



# Call for Papers

## **SCAM 2002**

### *Second IEEE International Workshop on Source Code Analysis and Manipulation*

1st October 2002,

**Montreal, Canada,**

Location: CRIM

address: 550 Sherbrooke Street West

(just a very short walk from the ICSM conference hotel)

Co-located with

IEEE International Conference on Software Maintenance, (ICSM 2002)

Fourth IEEE Workshop on Web Site Evolution, (WSE 2002)

Eighth IEEE Workshop on Empirical Studies of Software Maintenance, (WESS 2002)

Database Maintenance and Reengineering Workshop (DBMR 2002)

**Sponsored by** Institute of Electrical and Electronics Engineers (IEEE)

1.1	Group Photo . . . . .	4
1.2	Slides . . . . .	4
1.3	Top ten(ish) recommended books and papers . . . . .	4
1.4	Top ten(ish) myths of Source Code Analysis and Manipulation . . . . .	5
1.5	Summary of Discussions . . . . .	5
<b>2</b>	<b>Location</b>	<b>6</b>
<b>3</b>	<b>Keynote Speaker</b>	<b>6</b>
<b>4</b>	<b>The Journal of Automated Software Engineering Special Issue</b>	<b>6</b>
<b>5</b>	<b>Paper Submission Information</b>	<b>6</b>
<b>6</b>	<b>About SCAM</b>	<b>6</b>
6.1	Aims . . . . .	6
6.2	Topics Covered . . . . .	7
6.3	Definition of ‘Source Code’ . . . . .	7
6.4	Important Dates . . . . .	7
<b>7</b>	<b>Technical Program</b>	<b>7</b>
7.1	Keynote Address . . . . .	7
7.2	Testing, Metrics, Maintenance . . . . .	8
7.3	Source Transformation, Source Processing . . . . .	8
7.4	Slicing . . . . .	8
7.5	Dependence Graphs, Static Analysis . . . . .	8
<b>8</b>	<b>Registration</b>	<b>8</b>
<b>9</b>	<b>Accepted Papers</b>	<b>8</b>
<b>10</b>	<b>SCAM Committees</b>	<b>9</b>
10.1	Chairs . . . . .	9
10.2	Steering Committee . . . . .	10
10.3	Program Committee . . . . .	10
<b>11</b>	<b>Printable Call for Papers</b>	<b>10</b>
<b>12</b>	<b>History of SCAM</b>	<b>10</b>
<b>13</b>	<b>Enquiries about the workshop</b>	<b>11</b>

future SCAM workshops, to avoid us (as a community) repeating ourselves.

## 1.1 Group Photo

Not quite everyone made it to the photo call, but here is a group photo of all those who did.

On some screens this picture appears to be very dark. There is also a bigger version of this picture, which is optimised for lightness (thanks to Jim Cordy for the graphics manipulation). The bigger version may take some a few seconds to load on slower net connections. You will find it here.

## 1.2 Slides

Here are the scanned images (thanks to Lin Hu) of the slides that people made and put up to explain points during the workshop.

- Mutation Testing using Co-evolution  
by Mark Harman.
- Graph Transformations  
by Dave Binkley.
- Web structures  
by Andrea De Lucia
- Side effects  
by Mark Harman.
- Variable Dependence  
by Dave Binkley.
- Metrics as Fitness Functions  
by Mark Harman.
- Combining Slicing and Concept Assignment  
by Mark Harman.
- The Montréal Boat Example  
by Sebastian Danicic.
- Compiling and Maintenance  
by an unknown author (please contact Mark Harman if this is you).
- Knowledge Based Analysis Diagram  
by Ira Baxter and others.

Some of these slides may be assigned to the wrong (or unknown) author. If this has happened, please email `Mark.Harman@brunel.ac.uk` that he can fix it. Also, if the title is not the one the author wants, email `Mark.Harman@brunel.ac.uk`.

## 1.3 Top ten(ish) recommended books and papers

Top 10 ( or there about ) Books / Papers recommended by and for SCAMers.

