Testing Cloud Services. How to test Saas, Paas and Iaas

Kees Blokland
Test Architect at Polteq

Martin Pol
Test Architect at Polteq

Jeroen Mengerink
Test consultant, Teacher, Researcher at Polteq

EuroSTAR Software Testing Community

1. Preface

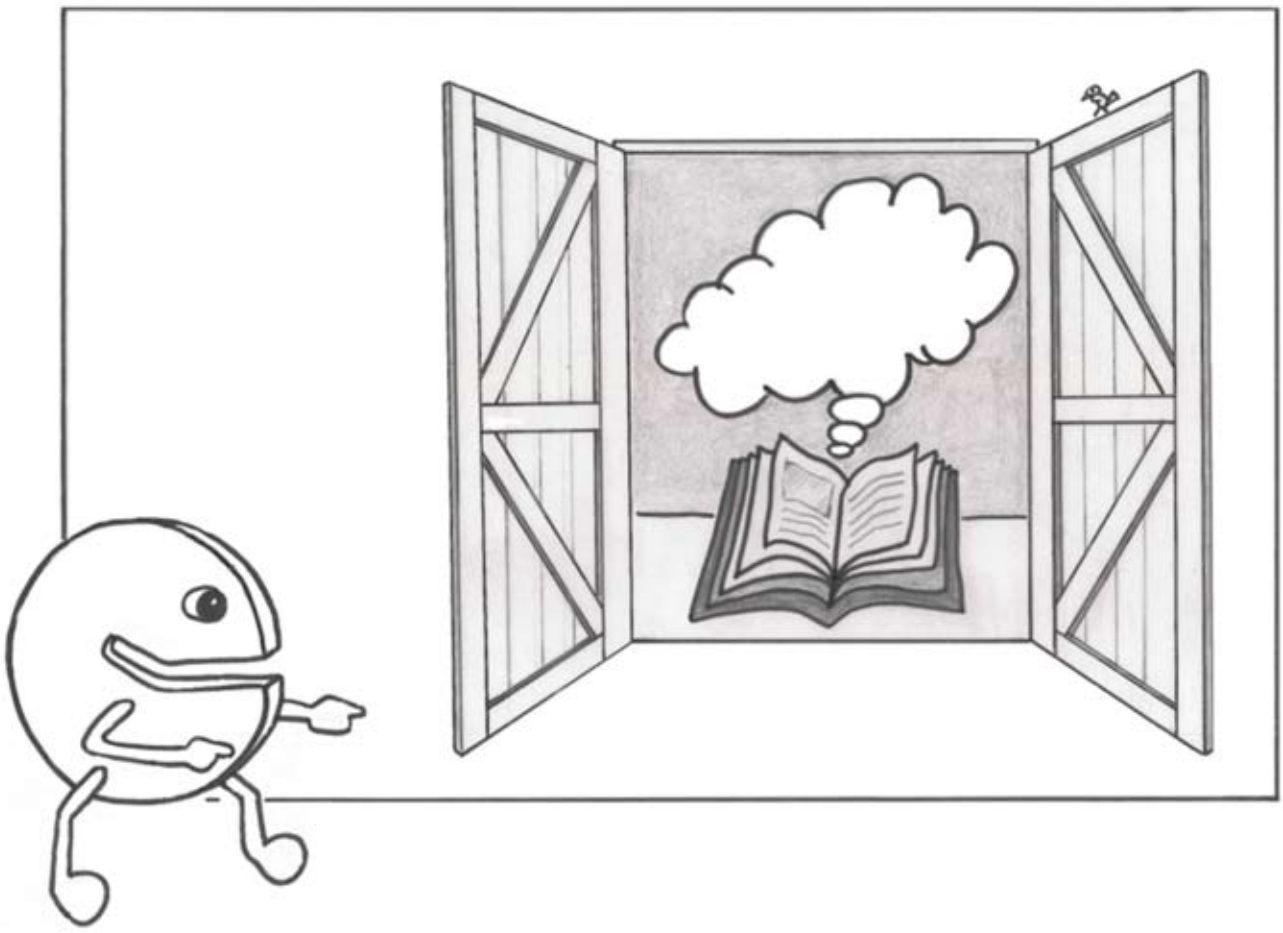
This e-book provides an introduction on testing cloud services and uses excerpts from the book "Testing cloud services. How to test Saas, Paas and Iaas.". The e-book is gives an impression on the general strategy for coping with the introduction of cloud services from risk and test perspective. It includes introductory text as well as detailed information from the book so that the reader gets an idea about the benefit the book can have.

2. Introduction

Cloud computing is a sea change in the way information systems are created and used. This simple fact is enough reason to investigate the phenomenon of cloud computing from a testing perspective. At the same time, it is useful to evaluate the testing profession from a cloud perspective because cloud computing offers interesting new options and solutions for a number of old bottlenecks.

More and more organizations are choosing to use services from the cloud. There are different reasons for this, but two come to mind immediately: new opportunities and cost reduction. An example of a new opportunity is worldwide access to data, at any time and with any device. And cost reduction occurs by sharing resources with other customers and not having to invest in resources (anymore).

This book is written mainly from the customer's perspective. People involved in introducing a cloud product and keeping it operational will gain a lot of knowledge, especially from the sections about mapping risks and taking measures to reduce those risks as much as possible. Therefore, project managers, test managers, and people in other test roles will benefit from reading this book. However, it does not limit itself to cloud computing customers



only. A supplier that is capable of taking on the viewpoints of customers is more able to offer successful services. By anticipating the risks a customer experiences, a test manager on the supplier side can achieve a competitive advantage for their company. In this book, a complete approach is described for the testing of and with services in the cloud. Innovations in the test profession come together with existing techniques and approaches. Testing Cloud Services contains a lot of test measures that are applicable within, but also outside, the cloud context.

Chapter 3 of this e-book provides a limited introduction to cloud computing.

