

SUITE 2009: First International Workshop on Search-Driven Development – Users, Infrastructure, Tools and Evaluation

Sushil Bajracharya
University of California, Irvine
sbajrach@ics.uci.edu

Adrian Kuhn
University of Bern
akuhn@iam.unibe.ch

Yunwen Ye
Software Research Associates, Inc.
ye@sra.co.jp

Abstract

SUITE is a new workshop series that specifically focuses on exploring the notion of search as a fundamental activity during software development. The goal of the workshop is to bring researchers and practitioners with special interest on search technology for software developers together. Participants will have broad range of expertise in topics ranging from building software tools and infrastructure, Information Retrieval, user studies and Human-computer interaction, benchmarking and evaluation.

The first edition of SUITE is held in conjunction with the 31st International Conference in Software Engineering (May 16-24, 2009. Vancouver, Canada).

1 Motivation

The workshop is motivated by the observation that software developers spend most of their times in searching pertinent information they need to solve their task at hand [8]. Past research has shown that code search is the most frequent activity software developers engage in [12]. They spend most of their time in navigation and search tools in their IDE [11]. More recently there has been some significant efforts both from academia and the industry in building specialized search engines for developers [2, 3, 1, 5, 4, 9, 6, 10, 13, 7]. Most of these leverage the huge amount of source code available in open source repositories. However, these tools are still exploring the tip of the iceberg. We know that source code is not the only artifact that developers need to search and that traditional search engine interfaces have limitations to serve as ideal tools for searching pertinent information for developers. Furthermore, along with the tools we still need a solid understanding of how developers are really using these systems.

2 Objectives

As software development is a process of both information creation and information gathering, software developers are constantly searching for the right information and person to solve their problems at hand. This workshop will focus specifically on exploring the notion of search as a fundamental activity during software development. The goal of the workshop is to bring researchers and practitioners with special interest on search technology for software developers together. Participants will have broad range of expertise in topics ranging from building software tools and infrastructure, information retrieval, user studies and HCI, benchmarking and evaluation.

The workshop will facilitate interested researchers to share their ideas and experience in understanding the search need and behavior of developers, building tools that addresses these various needs, and scientific ways to evaluate these tools.

3 Topics

The workshop addresses the problem of search as it occurs during software development. Search is related to software mining, but differs in its problems and challenges. For example, two of the important topics the workshop focuses on are: a) search-engines for public software repositories on the internet, and b) specialized search-engines for IDEs.

Areas of interests include, but are not limited to:

- Application of natural language processing on source code and related artifacts.
- Approaches, applications, and tools for software search.
- Case studies on setting up and running large software search-engines.
- Empirical studies of search and navigation in IDEs.

- How can industry and researchers collaborate?
- Information retrieval and machine learning techniques to search source code.
- Integration of specialized search engines into IDEs.
- Methods of integrating indexed data from various sources and histories.
- Query languages to search software and repositories.
- Search techniques to assist developers in finding suitable components and code fragments for reuse.
- Techniques for indexing large software repositories (and their history) efficiently.
- Static analysis and parsing of internet-scale code repositories.
- Crawling source code in the internet and code repositories.
- The use of visualizations to support software search.
- Validation of tools and software searching benchmarks (datasets).
- Ranking strategies and heuristics for code search.
- Slicing and generative techniques for code extraction and synthesis.

4 Submissions

This year's submissions to the workshop touches various themes as seen across the topics presented above. They range from tools and infrastructure to user studies and experiments. All, in one way or another, motivated by the goal of enhancing the search experience of developers during software development. The list of accepted papers is available from the workshop's website <http://smallwiki.unibe.ch/suite2009/>. Final versions of the papers appear in the ICSE proceedings.

5 Organizers

Sushil Bajracharya¹ is a PhD candidate in the Department of Informatics, Donald Bren School of Information and Computer Sciences, University of California Irvine, USA.

Adrian Kuhn² is a PhD candidate at the Software Composition Group, University of Bern, Switzerland.

Yunwen Ye is a manager in the Technology Strategy Division in Software Research Associates, Inc. Japan.

¹<http://www.ics.uci.edu/~sbajrach>

²<http://smallwiki.unibe.ch/adriankuhn>

References

- [1] Google code search home page. <http://www.google.com/codesearch>.
- [2] Koders web site. <http://www.koders.com>.
- [3] Krugle web site. <http://www.krugle.com>.
- [4] S. Bajracharya, T. Ngo, E. Linstead, Y. Dou, P. Rigor, P. Baldi, and C. Lopes. Sourcerer: a search engine for open source code supporting structure-based search. In *OOP-SLA '06: Companion to the 21st ACM SIGPLAN symposium on Object-oriented programming systems, languages, and applications*, pages 681–682, New York, NY, USA, 2006. ACM.
- [5] R. Hoffmann, J. Fogarty, and D. S. Weld. Assieme: finding and leveraging implicit references in a web search interface for programmers. In *UIST '07: Proceedings of the 20th annual ACM symposium on User interface software and technology*, pages 13–22, New York, NY, USA, 2007. ACM.
- [6] R. Holmes and G. C. Murphy. Using structural context to recommend source code examples. In *ICSE '05: Proceedings of the 27th international conference on Software engineering*, pages 117–125, New York, NY, USA, 2005. ACM.
- [7] O. Hummel, W. Janjic, and C. Atkinson. Code conjurer: Pulling reusable software out of thin air. *IEEE Softw.*, 25(5):45–52, 2008.
- [8] A. J. Ko, R. DeLine, and G. Venolia. Information needs in collocated software development teams. In *ICSE '07: Proceedings of the 29th international conference on Software Engineering*, pages 344–353, Washington, DC, USA, 2007. IEEE Computer Society.
- [9] O. A. L. Lemos, S. K. Bajracharya, J. Ossher, R. S. Morla, P. C. Masiero, P. Baldi, and C. V. Lopes. Codegenie: using test-cases to search and reuse source code. In *ASE '07: Proceedings of the twenty-second IEEE/ACM international conference on Automated software engineering*, pages 525–526, New York, NY, USA, 2007. ACM.
- [10] D. Mandelin, L. Xu, R. Bodík, and D. Kimelman. Jungloid mining: helping to navigate the api jungle. In *PLDI '05: Proceedings of the 2005 ACM SIGPLAN conference on Programming language design and implementation*, pages 48–61, New York, NY, USA, 2005. ACM.
- [11] G. C. Murphy, M. Kersten, and L. Findlater. How are java software developers using the eclipse ide? *IEEE Softw.*, 23(4):76–83, 2006.
- [12] J. Singer, T. Lethbridge, N. Vinson, and N. Anquetil. An examination of software engineering work practices. In *CASCON '97: Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research*, page 21. IBM Press, 1997.
- [13] S. Thummalapenta and T. Xie. Parseweb: a programmer assistant for reusing open source code on the web. In *ASE '07: Proceedings of the twenty-second IEEE/ACM international conference on Automated software engineering*, pages 204–213, New York, NY, USA, 2007. ACM.