

*Proceedings*

# *ICPC 2007*

## *15th IEEE International Conference on Program Comprehension*

*26-29 June 2007  
Banff, Alberta, Canada*

### *Sponsoring Organizations*

IEEE Computer Society  
IEEE Computer Society Technical Council on Software Engineering

### *Supporting Organizations*

Informatics Circle of Research Excellence (iCORE)  
University of Alberta, Canada  
ITC-irst, Italy



Los Alamitos, California  
Washington • Tokyo





*15th IEEE International Conference  
on Program Comprehension*

*Table of Contents*

Message from the Chairs .....ix  
Conference Committee.....x  
Steering Committee .....xi  
Program Committee .....xii  
Additional Reviewers.....xiv

*Keynotes*

Search Based Software Engineering for Program Comprehension ..... 3  
*Mark Harman, King's College London, United Kingdom*

Making the Code Look like the Design--Aspects and Other Recent Work ..... 14  
*Gregor Kiczales, University of British Columbia, Canada*

*Technical Presentations*

*Technical Session 1: Feature and Concept Analysis*

Reducing Program Comprehension Effort in Evolving Software by Recognizing  
Feature Implementation Convergence ..... 17  
*Jay Kothari, Trip Denton, Ali Shokoufandeh, and Spiros Mancoridis*

Recovering Concepts from Source Code with Automated Concept Identification ..... 27  
*Maurice M. Carey and Gerald C. Gannod*

Combining Formal Concept Analysis with Information Retrieval for Concept Location in Source Code .....	37
<i>Denys Poshyvanyk and Andrian Marcus</i>	

### ***Technical Session 2: Dynamic Analysis***

Understanding Execution Traces Using Massive Sequence and Circular Bundle Views .....	49
<i>Bas Cornelissen, Danny Holten, Andy Zaidman, Leon Moonen, Jarke J. van Wijk, and Arie van Deursen</i>	
Tracking Objects to Detect Feature Dependencies.....	59
<i>Adrian Lienhard, Orla Greevy, and Oscar Nierstrasz</i>	
WAD: A Feasibility Study Using the Wicked Audio Debugger .....	69
<i>Andreas Stefik, Roger Alexander, Robert Patterson, and Jonathan Brown</i>	

### ***Technical Session 3: Conceptual Models***

A Hybrid Program Model for Object-Oriented Reverse Engineering .....	81
<i>Xinyi Dong and Michael W. Godfrey</i>	
From Reality to Programs and (Not Quite) Back Again.....	91
<i>Daniel Ratinu and Florian Deissenboeck</i>	

### ***Technical Session 4: Empirical Studies***

A Comparative Study of Three Program Exploration Tools .....	103
<i>Brian de Alwis, Gail C. Murphy, and Martin P. Robillard</i>	
Assessing the Comprehension of UML Class Diagrams via Eye Tracking .....	113
<i>Shebnaaz Yusuf, Huzefa Kagdi, and Jonathan I. Maletic</i>	
Empirical Evaluation of a UML Sequence Diagram with Adornments to Support Understanding of Thread Interactions .....	123
<i>Shaohua Xie, Eileen Kraemer, and R. E. K. Stirewalt</i>	

### ***Technical Session 5: Mining Software Repositories***

Detecting Interaction Coupling from Task Interaction Histories .....	135
<i>Lijie Zou, Michael W. Godfrey, and Ahmed E. Hassan</i>	
Mining Software Repositories for Traceability Links .....	145
<i>Huzefa Kagdi, Jonathan I. Maletic, and Bonita Sharif</i>	

Characterizing and Understanding Development Sessions.....	155
<i>Romain Robbes and Michele Lanza</i>	

### ***Technical Session 6: Aspects and Change***

Evaluating Aspect Mining Techniques: A Case Study .....	167
<i>Chanchal Kumar Roy, Mohammad Gias Uddin, Banani Roy, and Thomas R. Dean</i>	
Using Bayesian Belief Networks to Predict Change Propagation in Software Systems.....	177
<i>Siavash Mirarab, Alaa Hassouna, and Ladan Tabvildari</i>	

### ***Technical Session 7: Static Analysis***

A Non-conservative Approach to Software Pattern Detection.....	189
<i>Niklas Pettersson and Welf Löwe</i>	
Recovering Workflows from Multi Tiered E-Commerce Systems .....	198
<i>Maokeng Hung and Ying Zou</i>	
Error Report Driven Post-Mortem Analysis .....	208
<i>Yi Zhang and S. Purushothaman Iyer</i>	

### ***Technical Session 8: Visualization***

Interactive Views to Improve the Comprehension of UML Models – An Experimental Validation.....	221
<i>Christian F. J. Lange and Michel R. V. Chaudron</i>	
Program Comprehension through Software Habitability.....	231
<i>Richard Wettel and Michele Lanza</i>	
Scenario Explorer: Interactive Visualization of Use Cases.....	241
<i>Rogardt Haldal, Jenny Samuelsson, and Ola Sundin</i>	

### ***Short Papers Session***

Metrics for Measuring the Effectiveness of Decompilers and Obfuscators.....	253
<i>Nomair A. Naeem, Michael Batchelder, and Laurie Hendren</i>	
Constructing Usage Scenarios for API Redocumentation.....	259
<i>Juanjuan Jiang, Johannes Koskinen, Anna Ruokonen, and Tarja Systä</i>	
Software Comprehension through Concern-Based Queries .....	265
<i>Tommi Reinikainen, Imed Hammouda, Juba Laibo, Kai Koskimies, and Tarja Systä</i>	

Enforcing Constraints between Documentary Comments and Source Code .....271  
*C. Dylan Shearer and Michael L. Collard*

### ***Working Sessions***

15 Years of Program Comprehension .....279  
*Scott Tilley*

Designing Your Next Empirical Study on Program Comprehension .....281  
*Massimiliano Di Penta, R. E. K. Stirewalt, and Eileen Kraemer*

Comprehending Aspect-Oriented Programs: Challenges and Open Issues .....286  
*Giuseppe A. Di Lucca, Mike Smit, Bruce Fraser, Eleni Stroulia, and H. James Hoover*

### ***Tool Demonstrations***

Lagrein: Visualizing User Requirements and Development Effort.....293  
*Andrejs Jermakovics, Marco Scotto, Alberto Sillitti, and Giancarlo Succi*

Use of a Genetic Algorithm to Identify Source Code Metrics Which  
Improves Cognitive Complexity Predictive Models .....297  
*Rodrigo Vivanco*

Author Index .....301