2012 Sixth International Conference on Internet Computing for Science and Engineering

Ontology-based Testing Platform for Reusing

Xuexiang Li
School of Software Technology
Zhengzhou University
Zhengzhou, China, 450002
lxx@zzu.edu.cn

Wenning Zhang
Software College
Zhongyuan University of Technology
Zhengzhou, China, 450007
smile8211@163.com

Abstract— **With more and more attention on the software quality, the test case reuse has become a focus in current research. By integrating knowledge management and software reuse theory, several design guidelines for reusable test cases are identified according to the characteristics of software components. A reusable test case knowledge management model is proposed to support the knowledge reuse based on the ontology representation of reusable test cases in this paper. With the ontology and knowledge management model, test engineers can retrieve and reuse test cases flexibly.**

Keywords-test case; knowledge management; ontology representation; test case reuse

I. INTRODUCTION

With the rapid development of information technology, the software testing has become a necessary and important part of the software development lifecycle. As one of the most decisive factors, the quality of test case has a direct and significant impact on the software testing quality and hence the quality of the software. By introducing the reuse theory into testing area, the test case reuse will be very effective [1] and can promote the test case design efficiency.

But there are some general issues [2] in the test case design activity, such as:

- Test engineers reuse test cases randomly and occasionally;
- The transformation of test case design knowledge among teams is not fluent enough;
- The loss of testing knowledge is serious especially in recent years;
- It is hard to select appropriate test engineers for the specific testing projects based on their knowledge.

To solve the issues mentioned above and the lower productivity payoffs [3] from the informal test case reuse, this paper focuses on systematical analysis of the ontology representation and the knowledge management model for reusable test cases to satisfy various reusing requirement.

II. KNOWLEDGE MANAGEMENT AND TESTING

Knowledge management is a large interdisciplinary field, encompassing anthropology, social psychology, organization theory, and economics etc. According to Davenport in 1998, the knowledge management is a method that simplifies the process of sharing, distributing, creating, capturing and understanding of knowledge. And it has been viewed as a source of competitive advantage for organizations [4].

In order to build effective technologies for knowledge management, the first issue we need to address is how to represent knowledge. Historically, the term ontology arises out of the branch of metaphysics, one of domain of philosophy. Ontology deals with the nature of reality – of what exists. Now, it has been applied to domains or disciplines. Within computer science and artificial intelligence, the term ontology was coined in the knowledge sharing and software reuse area for efficient engineering of knowledge-based systems. The ontologies are knowledge bodies that provide a formal representation of a shared conceptualization of a particular domain. The ontology engineering is a subfield of knowledge engineering. It studies the methods and methodologies for building ontologies: representation of a set of concepts and the relationships between those concepts. It can be used to describe and reason about the entities within the specific domain.

In the software testing, known as a knowledge-intensive activity, all test engineers are knowledge workers. In spite of the daily work of testing software according to the testing plan, the knowledge, skills, experience and inspiration related to the testing also play an important role. Managing the knowledge in software testing design activities can lead to greater competitive advantage, improved design quality, more effective management [5] and promote the reuse activity. Meanwhile it can help organizations to establish a formal process to identify, capture, store and retrieve critical knowledge and improve the work situation of test engineers and managers dramatically [6].

In recent years, the knowledge management in software testing related research covered that the knowledge management for reuse[5], the ontology representation and

978-0-7695-4705-3/12 $26.00 © 2012 IEEE
DOI 10.1109/ICICSE.2012.18

retrieval mechanism for reuse[7], the knowledge management model in testing[2], the knowledge management in software testing process[8] and the KBSE (knowledge-based software engineering conference) held annually etc. Although a great deal of research has investigated, the deep research on the reusable test case is needed for its own specific features.

III. REUSABLE TEST CASE

A software test case is a set of testing steps, inputs, and expected results under which test engineers can determine whether an application is working correctly or not. A reusable test case is the test case designed for the reusing purpose to capture common features among different test cases [9]. To satisfy the various reusing needs, a reusable test case should be usable, encapsulated, and adaptable. A well-designed reusable test case provides the users with a foundation from which more sophisticated methods can be learned. Based on the analysis of software components design criteria [10] and rich experience on test case design, we identified several important aspects to consider when designing reusable test cases as followed:

- A reusable test case should be well defined, reflecting the testing methods used and domain knowledge of test engineers. If the test case is more powerful to expose bugs, more credible and more understandable, it is said to be well-defined.
- A reusable test case should be designed for a special and distinct test requirement. The extensibility and reusability of test cases will be improved dramatically by following the "high cohesion, low coupling" design strategy.
- Test data and execution conditions can be separated to facilitate modification. By applying the idea reuse into test case design activity, the abstraction of test data should be conducted to focus on essentials for its better adaptability. The right level of abstractions will give us more coherent and more robust [11].
- A reusable test case should be fully documented, complete, correct, and follow the template of reusable test cases provided in the organization. The more detailed information test cases can provide, the more users can learn from them.

