

Information Sources and Their Importance to Prioritize Test Cases in the Heterogeneous Systems Context

Ahmad Nauman Ghazi¹, Jesper Andersson², Richard Torkar³,
Kai Petersen¹, and Jürgen Börstler¹

¹ Blekinge Institute of Technology, Karlskrona, Sweden

² Linnaeus University, Växjö, Sweden

³ Chalmers University of Technology, Gothenburg, Sweden
{nauman.ghazi, kai.petersen, jurgen.borstler}@bth.se,
jesper.andersson@lnu.se, richard.torkar@chalmers.se

Abstract Context: Testing techniques proposed in the literature rely on various sources of information for test case selection (e.g., requirements, source code, system structure, etc.). The challenge of test selection is amplified in the context of heterogeneous systems, where it is unknown which information/data sources are most important.

Contribution: (1) Achieve in-depth understanding of test processes in heterogeneous systems; (2) Elicit information sources for test selection in the context of heterogeneous systems. (3) Capture the relative importance of the identified information sources.

Method: Case study research is used for the elicitation and understanding of which information sources are relevant for test case privatization, followed by an exploratory survey capturing the relative importance of information sources for testing heterogeneous systems.

Results: We classified different information sources that play a vital role in the test selection process, and found that their importance differs largely for the different test levels observed in heterogeneous testing. However, overall all sources were considered essential in test selection for heterogeneous systems.

Conclusion: Heterogeneous system testing requires solutions that take all information sources into account when suggesting test cases for selection. Such approaches need to be developed and compared with existing solutions.

1 Introduction

With the technological advancement in the software industry, more and more heterogeneous systems are introduced in the market. A heterogeneous system is comprised of multiple subsystems. A review of literature on the topic conducted by us did not reveal a commonly agreed definition of what a heterogeneous

system is. Though, literature provides examples, such as, heterogeneity in this context can refer to that systems are implemented on different platforms, being developed using different processes, be of different size, etc.

A subsystem can exhibit heterogeneity in terms of both hardware and software. It does not limit itself to these aspects, though. Heterogeneity can also occur at different levels within the software development process. Heterogeneous systems are inherently complex and pose certain challenges to the verification and validation activities, such as specification, selection and execution of tests.

Testing of heterogeneous systems has received vast attention in recent years. In large heterogeneous systems it was observed that regression test suites grow, and hence require too much time to execute. In response, there is a need to prioritize and select test cases [1]. The challenge of test selection has been thoroughly investigated in previous research (e.g., in systematic reviews [4, 10]), but there still is a need to understand which information needs and sources are of relevance to guide practitioners of heterogeneous systems in selecting tests.

In this research, we identify the information sources required by practitioners involved in developing heterogeneous systems to prioritize test cases. This is done in a two step process. In the first step an industrial case study is conducted to understand how heterogeneous systems are tested and to elicit information sources, followed by an exploratory survey. The findings are compared with the literature investigating test selection independently of heterogeneous systems. The information gathered could be used in organizations to assure that the required information is available to testers to support them during the selection process. From an industrial perspective, identification of these information sources will further help to develop a framework to initiate a search space for automating test selection in different stages of development using search-based software testing techniques.

The remainder of the paper is structured as follows: Section 2 presents the related work. Section 3 outlines the research method, followed by the results in Section 4. Section 5 concludes the study by presenting a discussion of observations from the results.

2 Related Work

Heterogeneous Systems: Testing a heterogeneous system implies that several possible configurations must be tested. Reuse of artifacts is one way to speed up such repetitive activities considerably [6]. Otani et al. propose a framework that depends heavily on UML artifacts, which are used to automate independent verification and validation practices using generative technologies. These reusable artifacts are stored as XML data and reusable for other activities as well as other testing projects [5]. Otani et al. extend this work further [6] and introduce goal-driven reuse of artifacts in the context of heterogeneous systems. Changing configurations pose challenges to combinatorial testing techniques. To that end, Cohen et al. [3] conducted an empirical study to quantify the effectiveness of test suites. The study shows that there is an exponential growth of test cases when

