A Risk-Based Test Strategy

Dr. Ingrid B. Ottevanger

IQUIP Informatica B.V.

PO Box 263, 1110 AG Diemen, The Netherlands

Tel: +31 20 660 6600

Fax: +31 20 695 3298

E-mail: i.b.ottevanger@iquip.nl

1. Introduction

The development of a test strategy is a means of communication with the customer commissioning the test on such matters as the organization of testing and the strategic choices that go with it. The test strategy indicates how testing is to be carried out. In order to make the best possible use of resources and time, it is decided on which parts and aspects of the system the emphasis should fall. The test strategy forms an important basis for a structured approach to testing and makes a major contribution to a manageable test process.

The customer who commissions the test will expect specific qualities of the system when in production, and wants to know whether the released system will meet these requirements. If the system qualitatively does not meet the requirements or only to a limited extent, this implies high damage for the organization, for instance since high rework costs will be needed or clients/users will be unsatisfied. Therefore, this situation forms a risk for the organization. 'Risk' in this paper is defined as:

A risk is the chance of an error[1] occurring (chance of failure) related to the damage expected when this error does occur

---

Testing covers such risks by giving insight into the extent to which the system meets the quality demands. When quality turns out to be insufficient timely measures can be taken, e.g. rework by developers. If the shipping of the system implies many risks for the organization, better testing is obvious as a solution. And the reverse also holds:

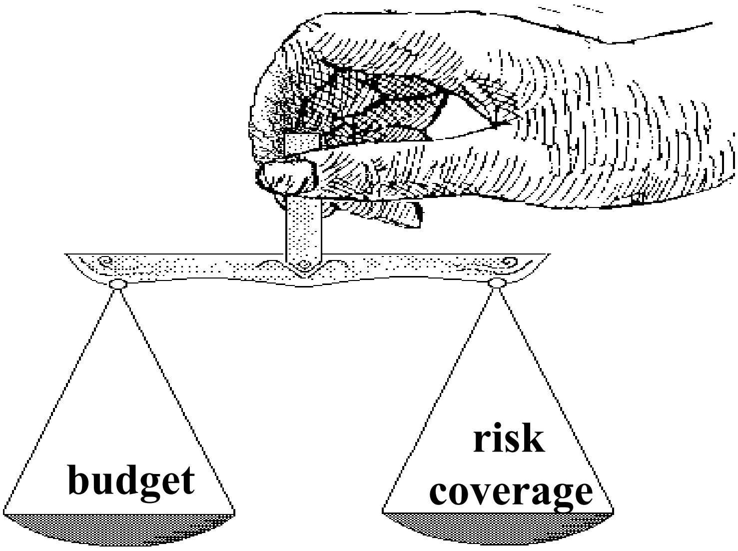
No risk, no test

Although in the above we refer to quality and risks in a general sense, there may be large differences depending on the situation. It is of great importance to

[1] The terms error, defect and failure are not used as exactly as IEEE advocates. In this paper error = fault or mistake (latent); defect = the manifestation of an error (overt); failure = the result or manifestation of one or more faults.

discuss this with the customer, and to translate the customer's wishes in this respect into the way testing will be performed. Thus, the test strategy is directed towards finding the optimal balance between the test effort to be exerted and the coverage required for the risks. To this purpose the risks are specified up to the level of quality characteristics and separate subsystem. In doing so it becomes possible to find a suitable test coverage for the assessed risks. Here a higher test coverage usually results in more test effort. In order to reach at the variation in test coverage needed, the use of more than one test specification technique (test design technique), each offering a specified test coverage, is crucial.

An analogy with insurances may clarify this matter a bit more. A person wants to cover a relevant risk and takes an insurance with a coverage fitting this risk as best as possible. This insurance takes a certain premium. If the person wants to pay less, an insurance with a lower coverage is bought. The consequence is that there will be no payment if the uncovered risk occurs. On the other hand, if coverage were to large, then too much premium is paid, since a situation has been insured which is unlikely to occur for this person.



The balance between budget and risk coverage

2. Risk Assessment

Test strategy is based on risk assessment. This means assessing the damage of the consequences of defects, both undetected prior to operation and occurring during operation.

