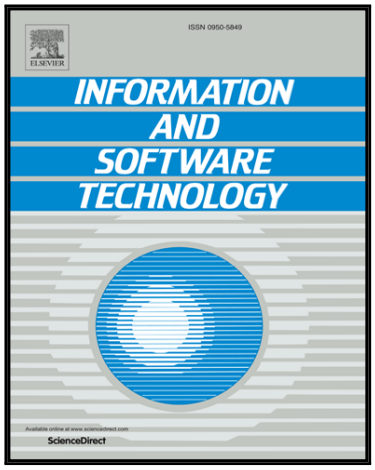

Introduction to Special Issue on Visualization Applied to Software Engineering

Paul Leger, Alexandre Bergel, Juan Pablo Sandoval Alcocer, Leonel Merino

PII: S0950-5849(22)00227-0
DOI: https://doi.org/10.1016/j.infsof.2022.107118
Reference: INFSOF 107118

To appear in: *Information and Software Technology*

Please cite this article as: Paul Leger, Alexandre Bergel, Juan Pablo Sandoval Alcocer, Leonel Merino, Introduction to Special Issue on Visualization Applied to Software Engineering, *Information and Software Technology* (2022), doi: https://doi.org/10.1016/j.infsof.2022.107118

Introduction to Special Issue on Visualization Applied to Software Engineering

Paul Leger[a], Alexandre Bergel[b], Juan Pablo Sandoval Alcocer[c], Leonel Merino[d]

[a]*Escuela de Ingeniería, Universidad Católica del Norte, Coquimbo, Chile*
[b]*RelationalAI, Switzerland*
[c]*DCC, School of Engineering, Pontificia Universidad Católica de Chile, Chile*
[d]*DILAB, School of Design and School of Engineering, Pontificia Universidad Católica de Chile, Chile*

Abstract

Software visualization is a broad research area whose general goal is to enhance and promote the theory, realization, and evaluation of approaches to visually encode and analyze software systems, including software development practices, evolution, structure, and software runtime behavior. Software visualization is inherently interdisciplinary, drawing on theories and techniques from information visualization and computer graphics and applying these in the software engineering domain.

This special issue on software visualization aims to bring together a community of researchers from software engineering, information visualization, computer graphics, human-computer interaction, and data science to discuss theoretical foundations, algorithms, techniques, tools, and applications related to software visualization. The special issue received 17 submissions of which 6 were accepted for publication (i.e., acceptance rate of 35.3%). Amongst the accepted papers, three correspond to extended versions of papers published in the IEEE Working Conference on Software Visualization (VISSOFT) 2021.

Keywords: Software visualization, Software evolution, Software comprehension, User studies

1. Introduction

Software developers can spend great efforts navigating and exploring the multiple components of a system. For instance, they navigate the system in order to understand it, and therefore, create a mental model so they can predict how a change will impact the codebase. Visualizations have the advantage that they can present in a comprehensive manner a great deal of information, which makes them good candidates for software analysis. Consequently, since several decades there is an increasingly active community of researchers and practitioners devoted to software visualization. The community annually gathers at the IEEE Working Conference on Software Visualization (VISSOFT). At this event, the community presents and discusses novel systems and techniques. The conference attendees also analyze the impact of visualizations by discussing the results of empirical evaluations.

For this Special Issue (SI), we invited selected papers from the VISSOFT 2021 conference to submit extended versions of their conference papers. We also put out to engage

other relevant work in the domain. All submissions underwent a thorough review process. First, we screened submissions to ensure their quality and scope. We either desk rejected or assigned to reviewers. Second, we looked for reviewers from the VISSOFT community. In the particular case of VISSOFT extended papers, we invited the same reviewers who reviewed the conference paper. Each submission received at least two reviews from reviewers who are acknowledged experts in the field. After the review process, we rejected some submissions and returned others for major or minor revisions. After the second round of reviews for those papers that required major revisions, we accepted some papers and rejected the remainder. Again, a few submissions were rejected and the rest were accepted.

This special issue provides an opportunity for both researchers and practitioners. Researchers will benefit from increased exposure of their works. Practitioners can find a selection of highly relevant works on software visualization research that can serve as input for adopting an available system or technique.

2. Accepted Papers

For this Special Issue, we received 17 submissions. After the review process, we accepted 6 papers, meaning an acceptance rate of 35.3%. Of these 6 accepted papers, three are extensions of VISSOFT papers and three are new papers submitted directly to the special issue. We next list the accepted papers:

- Papers that extend VISSOFT:
  - **Trace Visualization within the Software City Metaphor: Controlled Experiments on Program Comprehension** (Veronika Dashuber and Michael Philippsen). This paper explored more effective visualizations to support program comprehension based on runtime data of software execution, *e.g.,* program trace executions.
  - **CodeCity: A Comparison of On-Screen and Virtual Reality** (David Moreno-Lumbreras, Roberto Minelli, Andrea Villaverde, Jesus M. Gonzalez-Barahona, and Michele Lanza). This paper aimed to answer the question "Is Virtual Reality (VR) well suited for CodeCity, compared to the traditional on-screen implementation?", where CodeCity is an interactive 3D software visualization that represents software systems as cities, where, for example, buildings represent classes (or files) and districts represent packages (or folders).
  - **Exploring Granular Test Coverage and its Evolution with Matrix Visualizations** (Kaj Dreef, Vijay Krishna Palepu, and James A. Jones). Existing software-development tools rarely offer global and historical information about test case executions. This work approached such challenges with an interactive and matrix-based visual interface that provides a global overview of a software project's test suite, specifically in the context of the methods available in the project's codebase.
- New papers submitted directly to the Special Issue:

– **CADV: A Software Visualization Approach for Code Annotations Distribution** (Phyllipe Lima, Jorge Melegati, Everaldo Gomes, Nathalya Stefhany Pereira, Eduardo Guerra, and Paulo Meirelles). This paper aimed at designing a software visualization approach that graphically displays how code annotations are distributed and organized in a software system and developing a tool, as a reference implementation of the approach, to generate views and interact with users.
– **Collaborative Program Comprehension via Software Visualization in Extended Reality** (Alexander Krause-Glau, Malte Hansen, and Wilhelm Hasselbring). This paper explored the use of a web-based approach to enable (location-independent) collaborative program comprehension using desktop, virtual reality, and mobile augmented reality devices. To achieve this proposal, the authors designed and implemented three multi-user modes in our web-based live trace visualization tool ExplorViz. Users can employ desktop, mobile, and virtual reality devices to collaboratively explore software visualizations.
– **Visualization for the Evolution of Variant-Rich Systems: A Systematic Mapping Study** (Raul Medeiros, Jabier Martinez, Oscar DÃŋaz, and Jean-RÃľmy Falleri). This paper describes a mapping visualization interventions' current state in Variant-Rich Systems (VRS) evolution. The authors tackle evolution in both the functionality and the variability management architecture. Three research questions are posed: "What sort of analysis is being conducted?" (Analysis perspective); "What sort of visualizations are displayed?" (Visualization perspective); "What types of research have been reported and how have they been evaluated?" (Maturity perspective).

3. Conclusion

The articles in this special issue offer various visualizations applied to software engineering. The visualizations described in the articles target a variety of software development activities and artifacts, for instance, program comprehension, test coverage, and software evolution. A few articles describe evaluations that involved a virtual or extended reality device as a medium for displaying and interacting with software visualizations. Finally, an article presents a systematic mapping review of visualizations for the analysis of the evolution of variant-rich systems, which shows that there is a significant amount of articles that focused on this topic.

Acknowledgments

The Guest editors would like to thank all the authors who submitted to the Special Issue on Visualization Applied to Software Engineering. Additionally, we would like to thank all the reviewers for their effort in reviewing the assigned articles to provide valuable comments for improving the quality of the submitted articles. Lastly, we are grateful to Jeffrey Carver, Special Content Editor for the journal Information & Software Technology, for his assistance in managing this special issue.

*Biographies of authors

Paul Leger

Dr. Paul Leger, an associate professor at the Universidad CatÃşlica del Norte (Chile), holds a Ph. D. in Computer Science (University of Chile). Within Computer Science, LegerâĂŹs research interests include topics related to software engineering, programming languages, and different programming approaches like context-oriented programming. Leger works collaboratively with professors at the Tokyo Institute of Technology (Japan), Shibaura Institute of Technology (Japan), and the University of the Andes (Colombia). In addition, he leads the Pragmatics laboratory, which focuses on Computer Science and its business applications. Among his publications, we can count several indexed journal and conference publications. His web page is `http://pleger.cl`.

Alexandre Bergel

Dr. Alexandre Bergel is computer scientist at RelationalAI, Switzerland. Until 2022, he was an Associate Professor and researcher at the University of Chile. Alexandre Bergel and his collaborators carry out research in software engineering. His focus is on designing tools and methodologies to improve the overall performance and internal quality of software systems and databases by employing profiling, visualization, and artificial intelligence techniques.

Alexandre Bergel has authored over 170 articles, published in international and peer-reviewed scientific forums, including the most competitive conferences and journals in the field of software engineering. Alexandre has participated in over 175 program committees of international events. Several of his research prototypes have been turned into products and adopted by major companies in the semiconductor industry, certification of critical software systems, and aerospace industry. Alexandre is a member of the editorial board of Empirical Software Engineering. Alexandre authored 4 books: Agile Visualization with Pharo, Agile Artificial Intelligence in Pharo, Agile Visualization, and co-authored the book Deep Into Pharo. His webpage is `https://bergel.eu`.

Juan Pablo Sandoval Alcocer

Dr. Juan Pablo is an Assistant Professor at the Department of Computer Science, School of Engineering, Pontificia Universidad Católica de Chile. He received a Ph.D. degree in computer science from the University of Chile, Chile, in 2016. He is part of the Software Engineering and Intelligent Systems Laboratory (SEIS Lab). His research interests lie in software engineering, more specifically in the fields of software maintenance, mining software repositories, software performance, software visualization, and search-based software testing. He participated as a reviewer expert in various prestigious conferences and journals in the field including EMSE, IEEE VIS, VISSOFT, JSS, and IST. He is also a member of the Pharo community. His webpage is `https://jpsandoval.github.io/`.

Leonel Merino

Dr. Leonel Merino is Assistant Professor of Engineering Design at the School of Design and the Faculty of Engineering of the Pontifical Catholic University of Chile, since 2021. He is part of the Design Engineering Lab). Between 2018 and 2021, he was a Postdoctoral Research Fellow at the Visualization Research Center (VISUS), Universitaet Stuttgart. Since 2020, Prof. Merino is a member of the steering committee of VISSOFT, the leading international conference in software visualization. From 2014 to 2018, he worked as a Research Assistant at the Software Composition Group (SCG) and obtained a Ph.D. degree in Computer Science at Universitaet Bern. He worked as a Project Engineer at ENTEL between 2009 and 2014. Previously, he received a MasterâĂŹs degree in Computer Science, in the EMOOSE program, from the ÃĽcole des Mines de Nantes and Vrije Universiteit Brussel, in 2008. He obtained a B.Sc. degree in Engineering Sciences and a Professional Degree of Civil Engineer in Computer Science from the University of Chile, in 2006. His research interests lie at the intersection of the use of virtual and augmented reality, software engineering, human-computer interaction, data visualization, and empirical evaluation. His webpage is `https://leonelmerino.github.io/`