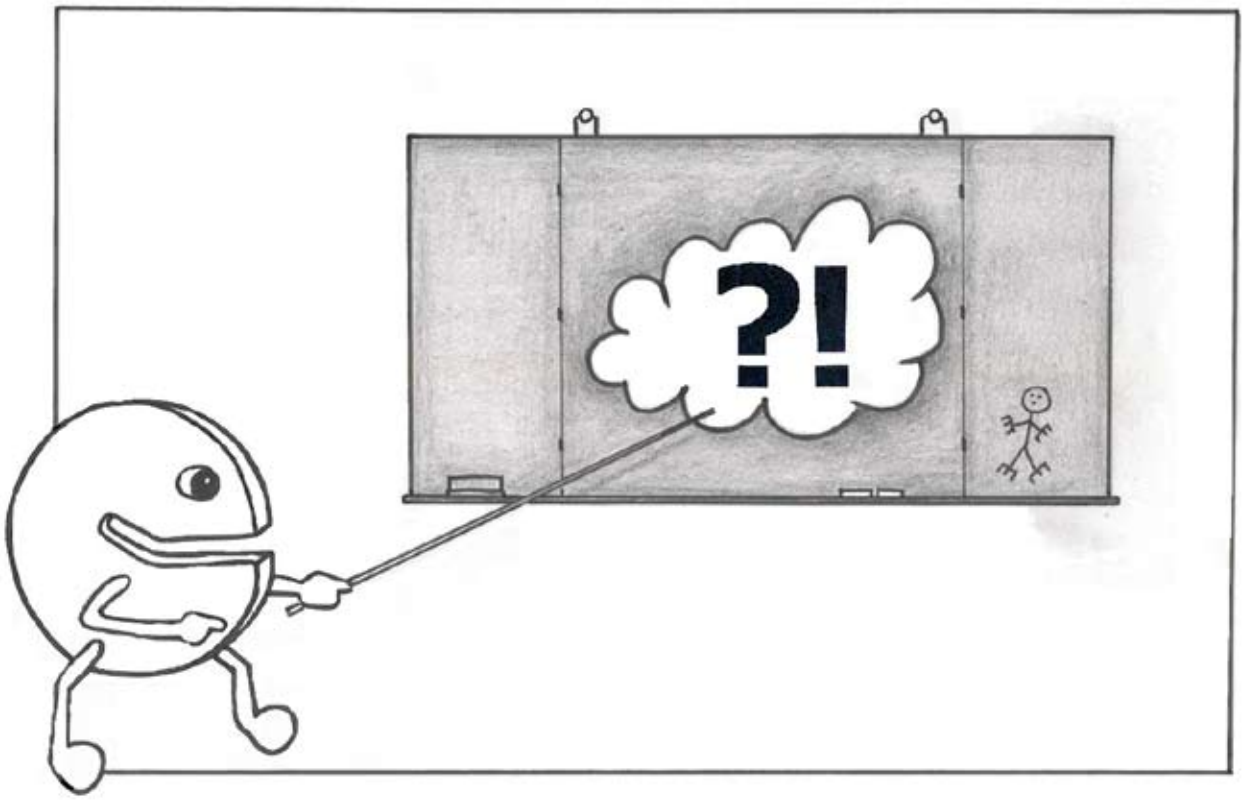
Chapter 4 is an introduction to the role of the test manager when dealing with the introduction of cloud services.

Chapter 5 explains the risk based strategy to determine the test measures that could help mitigating cloud related risks. Two risk areas (performance risks and legislation/regulations risks) are included to give examples of typical cloud risks.

Chapter 6 contains three examples of test measures (testing during selection and multi platform testing). The test measures chapter is the biggest chapter in the book (about two-third of the whole book).

3. What is Cloud Computing?

With cloud computing, a world of new opportunities that can be fitted into the current IT landscape arise. For a good understanding of the terms and concepts used in this book, a practical definition for cloud computing is essential. For this, we use the definition that is put forward by the American National Institute



of Standards and Technology (NIST).

NIST definition of cloud computing

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model is composed of five essential characteristics, three service models, and four deployment models.

The characteristics, service models, and deployment models (henceforth called implementation models) mentioned in the definition are separate dimensions of cloud computing that can occur in any combination. The risks that are described in this book are a direct result of one or more of the essential characteristics of cloud computing, the service model that is chosen, and the implementation model.

The five essential characteristics are:

On-demand self-service

Customers can configure computer facilities themselves, without human interaction with the service supplier. The services are easily available and can be obtained directly over the Internet.

Broad network access

Services are offered on a network. When standard protocols and standard formats are used, it is possible to obtain these services on different resources, such as PCs, laptops, tablets, and mobile phones.

Resource pooling

Multiple customers share the supplier's infrastructure through a rental model. The resources are appointed dynamically. This is done depending on demand from the customer. The exact location of the infrastructure is not important to the customer, though in general, the customer can set some preconditions, such as, for example, a particular country or a specific data center. The types of computer resources one has to think about—among others—are storage, computing capacity, memory, network bandwidth, and virtual environments.

Rapid elasticity

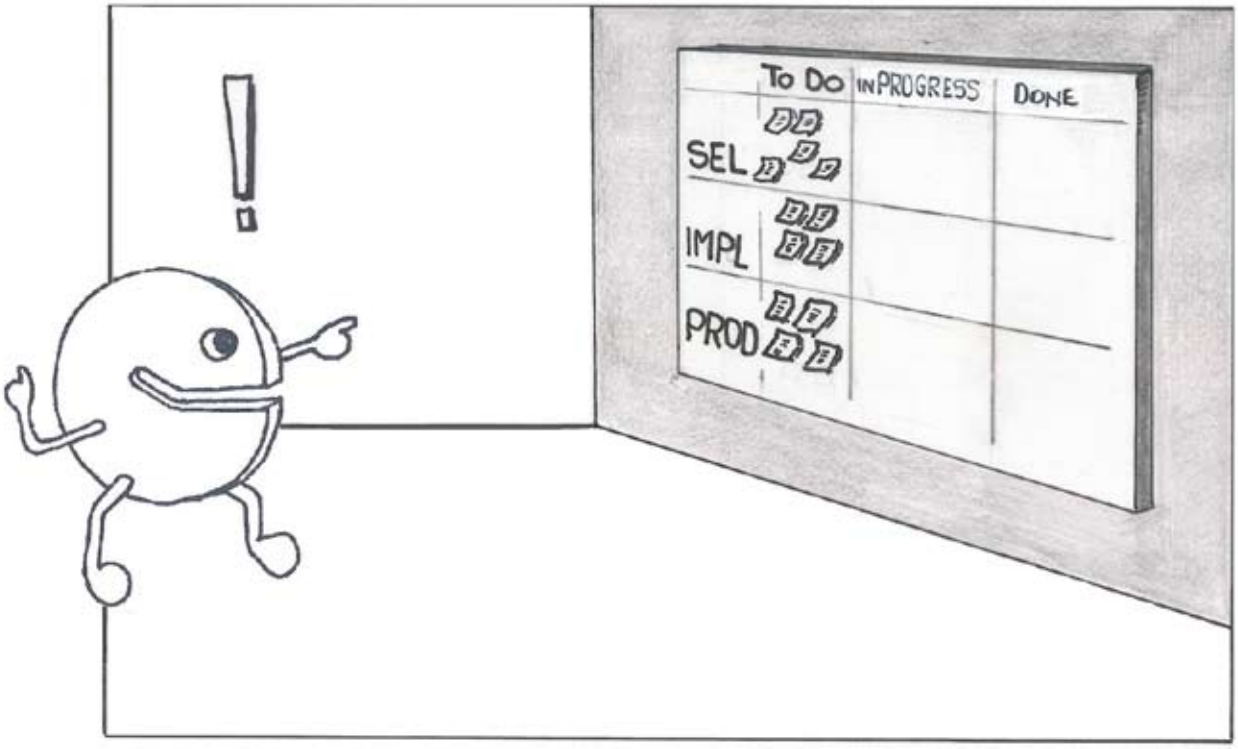
Services can be configured and released quickly and often automatically in an elastic fashion. This offers the capability of quickly scaling up and down. The customer experiences this as the apparently unlimited ability to obtain services at any moment and in any desired quantity.