IV. THE ONTOLOGY OF REUSABLE TEST CASE

To establish a common understanding and reduce terminological ambiguity among test engineers when reusing software test cases, we give the ontology description of reusable test cases to represent the concepts set and its inner relationships. In theory, ontology is a formal explicit specification of a shared conceptualization [12] and has been applied to describe a variety of knowledge domains. Compared to traditional approaches, some generic advantages [13] of ontology description in the context of software testing can be derived: first, it can help to join information that normally resides isolated in several separate component descriptions; second, it can provide background knowledge that allows non-experts to query from their point of view. Once the knowledge is available in an ontology format, it might be feasible and flexible to reuse test cases and the related knowledge.

In the field of software engineering, the specific ontologies have been identified and used widely [12]. To develop the reusable test case ontology, shown in Fig.1, we established an analogy between software testing process and test case.

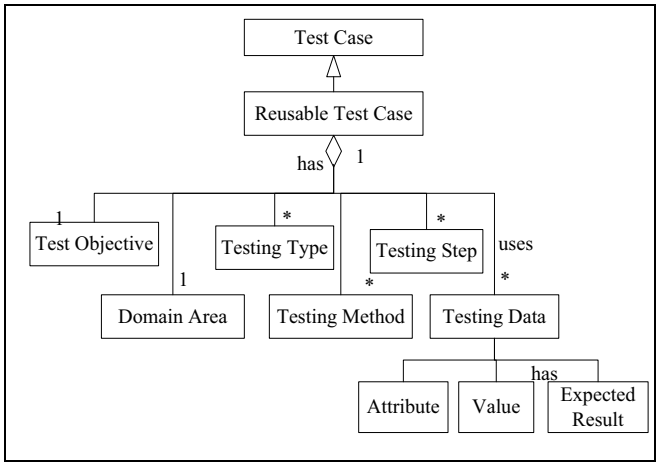

Figure 1. Ontology Representation of Reusable Test Case

A reusable test case is the sub-class of a traditional test case. It has its own test objectives, belonging to the specific domain area and testing type. By using several testing methods and following some testing steps, the testing data are designed based on the designers' skills and experience. Moreover, when designing a reusable test case, it is important to abstract the test data. To do so, we must establish the attribute, value and expected testing result for each test data group.

The testing types mentioned may be functional testing, usability testing, load testing, stress testing, installation testing and uninstallation testing etc.

The testing method mentioned may be equivalence partitioning, boundary value analysis, scenario analysis and random testing etc.

To formally define the reusable test case ontology, we establish a set of axioms. Axiom A1 shows the composition relationship (whole-part) between a test case and its elements. Such relation was formalized by the predicate ReusableTestCaseComposition (rtc, e_1, ..., e_n), denoting a reusable test case rtc is composed by elements e_1, …, e_n.

$$(\forall rtc, e_1, \ldots e_n)(ReusableTestCaseComposition(rtc, e_1, \ldots e_n) \rightarrow ReusableTestCase(rtc) \wedge Element(e_1) \wedge \ldots \wedge Element(e_n)) \quad (A1)$$

The consumption relationship between a reusable test case and its test data can be formalized by the predicate ReusableTestCaseConsumption (rtc, $attribute_1$, $value_1$, …., $attribute_n$, $value_n$), shown in axiom A2, denoting a reusable test case consumes n groups of testing data.

$$(\forall rtc, attribute_1, value_1, \ldots attribute_n, value_n)(ReusableTestCase Consumption(\forall rtc, attribute_1, value_1, \ldots attribute_n, value_n) \rightarrow ReusableTestCase(rtc) \wedge TestData(attribute_1, value_1) \wedge \ldots \wedge TestData(attribute_n, value_n)) \quad (A2)$$