configurations change and subsets of test suites are used, similar to what is common in regression testing. Vega et al. [11] propose a TTCN-3 based framework to test HL7 health-care applications. The technique supported by the framework is generic and does not need customization every time a configuration changes. Brahim et al. [2] provide a technique to specify test cases in globally distributed environments. This framework uses the UML 2 testing profile and TTCN-3 for test specification and generation. The authors claim that the use of TTCN-3 in combination with other languages and test notations ensures transparency and cost benefits. Overall, testing of heterogeneous systems involves testing multiple configurations and dealing with complex systems across a variety of platforms.

Test Case Selection: We identified two recent secondary studies on test case prioritization and selection (cf. [4, 10]). Singh et al. [10] conducted a systematic literature review on test case prioritization in the context of regression testing. The authors implicitly mentioned some information sources while categorizing the techniques for test case specification and prioritization. Singh et al. categorized techniques, which implicitly point to the following information sources: requirements, source code structure, historical information (with respect to changes made to the system or execution history), fault-driven approaches (e.g., fault proneness), as well as cost. Furthermore, combinations of approaches have been evaluated. Engström et al. [4] identified code-based techniques as the most commonly investigated in the literature.

3 Research Method

We first conducted a case study [9] to gain an in-depth understanding of information sources and their relevance for heterogeneous system testing. Thereafter, based on the case study results, a survey is conducted to explore the importance of the identified information sources in a more broader perspective.

3.1 Case Study Design

Objectives. The case study took place in close collaboration with industry. The long-term expectation of our industrial partner from this research is to optimize their overall test methodology and practices for the organization's software product line. The overall long-term objectives of our research are to understand the state of practice in testing heterogeneous systems, and how this relates to a system's heterogeneity characteristics. In the long-term perspective a process for optional test selection should be defined. Identifying the relevant sources for test selection is the first step towards that overall goal. The goal of this study is formulated by following the Goal-Question-Metric approach: To gain an in-depth understanding of the test process and relevance of information sources (Purpose) for test selection (Issue) in the context of heterogeneous systems (Object) from an industrial point of view (Viewpoint).

Research Questions. In this exploratory case study we intend to answer the following research questions:

- *RQ1: How are complex heterogeneous systems tested in practice from a process perspective?*
- *RQ2: What are the different information sources used in test selection?*
- *RQ3: Which information sources are most relevant in selecting and prioritizing test cases for testing complex heterogeneous systems in that process?*

Case and Context: Petersen and Wohlin [8] suggest a checklist to report context in relation to an object of study. The test process and related information sources for test case selection are the object of study in this research. Table 1 provides an overview of context information for the test process studied, derived from process documentation and the interviews conducted.

Table 1. Context

Context	Description
Product	<i>System type:</i> System of systems (multiple subsystems developed autonomously) with a total of 22 subsystems <i>Domain:</i> Telecommunication systems <i>Customization:</i> Highly customizable based on individual customer needs <i>Programming language:</i> Java <i>Quality:</i> The most highly prioritized quality attributes driving tests are: (1) scalability, (2) usability, (3) performance, (4) robustness and recoverability, (5) throughput, (6) stability, (7) variability, and (8) maintainability of a total of 33 attributes prioritized.
People	See Table 2
Organization	ISO 9001:2000 Certified
Size	More than 5000 employees
Market	Market-driven development (high number of potential customers)
Process	Agile software development

Data Collection: Multiple methods of data collection are used in this case study. However, the main source of data collection in this study are semi-structured *interviews* with selected practitioners from the case company. These interviews mainly resulted in identifying different dimensions of heterogeneity, the test selection process, identification of multiple key information sources that lay the foundations for the test specification process, important quality criteria, weaknesses in the test process and factors that influence test case selection.

Although the interviewed practitioners have diverse roles, the information regarding different information sources for test selection, test prioritization based on quality attributes, and challenges in the test process converged after the third interview. Table 2 provides a brief profile of the interviewed practitioners.