Measured service

Systems check and optimize the use of the underlying infrastructure. Here, for example, the usage of the following is measured: storage capacity, computing capacity, bandwidth, and active user accounts. The result is transparent for the supplier and the customer and as a result is a fair basis on which to invoice.

4. The Role of the Test Manager

The use and development of software continually changes, and that affects the activities and position of the test manager, as seen in projects where Agile methods are applied. The rise of cloud computing provides a new impetus to the role of the test manager. What is striking in Agile, as well as in cloud computing, is a shifting and broadening of focus. Managing functional software testing, originally the core of testing, is just one of the many tasks of the modern test man- ager. On one hand, this is because less and less traditional testing is done under the supervision of the



test manager; instead it is done in Agile teams or by the service supplier. On the other hand, it is because nonfunctional requirements are claiming an increasing role. A documented test basis (requirements and specifications) is no longer the starting point for testing. The test manager increasingly bases measures on the desires and needs of the operation and users.

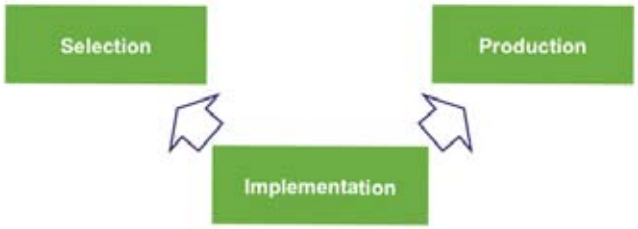
Thus, the test manager, as it were, climbs in the V-model and gets a broader responsibility: a process is successful when the business process and the users are supported properly.

Testing Cloud Services gives substance to this broadening of the role and offers test managers plenty of room to further develop themselves in the testing profession. The test manager, for instance, needs to be involved during the selection of services. In this stage, it becomes clear which requirements can be met and which cannot be met. After making a choice, one is stuck with the quality and options of the selected services, and their accompanying risks. However, this is not completely new: selecting a service is similar to selecting software packages. Here the test manager also provides a valuable contribution to the selection process.

The involvement of the test manager in the production phase is entering a new stage with cloud computing. In an IT landscape of connected systems, an increasing need for continuous system integration emerges. This is comparable to continuous component integration, where regularly (often daily) a new soft- ware build is compiled. An automated regression test in this case needs to ensure the continuity of the development process. With continuous system integration, something similar is going on: systems in production are updated all the time (patches, upgrades, new releases), which requires a more or less permanent regressive system integration test to ensure continuity in production. Where there is knowledge of the exact changes to the customer's own software, in cloud computing the customer can be confronted with unannounced and hardly documented changes. The continuous system integration test will be necessary as a backstop. The situation might even arise in production that makes choosing another service or service supplier necessary—for example, because testing shows that guarantees in continuity are not met.

In addition to a role during service selection

and in production, there is a job that needs to be done during service implementation. This is like the traditional role of the test manager: testing and giving advice on getting information systems into production (operation). All in all, the role of the test manager has broadened (see figure below).



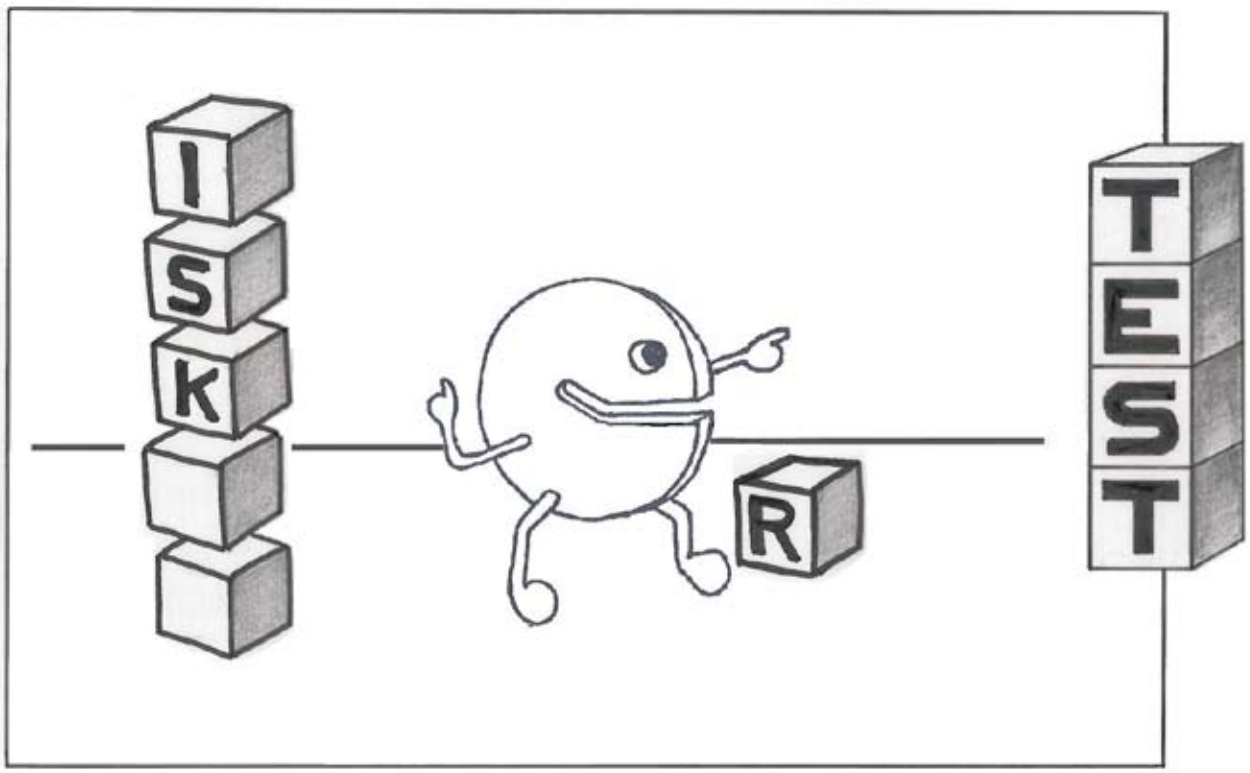
Broader role of the test manager

5. From Risk to Test

From the essential characteristics of cloud computing, a number of risks can be immediately determined. For example, broad network access means that most services are offered over the Internet, which introduces security risks. A further example is that resource pooling means that devices are shared with other customers, so response times of the service are affected by these other customers. This introduces a performance risk. The chosen implementation model also affects the risks, especially the severity of the risks. The security risk in the private cloud is less than in the public cloud where other customers have access to the same service. To determine which test measures are needed, all risks need to be mapped. For testing of services, this is no different from traditional test processes. By conducting a product risk analysis on the service, the areas that are important enough to test, and how stringent testing needs to be can be determined. This chapter contains a collection of cloud-related risks. We indicate which test measures can be taken to cover every risk. Risks and measures are based on practice and are meant as a source of inspiration; they are not exhaustive, but do provide guidance. This is the basis of Testing Cloud Services and, as such, the starting point of the test approach.

