Curriculum Vitae of Fausto Spoto

General Information

- Born on January 13, 1973 in Palermo, Italy
- Lives in Via Fincato, 12, 37131 Verona, Italy
- Phone: +39 3204352527
- e-mail: `fausto.spoto@univr.it`
- Italian national
- Speaks and writes English, Italian and French

Studies and Scientific Collaborations

7/1995 *Laurea cum laude* in *Computer Science* at the Dipartimento di Informatica of the university of Pisa, with a dissertation on the *Concrete and Abstract Semantics of Prolog with Cut*, under the supervision of Prof Giorgio Levi and Prof Egon Boerger.

10/1998-3/1999 Visits the School of Computer Studies of the University of Leeds, UK, during his PhD, under the supervision of Doctor Patricia M. Hill.

2/2000 PhD in Computer Science at the Dipartimento di Informatica of the university of Pisa, with a thesis on *Analysis of Logic Programs through Linear Refinement*, under the supervision of Prof Giorgio Levi. Referees: Prof Michael Codish e Prof Maurice Bruynooghe.

1-12/2000 Postdoc at the INRIA/CNRS research centre IRISA of Rennes, France, under the supervision of Prof Thomas Jensen.

1/2001-10/2002 Postdoc at the Dipartimento di Informatica of Verona, under the supervision of Prof Roberto Giacobazzi.

9-12/2001 Invited researcher at the School of Computing of the University of Leeds, where he works with Doctor Patricia M. Hill at the EPSRC GR/R53401 project.

6/2002 Visits the Nijmeegs Instituut voor Informatica en Informatiekunde of the Katholieke Universiteit Nijmegen, The Netherlands, where he works with Prof Erik Poll.

7-8/2002 Visits the IT University and the DIKU of Copenhagen, Denmark, where he works with Prof Fritz Henglein.

10-11/2002 Visits the School of Computing of Leeds, where he works with Doctor Patricia M. Hill.

11/2002-10/2005 Researcher at the Dipartimento di Informatica of Verona.

7-9/2003 Visits the Université de Rennes I, France, where he works with Prof Thomas Jensen.

10/2007, 10/2008 Visits the Université de La Réunion, France, where he works with Prof Frédéric Mesnard and Étienne Payet.

11/2005-Today Associate Professor at the Dipartimento di Informatica of Verona.

6/2006 Visits the Universitad Politécnica de Madrid in collaboration with Professors Elvira Albert and Samir Genaim.

12/2007-1/2008 Visits the Chiang Mai University (Thailand) with an ASEM-DUO fellowship, where he teaches classes on abstract interpretation.

7/2009-9/2009 CooperInt fellowship from the University of Verona to visit Professor Michael Ernst at the University of Washington.

10/2010-9/2011 Sabatical leave, spent between France (Réunion) and Germany (RWTH Aachen).

2/2012 Participation at the Dagstuhl Seminar on *Analysis of Executables: Benefits and Challenges*.

Fellowships and Research Projects

7/1999 European fellowship for Training and Mobility of Researchers (TMR) for taking part in the *Practice and Principles of Declarative Programming '99* conference in Parigi.

1999 Works at the EPSRC gr/m05645 UK research project *Software Support for Constraint Logic Programming*.

2000 Works, during his postdoc in Rennes, at the European research project *SECSAFE* on the security of information systems based on static analysis.

2001-2002 Works in Verona at the Italian MURST *Abstract Interpretation, Type-Systems and Control-Flow Analysis* project.

2001 He is co-author, with Patricia M. Hill, of the UK research project EPSRC gr53401 *Escape Analysis for Object-Oriented Languages*.

2002 Works, during his visit to Nijmegen, at the *VERIFICARD* European research project on the automatic verification of JavaCard microchips.

2002-2004 Works in Verona at the Italian MURST project *CoVer: Constraint-Based Verification of Reactive Systems*.

2004 He his co-author, with Patricia M. Hill, of the UK research project *Static Analysis for Numerical Stability* funded by the Royal Society of Sciences.