Interview questions were structured into six themes, namely: (1) Experience and current role of the interviewee, (2) verification model, (3) levels of heterogeneity in product and test process, (4) test prioritization based on quality attributes, (5) test selection process, and (6) tester’s perspective on weaknesses in test process. These interview themes are formulated to answer the research questions as well as to gain a better understanding of the current test activities at the organization which is the case under study and the context of each activity. The interviews were semi-structured and included open-ended questions. Each interview took approximately 60 minutes.

Table 2. Practitioners' profile

ID	Description
1	The interviewee is currently the head of functional node test team responsible for test activities within a subsystem, and has been working as a functional tester in this company since 2005.
2	The interviewee has worked in this company for 16 years overall. For the first 10 years worked as a requirement engineer and currently working as functional tester for last 6 years.
3	The interviewee is working as a system developer at the company and is exposed to design, development and test activities at the company. The interviewee has also worked in development of legacy system which is to be replaced with the current system.
4	The interviewee is part of the core test team and therefore is responsible for design decisions. The interviewee has responsibility for overall development and test strategy.

Documentation is also used as a data collection method for triangulation in this case study. Process and design documents of the product were obtained to capture the development and test processes at the company. These documents provided the researchers with a better understanding of different test levels and the strategy of the company.

Data Analysis: Interviews were recorded with the consent of the interviewees and later transcribed. For data extraction from these transcribed interviews, we used color coding, where unique colors were assigned to key areas important to answer the research questions. We identified and color-coded the following key areas:

- Test activities including test selection process as well as test execution
- Information sources for test selection
- Different levels of heterogeneity exhibited by the case under study
- Quality attributes that play an important role in test selection
- Weaknesses and challenges in the current test process

The *documentation* was analyzed using the same coding scheme.

Threats to Validity: A discussion of validity threats for software engineering is provided in [7].

Objectivity: An important threat is if the questions asked during the interviews are misunderstood. However, this threat was reduced by explaining the context to the interviewees, cross referencing the information gathered with the product documentation, and through member checking. Another threat to objectivity is that the interviewees may provide the information from a single perspective depending on their roles. This threat is minimized by carefully choosing the interviewees from different testing teams and development teams.

Theoretical Validity: Theoretical validity is concerned with not being able to capture what we intend to capture. In this case, we intend to prioritize the information sources used for test selection in the context of heterogeneous systems. To reduce this threat to theoretical validity, we first captured the information sources for test selection using both documentation and the interviews. We contacted further practitioners from other organizations involved in development of heterogeneous systems. These practitioners were asked to prioritize the list of information sources extracted from the documents and interviews to strengthen our findings.

Generalizability: The exploratory nature of the case study does not allow to generalize the results for all heterogeneous systems and all types of organizations. However, the results can be generalized to large scale telecommunication organizations involved in the development of system of systems that exhibit heterogeneity. Furthermore, even though not statistically generalizable, the results from the survey allowed to make qualitative reflections on the information gathered.

Interpretive Validity: Interpretive validity is concerned with researcher bias when drawing conclusions. Since the involved researchers have no particular preference for any of the solutions presented in the case study based on previous research, this threat can be considered low in this study.

3.2 Survey

Objective: The survey captured the relative importance of information sources in relation to test case selection, leading to research question *RQ3*.

Survey Distribution and Sample: A convenience sampling strategy was followed targeting practitioners that work with heterogeneous systems. We utilized personal contacts as well as communities (e.g., LinkedIn and Yahoo Groups) to acquire additional answers. Overall we obtained 42 answers of which 27 were complete and could be used for analysis.

Instrument: The survey¹ was capturing information about respondents, the characteristics of their organization and products, as well as test coverage goals and importance of the information sources for test selection and prioritization.

Analysis: For the analysis descriptive statistics are utilized.

Validity Threats: The same types of validity threats as for the case study.

Objectivity: To avoid possible misunderstandings of the survey questions, it was pre-tested and revised based on the feedback received. Furthermore, the survey was tested for duration to take at most 15 minutes to complete to avoid maturation.

