Software product line engineering for consumer electronics
Hartmann, Herman

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2015

Link to publication in University of Groningen/UMCG research database

Citation for published version (APA):

Copyright
Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

Take-down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): http://www.rug.nl/research/portal. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.
This chapter is an extended version of the paper that was published as:
H. Hartmann, J. Bokkerink, V. Ronteltap: “How to reduce your test process with 30%; The application of operational Profiles at Philips Medical Systems”. In: Supplementary proceedings of the 17th IEEE International Symposium on Software Reliability Engineering. This extension is submitted as: H. Hartmann: “A Statistical Analysis of Operational Profile Driven Testing and Risk Based Testing”.

Abstract.
Due to the increasing size and complexity of software the amount of faults is constantly increasing, thereby leading to a reduced reliability. Due to the pressure of time-to-market, it is not feasible to remove all faults, or even the majority of them, before market introduction. It is therefore crucial to focus testing on finding those faults that have the largest impact.

Within many organizations, operational profiles are used to focus the test activities on those areas that are most used by the end-user. Similarly risk based testing is used to focus testing on those areas that pose the largest risk to the user and the business.

In this paper a statistical model is introduced that captures the relation between faults and failures. Using the model the efficiency improvements of using operational profiles are simulated under different conditions. This analysis shows that using operational profiles improves the test efficiency, however not when a high reliability is required. For risk based testing a quantification is introduced and the analysis shows that an efficiency improvement can be realized when the usage frequency is treated as a separate dimension of the risk matrix, rather than making the usage frequency part of the impact dimension.

Because the model that is introduced is based on easy to understand probability theory, it has great explanatory power in illustrating the effects of allocating different amounts of test cases. A case study at Philips Healthcare is presented that shows the applications of the model and the experiences from using it.

The results of this paper can be used by practitioners to improve their test planning and can be used by researchers to further develop statistical models for software reliability.

8.1 Introduction

Due to the increasing size and complexity of software products, the number of faults, also known as defects, is constantly increasing [Siewiorek 2004]. This is observed during the test phase and after product release. Due to the pressure on time-to-market it is not feasible to
remove all faults or even the majority of them before market introduction. This results in a product with insufficient reliability [Ulrich 2004], and therefore an increasing field call rate for consumer electronics products [Ouden 2006A] or increasing maintenance efforts [Jacobs 2003]. The costs related to this, can become considerable. It is therefore imperative to make the test activities more effective and achieve the highest possible reliability within the available time for testing.

Testing within a project is usually defined by a test plan [Veenendaal 2002, McGregor 2010, Jørgensen 2013]. This plan describes what tests need to be executed and what coverage is required, which is a measure of how thorough each test should be executed. The challenge in defining the right coverage is to find the optimal balance between time-to-market and costs versus the required reliability [Veenendaal 2014].

The concept of operational profiles was introduced to guide the test planning process [Musa 1993]. In this approach the tests are allocated to different operations according to the operational profile, which is a set of operations along with the probability with which they occur. In many situations different operations will have a different criticality and therefore different reliability requirements. In these situations the operations are classified according to their criticality and operational profiles are generated for each class [Musa 1999].

The notion of different criticality is the rational for risk based testing [Amland 2000, Alam 2013, Schaefer 2014]. Risk based testing is used to prioritize test effort to those development artifacts that pose the greatest risk. These risks are usually based on two dimensions [Veenendaal 2014, Amland 2000, Schaefer 2014]: (1) the probability that a function or feature contains faults and (2) the impact that a failure causes, where the impact is a combination of the damage, or costs, of a failure and the usage frequency of each function or operation.

This paper focuses on test efficiency and answers the following research questions:

- What are the benefits for the test efficiency when using operational profiles?
- Which dimensions in risk based testing require most attention to reduce the overall risks?

In this paper a statistical model is introduced which estimates the efficiency improvements for testing when using operational profiles, and which analyzes the dimensions in risk based testing. The inputs for the model are the amount of faults that are expected in a function or operation, the chance that a fault leads to a failure and which analyzes the dimensions in risk based testing. The inputs for the model are the amount of faults that are expected in a function or operation, the chance that a fault leads to a failure and the operational profile. From this input the expected mean time between failures is calculated and the amount of faults that are found during testing.

A case study from Philips Healthcare describes the experiences using this model to derive the efficiency improvements for the test process.

This paper is structured as follows. Section 8.2 presents background and in Section 8.3 the statistical model is presented followed by Section 8.4 that shows simulations of using operational profiles. Section 8.5 describes the experiences at Philips Healthcare, while Section 8.6 analyses risk based testing. Section 8.7 provides a discussion and areas for further research and Section 8.8 gives a comparison with related art. The paper finishes with the conclusions in Section 8.9 and acknowledgments.
8.2 Background

This section provides background on software reliability and testing.

8.2.1 Software reliability, faults and failures

Reliability is defined as [Lyu 1996]: “The probability that a system functions in a specified environment without failure, for a specified time or a number of natural units”.

Example: The reliability of a car can be expressed in the number of years it drives without problems, or the number of kilometers.

The most used measure for reliability is Mean Time Between Failures (MTBF), which is the average time between two failures.