1. "The Psychology of Computer Programming", Silver Anniversary Edition by Gerald M. Weinberg. ISBN: 0932633420.
2. ENGLER et al. "Bugs as Deviant Behavior".
3. "An Introduction to General Systems Thinking", by Gerald M. Weinberg. ISBN: 0932633498.
4. Aho, Seti, Ullman, "Compiler Construction", "The Dragon book". ( or Fischer and LaBlanc ).
5. "Program Slicing", Weiser 1982 and 1984.
6. "Interprocedural slicing using dependence graphs." ACM Transactions on Programming Languages and Systems 12, 1 (January 1990), pp. 26-60.
7. Algebra I ( V.D. Liaerden)

It would be nice to have fuller references; these may appear here soon.

1. Tools needed for "What-if " and "Why-is " analysis.
2. How to compare / evaluate / standardize program transformation systems?
3. Safe Systems. How safe ( secure / reliable ) is a given system?
4. Myth: My system is automatic.
5. Myth: My system generates the right answer.
6. Myth: My system makes the whole answer.
7. Myth: My AST / ASG / My Graph is the structure of the program.
8. Myth: General understanding helps solve my problem.
9. Myth: You can analyze the program.

## 1.5 Summary of Discussions

Dave Binkley took these notes on the discussions and points made at the workshop. Mark Harman slightly edited them, as they appear here (below).

### SCAM 2002 Discussion Points

1. What are the enabling techniques for SCAM:
  - (a) Parsing
  - (b) Slicing
  - (c) Data Flow Analysis
2. Many analyses start with information obtained from a compiler. However, compiler is prejudiced towards compilation and thus may not be the best source of information.
3. Related question: what are the lex and yacc of analysis tools?
4. We need to include knowledge known before the source was created / modified (e.g. what was in the designers / maintainers head).
5. Complete automation is a mistake ... better to combine machine and human effort in a symbiotic relationship.
6. Information is available in coding style imposed by framework (e.g. OO) and management; in general, there is idiomatic information available in the code (e.g. UML has a structure that might be exploited). Thus, there are abstractions other than the source that are useful.
7. Given source is executable, it's the final "word" on the meaning of the system; even if documentation is out of date it still provides some useful information.
8. Engineer's needs are specific; therefore, they need tools to focus on particular questions not tools that produce exhaustive solutions.
9. A quick vague answer may be sufficient and even better than, a more precise answer that takes a long time to compute.
10. What is the correct (useful) combination of static and dynamic information?
11. How does one account for things that are unknown while performing an analysis (e.g. the affect of a call to an unknown procedure)? Here, conservatism is not necessarily the right choice. Algorithms that are correct most of the time, but incorrect sometimes, may be useful if they save a programmer time.
12. A programmer uses a library by chaining together calls to build a program. How do we treat software tools in the same way so that the user can pick the tools s/he needs and chain them together.
13. Transformations can be specified using source code templates. This allows programmer to work with a "formalism" that s/he is familiar with. (i.e. it is the "Lingua franca").
14. What are the use-cases that drive the need for slicing? Have slicing proponents have put the cart before the horse?
15. Chopping is often more useful than slicing.

The address is 550 Sherbrooke Street West, 1st floor, Montréal (QC) Canada, H3A 1B9. . This is just a very short walk from the ICSM conference hotel (The Holiday Inn), which is at 420 Sherbrooke Street West.

A map is available [here](#).

A list of Hotels in Montréal is available [here](#).

(Thanks to WSE 2002 for these details, which have been copied from their website.)

### 3 Keynote Speaker

The SCAM 2002 keynote speaker will be Ira Baxter.



Ira Baxter, Semantic Designs, USA

Ira Baxter has been involved with computing since 1966, and implemented one of the first minicomputer timesharing systems on a Data General Nova in 1970. He received his B.S. in Computer Science (1973), and worked for a number of years in industry both as a consultant and as owner of Software Dynamics, a systems software house, where he designed compilers, time-sharing and network operating systems. In 1990, he received a Ph.D. in Computer Science from University of California, Irvine, where he studied design reuse using transformational methods.

In 1995, he founded Semantic Designs. He is the principal architect of Semantic Designs' Design Maintenance System (DMS), a meta-system for developing and using flexible and scalable program transformation systems.

### 4 The Journal of Automated Software Engineering Special Issue

There will be a special issue of The Journal of Automated Software Engineering containing extended versions of selected papers from SCAM 2002.