Theoretical validity: One threat to inference is the number of participants. The present results are not statistically generalizable to a whole population. However, for the given context information gathered about the participants, some interesting qualitative observations can be made.

Generalizability: The surveyed companies have specific context characteristics that limit generalizability. The majority of respondents is related to consultancy (35.7%), followed by computer industry (28.9%) and communications (25.0%); other industries are under-represented. Agile and hybrid processes have the highest representations. Another possible bias is that only persons with a specific interest may have answered the survey.

Interpretive validity: Given that only quantitative data is studied in the survey part, the risk of bias is reduced.

¹ The supplementary information about interview and survey can be found at <http://www.bth.se/tek/aps/kps.nsf/pages/sources-for-test-case-selection>

4 Results

4.1 Test Process for Testing Heterogeneous Systems (RQ1)

Heterogeneity in the System under Test: The system of systems approach [1] used in the development of software products at the case company leads to heterogeneity. We identified three different dimensions of heterogeneity: (1) hardware heterogeneity, (2) software heterogeneity, and (3) process heterogeneity. These dimensions of heterogeneity are important to consider in an overall strategy to minimize the challenges they pose to the overall system(s).

During the interviews with the practitioners, we found that third-party open-source components are also used throughout the development of subsystems of the case company's next generation billing system. As the next generation billing system is developed to replace an existing telecom billing system that is used by a large number of globally distributed customers, there exist some legacy software components that are reused in the new system. Although, the complete product is developed using Java, due to legacy software there exist heterogeneity on the platform level.

It is also important to note, that the company recommends the use of certified hardware, but does not limit the customers to use the recommended hardware, which leads to hardware heterogeneity. Another factor that leads to hardware heterogeneity is the notion of variability. The next generation billing system is shipped to customers as part of different commercial offerings customized as per customer requirements. Therefore, this system can be used on multiple clusters configured to function as a single entity.

We also found, that due to multiple test levels, the practitioners perceive the underlying heterogeneity as an important factor while designing the test strategy. Heterogeneity at the verification level at telecom grade systems is very important to be considered. Otherwise, it may lead to the challenge of dealing with a huge number of tests to be executed to reach a thorough test coverage.

Test Levels and Selection Process: The case company is involved in the development of a large system of systems that exhibits heterogeneity at multiple levels, therefore optimal test selection is an important challenge. The case under study involves a test strategy with four different test levels: (1) software component test, (2) application component test, (3) subsystem component test, and (4) offering test. Each test level involves different test activities to ensure software verification.

Software component test: Test activities at this test level comprise interface and unit testing. The test tool JUnit is used for unit testing.

Application component test: This test level is targeted for testing of sub-subsystems that comprise several software components. The major challenge at this test level is the integration of software components to form functionally independent sub-subsystems. Integration testing at this level is performed using Pax-Exam, which is a test tool specialized for integration testing of osgi components.

Subsystem component test: To test a functionally independent subsystem comprising several sub-subsystems, this test level includes multiple test activities. These test activities are functional testing, unit testing, tests for installation, testing upgrades, integration testing and tests for stability. A virtual test environment is set up for the continuous subsystem component integration testing. This environment consists of subsystems that comprise more components. Each subsystem team owns the component and is thus responsible for its delivery, update and installation.

Offering test: Once a subsystem component is verified, it is moved into the next integration environment for commercial offering validation. At this test level, in the integration environment different configurations and end-to-end communication are tested. This test level also validates the deployability on commercial offering environments. The deployment environment consists of delivered subsystem components as per the requirement of commercial offering. Each subsystem component is evaluated upon deployability on certified hardware.

Test case selection is performed for each of the test levels mentioned above. A core test team is responsible for developing the overall test strategy for the complete product. We also found that, apart from the core test team, there are dedicated test teams for each test level that are responsible for both test selection and execution. Test prioritization at each level is also done by these dedicated teams.

For test case selection and test prioritization, an in-house software is developed. This software, based on features, tags the test cases with certain labels by analyzing the keywords in each of the test cases. These test cases are further categorized into various groups to test certain features of the system. This software also assigns weights to different feature categories taking in consideration the system's requirements to prioritize the categories to be tested more frequently under constraints. However, under the time constraints, functional system testing is given the highest priority for verification of a commercial offering.