Risk assessment takes place on the basis of quality characteristics and subsystems. For instance, if the system is insufficiently user-friendly, what will be the negative consequences. And what will be the damage when the salary calculation module in a payroll system does not work correctly.

In order to be able to perform this assessment well, the separate aspects of a risk are considered:
Risk = chance of failure x damage,

where chance of failure is related to aspects including frequency of use and the chance of an error occurring.

These aspects are listed below:

- Frequency of use

In a function which is used dozens of times each day the chance of an error demonstrating itself is much bigger than with a function used once a year.

- Chance of error

For the assessment of the chance of errors the following list can be helpful. It presents the locations where errors tend to cluster. It is partly based on H. Schaefer, 1996 (*Surviving under time and budget pressure*, in: Conference Proceeding EuroSTAR1996, Amsterdam, the Netherlands):

◊ Complex functions;
◊ Completely new functions;
◊ (Especially frequently) adjusted functions;
◊ Functions for which certain tools or techniques were employed for the first time;
◊ Functions which were transferred from one developer to another during development;
◊ Functions that were realized under extreme time pressure;
◊ Functions which had to be optimized more frequently than on average;
◊ Functions in which many defects were found earlier (e.g. in previous releases or during earlier reviews);
◊ Functions with many interfaces;
◊ Inexperienced developers;
◊ Insufficient involvement of users;
◊ Insufficient quality assurance during development;
◊ Insufficient quality of low-level tests;
◊ New development tools and development environment;
◊ Large development teams;
◊ Development teams with sub-optimal communication (e.g. owing to geographical spread or personal causes);

- Damage

If and when the error manifests itself, what will be the damage for the organization. Aspects are costs of repair (both of the system and of the consequences), forgone income and loss of clients or of confidence. Usually the damage increases if the error has its impact on other functions or systems. In the case of errors occurring in batch processes there may be a possibility to prevent them from hampering users, so that the eventual damage will be smaller than with similar on-line processes. Of course, this only holds if errors are detected on time.

Because of the complexity of the matter, it is impossible to assess risks with complete objectivity and in detail: it is a global assessment. It is therefore important for the risk assessment not to be carried out by the test manager alone. A large number of people involved in the scheme should contribute: customer, users, development team, accountants, IT auditors and so on. This not only increases the quality of the strategy, but it also has the advantage that the different parties are more aware of the risks and the extent to which testing contributes to making these risks manageable in a better way.

The developer of the test strategy should realize that 'users' are the best people to assess the damage and the frequency of use when valuing the risks (end-users, system managers and application managers, line management), whereas project team members are best to assess the chance of error (project managers, designers, programmers, project quality staff, test manager).

The focus in risk assessment is on product risks, or, in other words, what is the risk for the organization if the product does not demonstrate the expected quality. In addition to this, there are also (test) project risks. If the system must be in production on January 1st, if functional specifications are produced too late, if no experienced testers are available, or if the test infrastructure is not ready on time, then we speak of (test) project risks. These are not taken into account in determining the test strategy; they do play a role in the test plan.

In developing a test strategy the aim is to see to it that the test will be organized in such a way that with a certain extent of reliability

- the most important problems will be found;
- the problems will be found in an early stage;
- the problems that require the most rework time will be found first:
- efficient use is made of resources;
- and eventually an accurate quality advice can be given.

This can be summarized as:

Test strategy aims at finding the **most important** errors as **early** as possible against the **lowest costs**

In practice, the development of a test strategy is often planned to coincide with preparing the budget, for example with the help of test point analysis. The advantage is that the consequences of the adopted strategy are immediately translated into time required for testing, and consequently the cost of testing, which makes the strategic choices manageable. If the time available for testing is more or less fixed, it is also possible to use test strategy combined with test point analysis to determine what is achievable within the time limits. It is probably even more important to make it clear at this time which parts cannot be tested, or cannot be fully tested, and what risks will therefore be incurred.

