# Assessment of Contemporary Modularization Techniques – ACoM'07: Workshop Report

Alessandro Garcia[1], Phil Greenwood[1], George Heineman[2], Robert Walker[3], Yuanfang Cai[4], Hong Yul Yang[5], Elisa Baniassad[6], Cristina Videira Lopes[7], Christa Schwanninger[8], Jianjun Zhao[9]

[1]Lancaster University, UK; [2]Worcester Polytechnic Institute, USA; [3]University of Calgary, Canada; [4]Drexel University, USA; [5]University of Auckland, New Zealand; [6]The Chinese University of Hong Kong, China; [7] University of California at Irvine, USA; [8]Siemens AG, Germany; [9]Shanghai Jiao Tong University, China

## Abstract

The effective assessment of emerging modularization technologies plays a pivotal role on: (i) a better understanding of their real benefits and drawbacks when compared to conventional development techniques, and (ii) their effective transfer to mainstream software development. This report is intended to summarize the results of the 1st International Workshop on Assessment of Contemporary Modularization Techniques (ACoM'07) held in Minneapolis, USA, May 22, 2007, as part of the 29th International Conference on Software Engineering (ICSE'07). The main purpose of this workshop was to share and pool the collective experience of people interested in and actively working on assessment of innovative modularization techniques. The workshop consisted of an opening presentation, several paper presentations organized into three technical sessions, and four discussion groups. During the workshop presentations and discussions, the authors and participants directly and indirectly reviewed ongoing and previous work and debated a number of important issues on contemporary modularity assessment. The ACoM'07 website, including the electronic version of this report, can be found at <www.comp.lancs.ac.uk/computing/ACoM.07/>. We begin by presenting an overview of our goals and the workshop structure, and then focus on the workshop technical program and results.

## 1. Introduction

A plethora of new modularization techniques are emerging to cope with the challenges of contemporary software engineering, thereby addressing shortcomings of established development paradigms, such as object-oriented and component-based software engineering. For example, modern programming languages and modeling techniques (e.g. UML) are being enhanced with additional modularity mechanisms and abstractions, such as aspects, features, and the like. With new software engineering paradigms fast gaining wide attention in both research and industry environments, there is a pressing need to define proper assessment mechanisms, techniques and methods.

Effective evaluation of contemporary modularization techniques deeply concerns software engineering researchers and practitioners and can be characterized according to two different perspectives: an inter-paradigm perspective, and an intra-paradigm perspective. In the first perspective, 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, especially when compared to techniques from other development paradigms. According to an intra-paradigm perspective, systematic assessment of artefacts decomposed according to a specific paradigm is imperative through all the software lifecycle phases, from requirements engineering to implementation and maintenance. For example, estimation models and measures of software internal properties are useful assessment techniques that can assist software developers, managers, and customers to characterize and improve the software modularity attributes.

Since new modularization techniques, such as Aspect-Oriented Software Development (AOSD) and Feature-Oriented Programming (FOP), are starting to reach some level of maturity, assessment is becoming a central issue to software engineering researchers and practitioners. The relevance of the topic becomes even more evident when we look at the number of qualitative and quantitative case studies [5-11, 15, 16], software metrics [8, 12-14], and assessment frameworks [4, 7] involving new modularity techniques. They are consistently appearing in the software engineering literature.

In this context, the 1st Workshop on Assessment of Contemporary Modularization Techniques (ACoM) is the first initiative to put together researchers and practitioners with different backgrounds in order to discuss the multi-faceted issues that emerge in the assessment of new modularization technologies. Because there is a limited set of resources, methods, and knowledge on how to support the evaluation of such new modularity mechanisms, a number of complementary questions related to both intra-paradigm and inter-paradigm assessment issues still need to be addressed, such as:

1. what are the proper assessment mechanisms and methods to compare new modularity techniques and conventional ones (e.g., OO);
2. how to empirically assess the usability and usefulness of new modularization techniques within industrial settings and cost bounds;
3. how to support the quality assessment of artefacts decomposed according to new modularity mechanisms through the software lifecycle;
4. to what extent should software engineers using such contemporary modularity techniques rely on traditional metrics and quality indicators; and
5. how to validate new assessment mechanisms.

