3rd Workshop on Assessment of Contemporary Modularization Techniques (ACoM 2009)

Phil Greenwood
Lancaster University, UK
greenwop@comp.lancs.ac.uk

Alessandro Garcia
PUC-Rio, Brazil
afgarcia@inf.puc-rio.br

Yuanfang Cai
Drexel University, USA
yfcai@cs.drexel.edu

Claudio Sant'Anna
Federal University of Bahia, Brazil
santanna@dcc.ufba.br

Kevin Sullivan
University of Virginia, USA
sullivan@cs.virginia.edu

Thomas Cottenier
Hengsoft LLC, USA
thomas.cottenier@hengsoft.net

James Noble
Victoria University of Wellington, New Zealand
kjx@ecs.vuw.ac.nz

Abstract

Contemporary modularization techniques, such as Aspect-Oriented Programming and Feature-Oriented Programming, allow designers to improve changeability and adaptability. However, the impact of advanced modularization techniques on productivity and quality remains unclear. This workshop emphasizes the need to improve the assessment of modern modularization techniques, particularly in the emerging domains of collaborative software engineering, ultra-large systems, cyber-physical systems, and cloud computing.

The purpose of the 3rd ACoM workshop is (i) stimulate innovative ideas of modularity assessment methods; (ii) solicit experience reports from practitioners to understand the impact of contemporary modularization techniques; (iii) discuss the impact of assessment techniques; (iv) understand the applicability of assessment techniques; and (v) foster collaboration between practitioners and researchers.

Categories and Subject Descriptors D.2.8 [*Software Engineering*]: Metrics; D.1.0 [*Programming Techniques*]: General

General Terms Design, Experimentation, Measurement

Keywords adoption, assessment, metrics, modularity

OOPSLA 2009, October 25–29, 2009, Orlando, Florida, USA.
ACM 978-1-60558-768-4/09/10.

1. Main Theme

The continual increase in complexity and scale of software systems demand new development techniques to manage such characteristics. The application of contemporary modularization techniques, such as Aspect-Oriented Programming (AOP) and Feature-Oriented Programming (FOP), can help tame such complexity through improved modularization. However, it remains unclear the impact that these advanced modularization techniques have on software productivity and quality. It is even more challenging to determine their impact on emerging domains such as collaborative software engineering [1], ultra-large systems [2], cyber-physical systems [3] and cloud computing [4]. This workshop promotes the need for pushing the boundaries on the assessment of modern modularization techniques, particularly in the emerging domains of software development.

Even though there is evidence that conventional modularization techniques are overly constrained, some contemporary techniques have been criticized for promoting non-modular solutions in certain cases (e.g., [5–8]). One approach may be optimal in some circumstances, but not in others [9]. To maximize the benefits of contemporary modularization mechanisms, we seek assessment techniques that can lead to principled guidelines to facilitate the comparison, reconciliation, and synthesis of these techniques in practice.

It is not clear whether traditional assessment methods are sufficient to evaluate new modularization techniques in emerging domains. Traditional metrics do not adequately measure how well concerns are separated; do not measure how well modules can evolve independently to generate op-

tion values [10]; and do not measure how well tasks can be separated to improve task parallelism and collaboration through modularized architecture. We seek for novel assessment techniques to evaluate modern modularization techniques in these areas.

Contemporary modularization techniques call for well-designed empirical studies. It is necessary to perform such studies to determine how such techniques need to be improved and developed in practice. We aim at understanding their current impact in real enterprise industrial settings, fostering their well-informed adoption through effective assessment methods, and boosting the innovation of both new modularization and assessment techniques.

We also solicit innovative ideas regarding assessment-based improvement of modularization techniques to advance their suitability for emerging systems (e.g., ultra-large and cyber-physical systems). Assessing modularization techniques will reveal their benefits and drawbacks, and may reveal the need for advances in programming languages, architecture styles, or the novel combination of existing techniques. In particular, empirical studies along with supporting assessment techniques provide the basic means to improve our understanding of the benefits and drawbacks of new software decomposition techniques.

Designing effective experiments is notoriously difficult, even when using traditional applications [11] due to the high number of variables and the human factors that influence software development. We anticipate that performing studies involving complex systems, such as ultra-large or cyber-physical systems, will not only magnify existing difficulties but also introduce new ones. A dialogue between researchers and practitioners needs to be initiated to identify and understand such issues.

2. Relevance and Timeliness

Determining the effectiveness of contemporary modularization techniques is relevant to a wide range of emerging software development domains and communities. Collaborative software engineering, ultra-large systems, cyber-physical systems and cloud computing are just some examples of such domains. They all have unique wide-reaching software properties where contemporary modularization techniques may prove useful to tame complexity encountered in such domains. Alternatively, these domains raise interesting challenges to the developers of contemporary modularization techniques. The benefits and drawbacks of applying contemporary modularization techniques need to be determined to guide research and practice in these areas and develop techniques which can fulfill the needs of these emerging domains.

The ACoM workshop series is an initiative to bring together researchers and practitioners in order to discuss the multi-faceted issues that emerge in the assessment and/or comparison and application of new modularization techniques. The theme of the 3rd ACoM workshop[1] extends the first two editions and intends to stimulate discussions on important open questions:

1. What role do contemporary modularization techniques play in emerging system domains and what assessment strategies are needed?
2. What attributes of these complex systems need to be measured and assessed?
3. Are current conventional metrics sufficient to assess software quality in such domains?
4. What new assessment mechanisms are necessary to assess contemporary heterogeneous modularization techniques in emerging system domains?
5. What new modularization techniques or improvements to existing ones are suggested by assessment results?

References

[1] Adele Goldberg. Collaborative software engineering. *Journal of Object Technology*, 1(1):1–19, 2002.

[2] L. Northrop, et al. Ultra-Large-Scale Systems - The Software Challenge of the Future. Technical report, Software Engineering Institute, Carnegie Mellon, June 2006.

[3] Edward A. Lee. Cyber physical systems: Design challenges. In *Int. Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing (ISORC)*, May 2008.

[4] Brian Hayes. Cloud computing. *Comm. ACM*, 51(7):9–11, 2008.

[5] S.Apel, T. Leich, and G. Saake. Aspectual mixin layers: aspects and features in concert. In *ICSE '06: Proceedings of the 28th Int. Conf. on Software engineering*, pages 122–131, Shanghai, China, 2006.

[6] F. Filho, et al. Exceptions and aspects: The devil is in the details. In *Int. Conf. on Foundations of Software Engineering*, volume FSE-14, 2006.

[7] A. Garcia, et al. Modularizing design patterns with aspects: A quantitive study. In *4th Int. Conf. on Aspect-Oriented Software Development*, pages 3–14, Chicago, USA, 2005.

[8] P. Greenwood, et al. On the impact of aspectual decompositions on design stability: An empirical study. In E. Ernst, editor, *ECOOP*, volume 4609 of *Lecture Notes in Computer Science*, pages 176–200. Springer, 2007.

[9] K. Sullivan, et al. Non-modularity in aspect-oriented languages: integration as a crosscutting concern for aspectj. In *1st Int. Conf. on Aspect-Oriented Software Development*, pages 19–26, Enschede, The Netherlands, 2002.

[10] C. V. Lopes and S. K. Bajracharya. An analysis of modularity in aspect oriented design. In *4th Int. Conf. on Aspect-Oriented Software Development*, pages 15–26, Chicago, Illinois, 2005.

[11] Walter F. Tichy. Should computer scientists experiment more? *Computer*, 31(5):32–40, 1998.

[1] Workshop website: http://www.comp.lancs.ac.uk/computing/ACoM.09/