3. Quality Characteristics

The quality characteristics we distinguish can be divided into dynamic and static quality characteristics. The dynamic quality characteristics deal with features of the information system in use; examples are security, usability, continuity, traceability, functionality, userfriendliness, suitability, efficiency, performance. The static are concerned with intrinsic characteristics of the information system and the documentation, as considered from the standpoint of developers and future system managers. Examples are manageability, maintainability, connectivity, reusability, portability, testability.

4. Procedure

In developing a test strategy we distinguish between master test planning and a test plan for a specific test level, e.g. acceptance test or system test.

The procedure can be followed both for development of new systems and for maintenance situations. For the latter, however, it is best to make a few adjustments in the basic procedure (cf 4.4).

The development of a test strategy is not something that can be done purely methodically or formally. The below steps are aids and indicators. Experience and skills of the performer of this activity in the area of testing is a major success factor for a sound test strategy.

One should also realize that test strategies arise as a result of iterative processes and in connection with other activities for a test plan. If the first test strategy produces an amount of test effort needed or a certain time schedule which is unacceptable for the customer, the strategy should be adjusted. The lack of test skills or suitable infrastructure can also result in adjustments of the test strategy.

4.1 Strategy in Master Test Planning

The steps to be taken for a test strategy in master test planning are:

- Decide on the quality characteristics;
- Determine relative importance of quality characteristics;
- Attribute quality characteristics to test levels.

4.1.1 Step 1: Selection of Quality Characteristics

In close liaison with the customer and other parties involved a selection of quality characteristics is made on which the tests must focus. In doing so one should take risks for the business into account as well as aspects including system requirements, business objectives concerning the information system, directions and standards set by the computer centre. These quality characteristics are also used for reporting to the customer during test execution and completion.

Some characteristics are difficult to test. There may be a wish for a system to be user-friendly and flexible, for instance, but these wishes turn out not to have been translated into measurable requirements. That is why a substantial part of the effort here is devoted to formulating the relevant quality demands as measurably and unambiguously as possible. It is also the case that some quality characteristics demand relatively much effort in testing. Since it is not useful to offer possibilities which cannot be fulfilled, it should be determined beforehand what will be the estimated effort needed for a decision made.

For non-IT people our quality characteristics may be hard to handle. It helps when we translate them to the conceptual environment of our conversational partners. This can be done by finding illustrative examples of problems or errors that may occur in production and the damage that would be caused by this. This is one of the most difficult aspects of the formulation of a test strategy.

4.1.2 Step 2: Relative Importance of Quality Characteristics

On the basis of the results from Step 1 the importance of the selected quality characteristics is determined in relation to one another. This is done in the Matrix of Weights (see below), by weighing the relative risks per quality characteristic. Here the relative importance is indicated (in percentages). Note that it is not of importance to have exact percentages: the objective is to arrive at a general picture of the relative importance of the various quality characteristics. The filling in of the matrix helps evaluating the risks.

The customer should be forced to make choices. Therefore, as a directive we ask for a percentage of 5 as the minimum. The sum of all percentages should not exceed 100. An example of a Matrix of Weights is given below:

Quality characteristic	Relative importance
Manageability	5

Security	5
Usability	-
Connectivity	-
Continuity	10
Traceability	-
Flexibility	-
Functionality	50
Userfriendliness	10
Reusability	-
Infrastructure	-
Suitability	10
Maintainability	5
Performance	5
Portability	-
Testability	-
Efficiency	-
Total	100%

The Matrix of Weights

The high percentage for functionality in this matrix may strike the reader. This is in conformance with practical experience: generally 50% of the importance or more is attributed to this characteristic. The reason for this is that risks usually are larger for incorrect performing systems (Functionality) than for slow systems (Performance) or awkward systems (Userfriendliness).

4.1.3 Step 3: Quality Characteristics Attributed to Test Levels

With the aim of spending the total test effort as efficiently as possible, during test strategy development it is decided with which test level or combination of test levels the various selected quality characteristics will be tested. Also inspections may fall under the scope of the master test plan and under the test strategy. In the remaining sections when 'test' is used, inspections are also included.

In this way the various test levels within a project are brought into balance. It is obvious that the different responsibilities and authorities remain intact.