### 5 Paper Submission Information

Papers should be submitted electronically. There is an electronic submission site at <http://serg.ing.unisannio.it/scam2002/subm>

There will be a special issue of The Journal of Automated Software Engineering containing extended versions of selected papers from SCAM 2002.

#### Important Dates

<b>Deadline for submission</b>	<b>: 19th April 2002</b>
<b>Notification</b>	<b>: 14th June 2002</b>
<b>Camera Ready Due</b>	<b>: 19th July 2002</b>
<b>Workshop</b>	<b>: 1st October 2002</b>

### 6 About SCAM

#### 6.1 Aims

The aim of this workshop is to bring together researchers and practitioners working on theory, techniques and applications which concern analysis and/or manipulation of the source code of computer systems. While much attention in the wider

## 6.2 Topics Covered

Topics include, but are not limited to: program transformation, abstract interpretation, program slicing, source level software metrics, decompilation, source level testing and verification, source level optimisation and program comprehension. However, the emphasis of the workshop is on the analysis and manipulation techniques *themselves*.

## 6.3 Definition of 'Source Code'

For the purpose of clarity 'source code' is taken to mean any fully executable description of a software system. It is therefore so-construed as to include machine code, very high level languages and executable graphical representations of systems. The term 'analysis' is taken to mean any automated or semi automated procedure which takes source code and yields insight into its meaning. The term 'manipulation' is taken to mean any automated or semi-automated procedure which takes and returns source code.

## 6.4 Important Dates

### Important Dates

<b>Deadline for submission</b>	<b>: 19th April 2002</b>
<b>Notification</b>	<b>: 14th June 2002</b>
<b>Camera Ready Due</b>	<b>: 19th July 2002</b>
<b>Workshop</b>	<b>: 1st October 2002</b>

## 7 Technical Program

A printer-friendly version of the program is also available in postscript and PDF formats.

In keeping with the spirit and format of a workshop, SCAM will have a highly discursive nature, with four theme-based discussion tracks and a keynote presentation, aimed at structuring and stimulating discussion.

Authors will have a 15 minute slot to present their work. This is 10 minutes (maximum) for talking with 5 minutes (minimum) for questions. Authors are encouraged not to attempt to present the details of their paper in this time. Rather, respecting the discussion-centered goal of SCAM, authors are encouraged to use a **few** slides to present points, claims, issues and topics for discussion and to use their time allocation to attempt to set the agenda for the ensuing discussions.

Each session has a specifically allocated discussion time at the end of the presentations to allow for this.

8:00 - 9:00	Continental breakfast
9:00 - 9:45	<b>Keynote address</b>
9:45 - 10:00	short break
10:00 - 11:30	<b>Session I: Testing, Metrics, Maintenance</b>
11:30 - 12:00	refreshment break
12:00 - 1:30	<b>Session II: Source Transformation, Source Processing</b>
1:30 - 2:30	lunch
2:30 - 4:00	<b>Session III: Slicing</b>
4:00 - 4:30	refreshment break
4:30 - 6:00	<b>Session IV: Dependence Graphs, Static Analysis</b>
6:00 - 6:30	<b>Wrap up / closing / elections</b>
7:00 - ?::??	SCAM reception (shared with attendess from WSE 2002, WESS 2002 and DBMR 2002)

### 7.1 Keynote Address

Parallel Support for Source Code Analysis and Modification

**Ira D. Baxter**

igation

**Sudipto Ghosh**

An Extensible Metrics Extraction Environment for Object-oriented Programming Languages

**Terrence Harmer, George Wilkie**

Evaluating Clone Detection Tools for Use During Preventative Maintenance

**John Bailey, Elizabeth Burd**

A Simple Mathematically Based Framework for Rule Extraction Using Wide Spectrum Language

**Frederick Ramsey, James Alpigini**

### 7.3 Source Transformation, Source Processing

VADA: A Transformation-based System for Variable Dependence Analysis

**Mark Harman, Lin Hu, Chris Fox, Sebastian Danicic, Joachim Wegener**

Combining Source Transformation and Operator Overloading Techniques to Compute Derivatives for MATLAB Programs

