Observing Software Testing Practice from the Viewpoint of Organizations and Knowledge Management

Ossi Taipale, Katja Karhu, Kari Smolander
Lappeenranta University of Technology
P. O. Box 20
FI-53851 Lappeenranta FINLAND
+358 5 621 2838
Ossi.Taipale@lut.fi, Katja.Karhu@lut.fi, Kari.Smolander@lut.fi

Abstract

The objective of this qualitative study was to understand the complex practice of software testing, and based on this understanding, to develop hypotheses on testing organizations and knowledge management. The population of the study consisted of organizational units (OUs) that develop and test technical software for automation or telecommunication domains. First, a survey of testing practices was conducted and 26 OUs were interviewed. From this sample, five OUs were further selected for an in-depth case study. The study used grounded theory as its research method and the data was collected from 41 theme-based interviews. The analysis yielded hypotheses that included that the business orientation of an OU affects testing organization, knowledge management strategy, and outsourcing of testing. Further, identifying and avoiding barriers and using enablers improve knowledge transfer between development and testing. The results of this study can be used in developing the testing organization and the knowledge management strategy of an OU.

1. Introduction

John et al. [1] state that human and social factors have a very strong impact on software development endeavors and the resulting system. This is in line with the study of Argote and Ingram [2], who state that there is a growing agreement that organizational knowledge explains the performance of organizations. In our earlier quantitative study [3], respondents evaluated the development of testing organizations and their knowledge management as an important subject that improves the total efficiency of software testing.

Argote & Ingram [2] define knowledge transfer in organizations as "the process through which one unit (e.g. group, department, or division) is affected by the experience of another". The transfer of organizational knowledge (i.e. routine or best practices) can be observed through changes in the knowledge or performance of recipient units. According to Szulanski [4] the transfer of organizational knowledge can be quite difficult to achieve. This is because knowledge resides in organizational members, tools, tasks, and their sub-networks and, as Nonaka [5] shows, much knowledge in organizations is tacit or hard to articulate.

Knowledge transfer between development and testing, especially in the earlier phases of the software life cycle, is seen to increase efficiency. Both Graham [6] and Harrold [7] emphasize the need to integrate earlier phases of the development process with the testing process.

To our knowledge, literature on software testing does not contain studies concerning high abstraction level constructs of organizations and knowledge management. This background led us to observe software testing from the viewpoint of a testing organization and knowledge management.

In this study, our specific objective is to understand the practice of software testing, and based on that understanding, to derive hypotheses on testing organizations and knowledge management.

The grounded theory research method described by Glaser and Strauss [8] and later extended by Strauss and Corbin [9] was used. Grounded theory was selected because of its ability to uncover the issues from the practice under observation that may not have been identified in earlier literature [8].

The interviewees in this study represented companies that produce and/or test automation or telecommunication software. The criticality of their
end products is above average and the software is used in real-time environments.

The study included four theme-based interview rounds. We personally visited companies and carried out 41 tape-recorded interviews. The themes of each interview round are available at http://www.it.lut.fi/project/anti/.

The paper is structured as follows: Firstly, we introduce the related research. Secondly, the research process and the grounded theory method are described in section 3. The analysis results are presented in section 4. Finally, the discussion and conclusions are given in section 5.

2. Related research

Organization models of testing are discussed and compared in [10, 11]. Kit [10] describes seven approaches to organizing the testing function. Kit’s testing organization models include 1 “testing is each person’s responsibility”, 2 “testing is each unit’s responsibility”, 3 “testing is performed by dedicated resources”, 4 “the testing organization in quality assurance (QA)”, 5 “the testing organization in development”, 6 “centralized testing organization”, and 7 “centralized testing organization with a testing technology center”. This classification proved to be usable also in this study.

Knowledge management has been discussed widely from various perspectives, e.g. in [12], but its association with software testing requires still exploration. In organizations, knowledge is recognized as the principal source of economic rent and competitive advantage [2, 13]. According to Nonaka [14] organizational knowledge is created through a continuous dialogue between tacit (personalized) and explicit (codified) knowledge. Tacit knowledge is embedded in the employees. Explicit knowledge is codified and transferable, e.g. in documents, independently of people in whom tacit knowledge is embedded [15].