Every software component undergoes the unit test and application test, separately. Once these components are integrated to form an independently functional subsystem, functional node testing is done to identify possible defects in the system. Sub-net testing and network integration testing are later carried out, when all subsystems are integrated in the final product.

For different test activities and levels, the case company uses different testing tools. JUnit is used for unit testing of software components and Pax-Exam is used for the purpose of integration testing. However, there are other in-house testing tools used for functional testing of the functionally independent subsystems, as well as for system testing. Other than functional testing, non-functional testing for performance is also carried out at different levels.

Weaknesses in the Process for Test Case Selection: From the interviews we found the main weakness in the current test process to be a large regression test base. There are many test levels and each test level generates a huge number

of test cases. The heterogeneous nature of the system also poses a huge challenge to the testing of the complete system.

To maximize the test coverage, functional test cases to cover every subsystem, sub-subsystem and its components are developed. These test cases are most of the time overlapping with the unit tests, therefore a good test selection process is required to avoid this overlapping of functional test cases and the unit test cases. The maximum test coverage is an important test objective stated by the practitioners. However, it also leads to a combinatorial explosion and in this case a robust test selection process that can help avoid executing the same test cases again and again.

It is also a challenge that each commercial offering has a different configuration and testers are required to test all possible configurations. Therefore, a process for efficient system configuration is needed. However, this challenge is more related to the design and the development process but it affects the overall test process. There shall be a procedure to facilitate functional testing to be self contained so they can be installed and configured automatically for each customizable commercial offering.

4.2 Information Sources (RQ2)

We identified a number of information/data sources that are of vital importance for the test selection process. The information sources comprise important information sources needed for optimal selection and prioritization of test cases. Other than the information sources, different roles that are vital for test selection are included in the list of data sources. The information sources mentioned later in this section are extracted from the documentation provided by the company and further validated during the interviews with the practitioners. Practitioners are also asked to list additional information sources, which they perceive as important for test selection.

Functional and *non-functional requirements* serve as the foundation of the complete development process that also encompasses the testing. The *System model* of the complete product provides a detailed overview of the system and its constituent subsystems. This model also identifies the data flow between the subsystems as well as sub-subsystems and software components. *Configurations* are important, as the system we studied has multiple telecom operators as its target customer base. The system is developed to be customizable for different customer needs. Various system configurations need to be tested for each commercial offering. *Test objectives* are the reason or purpose that drive the design and execution of a test case. These test objectives are used to derive an effective test strategy for different test levels and also serve as an important data source for test selection under resource constraints. *Environment descriptions* and *high-level analyses*, based on tester's input, are also found to be data sources that shall be used in the test selection process.

From the interviews, we found that in *software component test*, an important data source for unit and integration test activities is the *tester's input*. However, the importance of *requirements* and *test objectives* as information sources, for

test selection at this test level, can not be overlooked. All data sources mentioned for *software component test* are also valid for *application component test*. Functional testing on the basis of *requirements* and *test objectives* is carried out at this test level. *Subsystem component test* comprises mainly integration and verification activities. Hence, *environment description*, *test objective* and *tester's input* serve as the prime data sources at this test level. As for the *offering test*, other than all data sources mentioned for preceding test levels, *configurations* play an important role. For testing of each commercial offering, specific configurations are pushed to the system to customize it according to the requirements. These configurations must be considered for test selection at this level to avoid execution of unrelated tests in the context of specific offerings and customer requirements.