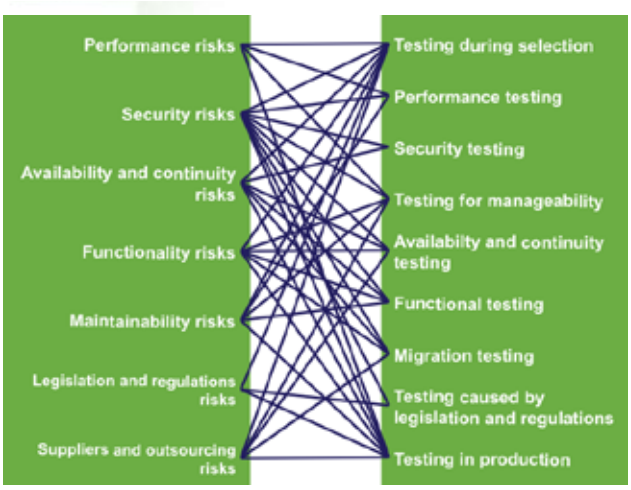
The following risk groups are identified in the book:

- Performance (example is this e-book)
- Security
- Availability and continuity



- Functionality
- Maintainability
- Legislation and regulations (example in this e-book)
- Suppliers and outsourcing

The relationships between risks and test measures are not one to one but vary. For one risk, more than one measure can be deployed, or one measure can cover different risks (see next figure).



Relationship between risks and test measures

5.1 Performance Risks

Risks concerning performance have been an important focus for a long time. With cloud computing, more communication is done over the Internet, which makes performance risks more significant. Think about the performance of a travel-booking website that is hosted in the cloud. When visitors to the website have to wait too long, they will quit and book their travel with another agency. The first travel agency suffers loss of revenue. In this example, it is easy to get an understanding of the importance of the Internet connection: it is used to reach the (systems of the) travel company. When different systems are used with the help of cloud computing, it is not clear right away that Internet connections are involved. The interfaces with external services and the use of the Internet intro- duce performance risks in all business processes that use these services. The risk of losing one customer is compounded by the risk that employees of the company are unable to book any travel and can't help any customers.

An important argument in choosing cloud computing is scalability of the service. When more capacity is needed (because performance or accessibility drops), it often can be obtained from the cloud quickly and automatically. Subsequently, it can be scaled down when the need decreases. This facility is called elasticity, and it calls for a new, cloud-specific test: does scaling capacity up and down work efficiently and effectively in practice?

The supplier's model for generating turnover is based on deploying IT resources as efficiently as possible: as many customers as possible using the least amount of resources will, of course, generate the most profit. The supplier will count on the fact that the peak load of the different customers will not coincide—whether this can be justified or not. This means that the service can be overbooked, especially when the service is more successful than anticipated, with the result that the performance of the service decreases. That is because the customers together stress the service more severely than was anticipated. This risk is larger for public and community clouds because the uncertainty of third-party use is largest. In the private cloud, IT resources are not shared with unknown customers and only one customer's services run on it, which lowers the risk in this situation. The table overleaf contains an example of performance-related risks.

Risk	Test measure (section number/s of the book)
Scaling does not suffice: – Scaling up manually does not work. – Scaling down manually does not work. – Manual scaling causes disruptions. – Scaling up automatically does not work. – Scaling down automatically does not work. – Automatic scaling causes disruptions. – Insufficient growth potential.	5.1.3, 5.2.4
<original book contains many other performance risks>	

Performance risks

5.2 Legislation and regulations risks

When using the new opportunities that cloud computing offers, companies need to take into account certain legislation and regulations, such as, for example, local and international privacy legislation for the storage and processing of personal data. Cloud computing is a worldwide market that does not consider national borders, and as a result resides under more than one jurisdiction. Because of this, the user of the service will have to obey different and sometimes even conflicting legislation. Not complying with a law will, of course, result in a risk.

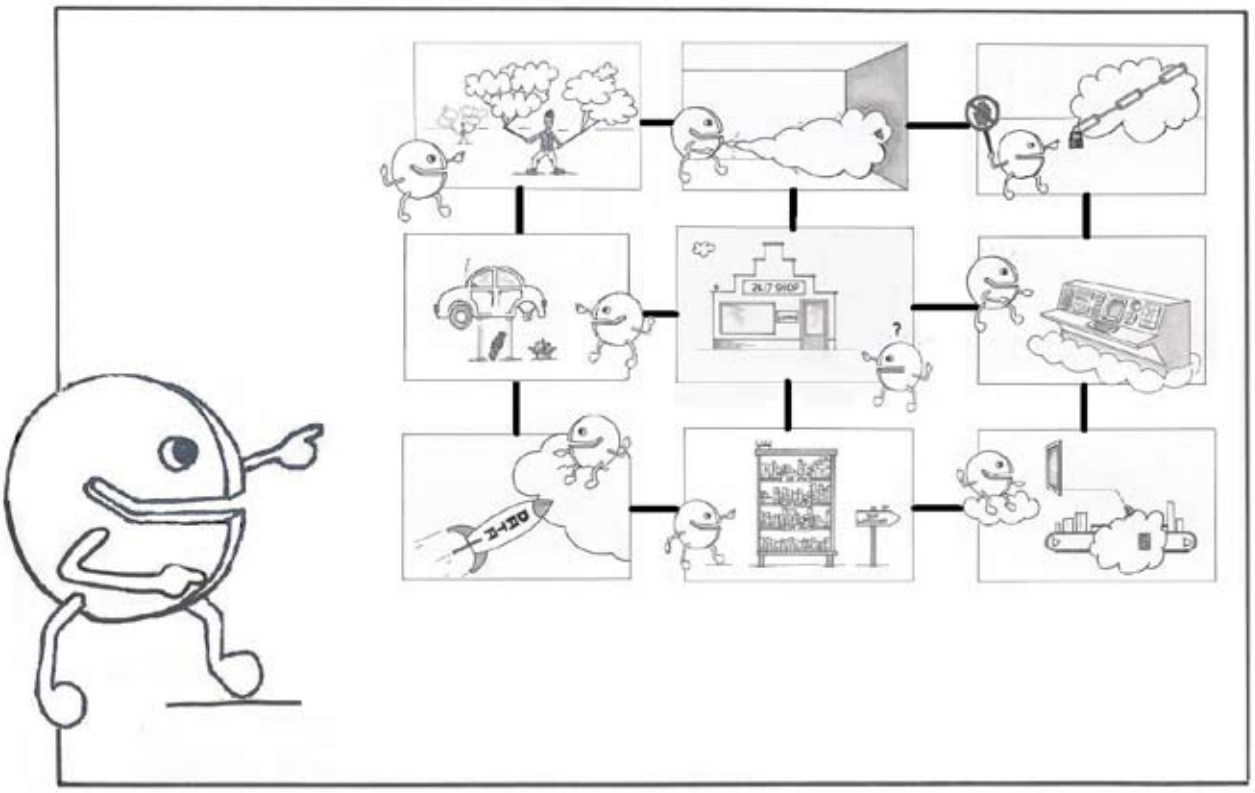
Legislation and regulations can also be risks in and of themselves. There are situations in which company data is securely stored but the government of the country in which the data is stored can still gain access.