2002-2005 Works in Verona at the Italian FIRB project *SPY-Mod: Interpretazione Astratta e Model Checking per la Verifica di Sistemi Embedded*.

2005-2007 Works in Verona at the Italian MURST project *AIDA*.

2008 ASEM-DUO fellowship to visit the Chiang Mai University (Thailand) and teach there classes on abstract interpretation.

2008 Works in Verona at the new edition of the Italian PRIN project AIDA.

2009 CooperInt fellowship of the University of Verona to visit the University of Washington.

2011 Von Humboldt Stiftung fellowship to visit the RWTH university in Aachen, Germany.

2018 CooperInt visit at the Complutense University, Madrid, Spain.

Scientific Activity

His research activity was originally centered on the analysis of computer programs written with a logic language, in order to derive properties such as sharing of data structures among variables, *freeness* and types. In that context, the development of new static analyses has been based on abstract interpretation and on the theory of linear refinement of domains, which lets one add directionality and compositionality to static analyses.

The research interest has then moved towards the static analysis of Java and Java bytecode programs, in order to verify properties of those programs before they are run, such as the classes of the objects bound to the variables, the escape of data structures from the syntactical context where they are allocated, the set of fields of objects which get modified during the execution of

a piece of code. The extensive use of Java for programming web or stand-alone applications and the use of Java bytecode as an exchange language for compiled programs, give this research a growing industrial interest; this is also the consequence of a growing feeling, among the software houses, that software verification is important before the software is sold. The use of Java and Java-like bytecode in *insecure* contexts such as mobile phones introduces new perspectives for the verification of software. From a theoretical viewpoint, all these results can be based on the theory of *abstract interpretation*, developed during the 1970's by Cousot and Cousot. The advantage of this theory is that it lets one build static analyses that are provably correct; moreover, it lets one compare different analyses wrt. their precision and derive optimal ones. Fausto Spoto has contributed to the application of abstract interpretation to programming languages that use dynamic data structures allocated in the heap (such as object-oriented languages) and to low-level programming languages (such as Java bytecode).

Fausto Spoto has developed a theoretical framework where many static analyses can be stated in a denotational fashion, so that they are naturally compositional, flow and context-sensitive and able to express functional properties of the code rather than just properties of the state of the computation. This has been achieved through a *magic-sets* program transformation which yields information on internal program points also when a denotational analysis is applied. This has allowed the theoretical development and the implementation of denotational analyses for information flow, sharing of data structures, nullness of program variables and fields, length of chains of pointers from the variables of the program and hence termination, for the non-cyclicity of the data structures bound to the variables and the initialisation of fields. These analyses have been adapted to the analysis of Java and Android programs.

These analyses have been improved by using information on the arguments that a method might modify during its execution. They have been implemented inside the Julia analyser, whose main author is Fausto Spoto, which lets one analyse quite large programs (up to 50000 methods). A web interface lets one use the analyser through the Internet in order to achieve nullness and termination analyses. The Julia analyser is now property of Julia Srl (`http://www.juliasoft.com`), a spin-off company of the University of Verona. Fausto Spoto is scientific consultant of the company. Julia Srl has been working in a collaborative project with the University of Washington, funded by the US Air Force, to the development of security analyses for Java and Android programs, and Fausto Spoto is the software project leader for Julia Srl.

Fausto Spoto has developed new static analyses for Java and Java bytecode based on constraints and automata as well. Constraint-base techniques are often not context-sensitive, but allow a more precise modelling of the fields of the objects. Automata-based techniques allow a simple abstraction of the *shape* of the traces of executions. They are ideal to approximate the sequence of operations that must be performed by such traces, and their relative ordering.

The results of the research of Fausto Spoto should be judged from the point of view of their formal cleaness and of the quality of the correctness proofs derived from the various static analyses. Moreover, one should consider their technological impact: such analyses have been implemented inside the Julia analyser and the quality and efficiency of such implementations have induced companies such as Aonix (`www.aonix.com`) and Google (`www.google.com`) to contact Fausto Spoto in order to use those analyses and develop new ones. As a consequence, it is more and more apparent that the static analyses of Julia and the new ones which are going to be developed will have an economical significance.

