Visualization Tools and Techniques Supporting Software Testing Activity

Marllos P. Prado[1]*, Auri M. R. Vincenzi[1], and Hugo A. D. do Nascimento[1]

Instituto de Informática, Universidade Federal de Goiás
Goiânia-GO, Brazil

Abstract. This report synthesizes the planning and conduction of a literature review performed to understand how visualization techniques and tools have been applied on software testing. Systematic Mapping was adopted as the method to guide the review process.

1 Research Planning

The main goal of this study was to identify and characterize papers on visualization tools and/or techniques to support software testing activity.

Besides goals definition, the Systematic Mapping planning is composed by inclusion/exclusion criteria and protocol to be adopted.

Protocol consists of research question definition, population, intervention, control, results and application. These parameters were established for the current study as described in Table 1.

1.1 Inclusion and Exclusion Criteria

The following restrictions were adopted to filter papers considered not relevant to the study, on identification step: (i) Studies not written in English language; (ii) Studies prior to year 2000.

The restrictions were established in order to make future study replication easier and in a time window considered feasible and enough representative.

The following inclusion criteria (I.C.) and exclusion criterion (E.C.) were adopted on the selection and extraction phase: (i) I.C.1. Applies visualization tool at software testing; (ii)I.C.2. Applies visualization technique at software testing; (iii) I.C.3. Experimental study of visualization tools or techniques in the context of software testing. (iv) E.C.1. Does not relate to visualization tools or techniques applied at software testing.(v) E.C.2. Visualization tools or techniques have a secondary focus at software testing. (vi) E.C.3. Not a paper, article, chapter, dissertation, thesis or is not accessible.

* The authors would like to thank the Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq - Brasil.

Table 1. Protocol Definitions

Topics	Description
Main Questions	RQ.1. How publications about visualization tools and techniques, applied to software testing, have evolved in the main digital libraries of the involved research areas? RQ.2 - What is the profile of visualization tools and techniques, applied to software testing, concerning testing, visualization and training aspects? RQ.3 - How visualization tools and techniques, which support software testing activity, have been evaluated?
Population	The population considered is composed by researchers and developers, who use/propose/evaluate studies in information visualization applied to software testing activity and publish their work at indexed electronic databases.
Intervention	The observed characteristic was the application of visualization tools and techniques to the software testing activity. Characteristic observation was made from software engineering researchers' point of view.
Control	A total of six relevant papers [4], [6], [5], [7], [1], [2]were previously set by authors to be used as the control for the search string, based on their expertise in the related areas. The search result should be considered adequate in case of returning at least 3 of these papers in each considered databases.
Results	The expected result of the mapping study was the characterization of how visualization tools and techniques are distributed over the testing activity. This should consider but is not restricted to: testing phases more and less supported by visualization; test level more and less considered (unit, integration, system); the distribution of visualization over testing techniques (functional, structural, error-based); how visualization proposals oriented to testing activity are being evaluated.
Application	i) Collaborate to a better understanding on how test activity is currently supported by visualization tools and techniques; ii) provide data to support the development of new works targeting the research opportunities found; iii) contribute with software engineers and visualization researchers in what are the solutions already established related to software testing activity.
Tools and Instrumentation	The digital libraries adopted in the systematic mapping were: "IEEE Xplore" (IEEE), "ACM" and "Science Direct". In order to conduct the process, the StArt tool [3] was used. StArt tool follows the process steps generally adopted in most secondary studies, which comprehends the phases of "Study Identification", "Selection" and "Extraction".
Search String	ACM: (Abstract:visuali*tion and Abstract:(technique or tool) and Abstract:(softwareor program) and Abstract:test*) IEEE: ("Abstract":visuali*tion AND (tool OR technique) AND ((softwareOR program) AND (test*))) Science Direct: tak(visuali*tion AND (tool OR technique) AND ((softwareOR program) AND (test*)))

In order to provide support to answer the research questions, relevant data must be collected during papers reading. These data are obtained observing predefined classification criteria (C.C.). Classification criteria shall represent objective and coherent properties to the intervention and results parameters defined in the protocol. To some criteria established, the value "Other" was created to classify papers which do not meet any of the predefined values but meets a possible existing value. The value "Not Adequate" was created to classify papers which do not meet a criterion because a possible adequate value does not exists or cannot be inferred. The value "Multiple" was applied to indicate that at least two existing values of the referred criterion are adequate for the judged paper. In this study, the classification criteria were adopted as presented in Table 2.