![Fault tree of software failures](image)

**Figure 46 Fault tree of software failures**

Figure 46 shows a fault tree of software failures [BSI 1998]. A failure originates from a person making a mistake. This can be a mistake made during specification, design or coding. Such a mistake may result in a fault in the software. A fault may lead to an internal error, depending on the way the system is used, i.e. the revealing mechanism. So, a fault in a part of the software that is not executed will not lead to an internal error and failure, and therefore has no influence on the reliability. In some cases, the software is fault tolerant meaning that it recognizes that an error occurs and the system chooses another way to perform the operation. However, an internal error in most cases leads to a failure which is experienced by the user (see Table 17 for the definitions of the used terminology).
Chapter 8

Table 17 Definitions in software reliability

<table>
<thead>
<tr>
<th>Mistake</th>
<th>Incorrect or missing action by a person or persons, that may cause a fault in a program</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fault</td>
<td>Defect in a system that may cause a failure when executed. In this paper the terms defect and fault will be used interchangeably</td>
</tr>
<tr>
<td>Error</td>
<td>Discrepancy between the internal state of a system and its desired state</td>
</tr>
<tr>
<td>Failure</td>
<td>Departure of system behavior from the requirements</td>
</tr>
<tr>
<td>Failure severity</td>
<td>Severity of a failure, i.e. the consequences of a failure for end-user or business</td>
</tr>
<tr>
<td>Failure intensity</td>
<td>Failures per natural or time unit. A way of expressing reliability, commonly as MTBF.</td>
</tr>
</tbody>
</table>

In other words: A software fault is a defect in the code, caused by a human mistake during the creation of the architecture, the design or during coding, and is developer oriented. A software failure occurs during execution, and is user oriented. Therefore, the purpose of testing should be to obtain a product which has the lowest failures intensity, i.e. highest MTBF, rather than focusing on the lowest amount of faults. There are alternative ways to create reliable software [Lyu 1996]. The primary way is to prevent that the software contains faults, e.g. by using formal methods. Secondly it is possible to remove the faults through testing and it is possible to prevent faults becoming failures during execution, known as fault tolerance [Pullum 1999]. This paper focuses on testing as one way to reduce the failure intensity.

8.2.2 Test planning

In many organizations the verification and test process starts with a planning phase in which test activities are defined, including test techniques and coverage [Veenendaal 2002, McGregor 2010, Jorgensen 2013]. Verification and test techniques may include code reviews, static analysis and dynamic testing, such as white- and black box testing. Test coverage is a measure of how thorough each test is executed. The test coverage criterion determines how many test cases need to be created. Since this is the most intensive part of test activities [Pohl 2005A], the test coverage largely determines the required effort. The challenge in test planning is to find the right balance between time-to-market and quality [Veenendaal 2014]. For safety critical systems, such as nuclear power plants, the reliability is essential thus requiring a high coverage, while for consumer products the time-to-market is more important and less coverage is needed [Musa 1999].

8.2.3 Software reliability engineering and operational profiles

Software Reliability Engineering (SRE) is defined as: “A practice for quantitatively planning and guiding software development and tests, with the emphasis on reliability and availability. It provides the solution to estimate, predict and measure the rate of failure occurrences in software [Musa 1999]”. This is done by determining when the software is
good enough to release, minimizing the risks of releasing with serious problems thereby avoiding excessive time-to-market due to over testing. Within SRE, two types of tests are used: the certification test and the reliability growth test:

The certification test is meant to establish that an application works the way it should. This test requires a stable system and does not involve debugging. The decision to be made is whether to accept or reject the software, based on the measured MTBF versus the target that is set.

The reliability growth test is a test aiming to improve the product’s reliability over time. It has the purpose of finding and removing faults. It includes feature, load and regression tests.

Both these tests use operational profiles to guide the test. An operational profile is a quantitative characterization of how a system is used. The operation profile is a set of software operations, along with the probability with which they occur.

For obtaining the operational profile, a number of steps have to be taken. First the customer profiles have to be identified, the users and the system modes, i.e. the mode in which the system operates. For each user the functions he/she uses have to be identified. The functions are usually further separated into operations. For each of these profiles, the occurrence probabilities have to be determined and are combined to obtain the overall operational profile [Musa 1993].

Musa claims that when the tests are allocated to the most-used operations, the reliability level will be the maximum practically achievable when the software needs to be shipped due to imperative schedule constraints before all the software faults can be removed [Musa 1993].

In many situations different functions and operations will have a different severity when this function or operation fails [Musa 1993]. In these situations the operations are classified according to their criticality and operational profiles are generated for each criticality class. Test time is allocated to each category, where most test time and resources are allocated to the categories with the highest criticality.

8.2.4 Risk based testing

Risk based testing is a widely adopted practice for setting priorities in testing [Amland 2000, Alam 2013]. Risk based testing is based on the assumption that not all functions and operations should be tested equally rigorously, but that (only) the most important faults should be found and removed. For areas with high risks, the tests are executed earlier, using more comprehensive test techniques and higher coverage targets are set [Schaefer 2014], while for non-critical areas lower coverage targets are set [daMotaSilveira 2011].

Usually a matrix visualizes the risks, which consists of two dimensions: The probability dimension captures the likelihood that a function fails and the impact dimension captures the usage frequency and consequence that a function fails [Schaefer 2014].

