of using use cases (including alternative flows) and the thorough test design technique of decision tables. The approach is scaled down for the second highest risk level, quadrant IV. (Remember, system testing will primarily focus on business risk). Use cases (including alternative flows) are still applied in quadrant IV, but decision tables are now no longer considered applicable. Instead, equivalence partitioning is used as a test design technique which normally defines a lower number of test cases than the decision table technique. Use cases are still used for quadrant I, but only the main flow will be executed and equivalence portioning is again applied as a test design technique. For quadrant III only the main flow of the use cases will be tested.

Another useful example of varying the test design techniques based on risk level and risk type can be found in IEC 61508[3]. An excerpt of this standard showing how it differentiates test techniques according to the software integrity level (SIL), another term for risk level, is shown in Table 1. This standard covers both static and dynamic test techniques and has specific tables for the various test levels and also for maintenance testing.

| Test Techniques | SIL 1 | SIL 2 | SIL 3 | SIL 4 |
|---|---|---|---|---|
| Stress testing | R | R | HR | HR |
| Performance testing | HR | HR | HR | HR |
| C/E graphing | – | – | R | R |
| BVA/EP | R | HR | HR | HR |
| Error guessing | R | R | R | R |
| Structure-based testing | R | R | HR | HR |
| Error seeding | – | R | R | R |

*Table 1. IEC 61508 Software Integrity Levels (R: Recommended, HR: Highly Recommended)*

Yet another example comes from DO-178b [1]. This standard also uses the approach that the intensity of tests to be performed should be derived from the level of risk. These standards prescribe the test approach to be used for each safety level as well as adequate completion criteria (see Table 2 for an example). The test professional should be aware that there are useful standards available, e.g., IEC 61508 [3] and DO-178b [1] that can provide support and inspiration when defining a differentiated risk-based test approach using test design techniques.

| Risk level | Risk mitigation component test approach |
|---|---|
| Low | No requirements |
| Medium | Statement testing and coverage measurements |
| High | Decision testing and coverage measurements |
| Critical | Modified condition decision testing and coverage measurements |

*Table 2. Risk-based approach to component testing [1]*

## Focus on Product Risks

It goes beyond the scope of this paper to explain in detail all the test design techniques mentioned, how they relate to test intensity, how they relate to each other, and how they can be internally varied. It should, however, be clear that a detailed understanding of test design techniques is required in order to be able to define a thorough test approach.

Many testers are technically-oriented people and tend to sometimes lose themselves in the technicalities of test design techniques. They may well design and write great test cases, but are they really necessary and the right ones? The main theme of this paper is that product risks should be one of the main drivers in choosing whether test design techniques are needed, which ones are needed, and how they should be applied.

Always think about why you are applying test design techniques and what the objectives are. Test design techniques are never the objective, they are only a means to an end. Focus on things that matter in building a great product. I believe this is what is meant by the agile manifesto statement "working software over comprehensive documentation". The use of test design techniques is certainly not a bad thing (on the contrary), but use them where it matters and where they have added value. Use them in an agile, business-like and risk-focused way.

## References


[1] DO-178b (1992), *Software Considerations in Airborne Systems and Equipment Certification, Requirements and Technical Concepts for Aviation*, RTCA SC167

[2] D. Graham, E. van Veenendaal, I. Evans and R. Black (2008), *Foundations in Software Testing – ISTQB Certification, 2nd edition*, Cengage Learning

[3] IEC 61508 (1998), *Functional Safety for electrical/electronic/ programmable electronic related systems*, Industrial Electrical Committee

[4] E. van Veenendaal (2012), *Practical Risk-Based Testing, The PRISMA Approach*, UTN Publishing ■


## > about the author


**Erik van Veenendaal** (*www.erikvanveenendaal.nl*) is a leading international consultant and trainer, and a widely recognized expert in the area of software testing and quality management. He is the founder of Improve Quality Services BV (*www.improveqs.nl*). He holds the EuroSTAR record, winning the best tutorial award three times! In 2007 he received the European Testing Excellence Award for his contribution to the testing profession over the years. He has been working as a test manager and consultant in various domains for more than 20 years. He has written numerous papers and a number of books, including "Practical Risk-Based Testing: The PRISMA Approach" and "ISTQB Foundations of Software Testing". He one of the core developers of the TMap testing methodology and a participant in working parties of the International Requirements Engineering Board (IREB). Erik is also a former part-time senior lecturer at the Eindhoven University of Technology, vice-president of the International Software Testing Qualifications Board (2005–2009) and currently board member of the TMMi Foundation. You can follow Erik on twitter via @*ErikvVeenendaal*.