**Christian Bishof, Hans Martin Buecker, Bruno Lang, Arno Rasch, Andre Vehreschild**

Mechanized Operational Semantics of WSL

**Xingyuan Zhang, Malcolm Munro, Mark Harman, Lin Hu**

Handling Preprocessor-conditioned Declarations

**Lerina Aversano, Massimiliano di Penta, Ira Baxter**

Grammar Programming in TXL

**Thomas Dean, James Cordy, Andrew Malton, Kevin Schneider**

### 7.4 Slicing

An Interprocedural Amorphous Slicer for WSL

**Mark Harman, Lin Hu, Malcolm Munro, Xingyuan Zhang, Sebastian Danicic, Mohammed Daoudi**

Dynamic Slicing Object-Oriented Programs for Debugging

**Baowen Xu**

Construction of the System Dependence Graph for Web Application Slicing

**Filipo Ricca, Paolo Tonella**

Predicate-Based Dynamic Slicing of Message Passing Programs

**Juergen Rilling, Hon F. Li, Dhrubajyoti Goswami**

### 7.5 Dependence Graphs, Static Analysis

Using Dependence Graphs as a Support to Document Programs

**Francoise Balmas**

Precise Call Graph Construction in the Presence of Function Pointers

**Ana Milanova, Atanas Rountev, Barbara Ryder**

Semantics Guided Filtering of Combinatorial Graph Transformations in Declarative Equation Based Languages

**Peter Bunus, Peter Fritzson**

Visualization of Exception Propagation for Java using Static Analysis

**Byeong-Mo Chang, Jang-Wu Jo, Soon Hee Her**

Papers accepted for the workshop are listed in Section 9.

## 8 Registration

Registration is available through the  
ICSM registration page

## 9 Accepted Papers

- Using Dependence Graphs as a Support to Document Programs  
Francoise Balmas
- Towards Measurement of Testability of Concurrent Object-Oriented Programs using Fault Insertion: A Preliminary Investigation  
Sudipto Ghosh



- Dynamic Slicing Object-Oriented Programs for Debugging  
Baowen Xu
- A Simple Mathematically Based Framework for Rule Extraction Using Wide Spectrum Language  
Frederick Ramsey, James Alpigini
- Construction of the System Dependence Graph for Web Application Slicing  
Filipo Ricca, Paolo Tonella
- An Extensible Metrics Extraction Environment for Object-oriented Programming Languages  
Terrence Harmer, George Wilkie
- An Interprocedural Amorphous Slicer for WSL  
Mark Harman, Lin Hu, Malcolm Munro, Xingyuan Zhang, Sebastian Danicic, Mohammed Daoudi
- Mechanized Operational Semantics of WSL  
Xingyuan Zhang, Malcolm Munro, Mark Harman, Lin Hu
- Visualization of Exception Propagation for Java using Static Analysis  
Byeong-Mo Chang, Jang-Wu Jo, Soon Hee Her
- Predicate-Based Dynamic Slicing of Message Passing Programs  
Juergen Rilling, Hon F. Li, Dhrubajyoti Goswami
- Precise Call Graph Construction in the Presence of Function Pointers  
Ana Milanova, Atanas Rountev, Barbara Ryder
- Handling preprocessor-conditioned declarations  
Lerina Aversano, Massimiliano di Penta, Ira Baxter
- Evaluating clone detection tools for use during preventative maintenance  
John Bailey, Elizabeth Burd
- Semantics Guided Filtering of Combinatorial Graph Transformations in Declarative Equation Based Languages  
Peter Bunus, Peter Fritzson
- Combining source transformation and operator overloading techniques to compute derivatives for MATLAB programs  
Christian Bishof, Hans Martin Buecker, Bruno Lang, Arno Rasch, Andre Vehreschild
- Grammar Programming in TXL  
Thomas Dean, James Cordy, Andrew Malton, Kevin Schneider

## 10 SCAM Committees

### 10.1 Chairs

#### **General Chair**

Dave Binkley, Computer Science Department, Loyola College in Maryland, USA

#### **Program Chairs**

Andrea De Lucia, Research Centre on Software Technology, Università degli Studi del Sannio, Italy and  
Jim Cordy, School of Computing, Queen's University, Canada

#### **Local Arrangements Chair**

Tom Dean, Department of Electrical and Computer Engineering, Queen's University, Canada