The values for probability and impact are determined by the various stakeholders in a project, e.g. the product managers, developers and testers. These values are based on expert opinion, rather than rigorous statistics. Through a guided discussion the participants score the factors that determine the values [Veenendaal 2014].
For the *probability* the following factors are commonly used [Jorgensen 2013, Veenendaal 2014, Schaefer 2014]: Complex or changed areas, new technology and tools, number of people involved, experience of the developers, whether the software is developed by developers located in different geographical locations, time pressure and areas that revealed many defects in the past.

For the *impact* the following factors are commonly used [Jorgensen 2013, Veenendaal 2014, Schaefer 2014]: visibility of the function to the user, the impact on the business when a function fails and the usage frequency.

The values of the individual factors are added, sometimes using a weighted sum [Veenendaal 2014], and visualized in a matrix, of which an example is shown in Figure 47.

![Figure 47 Example of a risk matrix](image)

In this example the functions F1 and F3 are regarded as having a high risk and should be tested intensely, while less test effort should be allocated to function F5. The risk matrix is used to plan the test which includes the test techniques, coverage and the order in which functions should be tested.

### 8.3 A Statistical Model to Estimate the Improvement on Test Efficiency

In this section a statistical model is introduced that captures the relation between faults and failures, and which calculates the reliability improvement of allocating different amount of test cases to each function and operation.

#### 8.3.1 Model elements and assumptions

The statistical model is based on the definition of faults and failures as shown in Figure 1 and defined in Table 1. It uses the amount of faults as input as well as the usage frequency. Furthermore each fault has a chance of leading to a failure, later referred to as failure probability of a fault, which depends on the revealing mechanism. Mistakes and errors are not modelled. The model uses the following assumptions:

- Each fault can be assigned to exactly one operation.
• Each fault has probability that it manifests as a failure when this operation is used.

The following assumptions about the test process are made [Oshana 1994, [Walto 1995]:
• Testing is a stochastic process and a test case is a random selection of the input space as in black-box testing.
• Faults are repaired directly and correctly, no new faults are introduced and no new functionality is added during the test process.

8.3.2 Definitions:
The model uses the following definitions:
• A product consists of different operations Oj, with j = 1, 2, 3, etc.
• The software that implements an operation can contain a number of faults. Because not each fault has the same failure probability, groups of faults are used that have different probabilities of leading to a failure. So there are fault groups with i = 1, 2, 3,.... and in each group i every fault has the same failure probability.
• Pi,j is the failure probability, which is the probability that a fault from fault group i of operation j manifests during execution.
• Ni,j is the number of faults of group i in operation j.
• P(Oj) is the chance that operation Oj fails.
• Rtot is the total number of test cases that are executed during a test cycle.
• Rj is the amount of test cases allocated to operation j.
• N(Oj) is the number of faults that are found with Rj test cases.
• Tj is the time that an operation j takes to execute.
• OPj is the probability of occurrence of operation j as part of the operational profile, also referred to as usage frequency.

8.3.3 Calculation of the MTBF
The probability that an operation fails is the probability that one or more of its faults manifests as a failure. The chance that an operation Oj fails then is:

\[ P(O_j) = 1 - \prod_i (1 - P_{i,j})^{N_{i,j}}. \]  

The MTBF of a single operation is determined by the time to execute an operation divided by the chance that an operation fails:

\[ MTBF_j = \frac{T_j}{P(O_j)} \]  

The MTBF of the product is the weighted average of the times the operations take to execute, divided by the weighted average that these operations fail. The weighted averages are determined by the operational profile:

\[ MTBF = \frac{\sum_j OP_j \times T_j}{\sum_j OP_j \times P(O_j)} \]
8.3.4 Calculation of the number of faults found during testing:

The number of faults that are found during a test for operation \( O_j \) is the sum of the faults that are found for each fault group \( i \). The number of faults that are found for each group \( i \) is the probability that such a fault has occurred during this test multiplied by the number of faults from that group:

\[
N(O_j) = \sum_i (1 - (1 - P_{i,j})^{R_j}) \times N_{i,j}
\]  
(8.4)

With these formulas the MTBF before testing can be calculated, the number of faults that are found for each fault group and operation and the MTBF can be calculated after the faults have been removed.

8.4 Simulations of the Test Efficiency when using Operational Profiles

This section presents the effect on the test efficiency of using operational profiles with the following scenarios:

- Test allocation based on operational profiles versus a uniform test distribution.
- The effect when a large number of test cases are executed to obtain very high reliability.

8.4.1 Simulation of test time allocation based on operational profile versus a uniform distribution

For this example a fictitious product with 4 operations is used as shown in Table 18.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Percentage of use</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>50</td>
</tr>
<tr>
<td>B</td>
<td>30</td>
</tr>
<tr>
<td>C</td>
<td>10</td>
</tr>
<tr>
<td>D</td>
<td>10</td>
</tr>
</tbody>
</table>

Each operation contains a total number of initial faults, being 100, which are equally divided over three fault groups. The first group has a failure probability 0.01, the second group of 0.005 and the third group of 0.001.