The table below contains examples of legislation- and regulations-related risks.

Risk	Test measure (section number/s in book)
Processing and storing data do not comply with some laws: – Laws in home country – Laws in other countries in which the data can reside	5.1.3, 5.8
Countries have conflicting laws.	5.1.3, 5.8
<more risks in original book>	

Table 4–6 Legislation and regulations risks

6. Test Matters



To address the risks in the previous chapter, test measures are needed. In describing the test measures, we pay a lot of attention to test management. In addition, for certain test measures, test specification and test execution is addressed. Testing Cloud Services presents a pick and mix of test measures: depending on the risks, one or more measures are picked and mixed. The starting point for this chapter is the outcome of the product risk analysis of the service. The result is a list of relevant detailed risks with appointed classification (high/medium/low). For each risk group, this list allows the test manager to determine appropriate test measures. In this chapter, we will describe the following test measures:

- Testing during selection (with an example is this e-book)
- Testing performance (with an example is this e-book)
- Testing security
- Testing for manageability
- Testing availability/continuity
- Testing functionality (with an example is this e-book)
- Testing migrations
- Testing due to legislation and regulations
- Testing in production

6.1 Testing During Selection

Generally, during selection, two or more service providers are considered. Too many options cost too much time and energy; too few options limit choice. With new services, it is possible that there is little choice, simply because few suppliers have yet to provide these services.

When a customer is dependent on one supplier, there is what is called a vendor lock-in. When problems arise with the supplier, continuity becomes a risk because there are few or no alternative suppliers to whom the service could be transferred. A vendor lock-in situation puts a customer in an unfavorable situation regarding contract negotiations.

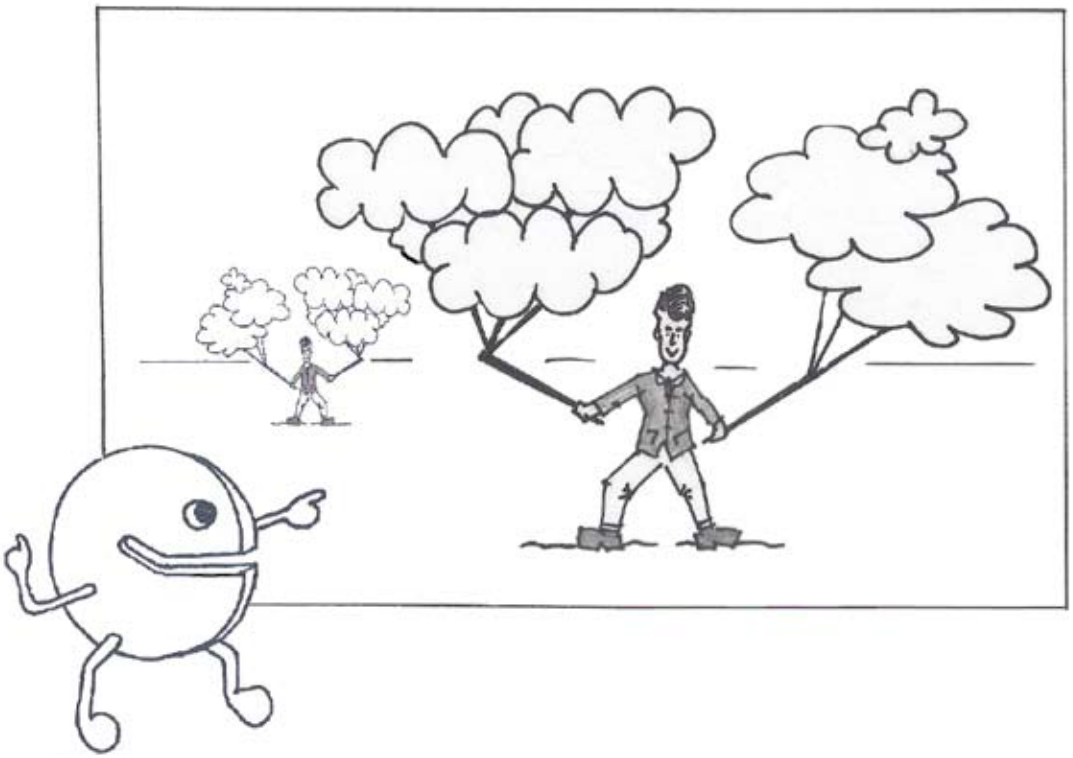
Selecting the most appropriate service and supplier is a time-consuming activity for which budget must be available. Sometimes senior managers or board members of suppliers and customers reach agreement beforehand, and lip service is paid to the selection process.

A pitfall in selection is that the initial objectives may fall by the wayside. This can happen when customers are presented with attractive new services with different options that do not prove to better fit the selection criteria.

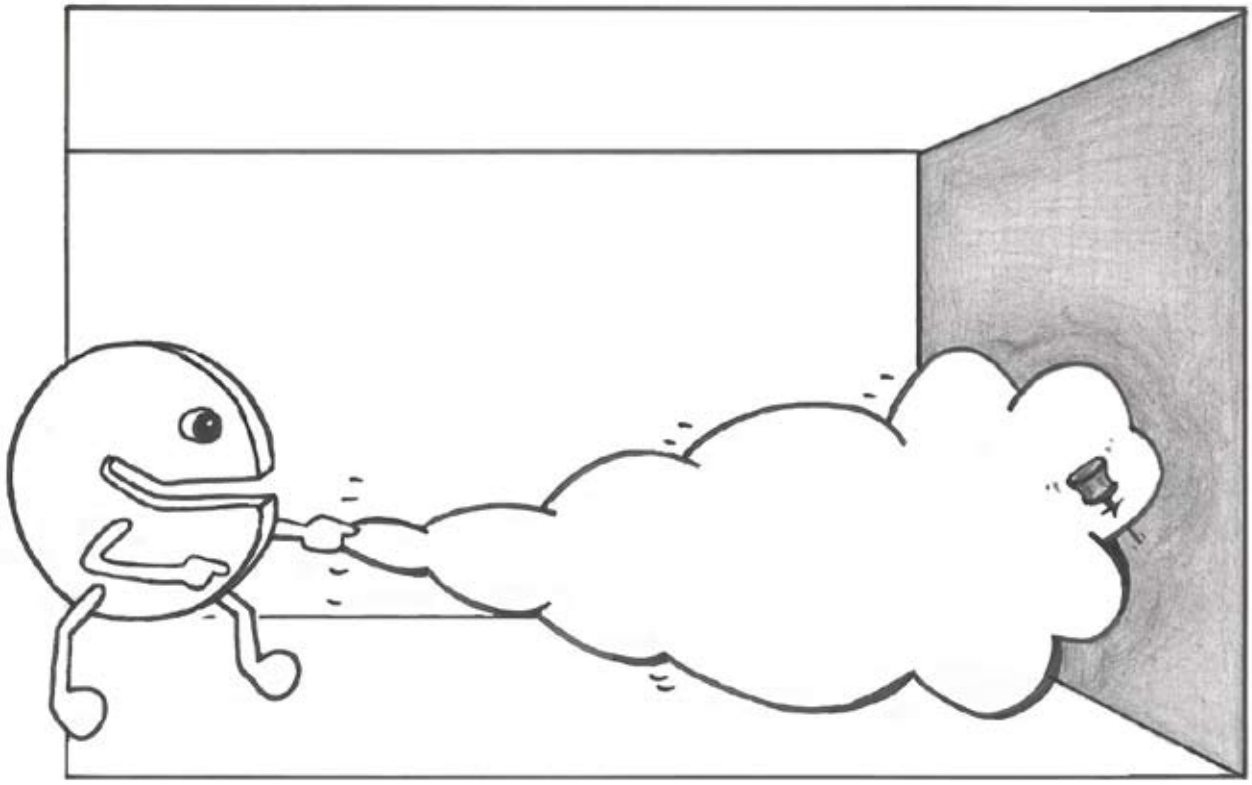
Another pitfall is to make concessions to the criteria so that a particular service better fits the requirements.