ACoM 2007 was built on the success of the preceding ASAT workshop (held at AOSD.07, March 2007). However, ACoM had a broader scope than ASAT, which concentrated exclusively on the assessment of aspect-oriented technologies. The goal of ACoM was to discuss modularity evaluation from a more general perspective, tackling also other emerging modularization

techniques. The workshop elicited two categories of submissions on the assessment of modularity techniques: (i) traditional position papers (up to 6 pages) related to workshop topics, and (ii) very short position statements (1–2 pages), where the authors described their "innovative thoughts", lessons learned, or points of view with respect to one or more workshop topics. Long position papers described work that is not yet advanced enough for a full conference paper. They were expected to have a more solid idea, even though it does not require strong validation ingredients. Short position statements were mainly reviewed for topicality, i.e., checked if they fit into the workshop topics, including some feedback from reviewers. Moreover we especially encouraged authors to present their experience and/or novel ideas on how to assess new modularization techniques (shorter paper format).

## 2. Workshop Proceedings and Program Committee

The papers were collected in the ACoM workshop proceedings, which was made available to the workshop participants during the conference. The Program Committee was composed of the following members:

Mehmet Aksit, University of Twente, The Netherlands
Elisa Baniassad, Chinese Univ. of Hong Kong, China - **Chair**
James Bieman, Colorado State University, USA
Paulo Borba, UFPE, Brazil
Lionel Briand, Carleton University, Canada
Christina Chavez, UFBA, Brazil
Yvonne Coady, University of Victoria, Canada
Robert France, Colorado State University, USA
Alessandro Garcia, Lancaster University (UK) - **Chair**
Holger Giese,University of Paderborn, Germany
Ian Gorton, Pacific Northwest National Laboratory, Australia
Rachel Harrison, Stratton Edge Consulting, UK
Arno Jacobsen, University of Toronto, Canada
Jörg Kienzle, McGill University, Canada
Cristina V. Lopes, University of California, Irvine - **Chair**
Mira Mezini, T.U. Darmstadt, Germany
Gail Murphy, University of British Columbia, Canada
Awais Rashid, Lancaster University, UK
Christa Schwanninger, Siemens AG, Germany - **Chair**
Peri Tarr, IBM Watson Research Center, USA
Tom Tourwe, CWI, The Netherlands
Robert Walker, University of Calgary, Canada
Jianjun Zhao, Shanghai Jiao Tong University, China - **Chair**

## 3. Workshop Organization and Structure

The organization was under the responsibility of the organizing committee members: Alessandro Garcia (Lancaster University), Elisa Baniassad (Chinese Univ. of Hong Kong), Cristina V. Lopes (University of California), Christa Schwanninger (Siemens AG), Jianjun Zhao (Shanghai Jiao Tong University), Eduardo Figueiredo, (Lancaster University), Cláudio Sant'Anna (PUC-Rio, Brazil) and with the assistance of the PC. We received submissions from countries in almost all the continents: Canada, USA, UK, Germany, The Netherlands, Spain, Brazil, New Zealand, and India. The submitted papers were reviewed and revised before the workshop. We selected 11 papers for presentation in the workshop. Each paper was reviewed by at least two members of the PC; the final selection was made by the workshop organizers based on the evaluation forms. The presented papers were chosen because they offered different or novel perspectives on the workshop topics and because they had a high potential for raising issues that would stimulate the discussions. An additional description of the selection process, as well as all the participants' position papers, can be found at the ACoM.07 Web site.

The meeting was composed of 28 participants, and provided a forum for the exchange of ideas on empirical studies and diverse approaches to the assessment of CoM techniques. In preparation for the workshop, participants were requested to read all other submissions and asked to prepare a clear position statement and questions that were likely to stimulate discussion. Moreover, most of the presenters tried to identify open questions that they will be facing in the near future. The presentations at ACoM'07 triggered highly interesting discussions between workshop participants – whose we sincerely want to thank for their active participation and the level of their contributions to the debate.

The workshop was structured into the following parts:

- An opening presentation by Alessandro Garcia was the starting point and introduction for the morning and the afternoon sessions. He reported on the meeting's topics and goals and the workshop organization process.
- Three technical sessions provided the framework to structure the paper presentations. The first session addressed emerging metrics for module dependencies. The second session was dedicated to modularity assessment for evolvability and reusability. The third session focused on topics related to experimental design challenges on assessing CoM techniques. At the end of each presentation, time was reserved for discussion. We appointed a chair for each session to coordinate the discussions. The most important topics presented or discussed in each session are briefly summarized in Section 5.
- Four discussion groups that debated themes selected by the own workshop participants. Each group focused in addressed one of the following categories of questions: (1) "How to measure heterogeneous types of module dependencies?", (2) "How to measure modularity for evolvability and reusability?", (3) "How to design empirical studies for emerging modularization techniques?", and (4) "What is the impact of composition techniques on modularity? What are the open problems (and potential solutions) for quantitative comparisons of AO vs. OO software?" The outcomes from the discussions are briefly summarized in Section 6.