An important area of knowledge management is knowledge transfer, which is also discussed widely in the literature [4, 16-18]. Becker and Knudsen [18] discuss barriers and enablers to knowledge transfer and divide knowledge transfer into intra- and inter-firm knowledge flows. Szulanski [4] explored the internal stickiness of knowledge transfer and found that the major barriers to internal knowledge transfer were knowledge-related factors such as the recipient’s lack of absorptive capacity, causal ambiguity, and an arduous relationship between the source and the recipient. Conradi and Dybå [16] studied formal routines to transfer knowledge and experience and concluded that formal routines must be supplemented with collaborative and social processes to promote effective dissemination and organizational learning.

Cohen et al. [17] have noticed that the result of testing ultimately depends on the interpersonal interactions of the people producing the software. The physical distance between development and testing may create new challenges for knowledge transfer. Cohen et al. [17] found in exploring the organizational structure that testers and developers ought to co-locate. When testers and developers worked in separate locations, communication, as well as personal relationships, was impaired, unlike when both groups worked in close proximity.


3. Research process

To understand the complex phenomenon of software testing, an exploratory and qualitative strategy following the grounded theory approach [8, 9] was considered suitable. According to Seaman [20], a grounded approach enables the identification of new theories and concepts, making it a valid choice for software engineering research.

We followed the process of building a theory from case study research described by Eisenhardt [21] and its implementation example [22]. Principles for an interpretive field study were derived from [23]. Other example studies included [24, 25].

3.1. Selecting case organizational units

The population consisted of organizational units (OUs) from small, nationally operating companies to large internationally operating ones.

The standard ISO/IEC 15504-1 [26] specifies an OU as a part of an organization that is the subject of an assessment. An OU deploys one process or more having a coherent process context and operating within a coherent set of business goals. An OU is typically a part of a larger organization, although in a small organization, the OU may be the whole organization.

The reason to use an OU as an assessment unit was that we wanted to normalize the effect of the company size to get comparable data. Based on the available information of the OUs, we estimated the criticality of their produced or tested software. The objective was to select OUs with software criticality above average
The criticality of the produced or tested software was measured during the first interview round to ensure that the process contents of the OUs were comparable. The criticality of the products was estimated by asking how severe problems the faults in their products can cause [27].

For the first interview round, the selection from the population to the sample was based on probability sampling. The population was identified with the help of national and local authorities. The OUs were in a random order in our database and every second organization was selected. The sample of the first interview round consisted of 26 OUs. From this sample, five OUs were further selected as case OUs for the second, third and fourth interview rounds. For the case study the sampling was theoretical [22] and the cases were chosen to provide examples of polar types [21], which means that the cases represent different types of OUs, e.g. different business, different size of the company, and different kind of operation. The goal of theoretical sampling [22] is not the same as with the probabilistic sampling. The researcher’s goal is not to collect a representative sample of all possible variations, but to gain a deeper understanding of analyzed cases and identify concepts and their relationships. The OUs and interviewees are described in Table 1.

3.2. Data collection

Managers of development and testing, testers, and systems analysts were selected as interviewees because these stakeholders face the daily problems of software testing. The interviews were conducted by two researchers to avoid the bias caused by the observer (observer triangulation). The duration of the interviews varied between one and one and half hours and they were all tape-recorded and transcribed to text. A memo containing the emphasized issues was written on each interview.

The first interview round contained both structured and semi-structured (theme-based) questions. The objective was to understand the basic practice of testing, identify case OUs (representative polar points) for the next round, and identify problems and improvement proposals. The interviewees were managers of development or testing, or both. The questions of the first round concerned general information on the OU, processes, knowledge transfer between development and testing, and the development environment of the OU. Results of the survey conducted during the first interview round are reported in [3, 27, 28].

The interviewees of the second round were managers of testing, those of the third round were testers, and those of the fourth round were systems analysts. The objective of these interview rounds was to achieve a deeper understanding of software testing in practice. The questions were theme-based and concerned problems of testing, the use of software components, the influence of the business orientation, knowledge transfer, schedules, organization and knowledge, testing automation, and economy.