By using multiple fault groups, rather than one, it is avoided that the simulations show extreme results. For instance when a single group is used with a high probability of failure then a few test cases will already find many faults and not many test cases are needed to find all faults. On the other hand, when faults have a low failure probability, then the MTBF is small and many tests are needed to reveal faults.

The time that a test case runs is set to 1, in this example without dimension, so could be seconds, minutes, or hours.
A Statistical Analysis of Operational Profile Driven Testing and Risk Based Testing

Table 19 shows a comparison between three scenarios.
1. The first scenario is a uniform distribution, i.e. to each operation the same amount of test cases is allocated, using a total of 250 test cases.
2. In the second scenario the amount of test cases is also 250 but is allocated according to the operational profile, so to operation A and B more test cases are allocated than to operations C and D.
3. In the third scenario only 70% of the total 250 test cases is used but also allocated according to the operational profile.

Table 19 Comparison of MTBF with uniform versus OP distribution

<table>
<thead>
<tr>
<th>Operation</th>
<th>Faults</th>
<th>Tests</th>
<th>Faults found</th>
<th>remaining Faults</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>100</td>
<td>62.5</td>
<td>27</td>
<td>73</td>
</tr>
<tr>
<td>B</td>
<td>100</td>
<td>62.5</td>
<td>27</td>
<td>73</td>
</tr>
<tr>
<td>C</td>
<td>100</td>
<td>62.5</td>
<td>27</td>
<td>73</td>
</tr>
<tr>
<td>D</td>
<td>100</td>
<td>62.5</td>
<td>27</td>
<td>73</td>
</tr>
<tr>
<td></td>
<td>400</td>
<td>250</td>
<td>108</td>
<td>292</td>
</tr>
<tr>
<td>MTBF</td>
<td>2.4</td>
<td></td>
<td></td>
<td>MTBF After 3.5</td>
</tr>
</tbody>
</table>

Test case distribution using OP

<table>
<thead>
<tr>
<th>Operation</th>
<th>Faults</th>
<th>Tests</th>
<th>Faults found</th>
<th>Remaining faults</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>100</td>
<td>125</td>
<td>43</td>
<td>57</td>
</tr>
<tr>
<td>B</td>
<td>100</td>
<td>75</td>
<td>31</td>
<td>69</td>
</tr>
<tr>
<td>C</td>
<td>100</td>
<td>25</td>
<td>12</td>
<td>88</td>
</tr>
<tr>
<td>D</td>
<td>100</td>
<td>25</td>
<td>12</td>
<td>88</td>
</tr>
<tr>
<td></td>
<td>400</td>
<td>250</td>
<td>98</td>
<td>302</td>
</tr>
<tr>
<td>MTBF</td>
<td>2.4</td>
<td></td>
<td></td>
<td>MTBF After 4.1</td>
</tr>
</tbody>
</table>

Test case distribution using OP with 70% of test cases

<table>
<thead>
<tr>
<th>Operation</th>
<th>Faults</th>
<th>Tests</th>
<th>Faults found</th>
<th>Remaining faults</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>100</td>
<td>87.5</td>
<td>34</td>
<td>66</td>
</tr>
<tr>
<td>B</td>
<td>100</td>
<td>52.5</td>
<td>23</td>
<td>77</td>
</tr>
<tr>
<td>C</td>
<td>100</td>
<td>17.5</td>
<td>9</td>
<td>91</td>
</tr>
<tr>
<td>D</td>
<td>100</td>
<td>17.5</td>
<td>9</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>400</td>
<td>175</td>
<td>75</td>
<td>325</td>
</tr>
<tr>
<td>MTBF</td>
<td>2.4</td>
<td></td>
<td></td>
<td>MTBF After 3.5</td>
</tr>
</tbody>
</table>

Table 19 shows the initial amount of faults and the amount that have been found during testing. This example shows that the MTBF after the fault removal with the number of test cases allocated according to the operational profile (OP) is higher than when allocated according to a uniform distribution, i.e. 4.1 versus 3.5 while fewer faults need to be repaired.
Chapter 8

The third scenario shows the results using OP with only 70% of the test cases, which results in the same MTBF and significantly less faults need to be repaired, i.e. 75 versus 108. In other words, when using operational profiles the test effort could be reduced with 30% for this example.

The result of this example cannot be generalized, since they depend on the number of faults, the amount of test cases and failure probabilities. The operational profile in this example has a moderate ratio between the occurrences probabilities of the operations with the highest occurrence probability versus the lowest, i.e. only a ratio of 1 to 5. In reality the ratio is much larger, e.g. 10 times or even 100 times as much. With such operational profiles the benefits will be higher as well when the same amount of faults per operation is assumed.

8.4.2 A large amount of test cases to obtain a high MTBF

In the simulations in the previous section still a significant proportion of faults remained in the system. In practice it might be needed to remove more faults to obtain a higher MTBF.

In Table 20 a comparison is shown of the MTBF of the uniform distribution versus an allocation according to the operational profiles, using different amount of test cases. The MTBF is shown after the test and removal of faults, and the efficiency improvement. The efficiency improvement is defined by the difference in percentage of test cases that is needed to obtain the same MTBF. E.g. an efficiency improvement of 23% means that 77% of the test cases according to the operational profiles results in the same MTBF as the uniform distribution.

In this example, the same operational profile, number of faults, fault groups and failure probability are used as in the previous example, in Section 8.4.1.