A +-sign in a matrix (for an example, see matrix below) indicates whether the test strategy takes a quality characteristic into account. '++' or '+++' indicate that relatively much attention is to be paid to the quality characteristic for the specified test level. It is obvious that one quality characteristic can be in effect for more than one test level, but depth will often vary. If structured test specification techniques are used, the acceptance test, for example, may use results of previous tests levels, on the basis of which it may be decided to test with less depth.

	Insp RQMS	Insp Specs	Insp Design	PT	IT	ST	FAT	PAT	Relative importance
Manageability	+	+	++					+	5
Security	+	+	+				+	+	5
Usability									-
Connectivity									-
Continuity	+		+					++	10
Traceability									-
Flexibility									-
Functionality	++	++		+	+	+++	++		50
Userfriendliness		++					++		10
Reusability									-
Infrastructure									-
Suitability	+	++					++		10
Maintainability		+				+			5
Performance			+				+	+	5
Portability									-
Testability									-
Efficiency									-
									100%

Example of a Test Strategy for Test Levels

Legenda:

Insp RQMS	Inspection /review of Requirements
Insp Specs	Inspection /review of Functional Specification
Insp Design	Inspection /review of Technical Design
PT	Program Test
IT	Integration Test
ST	System Test
FAT	Functional Acceptance Test
PAT	Production Acceptance Test

4.2 Strategy for a Test Level

The steps to be taken for a test strategy for a specific test level are:

1. Decide on the quality characteristics;
2. Determine relative importance of quality characteristics;
3. Divide the system into subsystems;
4. Determine relative importance of subsystems;
5. Specify test importance per subsystem and quality characteristic;

6. Establish test techniques to be used.

The strategy determination for a specific test level naturally has the master test plan strategy as a precondition and a starting point. If a master test plan, including a test strategy, is there, step 1 can be omitted and step 2 will be an easy and fast performed activity. Nevertheless, all steps are worked out below.

4.2.1 Step 1: Decide on Quality Characteristics

In collaboration with the customer and perhaps other parties concerned the quality characteristics are determined on which the test will focus, in relation to business risks. During the test and in the completion phase, results are reported on the basis of these quality characteristics.

4.2.2 Step 2: Determine Relative Importance of Quality Characteristics

Based on the results of step 1 the relative importance of the selected quality characteristics is determined. Determination of the importance takes place by weighing the risks per quality characteristic. This is shown in a Matrix of Weights by a percentage in the column Relative importance. In order to force the customer to make choices, a percentage of 5 is the minimum.

An example of a matrix for a functional acceptance test is given below:

Quality characteristic	Relative importance
Security	5
Functionality	60
Userfriendliness	10
Performance	5
Suitability	20
Total	100%

The Matrix of Weights for a Functional Acceptance Test (Example)

4.2.3 Step 3: Divide System into Subsystems

During this step and the following steps the test strategy is refined more and more. This implies that the quality characteristics and their relative importance as indicated in the Matrix of Weights are to be broken down for the combination of test specification technique and subsystem, later even for test specification technique and test unit.

The information system is divided into subsystems. The reason for this is that the same quality demands do not have to be valid for each subsystem. Moreover, the various subsystems may have different risks for the organization. In principle the division is the same as given in the design documentation. If we deviate from this one, we must clearly indicate the motivation for this. Examples of alternative divisions are on the basis of extent of risk or on the basis of order of release by the developer. If a conversion module is there, this is to be treated as a separate subsystem. Often the subsystem 'Total system' is distinguished. This serves the purpose of indicating that some quality characteristics can be evaluated effectively only with the help of an integral test, testing the coherence of the various subsystems.

In a later stage the various subsystems are further divided into independent test units. E.g. in a logistics system the subsystem Sales may be divided into the test units Quotations (all functions regarding quotations) and Orders.

4.2.4 Step 4: Determine Relative Importance of Subsystems

On the basis of the result of the previous step the relative importance (in percentages) of the subsystems should be indicated in the Matrix of Weight. This is done by weighing the risks per subsystem. It is not a matter of exact percentages; rather it is a matter of getting a general image of the importance of the subsystems as seen through the eyes of the customer and other parties concerned. This step helps in asking people to form an opinion of this.