The themes between the interview rounds remained similar, but the questions evolved from general to detailed.

Before proceeding to the next interview round, all interviews were scripted and analyzed because new ideas emerged during the data analysis. These new ideas were reflected in the next interview rounds. The data collection process of all 41 interviews generated a transcription of 946 pages.

<table>
<thead>
<tr>
<th>Interview round(s)</th>
<th>OU Business</th>
<th>Company size</th>
<th>Interviewees</th>
</tr>
</thead>
<tbody>
<tr>
<td>First</td>
<td>All 26 OUs including cases from A to E</td>
<td>Automation or telecommunication domain, products represented 48.4% of the turnover and services 51.6%.</td>
<td>The interviewed OUs were parts of large companies (53%) and small and medium-sized enterprises (47%). The average size of an OU was 75 persons.</td>
</tr>
<tr>
<td>2nd, 3rd, and 4th</td>
<td>Case A A MES producer and integrator</td>
<td>Large/international</td>
<td>Testing manager, tester, systems analyst</td>
</tr>
<tr>
<td>2nd, 3rd, and 4th</td>
<td>Case B Software producer and testing service provider</td>
<td>Small/national</td>
<td>Testing manager, tester, systems analyst</td>
</tr>
<tr>
<td>2nd, 3rd, and 4th</td>
<td>Case C A process automation and information management provider</td>
<td>Large/international</td>
<td>Testing manager, tester, systems analyst</td>
</tr>
<tr>
<td>2nd, 3rd, and 4th</td>
<td>Case D Electronics manufacturer</td>
<td>Large/international</td>
<td>Testing manager, 2 testers, systems analyst</td>
</tr>
<tr>
<td>2nd, 3rd, and 4th</td>
<td>Case E Testing service provider</td>
<td>Small/national</td>
<td>Testing manager, tester, systems analyst</td>
</tr>
</tbody>
</table>

*SME Definition [29]*
3.3. Data analysis

The grounded theory method contains three data analysis steps: open coding, where categories of the study are extracted from the data; axial coding, where connections between the categories are identified; and selective coding, where the core category is identified and described [9]. In practice, these steps overlapped and merged because the process proceeded iteratively. The categories and their relationships were derived inductively from the data. The process of grouping concepts that seem to pertain to the same phenomena is called categorizing, and it is done to reduce the number of units to work with [9].

The objective of the open coding was to classify the data into categories and identify leads in the data. The open coding of the interviews was done using ATLAS.ti software [30]. The process started with “seed categories” [31] that contained essential stakeholders, phenomena, and problems.

Seaman [20] notes that the initial set of codes (seed categories) comes from the goals of the study, the research questions, and predefined variables of interest. In the open coding, new categories appeared and existing categories were merged because especially in the beginning of the coding new information sprang up. The open coding of all 41 interviews yielded 196 codes which were classified into 5 categories.

The objective of the axial coding was to further develop categories, dimensions, and causal conditions or any kinds of connections between the categories. The categories were used to identify factors that affect testing organizations and knowledge management. The phenomenon represented by a category was given a conceptual name [9]. The created categories contained the business orientation of the OU, testing organization, knowledge management strategy, knowledge transfer, and outsourcing of testing. The categories were further developed by defining dimensions. The dimensions represent the locations of the property or the attribute of a category along a continuum [9]. Our inductive data analysis of the categories included Within-Case Analysis and Cross-Case-Analysis, as in [21]. We used the tactic of selecting dimensions, and looking for within-group similarities coupled with inter-group differences [21].

The objective of the selective coding was to identify the core category [9] and systematically relate it to the other categories. Strauss and Corbin [9] write that sometimes the core category is one of the existing categories, and at other times no single category is broad enough to cover the central phenomenon. In that case, the central phenomenon must be given a name. In this study, the creation of the core category meant the identification of the affecting factors (categories) that explain software testing practice from the organization and knowledge viewpoint, and finding the relationships between these categories.