Table 20 Comparison with large amount of test cases

<table>
<thead>
<tr>
<th>Number of test cases</th>
<th>Uniform</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MTBF After</td>
<td>3.5</td>
<td>5.2</td>
<td>10.4</td>
<td>102</td>
</tr>
<tr>
<td>Remaining faults</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Operation A</td>
<td>73</td>
<td>57</td>
<td>38</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>Operation B</td>
<td>73</td>
<td>57</td>
<td>38</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>Operation C</td>
<td>73</td>
<td>57</td>
<td>38</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>Operation D</td>
<td>73</td>
<td>57</td>
<td>38</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>Total remaining faults</td>
<td>292</td>
<td>128</td>
<td>156</td>
<td>40</td>
<td>12</td>
</tr>
<tr>
<td>OP</td>
<td>MTBF After</td>
<td>4.1</td>
<td>6.3</td>
<td>11.7</td>
<td>93</td>
</tr>
<tr>
<td>Remaining Faults</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Operation A</td>
<td>57</td>
<td>38</td>
<td>23</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>Operation C</td>
<td>69</td>
<td>52</td>
<td>34</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>Operation C</td>
<td>88</td>
<td>78</td>
<td>63</td>
<td>23</td>
<td>13</td>
</tr>
<tr>
<td>Operation D</td>
<td>88</td>
<td>78</td>
<td>63</td>
<td>23</td>
<td>13</td>
</tr>
<tr>
<td>Total remaining faults</td>
<td>302</td>
<td>246</td>
<td>183</td>
<td>55</td>
<td>28</td>
</tr>
<tr>
<td>Efficiency improvement in %</td>
<td>30</td>
<td>23</td>
<td>12</td>
<td>-5</td>
<td>-12</td>
</tr>
</tbody>
</table>
This example shows that when the amount of test cases increases, the benefits of using operational profiles are smaller in situations with many test cases, i.e. when a high reliability is required, and using operational profiles is even less efficient than using a uniform distribution.

The reason is that testing according the operational profiles leaves defects in the system for those parts that have a low occurrence probability, operation C and D, while for the operations with higher occurrence probability, A and B further testing will hardly reveal any additional faults and these operations are basically over tested.

This effect differs per situation, however, with a growing number of test cases for some operations no faults will be present anymore, to start with for those to which many test cases are allocated, i.e. have a higher occurrence frequency. Applying more tests to these operations will not have an effect on the overall reliability anymore.

Therefore, for high and ultrahigh reliability the use of operational profiles is less efficient for reliability growth testing. To obtain a very high MTBF the tests should be aimed at finding ALL faults most efficiently rather than allocating more test cases to operations that are most used, since this lead to over testing. Therefore a better strategy to obtain very high reliability is to allocate more test time to operations that contain many defects and/or are hard to find, i.e. have a lower failure probability.

8.5 Case Study: Philips HealthCare

This section describes the experiences at Philips Healthcare with operational profiles and using the statistical model.

8.5.1 Problem description and objective

Philips Healthcare manufactures a wide range of systems with a large portfolio of imaging equipment, i.e. systems for image acquisition, such as Magnetic Resonance Imaging (MRI) and Computer Tomography (CT). Other products are used to retrieve and visualize the medical images which are used by medical experts to create a diagnosis.

Healthcare Informatics is one of the business groups of Philips Healthcare. Healthcare Informatics is responsible for the development and marketing of software solutions for visualization, processing, archiving and distribution of medical information. Delivering software products with high quality is essential for Healthcare Informatics to remain competitive in this market.

Despite ample testing activities of Healthcare Informatics software products, still too many field problems are encountered and consequently too much effort has to be spent on maintenance activities. To reduce maintenance effort, more relevant faults, i.e. faults that lead to failures during the usage by the end-user, must be found during the test phase where the effort for testing cannot be extended since new product features need to be delivered to the market in a timely fashion.

The application of the software reliability engineering method at Healthcare Informatics had the aim to define the process of establishing and applying operational profiles and to investigate the benefits and consequences.
As part of this project, the operational profile has been determined for two products. The product MIP DICOM Viewer was used as a pilot project to gain experience. After the pilot project was finished, the operational profile has been determined and applied in the development of a new product, EasyWeb 4.1. For reasons of clarity, the experiences from MIP DICOM Viewer are described rather than for EasyWeb 4.1.

8.5.2 Obtaining the operational profile

Since there were no end-users for MIP DICOM Viewer available for determining the profiles, the operational profile was determined with the help of applicants from Healthcare Informatics. The applicants represent end-users, are often former radiologists and have close contacts with the end users in hospitals and are therefore knowledgeable with the way the Healthcare Informatics products are used.

To determine the occurrence rates of all functions and operations, all steps that are executed by the applicants while using the product, were recorded. For MIP DICOM Viewer, a record-and-playback tool, originally meant for the creation of automated tests, has been used to register the steps. Every executed step has been recorded and a snapshot of the screen has been taken. These snapshots provided the usage frequencies of the different functions and operations for the defined user types.

The user profile consists of the following group:

- Radiologists, e.g. orthopedists.
- General Practitioners, who are receiving results from specialists.
- Two types of Cardiologists.
- Patients, who are getting the possibility to view a selection of the images.