## 4. Workshop Presentations

As we explained above, 11 papers were accepted for presentation. Unfortunately, one of the speakers was not able to travel to Minneapolis due to personal reasons, so only 10 papers were presented. Each speaker had 10 minutes (for short position statements) or 15 minutes (for long position papers) per presentation, followed by 5 minutes for discussion. Paper discussants were assigned before the workshop, and they brought questions to the speakers in addition to those of the other participants. The papers and their authors were as follows; summaries of these presentations are presented in the following section of this workshop report.

- *Towards Assessing Modularity*; Hayden Melton, Ewan Tempero (U. Auckland, New Zealand)

  Discussant: Jianjun Zhao (Shanghai Jiao Tong U., China)

- *Semantic Dependencies and Modularity of Aspect-Oriented Software*; Alberto Costa Neto, Márcio Ribeiro, Marcos Dósea, Rodrigo Bonifácio, Paulo Borba, Sérgio Soares (Federal U. Pernambuco & Pernambuco State U., Brazil)

  Discussant: Hayden Melton (U. Auckland, New Zealand)
- *Indirect Coupling as a Criteria for Modularity;* Hong Yul Yang, Ewan Tempero (U. Auckland, New Zealand)

  Discussant: Chris Lüer (Ball State U., USA)
- *Identifying, Assigning, and Quantifying Crosscutting* Concerns; Marc Eaddy, Alfred Aho, Gail C. Murphy (Columbia U., USA & U. British Columbia, Canada)

  Discussant: Robert J. Walker (U. Calgary, Canada)
- *Modularization with Externalization of Control Flow*; Urjaswala Vora (CDAC Mumbai and IIT, India)

  Discussant: Scott Hendrickson (U. California at Irvine,USA)
- *An Evolution Model for Software Modularity Assessment*; Yuanfang Cai, Sunny Huynh (Drexel U., USA)

  Discussant: Alberto Neto (Federal U. Pernambuco,Brazil)
- *Assessing Module Reusability*; Chris Lüer (Ball State U, USA)

  Discussant: Marc Eaddy (Columbia U., USA)
- *On the Necessity of Empirical Studies in the Assessment of Modularization Mechanisms for Crosscutting Concerns*; Sven Apel, Christian Kaestner, Salvador Trujillo (U. Magdeburg, Germany & U. the Basque Country, Spain)

  Discussant: Phil Greenwood (Lancaster U., UK)
- *Using Design Structure Matrices to Assess Modularity in Aspect-Oriented Product Lines*; Pedro Matos, Rafael Duarte, Ivan Cardim, Paulo Borba (Federal U. Pernambuco, Brazil)

  Discussant: Yuanfang Cai (Drexel U., USA)
- *Performing and Reviewing Assessments of Contemporary Modularization Approaches: What Constitutes Reasonable Expectations?* Robert J. Walker (U. Calgary, Canada)

  Discussant: Thomas LaToza (Carnegie Mellon U., USA)
- *Using Program Families for Maintenance Experiments*; Scott D. Fleming, R. E. Kurt Stirewalt, Laura K. Dillon (Michigan State U., USA)

  Discussant: Salvador Trujillo (U. the Basque Country. Spain)

## 5. The Sessions

There were three sessions of presentations and discussions. Each of the sessions was organized according to common themes in the position papers. The session summaries as produced by the respective session chairs are presented below.

**Session 1: Innovative Metrics for Module Dependencies**
***Chair: Robert Walker (University of Calgary, Canada)***

This session included four presentations. The first presentation given by Hayden Melton, was entitled "Towards Assessing Modularity". The talk started by re-examining the definition of modularity, and discussing the inherent shortcomings of existing definitions. Hayden argued that the first step in assessing a modularization technique should be to determine if it is even correctly classified as such. Then, he advocated that an explicit definition of modularity is required for such a classification, rather than the implicit definitions common in the field, based on external quality attributes (i.e., evolvability, comprehensibility, parallel development). He presented a definition for modularity and argued that such a definition is practical to demonstrate an approach for assessing the effect two design principles have on it: (1) to avoid dependency cycles among source files, and (2) to favour a "flatter" rather than "taller" source file dependency graph. Participants questioned the need for an explicit definition of modularity, and the idea that a flatter dependency graph is better than a taller one.