4.3 Relative Importance of Information Sources (RQ3)

Demographics: With respect to the roles of subjects, mostly technically oriented roles answered the survey, in particular developers (22.2%), software architects (18.5%), software verification and validation (18.5%), and quality control/management (14.8%). Other roles were only represented by less than 10% of the respondents (system analysts, project managers, product managers, and software process engineers). With respect to experience in software engineering the average experience was 10.55 years. The average experience in testing heterogeneous systems was 4.63 years. Companies of various sizes participated: less than 50 employees (18.5%), 50 to 249 employees (29.6%), 250 to 4499 employees (29.6%), and 4500 and more (22.6%). The most common system type was data dominant software (63.0%), followed by control-dominant software (25.9%), computation-dominant software (25.9%), and systems software (22.2%). Other systems were represented by 14.8%. The most common development models were agile (29.6%) and hybrid processes dominated by agile practices (29.6%). Other processes were represented by 11% or less (e.g., waterfall, V-model, spiral model).

Coverage and Information Sources Prioritization: We analyzed the importance of coverage criteria (see Figure 1), and found that specification-based criteria are considered most important, followed by fault-based coverage. All coverages are considered overall important.

We also analyzed how practitioners prioritize different information sources (see Figure 2), and found that most practitioners consider functional requirements as the most important information source followed by test objectives and system model. Each identified information source is considered important to a significant number of practitioners and therefore cannot be neglected. One respondent mentioned that they are utilizing 37 different information sources, but did not specify which these are.

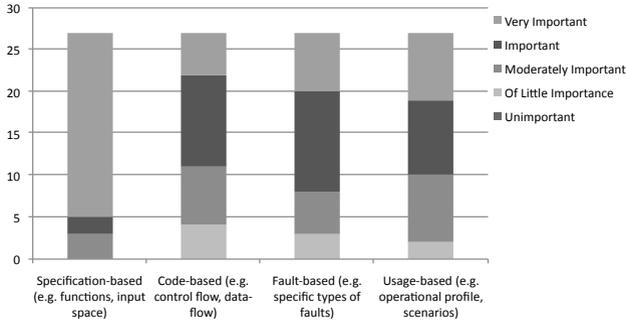


Fig. 1. Importance of test objectives

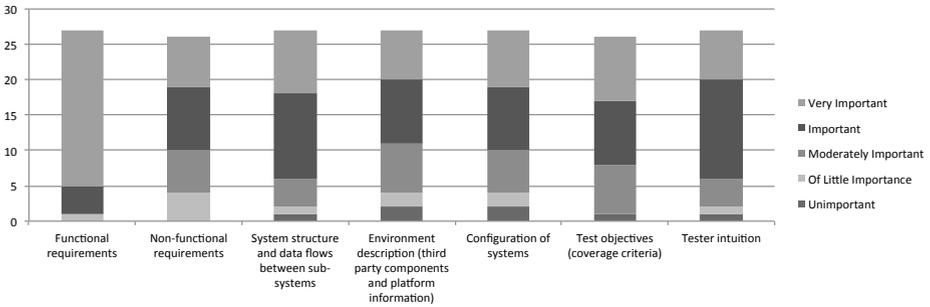


Fig. 2. Importance of data sources in test specification

5 Discussion and Conclusion

In this study, we investigated a heterogeneous testing process to understand how complex heterogeneous systems are tested. The main focus of this research was to identify information sources for test case selection. The literature as well as the results of our case study indicate that it is essential to reduce the size of continuously growing test suites.

We conducted a case study to gain an in-depth understanding of test selection in heterogeneous systems, followed by a survey to understand the relative importance of information sources. The following summarized our findings with respect to our research questions:

Test Process (RQ1): Multiple hierarchies of test levels have been identified that are relevant to test selection. Overall, the test levels map well to test levels one would expect from the V-model. Functional testing was given the highest priority, though at the same time quality characteristics were prioritized highly. The complexity of the overall system led to having several testing teams focusing on their specific test levels. Overall, a situation of overlapping tests occurred. Furthermore, core challenges were growing regression test suites and a high number of configurations to be tested when combining different (sub-)systems. The

lessons learned regarding RQ1 are: First, a systematic investigation is needed to understand responsibilities of different test levels to avoid overlapping test cases. That is, one has to clearly determine which value each test level adds in terms of the kind of quality that is assured with it. Having a good understanding of this could potentially reduce the size of test suites significantly. Second, solutions from software product-line testing might be applicable to the testing of heterogeneous systems, since product-line testing faces similar challenges: when different (sub-)systems/features are combined, the number of configurations to be tested increases.