In addition to the user profile frequency and usage frequency also the user weights have been determined to capture the relative importance of different user types [Aggarwal 1993] since the reliability for some users, especially the orthopedists is more important than for other users. The reliability for the patients is regarded as much less important.

Table 21 shows a subset of the operational profile of MIP DICOM Viewer for the Orthopedist. This gives a good representation, i.e. the final operational profile of the different user and user weights hardly differs.
The profiles show that there are functions and operations that were hardly used while others were more heavily used, i.e. some operations are used more than 10 times than other operations. Some operations are not used at all.

### 8.5.3 Estimation of benefits for test efficiency

Since data of the number of faults and failures from Philips Healthcare are confidential, only the method to determine the benefits can be shared in this paper.

- To use the model, described in Section 8.3, the following data is required:
  - The initial number of faults $N_{i,j}$.
  - The number of tests executed $R_{tot}$.
  - The failure probability $P_{i,j}$ of a fault and the fault groups with the number of faults per group.
  - The operational profile.
  - The time than an operation $j$ takes to execute $T_j$.

The MTBF and the number of faults found during the test can be calculated from the above data. The operational profile is obtained and the time to execute an operation can be measured. Finally the number of test cases can be defined. So the initial number of faults $N_{i,j}$ and $P_{i,j}$ the chance a fault manifests need to be estimated.

Data from MIP DICOM viewer was used to estimate the benefits for EasyWeb 4.1; the product under development. Since these products have similar functionality and are based on the same technology the fault density and failure probability of a fault are expected to be similar.
Chapter 8

The initial number of faults was estimated based on comparing the code size of the EasyWeb 4.1 in relation to that of MIP DICOM Viewer. A uniform distribution of faults across the operations and an average failure probabilities $P_{i,j}$ for the faults was assumed, so only one fault group was used. For MIP DICOM viewer the number of faults were estimated based on the problem reports that were entered in the various stages of the project, furthermore the amount of test cases were known. So using formula (8.4) only $P_{i,j}$ was the remaining unknown and could be determined by simulation. The benefits could then be estimated by using the total amount of test cases and comparing a uniform test case distribution versus a distribution according to the operational profile. In this analysis a separation was made between operations that belong to different criticality classes.

8.5.4 Presentation of the results

It was experienced that, when explaining the use of operational profiles to an audience that is unfamiliar with this method, the model turned out to be very useful for demonstrating the effects. Because the simulations not only show the outcome, but also show the amount of faults and faults found, it was regarded as very convincing even though there was a general agreement that the results of the simulations should not be regarded as highly accurate.

The difference between faults and failures was made clear through this model and the effect on the reliability. Whereas, in the past, test architects were typically aiming to find as many faults as possible, the test architects realized that the tests should be focused on finding the most relevant faults. As a result of the presentation of the results, Philips Healthcare decided to adopt the use of operational profiles to guide the test effort.

8.6 A Statistical Analysis of Risk Based Testing.

In this section the statistical model will be used to evaluate the priority setting of test activities using risk based testing. This section starts with experiences from the Philips Healthcare, a comparison between risk based testing and operational profile driven testing and a quantification of risks in risk based testing.

8.6.1 Different reliability requirements in healthcare products

While reliability is extremely important for medical devices, there is a large difference for each function or operation. For instance, when an image acquisition is used during medical interventions, e.g. surgery, the reliability requirements are now extremely high because it is immediately life threatening when a system breaks down.

For other systems, such as the retrieval and storing of data the reliability requirements are lower and differ per operation. For instance it is crucial that the correct image is stored and retrieved because this might lead to a wrong diagnosis. For operations such as printing or zooming, the reliability requirements are lower.

Therefore Philips Healthcare has adopted risk based testing for planning the test cases and resources.
8.6.2 A comparison of risk based and operational driven testing

In software reliability engineering the functions and operation are classified according to their severity and operational profiles are created for each class. For certification tests, the reliability targets are used to accept or reject the software. For growth testing most tests are allocated to the highest severity classes and the operational profiles are used to allocate test cases within those classes.

In risk based testing the priorities are based on expert opinion, which includes an analysis of areas where many faults are expected, i.e. the probability dimension. For each function it is then defined how much test time should be allocated as part of reliability growth testing where the functions with the highest risk obtain most test cases. Risk based testing does not include reliability certification tests.

So the main differences are, that in operational profile driven testing the prioritization does not include an estimation of the faults, while risk based testing is based on expert opinion and does not include reliability certification tests.

8.6.3 Quantification of the risks

For a quantification of the risks and application of the statistical model a quantification of the risk dimensions and factors is introduced.

1. The probability dimension is the area where most problems are expected [Veenendaal 2014]. These areas are estimated based on a number of factors [Veenendaal 2002, Schaefer 2014]: Complex areas, changed area, new technology, time pressure and so on. These factors are also used for defect density predictions [Neufelder 2000]. In the statistical model these will be represented by the number of faults from which, together with the failure probability of the faults, the failure probability per operation \( P(O_j) \), is calculated.