Alberto Costa Neto gave the second talk with the title "Semantic Dependencies and Modularity of Aspect-Oriented Software". He presented an analysis of the different OO and aspect-oriented (AO) versions of a web-based information system based on design structure matrices (DSMs). First, Alberto presented the notion of syntactic and semantic coupling and showed a different DSM including both of them. Syntactic coupling is often only considered for coupling measurement, but in his group's opinion it is not enough for a complete study of the coupling when considering AO systems. Their DSM-based analysis showed that without considering the semantic coupling, it is impossible to achieve parallel development, an essential property according to Parnas, to categorize software as modular. At the end, Alberto presented the analysis of a version with design rules that considered semantic coupling, which confirmed that it provides a better modularization than previous versions. Participants questioned whether the DSM analysis was correct, and whether considering additional forms of coupling would significantly alter the analysis yet further.

Hong Yul Yang gave the talk "Indirect Coupling as a Criteria for Modularity". The main goal of his presentation was to argue the need for an additional criterion for modularity, coupling due to transitive data- or control-flow. He first explained that his group has explored the potential impact of indirect coupling on quality and studied its effect on existing applications through initial empirical studies. They strongly believe in the need to devise modularity techniques to focus explicitly on the control of indirect coupling, and to do this, it will be necessary to first determine the exact effect of indirect dependence on external quality attributes. Participants questioned whether the authors' notion of indirect coupling would aid modularity.

The last presentation in this session, titled "Identifying, Assigning, and Quantifying Crosscutting Concerns", was given by Marc Eaddy. According to Marc, before we can modularize concerns, we must be able to locate and quantify them. He presented a systematic method for manually identifying concerns and their associated code fragments. He and his colleagues believe that the method is more accurate and easier to apply consistently than previous approaches. He also introduced a suite of metrics for quantifying the degree to which a concern is scattered across components and separated within a component. He discussed how their metrics are more descriptive than previous concern metrics and traditional object-oriented metrics. Participants questioned whether these were really metrics as opposed to measures, and whether they had been validated.

**Session 2: Modularity Assessment for Evolvability and Reusability**

***Chair: George Heineman (Worcester Polytechnic Institute, USA)***

Two papers (from the three assigned to this session) were presented. Urjaswala Vora, the author for the third paper of our session "Modularization with Externalization of Control Flow", was unable to present her paper. The session chair summarized this last paper. So, first, Yuanfang Cai of Drexel University presented her paper "An Evolution Model for Software Modularity Assessment". This paper continued the promising work on using design structure matrices (DSMs) and their more powerful counterpart, Augmented Constraint Networks (ACNs) to directly address changeability analysis. In the proposed framework, software designs and potential changes are uniformly modeled using ACNs, independent of the language paradigm and modularization mechanisms. The idea is to specify design modular structure using derived design structure matrices, and define evolution vectors to quantitatively reflect informal design principles. The authors analyzed the OO and aspect-oriented versions of the Observer pattern in terms of their ability to accommodate envisioned changes. The results have shown that the proposed model quantitatively and formally verified previously informal analysis results. The evolution vector has the potential to be extended with additional dimensions, such as net option values, having the potential to bridge the gap between software design modeling and rigorous economic analysis.

Next, Chris Luer presented his paper "Assessing Module Reusability". This paper sought to define benchmarks for both the assessment of individual reusability as well as the reusability of modularization techniques in general. According to Luer, the assessment of reusability of a modularization technique needs to also encompass an identification of the future reuse potential of actual modules using the given modularization mechanisms. In this process, some examples of identification guidelines are: (i) determine likely modules that can be implemented using the modularization technique, (ii) determine likely situations in which these modules can be reused, (iii) determine the functionality of the modules based on these situations, (iv) determine the applicability of the modules based on these situations, and (v) multiply functionality and applicability to calculate reusability.

**Session 3: Challenges on Experimental Design**
***Chair: Phil Greenwood (Lancaster University – UK)***

The papers presented in this session focused on highlighting some of the issues that have been and may be encountered when assessing contemporary modularization techniques. The issues raised were both quite general (in terms of being applicable to a variety of modularization techniques) and specific to one particular approach. Even when specific issues were raised these were still useful in that they provoked interesting discussions between the workshop's participants.

The first paper presented in the session was titled "On the Necessity of Empirical Studies in the Assessment of Modularization Mechanisms for Crosscutting Concerns". This paper outlined the differences between AspectJ-like Languages and Collaboration Languages. The paper hypothesizes that AspectJ-like languages are inappropriate for creating collaborations and Collaboration Languages should be used instead. To test this hypothesis, the authors put forward a series of suggestions describing how these two approaches should be compared using cognitive distance and ethnographic studies. The discussion for this paper centered on the classification/assessment of languages which are neither a pure AspectJ-like language nor a pure Collaboration-language (such as CaesarJ). The classification issue will be circumvented by ignoring such ambiguous languages. It was suggested that such languages could be used as a "comprehension-bridge" for participants in the study who are experts in one approach but not the other. Finally, it was highlighted that an interesting extension to this study could involve examining the benefits of features borrowed from the AspectJ-like languages and the benefits of features borrowed from Collaboration languages in these hybrid approaches.