The relative importance is determined of each subsystem within the information system. In the Matrix of Weights this is indicated with a percentage in the column Relative importance.

An example of a Matrix of Weight for a functional acceptance test (based on the Strategy Matrix for test levels in the master test plan, shown above) is given here:

	Relative importance
Subsystem 1	30
Subsystem 2	15
Subsystem 3	20
Conversion	15
System	20
Total	100%

Relative Importance of Subsystems for a Functional Acceptance Test (Example)

4.2.5 Step 5: Specify Test Importance per Subsystem and Quality Characteristic

Finally a refinement is made by assessing the importance of the combination quality characteristic - subsystem. E.g., a refinement may be that userfriendliness is important (relative importance of 10), but this holds predominantly for subsystem 1 and not at all for subsystem 3. Again it is emphasized that test strategy determination is not a mathematical affair: it is meant to get an image of the relative test importance of the various subsystems and quality characteristics. This is also the reason why we choose +, ++ and +++ as notational symbols, rather than opting for the pseudo-certainty of a mathematical formula. An example of this is the following: suppose both userfriendliness and a specific batch subsystem are very important, a mathematical formula would probably result in large test effort to be spent on the userfriendliness of the batch procedure. The Matrix of Weight may look like this:

	Subsystem 1	**Subsystem 2**	**Subsystem 3**	**Con-version**	**Total system**	**Relative importance**
Security	+	+				5
Usability						-

Continuity						-
Traceability						-
Functionality	++	+	+	++	+	60
Userfriendliness	++	+				10
Performance	+		+			5
Suitability	+	+	+		++	20
Efficiency						-
Relative importance	30	15	20	15	20	100%

Relative Importance of Subsystem x Quality Characteristic (Example)

4.2.6 Step 6: Establish Test Techniques to be Used

The final step in test strategy involves the selection of the test specification techniques that will be used to test the combination of the selected quality characteristics and subsystems. A high importance implies the use of techniques with a high coverage or the use of more techniques, a low importance implies the use of techniques with a lower coverage or the use of fewer techniques.

In choosing the techniques one should also take into account various other factors, a number of which are listed below.

- Quality characteristic to be tested

A technique is fit for testing one or more quality characteristics. Some quality characteristics can best be tested with one (set of) techniques, others with another one.

- Area of application

Some techniques are specifically suitable for testing the interaction (screens, reports, on-line) between the system and the users, others are better in testing the processing of systems (batch processes). There is a relation with the type of error to be found with a technique, e.g. false input checks, incorrect processing or errors of integration.

- Availability of test basis

Each techniques starts from a certain test basis. This may be the functional specification, the technical design, program code or descriptions of the end-user organization. The exact form of the test basis is also relevant to the choice of a technique, e.g. decision tables, pseudo-code, structured language or unstructured prose.

- Extent of formality

Informal test specification techniques offer more freedom for the tester in making the test cases than do formal techniques.

- Use of resources

The application of a techniques requires a specific amount of resources, in terms of man capacity as well as machine capacity. The use of resources has a direct relation with costs.

- Required knowledge and skills

Not each tester is equipped for each technique. For the useful application of some techniques much business knowledge is needed. For other techniques more analytic talent is required. Therefore, the knowledge and skills of the test staff also influences the choice for techniques.

For practical reasons one should attempt to cover all selected quality characteristics with a minimal set of test specification techniques.

The selection of the test specification techniques should be done in an early stage of the test process, for then the test team can take the appropriate actions in training for techniques and the necessary checklists can be made or adjusted for the specific situation.

As a result of this step the techniques that will be used per subsystem are defined. Optionally, especially with large test projects, this last step in the test strategy is performed slightly later in the process, namely during the preparation phase. As a part of this the priority order of the tests to be performed is determined. The aim of this is to have the most important tests take place as early as possible.