2. The impact dimension consists of two parts: The frequency of use and the damage [Veenendaal 2002, Schaefer 2014]:
   - The frequency of use will be represented by the occurrence probability of the operations as part of the operational profile [Jørgensen 2013].
   - The damage describes the impact to the business or end user when a function fails [Amland 2000, Schaefer 2014] also referred to a cost and consequences of a failure [Veenendaal 2014]. In general this is very difficult to quantify and in [Veenendaal 2014] it is suggested to use a logarithmic scale to represent the costs using values for costs of 1, 3 and 10. In the statistical model this scale will be used.

Using these quantifications for probability and impact, the risk per operation can be quantified. Here \( R(O_j) \) is defined as the risk for operation \( O_j \).

The risk per operation is defined as the multiplication of the chance that an operation fails, \( P(O_j) \), with the change that it is executed, \( OP_j \), and the damage that occurs when an operation fails \( D(O_j) \):

\[
R(O_j) = D(O_j) \times P(O_j) \times OP_j
\]  

(8.5)
The failure chance \( P(O_j) \), is defined in formula (8.1) while \( OP_j \) and \( D(O_j) \) are given. Note that in [Jorgensen 2013] a formula for risk is given that only consist of the operation profile and the damage, and in [Amland 2000] a formula is given that only includes costs and the chance that a function contains a fault. So both this related art miss a factor in the formula for risk.

**8.6.4 Simulation of different scenarios**

An estimation of the benefits of using risk based testing is obvious and will therefore be omitted. What is more worthwhile is to determine what the impact of test effort is on the different risk dimensions.

For evaluating effects of the factors in risk based testing, three different functions, i.e. set of operations, are compared that are all three classified as having medium risk. Using the statistical model the effect of adding extra test cases to each of these functions, \( F_2 \), \( F_3 \) and \( F_4 \), from the risk matrix shown in Figure 48 Risk matrix with 5 functions, is analyzed and compared. Here we assume that one operation is assigned to one function only as in [Jorgensen 2013].

![Figure 48 Risk matrix with 5 functions](image)

The Function \( F_2 \) has a high probability, i.e. high expected number of faults, while the impact is regarded as low. Function \( F_3 \) has medium probability and medium impact, while Function \( F_4 \) has low probability and high impact.
Table 22 shows the results of the simulations. Again the same fault groups have been used as in the examples in Section 8.4 and the time for a test case to run is 1. In the simulation 500 versus 100 test cases are executed and the faults have been removed.

The table shows the risk reduction, which is the reduction in risk, of allocating a larger number of test cases to a function, versus a lower amount of test cases, i.e. 500 versus 100.

This table shows that by allocating 500 test cases instead of 100 to function F4 reduces the risk by 0.51 per time unit. For the Functions F3 and F2, which have more faults but lower damage per failure and lower usage frequency, the risk reduction is substantially smaller, namely 0.20 and 0.03. In other words: although all three functions are classified as having medium risk, which would mean that for each of them about the same test cases would be allocated, the simulations show that having more test cases for function F4 leads to a much higher risk reduction than allocating this amount of test cases to F3 and F2.

The result of these simulations can be explained for two reasons:

- The impact consist of two parameters that both have a linear effect on the damage per unit of time, because these are based on the quantification of the risks as given in formula (8.5).
- The reduction of faults, due to the number of test cases, is nonlinear because increasing the number of test cases, e.g. by two times, will not reduce the number of faults two time, since the same faults may be found in two different test cases. Basically when a function contains many faults, few cases will already reveal a significant amount of faults. The example shows that 100 test cases reveal almost half of the amount of faults as 500 test cases.

Different values for damage, number of faults and usage frequency will give another value for risk reduction as results. In this example there is a ratio of 10 on usage frequency, damage and also number of faults, which is regarded as realistic.

Since the influence of damage and usage frequency on the reliability will be similar for other products, it is likely that the observed effect is applicable to many projects.
8.6.5 Implications for the application of risk based testing

The current practice of risk based testing would be improved, i.e. result in products with higher reliability, when the test cases are allocated using three dimensions:

1. **Probability**: this is the same as currently used and estimates the amount of faults.
2. **Frequency of use**, also the same as currently used, but now as separate dimension. For this dimension the operational profile can be used.
3. **Damage or Costs** of a failure. Again this is also the same as currently used but now also as separate dimension.

This means that the highest amount of test cases should be allocated to the product of these three dimensions. This new approach can be easily incorporated in current practice, since the required information is already gathered; only the prioritization will be set differently.

The effects will differ per situation. For instance, in the situation when the use frequency is close to uniform, this new approach will not lead to large differences in comparison with the current approach. However this new approach is more fine-grained, and based on the quantification of risks, and therefore lead to a better allocation of the amount test cases in most situations, and can never lead to a worse allocation of the amount of test cases.

While the proposed new practice will allocate fewer test cases to operations with high amount of faults with lower usage frequency and damage, these test will still reveal a higher amount of faults. So the time needed to resolve this higher amount of faults will be higher and needs to be planned a priori.

8.6.6 Conclusions on risk based testing

From the analysis the following conclusions are drawn:

- The two dimensional representation of risks, in which frequency of use is part of the impact of a failure, does not provide good guidance because there is a too strong focus on removing faults, rather than reducing the amount of (severe) failures.
- A representation using three dimensions, i.e. **Number of Faults**, **Frequency of use** and **Damage** gives a better representation.
- Incorporating this improved analysis in current practice will not require a large change in way of working since the required information is already available.