The general rule in grounded theory is to sample until theoretical saturation is reached. This means, until (1) no new or relevant data seem to emerge regarding a category; (2) the category development is dense, insofar as all of the paradigm elements are accounted for, along with variation and process; (3) the relationships between categories are well established and validated [9].

The data was analyzed after each interview round to collect new issues, and to see if it was worthwhile to continue the data collection procedure. The theoretical saturation was reached during the fourth interview round, because new categories did no more appear, categories were no more merged, shared, or removed, attributes or attribute values of the categories did not change, and relationships between categories were considered stable i.e. the already described phenomena begun to repeat in the data.

Within-group similarities and inter-group differences (dimensions) and relationships between the categories yielded the description of software testing in practice. Each chain of evidence in this description was established by having sufficient citations in the case transcriptions.

4. Analysis results

In the following, we will describe the results of the analysis. First, the case OUs and their characteristics are depicted by categories and their dimensions. The dimensions explain the similarities and differences between the case OUs.

In categorizing, the factors that most affected testing organizations and knowledge management were identified from the research data, grouped and named. We developed the categories further by focusing on the factors that explain organization and knowledge management, and abandoned categories that did not seem to have an influence on the phenomenon. Dimensions emerged as the result of the within-group and inter-group analysis. The categories are listed in Table 2.
The dimension of the category “business orientation” varied from a purely product oriented to a purely service oriented OU. The construct business orientation described the ratio between products and services of the OU’s turnover and it was derived from Sommerville [32]. Sommerville divides software products into two broad classes, generic products and customized products. We complemented and widened Sommerville’s classification with a finer granulation and added a purely service-oriented dimension “consulting and subcontracting” at the other end. The business orientation of an OU was measured by dividing the turnover of the OU into product and service items. The product items included a customized product (based on a product kernel), a uniform product kernel in all deliveries, a product family composed of distinct components, and a standardized online service product (e.g. product/service provider). The service items included training and consulting, subcontracting, system integration, installation service, and self service (e.g. service/service provider) [28]. In general, OUs can be positioned along a continuum that starts from a purely service oriented OU and ends at a purely product oriented OU.

The dimension of the category “testing organization” varied from 1 “testing is each person’s responsibility” to 7 “centralized testing organization with a testing technology center” and it described the approaches to organizing the testing function. The classification of the organization models was adopted from Kit [10].

The dimension of the category “knowledge management strategy” described the ratio between the codification and personalization strategies. In the codification strategy knowledge is handled in an explicit form and in the personalization strategy in a tacit form.

The dimension of the category “knowledge transfer” described the barriers and enablers of the formal and informal communication and interaction between development and testing. The dimension was expressed as a list of observed issues that either enable or hinder knowledge transfer between development and testing, e.g. physical distance, the recipient’s absorptive capacity, causal ambiguity, and the relationship between the source and the recipient [4].

The dimension of the category “testing outsourcing” denoted the extent to which an OU outsourced or offered testing resources.

4.1. Description of cases

4.1.1. A MES producer and integrator. In Case A, products represented 50% and services 50% of the turnover of the OU.

Case A developed and tested manufacturing execution systems (MES). Its services included systems integration and customizing of systems. Customized products supported services and were based on uniform product kernels. Testing was partly integrated with software development, but system testing was organized as a part of the quality assurance organization. Specialization was more emphasized than job rotation inside testing.

Codification of knowledge was not emphasized earlier because developers and testers co-located, but the situation was changing because case A had founded a new development center. The need for formal documentation and communication was growing in the knowledge management. Domain knowledge was especially emphasized in testing customized systems. Problems of knowledge management included that documentation was not up to date, troubleshooting documentation was inaccurate, and the provided schedule and release information from the testing organization to development was insufficient.

Knowledge transfer between development and testing was flexible because most of the developers and testers worked physically close to each other. The effect of distance as a barrier to knowledge transfer was growing because of the new development center.

Using outsourced testing resources had been considered, but not yet implemented. Close cooperation between development and testing was emphasized.

4.1.2. Software producer and testing service provider. Case B was a mostly service oriented OU. Products represented 25% and services 75% of its turnover.
Case B tested its own software products and offered testing services to external customers. Testing was partly integrated with software development; partly testing adapted to the processes of external software development. The testing organization was centralized. Case B tended towards job rotation inside testing and specialization was less emphasized.