The production relationship between a reusable test case and its expected results can be formalized by the predicate ReusableTestCaseProduction (rtc, $result_1$, …, $result_n$), shown in axiom A3, denoting a reusable test case should give the expected $result_1$, $result_n$ after execution, each for the input data group $attribute_1$, $value_1$, … $attribute_n$, and $value_n$.

$$(\forall rtc, result_1, \ldots, result_n)(ReusableTestCaseProduction(\forall rtc, result_1, \ldots result_n) \rightarrow ReusableTestCase(rtc) \wedge ExpectedResult(result_1) \wedge \ldots \wedge ExpectedResult(result_n)) \quad (A3)$$

V. THE KNOWLEDGE MANAGEMENT MODEL FOR TEST CASE REUSE

By using the ontology to represent the explicit specification of reusable test cases, test engineers can find the right knowledge at the right moment for a given testing project. To address the test case reuse issues, we need establish the positive and active knowledge transfer and sharing strategy among test engineers to promote the creation of learning organization. The knowledge management model for reusable test cases is presented in Fig. 2.

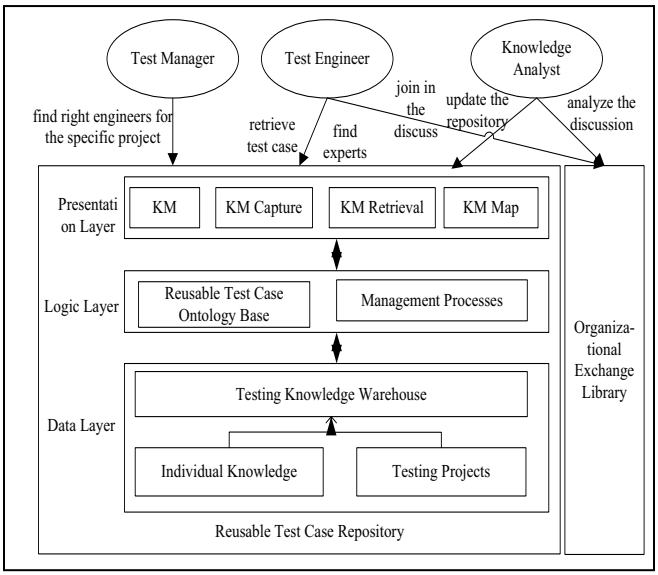

Figure 2. Knowledge Management Model for Test Case Reuse

The main elements of the model are test engineers, the reusable test case repository and the organizational exchange library. The organization can extract testing knowledge from members' tacit knowledge and testing projects' data to create the testing knowledge warehouse then establish the reusable test case repository according to the ontology representation. Test engineers may retrieve appropriate reusable test cases or find the right experts according to the knowledge map provided by the reusable test case repository. Additionally, test managers can find right test engineers to organize the team for the specific project in a short time based on the knowledge map too. All members in the organization can talk freely in the organizational exchange library which records the questions raised and their answers. Based on the systematic analysis on the discussions, the knowledge analysts submit the best practices into repository and update the knowledge level for each member in the organization.

The knowledge map mentioned in the model is a navigation aid to both explicit information and tacit knowledge. It shows the relationships between knowledge stores and the dynamics. Sometimes, it can be viewed as yellow pages directory to link the brief information and its detailed information. In our model, the expert database and the reusable test case indexes are maintained through the knowledge map by knowledge analysts.

In the reusing process, some non-technical problems are important, and maybe more difficult to solve than technical problems[3].To satisfy various reuse needs and improve the reuse efficiency, some effective management processes must be taken to ensure the quality of test cases:

As test cases evolve in applications, they may be changed for a variety of reasons. Thus the efficient and effective change management process is needed. Its sub-activities include documenting the change request, evaluation the changes impact, and conduction the change to the test cases.

Periodical review means the analysis of the repository periodically, typically once a month or quarter. People may remove duplicated test cases, merge similar test cases, adjust the architecture of the repository and analyze the using frequency etc.

All reusable test cases should be certified before being put into the repository. The review activity involves: review preparation, review performance and corrective actions identification etc. The related participants involves domain experts, test engineers, software specialists etc. It is the inherently an incremental process because it occurs throughout the lifecycle of the cases.

VI. CASE STUDY

We have applied the ontology representation and the knowledge management model for software test case reuse in a third-party testing center in a planned reuse mode which requires up-front investment and commitment.