#### **Publicity Chair**

Mark Harman, Department of Information Systems and Computing, Brunel University, UK and  
Jianjun Zhao, Department of Computer Science and Engineering, Fukuoka Institute of Technology, Japan

#### **Finance Chair**

Dave Binkley, Computer Science Department, Loyola College in Maryland, USA

#### **Electronic Submission Chair**

Silvio Stefanucci, Research Centre on Software Technology, Università degli Studi del Sannio, Italy

Gerardo Canfora, Research Centre on Software Technology, Università degli Studi del Sannio, Italy  
Andrea De Lucia, Research Centre on Software Technology, Università degli Studi del Sannio, Italy  
Keith Gallagher, Computer Science Department, Loyola College in Maryland, USA  
Mark Harman, Department of Information Systems and Computing, Brunel University, UK  
Malcolm Munro, Department of Computer Science, University of Durham, UK

### 10.3 Program Committee

Giulio Antoniol, Research Centre on Software Technology, Università degli Studi del Sannio, Italy  
Ira Baxter, Semantic Designs, USA  
Keith Bennett, Department of Computer Science, University of Durham, UK  
Dave Binkley, Computer Science Department, Loyola College in Maryland, USA  
Liz Burd, Department of Computer Science, University of Durham, UK  
Gerardo Canfora, Research Centre on Software Technology, Università degli Studi del Sannio, Italy  
Aniello Cimitile, Research Centre on Software Technology, Università degli Studi del Sannio, Italy  
Pascal Costanza, Institut für Informatik III, Universität Bonn, Germany  
Jim Cordy, School of Computing, Queen's University, Canada  
Tom Dean, Department of Electrical and Computer Engineering, Queen's University, Canada  
Andrea De Lucia, Research Centre on Software Technology, Università degli Studi del Sannio, Italy  
Anna Rita Fasolino, Dipartimento di Informatica e Sistemistica, University of Naples, Italy  
Keith Gallagher, Computer Science Department, Loyola College in Maryland, USA  
Michael Godfrey, Computer Science Department, University of Waterloo, Canada  
Mark Harman, Department of Information Systems and Computing, Brunel University, UK  
Rob Hierons, Department of Information Systems and Computing, Brunel University, UK  
Mariam Kamkar, Department of Computer and Information Science, Linköping University, Sweden  
Kostas Kontagiannis, Department of Electrical and Computer Engineering, University of Waterloo, Canada  
Rainer Koschke, University of Stuttgart, Germany  
Andrew Malton, Computer Science Department, University of Waterloo, Canada  
Hausi Müller, Department of Computer Science, University of Victoria, Canada  
Malcolm Munro, Department of Computer Science, University of Durham, UK  
Vaclav Rajlich, Department of Computer Science, Wayne State University, USA  
Jurgen Rilling, Department of Computer Science, Concordia University, Canada  
Paolo Tonella, Centro per la Ricerca Scientifica e Tecnologica, Istituto Trentino di Cultura, Italy  
Genny Tortora, Dipartimento di Matematica e Informatica, Università di Salerno, Italy  
Arie Van Deursen, Centrum voor Wiskunde en Informatica, Netherlands  
Michael Van De Vanter, Sun Microsystems Labs, USA  
Giuseppe Visaggio, Dipartimento di Informatica, Università degli Studi di Bari, Italy  
Martin Ward, Software Migrations Ltd, UK  
Hongji Yang, Department of Computer Science, DeMontfort University, UK  
Jianjun Zhao, Department of Computer Science and Engineering, Fukuoka Institute of Technology, Japan

## 11 Printable Call for Papers

A single sheet call is available in postscript and PDF. This document also contains the information authors will need regarding submission of papers (Section 5) and important dates (Section 6.4).

## 12 History of SCAM

The first SCAM workshop (SCAM 2001) was held in Florence, on November 10th 2001, co-located with ICSM 2001, WESS 2001 and WSE 2001. The workshop was a one day event with the emphasis on discussion. 48 people attended the workshop. The SCAM 2001 Proceedings are published by the IEEE and there will be a special issue of the journal Information and Software Technology to appear shortly featuring extended versions of selected papers from SCAM 2001.

Andrea De Lucia and Jim Cordy), please contact the general chair, Dave Binkley.