

Introduction to the special issue of selected articles from SOFTVIS'2010

Information Visualization
12(2) 105–106
© The Author(s) 2013
Reprints and permissions:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/1473871612474456
ivi.sagepub.com


Alexandru C. Telea¹, Carsten Görg² and Steven Reiss³

In this special issue of the *Information Visualization* journal, we are proud to present a selection of the recent state-of-the-art research that is going on in the field of Software Visualization. This special issue is dedicated to ACM SOFTVIS 2010, the 5th ACM Symposium on Software Visualization. SOFTVIS'2010 took place on 25–26 October 2010 in Salt Lake City, UT, USA. For the first time in its history, SOFTVIS was colocated with VisWeek, which is the premier forum for visualization advances for academia, government, and industry. VisWeek traditionally combines three main conferences: IEEE Visualization, IEEE Information Visualization (InfoVis), and IEEE Visual Analytics Science and Technology (VAST). The addition of SOFTVIS to the 2010 edition of VisWeek has represented a major opportunity for the data and information visualization to be exposed to the use cases, challenges, and solutions developed in the software visualization community, on the one hand, and for the software visualization community to get a closer look at the latest advances in information visualization techniques and tools, on the other hand.

During SOFTVIS'2010, 20 full articles were presented, as well as 9 posters and 3 tool demonstrations. Several notable observations can be made on the presented research. First, a large majority of the presented articles focused on visual and computational *scalability* as a central research challenge, as modern software systems become increasingly large and complex. Second, we saw increased effort put into *user evaluations* and *case studies* involving software visualization techniques and tools, a steadily rising trend already visible for a few years in the software visualization community. Finally, we noted an increasing interest in using and extending recent information visualization techniques such as space-filling methods and image-based rendering methods, which indicate that the gap

between the information visualization and software visualization communities is getting narrower.

For this special issue, we have selected five of the articles presented at SOFTVIS'2010 and invited their authors to submit extended versions. The selected articles reflect well the wide spectrum of techniques used in software visualization, the broad set of use cases for software visualization, the high concern for validation and user studies, and the highly applied nature of this discipline. Murphy-Hill and Black, recipients of an ACM SIGSOFT Distinguished Paper Award, present a novel visualization that combines source code text with code smell metrics and describe in extensive detail the experimental path for refinement and validation of hypotheses related to the effectiveness of their proposed visualization. Frisch and Dachsel, also recipients of an ACM SIGSOFT Distinguished Paper Award, address the visual scalability of displaying large class diagrams using a set of focus and context techniques and present a detailed evaluation setup for their method. Aftandilian et al. present Heapvis, a system that combines focus and context and automatic aggregation techniques to interactively visualize heap traces represented as graphs for program understanding and debugging. Beck and Diehl also use space-filling layouts, but for the different task of comparing several

¹Department of Mathematics and Computer Science, University of Groningen, Groningen, The Netherlands

²Department of Pharmacology, School of Medicine, University of Colorado, Aurora, CO, USA

³Department of Computer Science, Brown University, Providence, RI, USA

Corresponding author:

Alexandru C. Telea, Department of Mathematics and Computer Science, University of Groningen, Nijenborgh 9, Groningen, Groningen 9767 AG, the Netherlands.
Email: a.c.telea@rug.nl

instances of a software architecture, with a practical application for the analysis of software clustering. Steinbrücker and Lewerentz address the task of understanding the evolution of large-scale (millions of lines of code) software systems using a 2.5-dimensional visual layout and an image-based rendering approach that extends the well-known software city metaphor.

Acknowledgement

We are grateful to the authors and reviewers of the articles published in this special issue for their work that helped create a high-quality contribution to the

Information Visualization journal that reflects the state-of-the-art in Software Visualization. We also thank Chaomei Chen, Editor-in-Chief of *Information Visualization*, for his support in offering this opportunity to disseminate Software Visualization research results to the larger audience of Information Visualization practitioners. We believe that this initiative will strengthen the trend set by SOFTVIS'2010 to bring the two communities of Information Visualization and Software Visualization closer to each other and foster creative cross-fertilizations between these two disciplines.