The second paper of the session describes the experiences encountered when conducting studies to assess the modularity of software product lines. The modularity of a mobile game product line was assessed using design structure matrices (DSM) via the dependencies of the software. The DSMs were constructed via the package name dependencies of the various software modules in the mobile game product line. The DSMs were then used to identify cyclic dependencies and the analysis of the DSMs reinforces that more elaborate interfaces between classes and aspects are needed. One question asked regarding this presentation involved the semantic and syntactic gap that could manifest itself in the software modules and how this could be resolved when using DSMs.

The next paper presented in this session did not explicitly target one particular modularization approach but instead raises a series of questions and hypotheses regarding the road-ahead for assessing contemporary modularization techniques. These issues and hypotheses relate to the maturity (or immaturity) of the contemporary modularization techniques and how these can then be assessed (i.e., via what method) with reasonable expectations. The discussion for this paper centered on the relevance of comparing new (and immature) modularization techniques with more established techniques. For example, there might be some pitfalls in comparing Java and AspectJ due to the perceived differences in the level of maturity. There may be an unfair bias towards Java due to its increased maturity. However, if AspectJ is comparable with Java then this may indicate that AspectJ is superior due to its lower levels of maturity. However, it is not clear what conclusion can be drawn if AspectJ fairs worse. Questions were also asked regarding what studies can be performed between preliminary experiments and subsequent ones.

The final presentation of the session outlined an approach to conduct formal experiments for assessing a technique's impact when performing perfective maintenance. This experiment concentrates on the use of program families in order to reduce the number of issues involved in selecting an appropriate base-program (such as ensuring the base-program is realistic, does not favour one particular approach, ensuring the programs are comparable, and ensuring the experiment is replicable). Perfective maintenance changes are then performed on the program families using different treatments to compare and contrast each approach. The majority of the issues raised during the discussion of this paper focused on ensuring the fairness when performing the experiments and applying the treatment. It was suggested that it would be very easy for bias to be introduced especially when applying the different treatments. A suggested solution to this is to extensively collaborate with different external partners to ensure

that all experiments conducted are fair. Also, peer-review should also be used to ensure that all published results are fair and consistent.

## 6. Discussion Groups

In addition to the technical sessions, the workshop included four discussion groups that run in parallel. The discussion groups were a nice opportunity to engage in lively discussion and debate on important topics related to assessment of CoM techniques, and according to the interests of the group members. The format and the questions to be discussed are defined by the members of the groups according to their interest. Four discussion groups were organized around the themes described in the following subsections.

### 6.1. Measuring Heterogeneous Module Dependencies

The group members were: Robert Walker, S. Larsson, Neil Onfayco, Daqing Hou, and Hong Yul Yang. The group was initially dismayed to realize that each had joined in the hopes that someone else would have some strong opinions or good ideas about the question. When this turned out not to be the case, the discussion centred around four key questions: What kinds of module dependencies exist? Can we ever define a complete set of module dependencies? What means of measurement lie at our disposal? Can such measurements be useful in the real world to guide design decisions?

While no one was willing to go out on a limb and claim that a complete set of module dependencies could or could not be defined, the group had the disquieting sense that such a set might well be open-ended at least in the immediate future. Rather than adopt a bleak outlook, the group began to consider whether an incomplete set of module dependency types might suffice to guide design decisions in the real world. Anecdotes from real-world development were discussed in this context. This led to speculation that, even with a complete set, it might be profitable to restrict one's attention to a subset in certain situations, perhaps according to the development domain.

The question that naturally followed from the above was how to determine the appropriate subset of dependencies. It was agreed that this cannot be answered without a clear understanding of how exactly the dependencies differ, thus having some kind of classification of dependencies according to various characterizing facets was deemed necessary. We attempted to identify several of such facets, one of which is the nature of the "modules" that engage in the dependencies. They could be explicit program entities such as packages, classes and even functions or methods. On the other hand, the notion of modules can be extended to less explicit entities such as components, services and aspects. Another facet we identified was the nature in which the dependencies manifest themselves between modules. The group recognized the varying scales of "explicitness" of dependencies, ranging from syntactic, i.e. explicitly stated in the program, to semantic ones, which may not necessarily be automatically identifiable. Yet another facet identified was the level of abstraction the dependency applies to, which could either be at the code level, design level, or even at requirements level. There is also the question of whether the dependency is traceable across the different abstractions. While these facets are just initial examples, the group believed that such a mechanism for classifying dependencies will provide a systematic basis for decisions on the appropriate types of dependencies.