Information Sources for Test Selection and their Priorities (RQ2/3): In the research we identified multiple sources of information. Comparing with the literature, non-functional requirements and environment description are highlighted in the heterogeneous systems context. Generally, all sources were rated as either very important or important by the majority of respondents. But, the most important source of information were functional requirements. Looking at the existing techniques proposed for selection and prioritization, those techniques combining different information sources for prioritization are hence of particular interest, given that all sources appear to be of relevance when selecting tests in this context. Figure 3 shows that, given the high priority of information sources, they all have to be considered in the selection process. The selection process utilizes approaches to search for a set of good solutions for the next regression test run, taking the information sources into account.

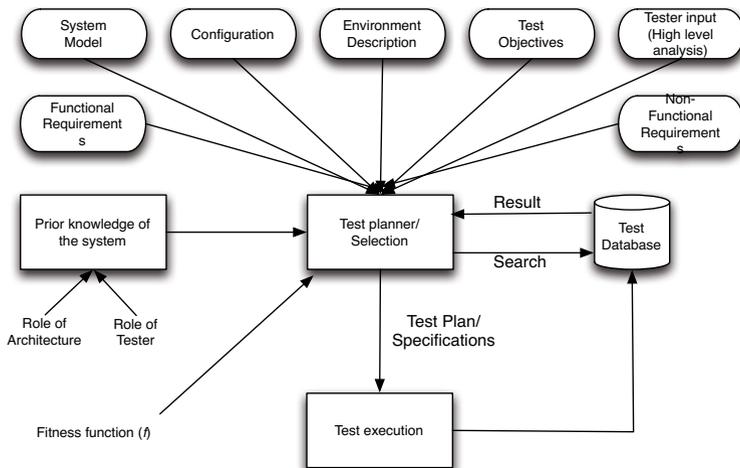


Fig. 3. Information Sources in the Test Process

In future work, we propose to focus on identifying and evaluating test selection approaches that are able to utilize all data sources for test selection, and comparing them with existing solutions on real systems.

References

- [1] Ali, N.B., Petersen, K., Mäntylä, M.: Testing highly complex system of systems: an industrial case study. In: ESEM, pp. 211–220 (2012)
- [2] Andaloussi, B., Braun, A.: A Test Specification Method for Software Interoperability Tests in Offshore Scenarios: A Case Study. In: ICGSE, pp. 169–178 (2006)
- [3] Cohen, M.B., Snyder, J., Rothermel, G.: Testing Across Configurations: Implications for Combinatorial Testing. *Sof. Eng. Notes* 31(6), 1–9 (2006)
- [4] Engström, E., Runeson, P., Skoglund, M.: A systematic review on regression test selection techniques. *Inf. & Soft. Tech.* 52(1), 14–30 (2010)
- [5] Otani, T.W., Michael, J.B., Shing, M.-T.: Software Reuse in the IV&V of System of Systems, May 30-June 3, pp. 1–5 (2009)
- [6] Otani, T.W., Michael, J.B., Shing, M.-T.: Goal-Driven Software Reuse in the IV & V of System of Systems, pp. 1–6 (June 2010)
- [7] Petersen, K., Gencel, C.: Worldviews, research methods, and their relationship to validity in empirical software engineering research. *Mensura* (2013)
- [8] Petersen, K., Wohlin, C.: Context in industrial software engineering research. In: ESEM, pp. 401–404 (2009)
- [9] Runeson, P., Höst, M.: Guidelines for conducting and reporting case study research in software engineering. *Emp. Soft. Eng.* 14(2), 131–164 (2009)
- [10] Singh, Y., Kaur, A., Suri, B., Singhal, S.: Systematic literature review on regression test prioritization techniques. *Informatica* 36(4), 379–408 (2012)
- [11] Vega, D.E.: Towards an Automated and Dynamically Adaptable Test System for Testing Healthcare Information Systems. In: ICST, pp. 331–334 (2010)