8.7 Discussion and Further Research

8.7.1 Accuracy and extensions to the statistical model

The model introduced in this paper is based on the assumption that a fault can be allocated to one function or operation, rather than assigning a fault to a development artifact, such as software code, that is used by multiple operations. This means that when testing one operation, this may also reveal faults in other operations and thus would reduce the effect of testing based on usage frequency. For a more precise model the design of the system is needed so that a fault can be allocated to different operations and thus manifests during the tests of these operations. In the case of Philips Healthcare a product line approach is adopted [Pohl 2005A] in which software components are allocated to functions (see [Linden 2007]
Chapter 13) and therefore such mapping would be possible between different functions and the expected number of faults that are shared. This type of modeling can be added to the statistical model by introducing fault groups that are shared by multiple operations.

The accuracy of the predictions also relies on the input to the model. For the number of faults a large amount of research is available, e.g. [Neufelder 2000], and also on the creation of operational profiles. The remaining input is \( P_{ij} \) the chances that a fault becomes a failure. In the case study this was based on a past project and only an average value was estimated. More research is needed to derive more accurate values. It may be expected that these values will differ per application area; e.g. for user interface applications or database applications, a fault may reveal itself quickly while for embedded systems, where a fault may only occur in very specific circumstances, e.g. related to timing aspects and different hardware configurations the failure chance for many faults will be much smaller [Gray 1985].

While the accuracy and the model can be improved, the author believes that the conclusions drawn on the use of operational profiles and risk based testing are valid since improvements to the model are not likely to results in large differences. Furthermore, the simulations with the current model show clear effects that for which the reasons behind these effects could be clearly argued.

### 8.7.2 Use of the model in practice

In this paper the estimated efficiency improvements are based on black box testing. Similarly as operational profiles and risk based testing are also used to plan the test activities for other test techniques [Veenendaal 2002], [Musa 1999], the planning of test activities in general and setting the priorities for other types of tests can also be based on simulations with the model. Here further research is needed.

The model in its current form is only able to give a rough estimate and provides general trends and conclusions. Model enhancements are identified in this paper which could make this model suitable for further improving the planning of test activities. For instance, when the values for the probability and impact in risk based testing are estimated more precisely, e.g. based on a fault density prediction and market analysis, the statistical model can be used for optimizing the distribution of test cases, in relation to the available test time.

From the experiences in demonstrating the simulations to experts as well as non-experts on testing, the author believes that the model may also be very suitable for educational purpose. From simulations with the model the difference between faults and failures, failure intensity and so on become clear and choices made during test planning become very tangible. Furthermore the model uses basic probability theory and can easily be implemented using spreadsheets.

### 8.8 Comparison with Related Art

The benefits of using operational profiles have been discussed in earlier work, e.g. by [Musa 1993], but this work focused on the benefits of having reliability targets and an improvement of the engineering process. In prior art it has been stated that using operational profiles also would lead to a higher efficiency by finding more relevant faults but none has
Chapter 8

provided an estimation, nor did that prior art recognize that for products that requires very high reliability, using operational profiles is less efficient.

In the work on decision support, e.g. [Rus 1999] different strategies are discussed, but no quantification for test efficiency for risk based testing and using operational profiles is given.

In none of the prior art on risk based testing a quantification of the benefits is given, nor has prior art compared the priorities for the different dimensions in risk based testing. Therefore prior art did not identify that the current practice of combining damage and usage frequency may lead to too much focus on reducing the number of faults rather than the risks.

The use of different targets for MTBF has been discussed [Musa 1999] but did not address the number of faults to prioritize the test cases. In the work on weighted operational profiles [Aggarwal 1993] it is suggested to allocate more test cases to functions and operations that are regarded as more important, but also this related art does not provide quantification.

8.9 Conclusions

In this paper a statistical model is introduced that estimates the benefits of using operational profiles and risk based testing. The simulations and case study gave two results:

- Guiding tests according to operational profiles improves the test efficiency however not when very high reliability is required. When a very high reliability is required, test cases should focus more on finding faults in operations that contain many faults.
- The priority allocation in risk based testing has a tendency of focusing too much on reducing the amount of faults rather than reducing the failure intensity. An efficiency improvement can be realized when the usage frequency is used as a separate dimension of the risk matrix rather than making the usage frequency part of the impact dimension. This improvement can easily be incorporated in current practice.

The case study at Philips Healthcare showed that the model can be applied in practice and resulted in the adoption of using operational profiles to allocate test priorities.

Because the model is easy to understand, it has great explanatory power in illustrating the influence of different amounts of test cases on the products reliability.

The results of this paper can be used by practitioners to evaluate and improve the prioritization of test time allocation and can be used by researchers as a basis to further develop statistical models on test efficiency.

The model in its current form is only able to give a rough estimate and provide general trends and conclusions. Further improvements are identified in this paper which could make this model suitable for a more precise planning and optimization of test activities.

Acknowledgments

The author thanks Vincent Ronteltap, Jarmila Bökkerink and Arjan den Hartog from Philips Healthcare for their contributions to the case study and Peter Sonnemans from the Technical University of Eindhoven for his help on the statistical model.