### 6.2. Measuring Modularity for Evolution and Reuse

The second discussion group, which involved Sunny Huynh, Yuanfang Cai, Chris Luer, and Alberto Costa, focused on answering three questions, with the first being: "What are the state-of-the-art techniques to assess software modularity?" Following are the points the group has summarized: (i) employ students to do empirical study, (ii) ask experts about which modularization techniques is better, (iii) assess through case studies, and (iv) use Design Structure Matrices (DSMs). DSM has been used to manifest the different evolution phenomena of Linux and Mozilla.

The second question was what are the problems that need to be solved as far as modularity measurement (for evolvability and reusability) is concerned? The members have explicitly agreed that it still remains difficult to precisely define such terms as modularity, high coupling, and low cohesion. Moreover, it is hard to predict future changes and module reuses, or to objectively measure these changes. The other points raised were: (i) what are design decisions? What are design properties? If a designer decides to make the program faster, is it a decision or property? (iv) can we formalize the role of design rules? Is it possible to develop programming language support for design rules? (v) which modularization techniques need to be compared? In what dimensions are they comparable? (vi) for a given module/requirement, how do we determine which modularization technique should be used to implement it? Classes, aspects, etc…? (vi) What are the proper objective measures for modularity? Time to accommodate a change? The number of bugs after the change?

The last question was "What are the possible research directions for modularity measurement towards reusability and maintainability assessment?" The group has elicited the following directions as relevant good candidates: (i) empirical studies of the decision tree model, (ii) develop benchmarks for comparing modularization techniques; possible options are a combination of: using Gof Design patterns; using Alessandro Garcia's testbed; and comparing OO vs. AO solutions with well-known and new metrics, (iii) exploring new programming paradigms: what comes after AOP?, (iv) language extension for design rules, (v) implementation competition, together with standardized test cases, and (vi) concretely, we first need to develop test cases, and then let the students implement the test cases using different modularization techniques, such as OOP vs. OOP. Finally, we evaluate and compare the reusability and evolvability of each implementation.

### 6.3. Empirical Studies and CoM Techniques

The third discussion group was formed by Phil Greenwood, Steffen Zschaler, Salvador Trujillo, Michele Sama, and Scott Fleming. The members debated the following question: how to design empirical studies for emerging modularisation techniques? Although several topics regarding the discussion topic were touched upon, the discussions generally focused on the issue of performing maintenance-related studies and how empirical studies should be performed in this context. The initial discussions centered on identifying the necessary elements to perform such

studies. The high-level elements that were identified to perform such a study include:

- Test subjects;
- Base programs/artefacts;
- Assessment treatments (i.e. tasks to perform to compare approaches such as maintenance changes);
- Assessment techniques (what to assess).

Following the identification of these key elements the group then went on to discuss the various elements and the issues surrounding them in more detail. The test subjects were identified as a key element to the success of any empirical study performed due to the key role they play when executing the experiment. There is potential for bias and error to be introduced to the empirical study by the test subjects. The group attempted to highlight the various attributes that could influence the outcome of the study and ways that this influence could be minimized.

The first variable of the test subjects identified was ensuring that each of the test subjects had similar levels of experience. This may be difficult to achieve when different test subjects are experts of different techniques. One member of the group had experience of this problem and they had successfully used aptitude tests to determine the ability of each of the test subjects prior to performing the empirical studies. These tests enabled them to group the test subjects more accurately according to their skill levels. A related issue touched upon involved discussing the suitability of using students in such studies and whether they have suitable levels of experience. It was mentioned that one of the reasons why they are so frequently used is because they are readily available and can be easily coerced to participate in such studies. The group also discussed how to select the test subjects to accurately model the population at large; again pre-testing was mentioned as a potential solution to this problem. One solution to increase the number of potential test subjects is to 'share' test subjects amongst various sites. This could be potentially achieved by dedicating workshop days at relevant conferences for performing certain empirical studies. Various groups performing related experiments could collaborate so that all necessary data can be collected in one session and each group could reuse the data for each of their own experiments.

The next discussion topic turned to the selection procedure used to determine the base program that the approaches to be assessed should be applied to. The group came to the conclusion that there should not be just on base program selected but instead a variety of base programs should be selected in order to cover a wide spectrum of domains and architectural styles. Doing this should reduce the bias towards one approach over the others and also allow broader conclusions to be drawn. Furthermore, the metrics used in such empirical studies are very low-level and so are dependent on one particular implementation. Using a range of implementations will again reduce the bias and generate broader conclusions.