Codification of the knowledge was more emphasized because e.g. test plans, test runs, and test results were reported using the templates of the customer’s quality system. Domain knowledge was more emphasized when testing the company’s own products and less emphasized when testing external software. Problems of knowledge management included a lack of codified knowledge and finding the right balance between personalized and codified knowledge.

Intra-OU knowledge transfer was described as flexible because testers and developers were able to communicate face to face, but in providing testing services knowledge transfer depended more on planned and formal meetings. We observed in case B that the knowledge transfer from the customers was controlled by the customers and it further emphasized codified knowledge.

Case B both used outsourced testing resources and offered testing services to other companies.

4.1.3. Process automation and information management provider. Case C developed customized systems. Services represented two thirds and products one third of the turnover of the OU.

The testing resources were partly integrated with software development. System testing formed a centralized testing organization with a testing technology center. In Case C, the testers’ work was balanced between specialization and job rotation.

Codification of the knowledge was less emphasized, but it was growing in importance because knowledge had to be transferable to the new development center. Domain knowledge was emphasized in testing customized systems. Problems of knowledge management included the lack of codified knowledge.

Knowledge transfer was seen flexible because testers and developers were able to communicate face to face. The effect of distance was minor, but it was growing because of a new development center. The distance between development and testing, and the fact that testers do not understand the technical language of the developers were found as the barriers to the knowledge transfer.

Case C did not use outsourced testing resources. Close co-operation between development and testing was seen important.

4.1.4. Electronics manufacturer. Case D was a purely product oriented OU.

High product orientation required high-quality end products because recalls in serial production are very expensive. D emphasized the measurement function of testing. System testing formed a centralized testing organization with a testing technology center. Case D tended towards job rotation inside testing and specialization was less emphasized.

Knowledge transfer between development and testing was planned, formal, and transparent as specified by the quality system. The effect of distance was medium because development centers were located around the world, but knowledge transfer was supported by the available technology and the quality system, which emphasized the codification of knowledge. Domain knowledge was less emphasized.

Case D concentrated on the core business and outsourced testing resources to a great extent.

4.1.5. Testing service provider. Case E was a purely service oriented OU. Working as an external testing organization required the adaptation to the processes of the customer. Testing formed a centralized testing organization. In Case E, the testers’ work was balanced between specialization and job rotation. There was more job rotation in short term projects than in long term projects.

Codification of the knowledge was emphasized, but the level of the customer documents was not satisfactory. Domain knowledge was emphasized and it was seen as a competitive advantage.

The knowledge transfer between the customer’s software development and the testing service provider was active and clear and it was specified as the task of the contact persons. A contact person was perceived as an enabler of knowledge transfer. The effect of distance was significant because of the work as an external testing organization. We observed that in Case E also the knowledge transfer between customers and the OU was controlled by customers, and it further emphasized codified knowledge.

Case E both used outsourced testing resources and offered testing services to other companies. Outsourcing of the testing resources depended on the customer project.
4.2. Summary and Hypotheses

Hypotheses were shaped according to Eisenhardt [21]. According to her, the central idea is that researchers constantly compare theory and data – iterating toward a theory which closely fits the data. As suggested by Eisenhardt, [21] we iteratively collected evidence for each category, explored the logic across cases and searched evidence from the data for each identified relationship between categories.

The summary of the factors that affect testing organizations and knowledge management is expressed in Table 3. The business orientation affected the testing organization, because the testing processes adapted according to the business orientation. The business orientation affected also the knowledge management strategy i.e. the emphasis between codification (explicit knowledge) and personalization (tacit knowledge). In addition, the business orientation affected testing outsourcing because product oriented OUs codified more knowledge that further enabled outsourcing of testing resources. Further, the business orientation affected knowledge transfer, because service oriented OUs suffered more from the barriers to knowledge transfer.

4.2.1. Hypothesis 1: Business orientation affects testing organization. In product oriented OUs, the organization model ought to support repetitive testing tasks that enable development of a deeper expertise. In service oriented OUs, the organization model ought to support adaptation to the processes of the customers that demand broader expertise. In product oriented OUs, the testing organization adapted to the processes of the OU’s own software development, and in service oriented OUs, the testing organization adapted to the processes of the customers.