The selection process is divided into the following steps:

- Include cloud-related aspects.
- Determine completeness and controllability of selection criteria.
- Assess services and suppliers. Examples of measures that can be taken during the selection process are: "Inspect specifications and terms", "Ask for references and certificates", "Perform audits and inspections", "Perform a Proof of concept", "Simulate E2E business processes".
- Issue selection advice. Every criterion that is not clearly met represents a risk. A test manager can produce a selection report, comparable to a test report, which summarizes these risks. Not all risks are directly related to IT, such as costs and legal risks. For these risks, input from others is required.



6.2 Performance Testing



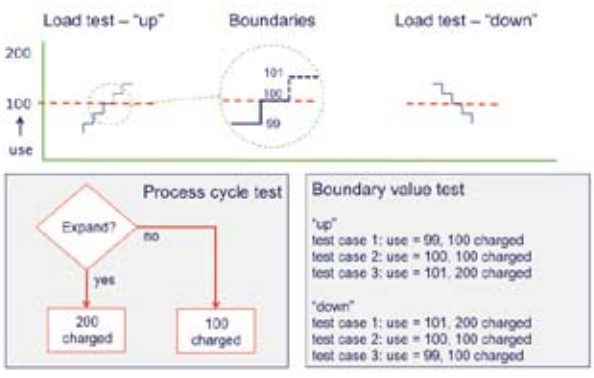
6.2.1 Testing elasticity

Testing elasticity is a new aspect of performance testing. The objectives of the test are to determine whether the performance of the service meets the requirements across the entire load spectrum and whether the appointed service capacity scales with the service load. The pay-per-use model is often linked to the elastic aspect of the service. The administrative (financial) handling can become part of testing elasticity.

The approach to this test comprises executing a load test with the load increasing beyond the scaling boundary (scaling up) and then the load decreasing below the scaling boundary (scaling down). Manual actions may need to be performed as part of the test procedure in a process cycle test. Behavior on possible boundaries of packages also needs to be addressed.

The following test approach covers a situation where scaling up and scaling down happens automatically (true elasticity). During the elasticity test, three test techniques are combined, as shown in the figure below.

- Load tests with a load profile to vary loads
- Boundary value analysis at the boundaries of scaling up and scaling down
- Process cycle test for the administrative process (this will include pay-per-view invoicing)

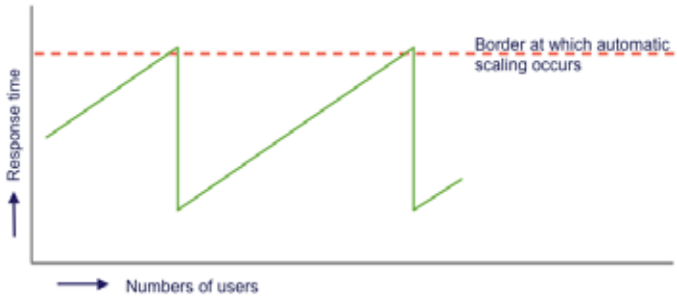


Combining test techniques

The starting point is the assumption that there is a direct connection between capacity use and invoicing. Preferably the load will be increased in such a way that on the supplier side, something actually happens to keep up with the increase and decrease of the load.

Stepwise, the load is increased to the determined maximum; this continues for long enough to be sure that the higher usage will be invoiced. After a predetermined amount of time, the load is decreased again. This also continues for long enough to be sure that the reduced usage will show up in the invoice.

The expected result with regard to the load is that performance complies with the requirements (choose an important operation from the load test as the test case) and that no functional problems occur. The response times are expected to increase and then show a significant decrease after automatic scaling up, as is shown in the figure below.