Supported by the management level, the testing center has constructed the reusable test case repository with more than 12,000 cases. And the organizational exchange library

was established on the intranet. The tacit knowledge and the explicit knowledge are captured and stored in the testing knowledge warehouse based on the analysis conducted by the knowledge analysts in the center. Additionally, the testing center has developed a comparatively sound knowledge management system such as the daily and weekly discussion and maintains the knowledge map frequently. Meanwhile the team members with good communication skills and strong team spirit play an important role in our applications.

We applied the reusable repository to the forewarning management system for coal mine and the meal ordering system to conduct documentation testing (DT), functional testing (FT) and usability testing (UT). Some testing data are listed as followed (Table I).

TABLE I. TESTING DATA

Testing projects	**Testing DATA**					
	DT test cases	FT test cases	UT test cases	Total test cases	design effort (person day)	Efficiency (number / effort)
Forewarning System	83	1478	60	1621	25	64.84
Meal Ordering	50	644	130	824	12	68.67
ERP System	50	826	80	956	20.5	46.3

In the above table, the forewarning system for the coal mine company and the meal ordering system were tested based on the knowledge management model and the reusable test case repository. The ERP system for the computer retailers was a historical testing project. It was tested before the construction of repository. The data show that the efficiency and productivity of test case design and the work situation of test engineers and managers has improved obviously. Also the test case quality and knowledge transform efficiency has improved according to the subjective judgment from test engineers.

VII. CONCLUSION

By introducing the reuse theory and knowledge management into software testing area, the ontology description and knowledge management model for reusable test cases are discussed based on the definition and design analysis of reusable test cases. With the ontology and knowledge management model, test engineers can retrieve and reuse test case flexibly and the design efficiency of test case has been improved. Regarding to the future work, we may focus on the development of the supporting management system and the related cultural factors analysis.

REFERENCES

[1] J. D. McGregor, D. Muthig, K. Yoshimura, P. Jensen, "Successful Software Product Line Practices," *IEEE Software*, Vol. 27, No.3, pp.16-21, May.2010.

[2] Y. Liu, Lei Z., M. Jin, X. Liu, "Research of Model and Knowledge Management in Software Testing," *Application Research of Computers*, Vol.26, No.1, pp.143-145, Jan.2009.

[3] Ruben Prieto-Diaz, "Status Report: Software Reusability," *IEEE Software*, Vol.10, No.3, pp. 61−66, May.1993.

[4] Linda Argote, Paul Ingram, "Knowledge Transfer: A Basis for Competitive Advantage in Firms," *Organizational Behavior and Human Decision Processes*, Vol. 82, No. 1, pp.150-169, May, 2000.

[5] R. Fruchter, P. Demian, "Knowledge Management for Reuse," *Proceedings of CIB w78 Conference*, Aarhus School of Architecture, Denmark, 2002.

[6] T. Dingsoyr, R. Conradi, "A Survey of Case Studies of the Use of Knowledge Management in Software Engineering," *International Journal of Software Engineering and Knowledge Engineering*, Vol.12, No.4, pp.391-414, May.2002.

[7] X. Liu, G. Yang, L. Cai, J. Zhang, "Ontology Description and Retrieval of Test Case Based on Reuse Behaviour," *Computer Application and Software*, Vol.28, No.10, pp.65-68, Oct.2011.

[8] I. Rus, M. Lindvall, "Knowledge Management in Software Engineering," *IEEE Software*, Vol. 19, No.3, pp.26-38, June, 2002.

[9] G. J. Myers, *The Art of Software Testing*, USA: Wiley, 1979.

[10] J. Chang, K. Li, L. Guo, H. Mei, F. Yang, "Representing and Retrieving Reusable Software Components in JB (Jadebird) System," *Acta Electronica Sinica*, Vol. 28, No. 8, pp.20-23, Aug.2000.

[11] B. Meyer, "The Reusability Challenge," *Computer*, Vol.29, No.2, pp.76-78, Feb.1996.

[12] E. F. Barbosa, E. Y. Nakagawa, J. C. Maldonado, "Towards the Establishment of an Ontology of Software Testing," *18th Int. Conf. on Software Engineering and Knowledge Engineering (SEKE'06),* San Francisco Bay, USA, Jul.2006.

[13] H. J. Happel, S. Seedorf, "Applications of Ontologies in Software Engineering," *In 2nd International Workshop on Semantic Web Enabled Software Engineering (SWESE 2006)*, USA, Nov. 2006.