“We have to adapt to the processes of customers. They are almost always different.” – Testing manager, the purely service oriented Case E.

4.2.2. Hypothesis 2: Business orientation affects the knowledge management strategy. OUs ought to adjust the knowledge management strategy according to the business orientation. The codification of knowledge was more systematic in product oriented OUs because of the repetitive

| Table 3. Factors affecting testing organization and knowledge management in Case OUs |
|---------------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| Case OU Category                | Case A          | Case B          | Case C          | Case D          | Case E          |
| Testing organization            | 4, System testing as a part of the software quality assurance organization [10]. Testing partly integrated with software development. More specialization, less job rotation between testers. | 6, Centralized testing organization [10]. Testing partly integrated with software development, partly adapts to the processes of external software development. Less specialization, more job rotation. | 7, System testing as a centralized testing organization with a testing technology center [10]. Testing partly integrated with software development. Medium specialization, medium job rotation. | 7, System testing as a centralized testing organization with a testing technology center [10]. Testing as a separated function, measurement function. Less specialization, more job rotation. | 6, Centralized testing organization [10]. Testing adapts to the processes of external software development. Medium specialization, more job rotation in short-term projects, less in long-term projects. |
| Knowledge management strategy   | Codification less emphasized, growing. Domain knowledge emphasized in testing customized systems. | Codification more emphasized in the customers’ projects. Domain knowledge more emphasized when testing own software. | Codification less emphasized, growing. Domain knowledge emphasized in testing customized systems. | Codification more emphasized. Domain knowledge less emphasized. | Codification more emphasized in the customers’ projects. Domain knowledge emphasized. |
| Testing outsourcing             | Case A has considered testing outsourcing, but not yet used. | Case B both uses outsourced testing resources and offers testing services to other companies. | Case C does not use outsourced testing resources. | Case D concentrates on core business. Testing is outsourced to a great extent. | Case E offers testing services to other companies. |
development and testing tasks, but also in OUs offering testing services because they had to adapt to the quality system of their customers. Codification was seen to improve the knowledge transfer.

“Making testing results (documents) available organization wide is an efficient way to avoid extra work and costs in development and testing.” – Testing manager, Case D

The dimension “job rotation” of the category testing organization was more associated with explicit knowledge. “Specialization” was more associated with tacit knowledge. In the product oriented Case D and in OUs offering testing services (cases B and E), jobs were rotated inside the testing organization, and in case OUs A and C jobs were rotated between development and testing. Job rotation enhanced knowledge transfer.

“We use job rotation to disseminate testing knowledge.” – Testing manager, Case B.

Testers’ domain knowledge was more emphasized in service oriented OUs than in product oriented OUs because the adaptation to the processes of the customers demanded broader expertise.

4.2.3. Hypothesis 3: Business orientation and knowledge management strategy affect outsourcing. Codification of knowledge enables testing outsourcing. In product oriented OUs the testing process was more repetitive, compact and easier to specify, which enabled the codification of knowledge and further outsourcing of testing.

“In product oriented companies the testing process is more compact and easier to specify. They are more potential customers.” – Testing manager, Case E (testing service provider).

Product oriented OUs were more willing to outsource testing because they were more prepared for outsourcing because of codified knowledge.

“During the last year we decided that testing is not our core business. We decided to use outsourced testing resources.” – Testing manager, Case D.

The lack of codified knowledge, e.g. product knowledge, limits outsourcing. External testing organizations do not have all the information concerning e.g. new products, methods, and tools.

“We can not offer all testing services. We do not have the needed information.” – Testing manager, Case B.

4.2.4. Hypothesis 4: Business orientation affects barriers and enablers of knowledge transfer. In the product oriented Case D knowledge transfer was formal, planned, and transparent because of the quality system which emphasized codification. The barriers that caused inefficiency in knowledge transfer in OUs developing customized systems included the growing distance between design and testing locations and the lack of absorptive capacity of the persons. In the OUs offering testing services the lack of planned and formal meetings and customer control of knowledge transfer were identified as barriers to knowledge transfer.