When discussing the assessment treatments (in terms of maintenance changes) the group concluded that there were two general approaches to selecting such treatments. Firstly, realistic changes could be selected by analysing real-life systems and selecting maintenance changes performed on them. Or, the maintenance changes could be created by the experiment developers. Each of these approaches is valid and each has their own benefits. For example, using real-life maintenance changes improves the validity of the results gathered due the maintenance changes having real-world relevance. However, selecting the maintenance changes in this way may not provide the most interesting results as the treatments may not be suited to create the desirable observations from the base program. It was decided that the most appropriate way of selecting the maintenance changes would be to use an approach that is a combination of the two. Whereby the experiment developers propose a series of changes that would cause interesting events to occur but verify that these changes are also realistic with the original developers of the base application.

Two of the most important questions that need answering when performing empirical studies are: what to assess and what metrics should be used to measure this? The first part of the question can only be determined by the people carrying out the empirical study and is dependent on what outcomes they wish to establish from performing the study. Answering the second question depends on what needs to be assessed; however, attributes often have a number of metrics that can be used to assess them. As a result, this question becomes what is the best metric to use? The precise answer to this question will be unclear, each metric will be more appropriate in different circumstances. Furthermore, developers are often unaware of all the metrics that are available for assessing a particular attribute and so they may unwittingly chose a metric which is not the most suitable. One possible solution to this would be to establish a publicly-available resource that metric developers can submit their metrics to and then experiment developers can browse this resource and compare metrics to enable them to select the most appropriate one. However, this resource will require each of the metrics submitted to be validated to ensure that the metrics are suitable for their use.

Finally, a general topic that was discussed was the validity/appropriateness of formal experiments and the issues surrounding them. For example, selecting the correct control attributes of the experiment to assess the desired properties and to be able to recreate the experiment. This obviously leads to the possibility of selecting the incorrect control attributes which could lead to an invalid experiment being performed. The group did come to the conclusion that exploratory experiments do have some uses when assessing contemporary modularization techniques. However, it is important that their limitations are acknowledged when presenting any results gathered using such experiments. This led to a discussion comparing formal experiments with 'marketing' exercises. For example, papers often present the positive side of approaches using basic experiments which we consider marketing exercises. Whereas, formal experiments are much more rigorous and controlled, this is not to say that marketing exercises are inappropriate but they should not be misrepresented.

### 6.4. Interplay of Tangling, Composition and Modularity

The fourth group discussed a heterogeneous set of questions, including: (i) what are the open problems (and potential solutions) for quantitative modularity comparisons of AO vs. OO software?, (ii) what is the impact of composition techniques on modularity?, and (iii) how to quantify tangling and its impact on software quality? The group members were: Hayden Melton, Marc Eaddy,

Kurt Stirewalt, George Heineman, Thomas LaToza, and Alessandro Garcia.

The focus of the discussions moved back and forth between Aspects (as, perhaps, the most popular specific mechanism for modularization of crosscutting concerns) and on individual contemporary modularization techniques. Eventually, we opened the discussion to a broader topic, whether a particular modularization technology empirically be shown to improve the production of that software. As we all agreed, this was not only the underlying theme of this workshop, it must be the question that must be answered for contemporary modularization techniques. There are numerous avenues that can pursued for this goal, such as showing (a) improved program comprehension; (b) maintenance improvements; (c) performance/efficiency gains; and (d) reuse. Based upon the paper presentations from session II, these different avenues must consider evolution and inherent reusability when assessing the individual modularization technique being investigated. We have also concluded that there is no de-facto agreed benchmark for evaluating the (best) use of AOP, neither for C++ or Java. Could this community provide some perspectives on upcoming benchmarks?

As far as tangling phenomenon is concerned, we have realized that aspect tangling can be both incidental and essential. In this context, we have raised the following points: (i) there is a recognition that two aspects cannot be independently treated due to the essential tangling, (ii) a typical measure for tangling (whether incidental or essential) is the number of concerns that impact the same module, and (iii) cohesion measures based on data flow and control flow could be used to determine the strength of relations of concerns realized by a particular module.

## 7. Conclusions

The particular focus of this meeting was identifying the role of existing and innovative techniques to assess contemporary modularization techniques. Altogether, the workshop was a large success due to the quality of the submitted position papers and the level of participation of the audience. ACoM'07 achieved its goal to provide a forum for interactive discussions on the research issues of evaluating CoM approaches. From the discussion groups' outcomes, it is clearly possible to infer items for a research agenda in both the short- and long-term.