4.3 Strategy during the Test Process

The test strategy determined in advance often will be put under pressure in a later stage of the test project. In such a situation the test manager is asked to perform less tests or shorter tests in order to conform to the adjusted schedule. The consequences are to be seen mainly in the last step of strategy development: suddenly some tests must be cancelled or must be carried out with less depth. Using the test strategy as a basis the test manager may discuss with the customer which tests can be dropped or where less thorough testing can be done. By indicating which parts are to be tested less in relation to the risks assessed (translated into importance levels in the strategy), the test manager can report in a solid fashion on the increased risks after the testing phase. Therefore it is essential not to change the steps 1 to 5: the risks and the importance levels do not change. The result is that, when testing is reduced, there will be more risks after the system has been implemented.

Apart from this there is also the situation that during testing it turns out that part of the system contains an excessive number of errors or excessively few errors. Both cases justify adjustments in the test strategy, namely the increase and decrease of test effort, respectively. Contrary to the situation in the previous paragraph here risks will remain the same after implementation of the system. The correction can be summarized as follows:

Testing should continue as long as the costs for finding and correcting errors during testing are lower than the costs connected to the error occurring in production

In 'finding and correcting errors' more costs than just test costs play a role; other, extensive costs may be concerned with the delay in shipping the product. For 'the error occurring in production' one should also take into account the chance that the error will actually occur: an error that will never occur is no error (defect)!

4.4 Strategy during Maintenance

The main difference between the development of new systems and maintenance for the test strategy is the chance of error. In the case of maintenance changes are made to an existing information system. These changes should be tested. During maintenance there is a risk that new errors are introduced, with a decrease in quality for the system as a result (regression).

The implication of this different chance of error in the case of maintenance implies for the strategy that the relative importance of the subsystems may change: a subsystem which had a high importance when it was developed, may be unchanged in maintenance. Since the chance of regression is the only risk in this case, the test importance is much lower. Therefore, test strategy development for a test level can be modified by changing the concept of 'subsystem' in the steps to 'change'. For each change it is analysed which system parts were mutated, which parts may have been influenced by the change, and which quality characteristics are relevant. There are various possibilities for testing each change, dependent on the risks:

- A limited test, only focused on the change;
- A complete (re)test of the function which had been changed;
- A test of the coherence of the changed function and the adjacent functions.

There should also be a regression test for the system as a whole. This test focuses on the coherence of the changed and unchanged parts of the system, since the chance of regression is largest here. If the test strategy for the new developed system is available, the importance levels attributed to the subsystems here can be of use.

Apart from the changed chance of error there are more differences between a system developed and a system under maintenance. However, they have no influence on the *technique* of test strategy development. Examples of these differences include:

- Test basis is missing, incomplete or not up-to-date

This situation, frequent in maintenance, has consequences for the test specification techniques to be chosen.

- Predictable versus ad-hoc maintenance

The majority of maintenance situations are predictable and can therefore be planned. The strategy determination is easily to be applied to this type of maintenance. The situation is more difficult in the case of ad-hoc maintenance, where the focus is on putting right a production break-down and getting the system in the air as soon as possible. A formal strategy determination costs too much time here. It is feasible, however, to have some test strategy scenarios available: if program x goes wrong and is repaired, what should be tested? These scenarios support an optimal test for ad-hoc maintenance.

5. Conclusion

The testing of information systems should be based on the business risks which the organization will experience in using these information systems. In practice, test managers often take the steps to come from risks to test coverage in an intuitive manner. In this paper, the steps needed for the definition of a test strategy are made explicit. The result of such a test strategy is better insight for all parties involved and a sound basis for negotiating testing depth.

Good risk assessment is a part of these steps. It is essential to realize that this explicit way of looking at risks cannot be done by a test manager or tester alone. It is necessary to ascertain for the involvement of users and managers of the client organization, of auditors, and of project people such as developers, testers, QA staff and project managers. In practice, the discussion of risks and related testing strategies in this way proves to be a real eye-opener for all parties concerned. It also enables negotiation of testing depth by having the customer decide which elements should be tested how thoroughly.

The stepwise definition of the test strategy can be used for any test level (e.g., system test, acceptance test) and also for an overall strategy (master test plan, including and co-ordinating all test levels and inspections/reviews.