Table 2. Classification Criteria

Criterion	Label	Purpose	Options
C.C.1	Name	Registers the name of the proposed tool or technique when applicable.	-
C.C.2	Source	Registers the study source.	(i) ACM; (ii) IEEE; (iii) Science Direct.
C.C.3	Focus	Registers the study focus.	(i) Tools; (ii) Technique; (iii) Multiple; (iv) Experimental Study.
C.C.4	Evaluation Context	Registers the context of investigation applied and reported on paper's proposal evaluation.	(i) Industry; (ii) Academic; (iii) Multiple; (iv) Poorly/Not Detailed; (v) Not Adequate.
C.C.5	Test Level	Registers the testing granularity attended by the proposal.	(i) Unit; (ii) Integration; (iii) System; (iv) Multiple; (v) Not Adequate.
C.C.6	Analysis Object	Registers the kind of software artifact represented or mapped by visualization.	(i) Code; (ii) Model/Diagram; (iii) Trace; (iv) GUI; (v) Procedure/Method call; (vi) Threads; (vii) Multiple; (viii) Other, (ix) Not Adequate.
C.C.7	Criterion's Technique	Registers the type of testing adequacy technique addressed in the paper.	(i) Functional; (ii) Structural; (iii) Error-Based; (iv) Multiple; (v) No Criterion; (vi) Not Adequate.
C.C.8	Visualization Substrate	Registers the type of substrate used to represent the visualization.	(i) 2D; (ii) 3D; (iii) Multiple; (iv) Not Adequate.
C.C.9	Life Cycle Scope	Registers the software development life-cycle phase at which paper proposal would be more suitable.	(i) Analysis, (ii) Design, (iii) Construction, (iv) Deployment, (v) Multiple, (vi) Not Adequate.
C.C.10	Training Support	Registers if the tool or technique proposed provides any support for training or learning it.	(i) Yes; (ii) No; (iii) Not Adequate.
C.C.11	Target Language/Platform	Registers the name of the programming language or platform addressed by the visualization tool or technique.	-
C.C.12	Relevant Notes	Registers any important information about the proposal, identified during paper reading.	-
C.C.13	Interaction	Registers the kind of interaction used to manipulate the visualization.	(i) Cursor; (ii) Prompt; (iii) Touchscreen; (iv) Sensor(s); (v) Other; (vi) Multiple; (vii) Not Adequate.
C.C.14	Test Phase	Registers the testing activity task better attended by the visualization tool/technique.	(i) Planning; (ii) Implementation; (iii) Analysis; (iv) Multiple; (v) Not Adequate.
C.C.15	Visualization Reference Model	Registers if a visualization tool/technique was built based in a Visualization Reference Model.	(i) No; (ii) Yes; (iii) Not Adequate.

2 Mapping Conduction

This section presents each step of the systematic mapping, based on the defined plan. The study started in July 2013 and was completed in August 2014.

In the Study Identification phase, restrictions set in Section 1.1 were firstly applied. A total of 649 papers were returned from the three digital libraries (232 from ACM, 348 from IEEE and 69 from Science Direct). 76 papers were detected as duplicated and a total of 573 papers were actually considered.

Selection step was carried as a review process based on the papers title and abstracts and eventually - when these were not enough - based on introduction and/or conclusion reading. In this step of the study papers were initially read by one reviewer, in a way to determine which paper should be kept or discarded. Then, reviewer gave his advising. After, another reviewer made his judgment considering the same set of papers. Papers accepted/rejected by both reviewers were automatically accepted/rejected for the next step. In case of divergent opinions, the paper was set as accepted to be read on the next phase. Additionally, in cases when two or more papers from the same author and describing the same

technique were found, only the most recent publication was considered. After completing the analysis of the 573 papers, 485 were considered rejected and 88 were accepted and passed to the following step.

In Extraction phase, each of the 88 identified papers on the previous phase were fully read and classified permanently according to the inclusion and exclusion criteria. From the 88 studies, 3 were considered duplicated, 21 were rejected and 64 were accepted. The classification criteria were also concomitantly evaluated at this step, after each accepted paper was read.

References

1. Dalton, A.R., Wahba, S.K., Dandamudi, S., Hallstrom, J.O.: Visualizing the runtime behavior of embedded network systems: A toolkit for TinyOS. Science of Computer Programming 74(7), 446–469 (2009)
2. Engström, E., Runeson, P.: Test overlay in an emerging software product line – an industrial case study. Information and Software Technology 55(3), 581–594 (Mar 2013)
3. Hernandes, E., Zamboni, A., Fabbri, S., Thommazo, A.A.D.: Using GQM and TAM to evaluate StArt - a tool that supports Systematic Review. CLEI Electronic Journal 15, 3 – 3 (04 2012)
4. Jones, J., Harrold, M., Stasko, J.: Visualization of test information to assist fault localization. In: Proceedings of the 24rd International Conference on Software Engineering, 2002. ICSE 2002. pp. 467–477 (2002)
5. Lawrence, J., Clarke, S., Burnett, M., Rothermel, G.: How well do professional developers test with code coverage visualizations? In: 2005 IEEE Symposium on Visual Languages and Human-Centric Computing. pp. 53–60 (2005)
6. Li, J., Horgan, J.: A scalable tool for efficient protocol validation and testing. Computer Communications 26(7), 766–773 (2003)
7. Mao, C., Lu, Y.: CppTest: A prototype tool for testing c/c++ programs. In: The Second International Conference on Availability, Reliability and Security, 2007. ARES 2007. pp. 1066–1073 (2007)