As such, the workshop achieved the following aims: (i) open issues on the assessment of contemporary modularization techniques were debated by researchers with different perspectives and backgrounds; (ii) the workshop contributed to bring the attention of the software engineering community to the importance of rigorous evaluation of emerging modularization techniques; (iii) we hope that this event motivated the expansion of research and practice associated with assessment of emerging modularization technologies; and (iv) the workshop offered a collaborative environment for both practitioners and researchers interested in effective assessment of new development techniques. All the position papers and presentation slides are available at the website. All the results obtained by the discussions were summarized through this report and made electronically available at the workshop website. We sincerely believe that the issues described here shall become part of the forthcoming research agenda on the assessment of CoM techniques.

## Acknowledgements

The organizers would like to thank all those who contributed with submissions to the workshop and the program committee members who invested their time on reviewing such submissions. In addition, we thank the session chairs for the fine work in coordinating the sessions and promoting interesting discussions, respectively. Finally, we would sincerely like to thank again all the ACoM'07 participants for their active involvement in the meeting and the level of their contributions to the groups' debate.

## References

[1] Murphy, G., Walker, R., Baniassad, E. Evaluating Emerging Software Development Technologies: Lessons Learned from Assessing AOP. IEEE Transactions on Software Engineering 25(4): 438-455,1999.

[2] Kersten, M. and Murphy, G. Atlas: A Case Study in Building a Web-based Learning Environment Using Aspect-Oriented Programming. Proceedings of OOPSLA'99, November 1999.

[3] Lippert, M., Lopes, C. A Study on Exception Detection and Handling using Aspect-Oriented Programming. Proceedings of ICSE 2000, May 2000.

[4] Lopes, C., Bajracharya, S. An Analysis of Modularity in Aspect Oriented Design. Proceedings of AOSD 2005, March 2005.

[5] Hannemann, J., Kiczales, G. Design Pattern Implementation in Java and AspectJ. Proceedings of OOPSLA 2002, pp 161-173.

[6] Garcia, A., Sant'Anna, C., Figueiredo, E., Kulesza, U., Lucena, C., von Staa, A. Modularizing Design Patterns with Aspects: a Quantitative Study. In Proceedings of AOSD'05, pp. 3-14.

[7] Filho, F., Cacho, N., Ferreira, R., Figueiredo, E., Garcia, A, Rubira, C. Exceptions and Aspects: the Devil is in the Details. Proceedings of the 14th Intl. Conference on Foundations on Software Engineering (FSE-14), Portland, USA, November 2006.

[8] Ceccato, M., Tonella P. Measuring the Effects of Software Aspectization. Proceedings of the 1st Workshop on Aspect Reverse Engineering (CD-ROM), The Netherlands, (2004).

[9] Kienzle, J., Guerraoui, R. AOP: Does It Make Sense? The Case of Concurrency and Failures. Proceedings of ECOOP 2002: 37-61.

[10] Soares, S., Borba, P. Implementing Distribution and Persistence Aspects with AspectJ. Proceedings of OOPSLA'02, pp. 174-190.

[11] Apel, S., Leich, T., Saake, G. Aspectual Mixin Layers: Aspects and Features in Concert. Proceedings of ICSE 2006, Shangai, China.

[12] Zhao, J. Measuring Coupling in Aspect-Oriented Systems. Proceedings of METRICS'2004, September 2004.

[13] Zhao, J. and Xu, B. Measuring Aspect Cohesion. Proceedings of FASE'04, LNCS 2984, Barcelona, March 2004, 54-68.

[14] Zakaria, A., Hosny, D. Metrics for Aspect-Oriented Design. Proceedings of the Workshop on Aspect-Oriented Modeling, San Francisco, USA (2003).

[15] Greenwood, P., Bartolomei, T., Figueiredo, E., Dosea, M., Garcia, A., Cacho, N., Sant'Anna, C., Soares, S., Borba, P., Kulesza, U., Rashid, A. On the Impact of Aspectual Decompositions on Design Stability: An Empirical Study. Proceedings of the 21st European Conference on Object-Oriented Programming (ECOOP.07), July 2007, Germany.

[16] Cacho, N., Sant'Anna, C., Figueiredo, E., Garcia, A., Batista, T., Lucena, C. Composing Design Patterns: A Scalability Study of Aspect-Oriented Programming. Proc. 5th Intl. Conference on Aspect-Oriented Software Development (AOSD'06), Bonn, Germany, 20-24 March 2006.