The distance between developers and testers affected the codification of knowledge. If developers and testers were not co-located, the need for codification was higher.

“We have started to develop software in a new subsidiary (in a foreign country). They have to document their work well.” – Tester, Case C.

In Case E, a contact person was mentioned as an enabler in increasing the efficiency of knowledge transfer with customers.

“There should be one and only one contact person in charge of knowledge transfer in both design and testing organizations.” – Testing Manager, Case E

5. Discussion and conclusions

The objective of this qualitative study was to understand the complex practice of software testing, and based on this understanding, to develop hypotheses concerning testing organizations and knowledge management. The affecting factors and their relationships were analyzed using the grounded theory method. Both the analysis of the software testing practice and the hypotheses were grounded on the data.

The hypotheses included that the business orientation of an OU affects testing organization, knowledge management strategy, and outsourcing of testing. Further, identifying and avoiding barriers and using enablers according to the OU’s business orientation improve knowledge transfer between development and testing.

Hansen et al. [33] discuss business orientation from the competitive strategy point of view. They divide companies according to their standardized or customized products. Companies that follow a standardized product strategy sell products that do not vary much, if at all. A knowledge management strategy based on reuse fits companies that create standardized products. In companies that sell customized products and services, most of the work goes toward meeting particular customers’ unique needs. Because those needs will vary dramatically, codified knowledge is of a limited value. Hansen et al. [33] conclude that companies that follow a customized product approach should consider the personalization model.

Only the product oriented Case D had clearly focused on a codification strategy throughout the entire
organization, because it was specified by their quality system. The case OUs A and C, that developed customized systems, relied more on the personalization model. The case OUs B and E followed the codification strategy more consistently, because the interaction with their customers required formal documentation. Hansen et al. [33] view that the company should focus on only one of the knowledge management strategies. In our study, OUs offering testing services made an exception. Although the OUs B and E offered services, their knowledge management strategy was more inclined towards codification. This does not seem to directly fit to the description by Hansen et al. [33] that companies providing services should focus only on the personalization strategy.

Product oriented OUs used more outsourced testing resources than service oriented OUs. Because product oriented OUs codify more knowledge, the development and testing processes are more repetitive, and the need for outsourcing is more predictable. Cowan et al. [34] state that codification will provide high benefits in stable systems characterized by specific requirements of knowledge transfer and communication, such as delocalization and externalization, i.e. outsourcing.

OUs strive for efficient knowledge transfer. Identifying and avoiding barriers and using enablers improve knowledge transfer. Szulanski [4] writes that the ability to transfer best practices internally is critical to a firm’s ability to build a competitive advantage.

To guarantee the validity of the study we used probability sampling when selecting the sample for the first interview round and theoretical sampling when selecting case OUs. Robson [35] lists three threats to validity in this kind of research: reactivity (the interference of the researcher’s presence), researcher bias, and respondent bias and strategies that reduce these threats. To avoid these threats, the interviews were conducted by two researchers and the data was analyzed by four researchers (observer triangulation) [36]. We compared the data (data triangulation) and the results (method triangulation) of this study with our earlier quantitative [27] and qualitative studies [37].

The limitation of the study is the number of the case OUs. It is obvious that increasing the number of cases could reveal more details. However, our target was not to create a comprehensive list of the factors that affect testing organizations and their knowledge management but to cover the most important factors from the point of view of our case OUs. The influence of testing schedules is not covered here because it is out of the scope of this study and it is already reported in [28].

We believe that paying more attention on the business orientation when organizing testing, selecting knowledge management strategy, outsourcing testing, and improving knowledge transfer is important for the practice of software testing. The results of this study can be used in developing testing organizations and their knowledge management strategy.

6. Acknowledgments

This study was supported by the ANTI-project (www.it.lut.fi/project/anti) funded by the Technology Agency of Finland (project numbers 40155/04 and 40191/05) and by the companies ABB, Capricode, Delta Enterprise, Metso Automation, Outokumpu, Siemens, and SoftuTest.

7. References