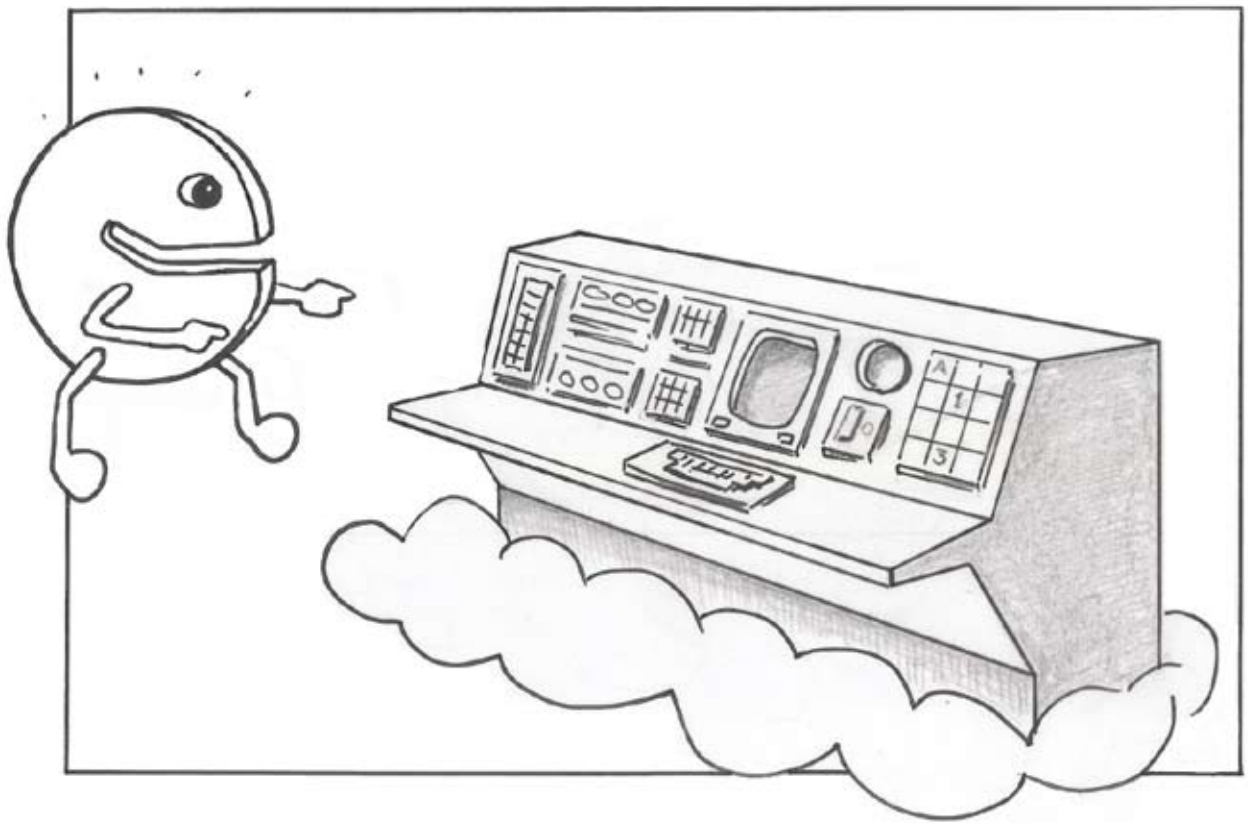


Expected result

When the boundary at which capacity will be scaled up is known, boundary value analysis can be performed. The objective is to determine whether there are any disruptions to the service around adjusting capacity.

In principle, one would have to wait for the supplier to send his invoice. As a result, it may take a while until the result of the test is known. When there is an online option to monitor the current invoice in real time, this can be evaluated during the test. Potentially, the supplier's help can be called upon to obtain a copy of the current invoice.

6.3 Functionality Testing



6.3.1 Multi-platform testing

One of the main advantages of working in the cloud is the opportunity to access the service using different resources. In addition to the new technology that makes this possible, two other coinciding trends have arisen. These are the New World of Work and the BYOD phenomenon. With the New World of Work, the working location is not, per se, bound to the office and people increasingly work outside office hours. This requires access to the IT facilities over the Internet. More and more people like to work with their own equipment rather than with the resources made available to them by their employer. Laptops, tablets, and smartphones are in this category. Testing different combinations of plat- forms and services is summarized here under the term multi-platform testing.

Which variations can be applicable (see figure below)?

- Browser types (IE, Safari, Chrome)
- Variations of apps
- Different operating systems (Windows, iOS, Linux, etc.)
- Different mobile devices (tablets, smartphones)
- Different versions of hardware

Considerations for the supplier

It is in the interest of the supplier to test the service with common platforms (portability test). A complete functional test per platform is not feasible. There are choices to be made. There are a number of considerations on which to base choices:

- User statistics. Which platforms and versions are used the most?
- New platforms. A supplier may want its profile connected with supporting a new platform that is not yet found in user statistics.
- Platforms that share the same kernel. Testing one platform per kernel can be sufficient to cover the most important risks.
- Focusing on the most-used paths through the service.
- Testing one of each kind of technical facility (client-side scripts, forms, media, control types).
- Different technical facilities in apps (global positioning system (GPS)/no GPS, camera/ no camera, Wi-Fi/no Wi-Fi).
- Requesting all screens and menus once.

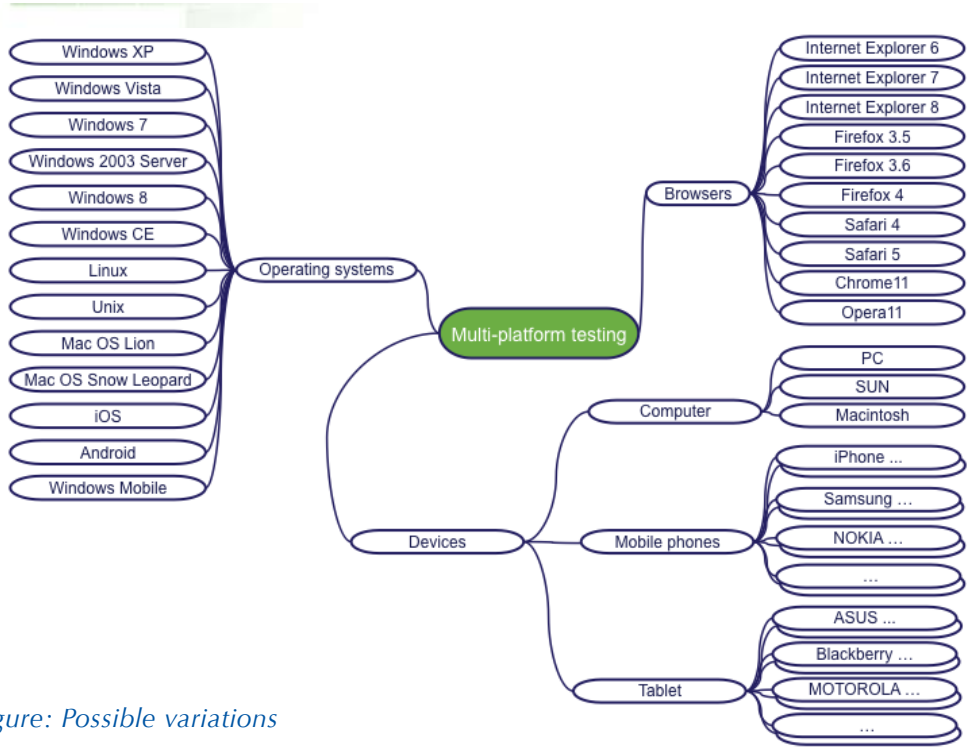


Figure: Possible variations

When the supplier follows W3C web standards closely in designing and building services and makes no or little use of specific properties and possibilities of platforms, portability risk is limited.

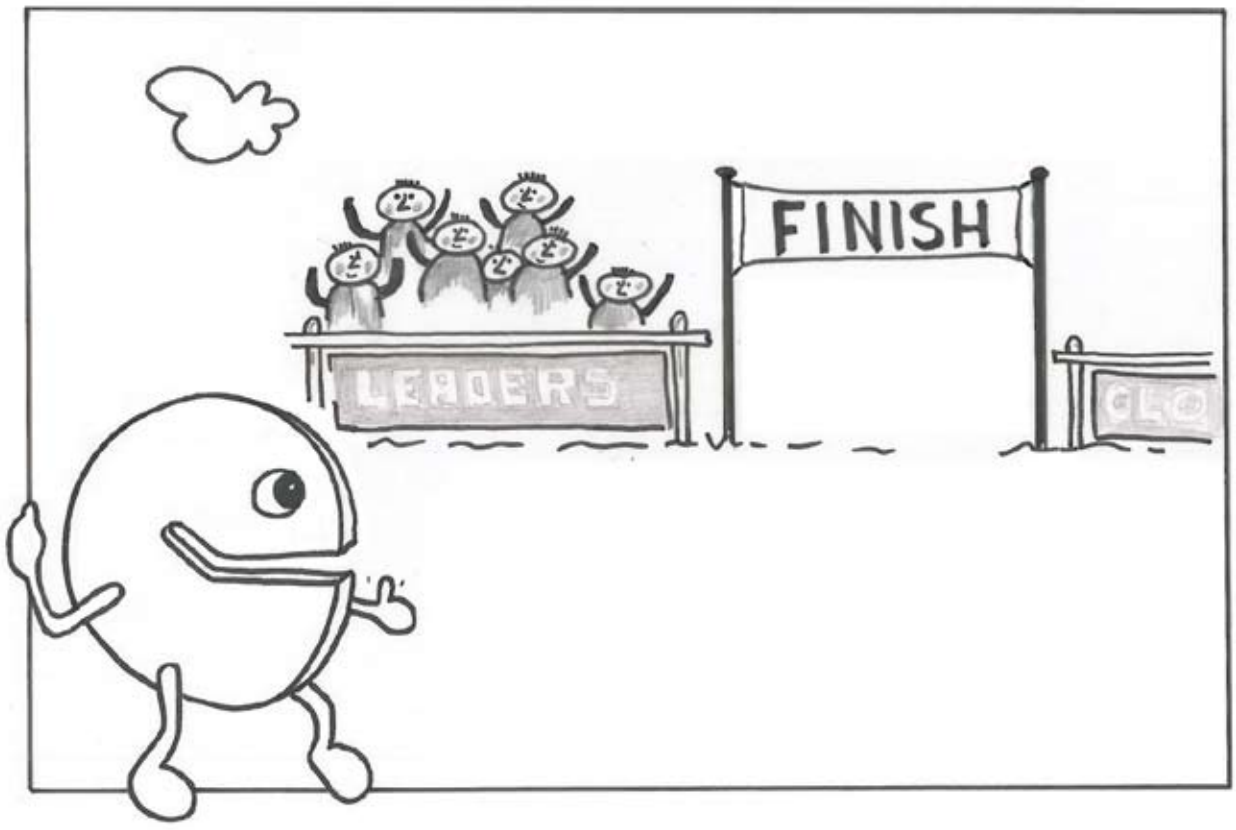
Testing the multitude of mobile devices is becoming a problem. In section 3.3 of "Testing Cloud Services", an example of test services that respond to this is described.

Considerations for the customer

A customer has different choices to make. Defining a BYOD policy is in most organizations still in its early stages. In this policy, a number of matters need to be defined. This leads to determining the following categories:

- The standard office setups used within the organization
- Other resources that are supported (for instance certain types of mobile devices)
- Resources that are allowed to be used but are not guaranteed
- Resources that are not allowed to be used The first and second categories are fully incorporated in the multi-platform test. For the third category, a sample can be applied to testing, or it is not incorporated in the test strategy at all. The fourth category is outside the scope, except when an organization wants to test if the "forbidden platforms" are denied. In determining what needs to be tested, the same considerations apply as for the supplier.

7. Completion



How can a book that is about something that still needs to reach full development be concluded? For starters, by establishing that practice will show to what extent the test measures fit and which further innovations are needed. More risks will undoubtedly be identified, existing risks will reveal to be less relevant than imagined, and the test measures will be adjusted accordingly. Further developments will be followed and various media additions to this book will appear. Testing Cloud Services offers the entire approach, from identifying risks to taking test measures.
With all the risks and test measures that are described in this book, the impression might be that going to the cloud is fairly risky. In other words, if this needs to be tested so extensively, is cloud computing such a good idea?

But here the universal rule is valid:

No Risk, No Test

Example. Based on the extent of the security risks, one can choose from the following options:

- **Not test at all.** This is valid for widely used and proven services such as email.
- **Hiring a test company.** This is advisable when the business stands or falls on proper functioning and/or it is plausible that malicious parties will try to gain access to company data.
- **Test in-house.** All other cases.

With the use of the approach in this book, the test manager can put together a generic test strategy that applies to the organization's context. Certain risks are, for instance, general and apply to every service that is introduced into the organization's IT landscape. Choices are made by each organization. Certain test measures will become a permanent part of the test strategies for services in the cloud, such as setting up a continuous E2E regression test. The good news for the test manager is that cloud computing offers plenty of opportunities. Broadening the role of the test manager fits in with the IT profession becoming more mature: the shift from "Do I get what is promised and are there no errors in it?" to "Is this a correctly functioning solution for the organization?" This requires involvement in all phases of the entire life cycle of providing IT services from the cloud. In addition, it also requires more knowledge in the field of testing nonfunctional requirements. The key is to select the tools that fit the context. This book is just the beginning of testing of and with services in the cloud: the warm-up is finished, but the match still needs to be played!

Biographies

Kees Blokland, Test consultant at Polteq

Prior to joining Polteq (www.polteq.com) in 2003 as a test consultant, Kees Blokland worked fourteen years for Lucent Technologies in various test positions, including test department manager. Now, as manager of R&D at Polteq, Kees develops their vision of testing and provides international test consulting. As senior test manager, Kees advises on test management, test improvement, and test outsourcing. He teaches a broad range of test classes, including ISTQB Advanced, TMap, and special topics, like Cloud, SOA, and outsourcing. Kees is a frequent speaker at conferences worldwide and one of the authors of the book: "Cloutest® Testing cloud services. The approach: from risk to test measures."

Martin Pol, Test Architect at Polteq

Martin Pol has played a significant role in helping to raise the awareness and improve testing worldwide. He has gained experience by managing testing processes and implementing structured testing in many organizations in different branches. He was responsible for the creation of the TMap testing approach and the Test Improvement Model, TPI. Books are available in many languages. Martin Pol chaired the Dutch SIGIST for 5 years and EuroSTAR for three times. He received the "European Testing Excellence Award". Martin is a regarded presenter at conferences and training sessions throughout Europe, North and South America, India, China and Australia. Martin is working for Polteq Test Services, based in The Netherlands and Belgium. He is still active in the testing practice every day.

Jeroen Mengerink, Test consultant, teacher, researcher at Polteq

Jeroen is a test consultant for Polteq. Next to his work for clients, he is involved in various test innovations. His main area of expertise is Agile. Jeroen teaches several test courses e.g. about Agile (CAT), SOA and Cloud. He is co-author of the book and approach Cloutest(r) on how to test when cloud computing is involved. He has contributed as a speaker to various events for Polteq and her clients. In international assignments he has presented the results of TPI assessments to senior management. He presented several times at events like Eurostar, ChinaTest and TestNet on a large variety of subjects.

Blokland/ Mengerink/Pol, "Testing Cloud Services, How to Test Saas, Paas & Iaas", Rockynook, ISBN 978-1-937538-38-5

Join the EuroSTAR Community...

Access the latest testing news! Our Community strives to provide test professionals with resources that prove beneficial to their day-to-day roles. These resources include catalogues of free on-demand webinars, ebooks, videos, presentations from past conferences and much more...

Follow us on Twitter @esconfs
Remember to use our hash tag #esconfs when tweeting about EuroSTAR 2013!

Become a fan of EuroSTAR on Facebook

Join our LinkedIn Group

Add us to your circles

Contribute to the Blog

Check out our free Webinar Archive

Download our latest eBook

www.eurostarconferences.com