Teaching Activity

He starts teaching in 1997 with lab classes on *Programming* in Pisa, with the help of Prof Dino Pedreschi, and then in Leeds for *Functional Programming*, with the help of Doctor Mark Tarver. Once in Verona, he teaches lab classes on *Computer Hardware* and then stabilises his teaching around the classes of *Programming*, *Logic Programming* and *Compilers*. For the latter, he needed to identify a few topics, compatible with the relatively small number of hours of teaching, yet significant enough to include both theory and compiler implementation. Such classes have been centered on the compilation of an object-oriented language, which is relatively different from what

is taught in traditional compilers courses. To that goal, a small didactic compiler has been developed for a simple object-oriented language called Kitten. During his visits to Réunion, he teaches imperative and logic programming. Up to 2008, he has also taught classes on *Software Engineering in Java* for students of the Master in Computer Game Development at the university of Verona. These classes teach the methodological development of software in Java, design patterns, concurrent programs development and testing. *Basic Computer Science* for students from humanistic disciplines at the university of Verona. Those are simple courses, but require a strong attention and a clean presentation in order to make them accessible, useful and non-trivial.

Software products

His research activity has induced the development of the following two software products (more information on the description of the scientific activity):

- Julia: this is a generic and localised software analyser for full Java bytecode. It includes analyses of class for the extraction of the application, analyses for non-interference, for escape, sharing and non-cyclicity of data structures, nullness and termination. It is by far the more evolved system for the analysis of Java bytecode and hence Java. It is now property of Julia Srl (`http://www.juliasoft.com`).
- Kitten: this is a didactic compiler for the students of the course on *Compilers* in Verona. His goal is to allow them to see the partial results of the compilation, from the list of tokens to the abstract syntax, from the type information to the intermediate code, and act directly on the compiler to add new functionalities, by exploiting the object-oriented nature of the same compiler.

Publications

International Journals

1. Fausto Spoto. *Operational and Goal-Independent Denotational Semantics for Prolog with Cut*. In **Journal of Logic Programming**, 42(1):1-46, January 2000.
2. Patricia M. Hill and Fausto Spoto. *Generalising Def and Pos to Type Analysis*. In **Journal of Logic and Computation**, 12(3), June 2002, pages 497-542.
3. Giorgio Levi and Fausto Spoto. *Pair-Independence and Freeness Analysis through Linear Refinement*. In **Information and Computation**, 182(1), pages 14-52, 2003.
4. Fausto Spoto and Thomas Jensen. *Class Analyses as Abstract Interpretations of Trace Semantics*. In **ACM TOPLAS**, volume 25, September 2003, pages 578-630.
5. Patricia M. Hill and Fausto Spoto. *Logic Programs as Compact Denotations*. In **Elsevier Computer Languages, Systems and Structures**, volume 29, issue 3, pages 45-73, October 2003.
6. Patricia M. Hill and Fausto Spoto. *Deriving Escape Analysis by Abstract Interpretation*. In **Higher Order and Symbolic Computation**, volume 19, pages 415-463, 2006.
7. Fausto Spoto. *Optimality and Condensing of Information Flow through Linear Refinement*. In **Theoretical Computer Science**, 2007, volume 388, pages 53-82. Elsevier.
8. Fausto Spoto, Fred Mesnard and Etienne Payet. *A Termination Analyser for Java Bytecode Based on Path-Length*. In **ACM TOPLAS**, volume 32, number 3, 2010.
9. Fausto Spoto and Etienne Payet. *Magic-sets for Localised Analysis of Java Bytecode*. In **Higher-Order and Symbolic Computation**, volume 23, number 1, pages 29–86, 2010.

10. Fausto Spoto. *Precise null-Pointer Analysis.* In **Software and Systems Modeling**, volume 10, number 2, pages 219–252, 2011.

11. Étienne Payet and Fausto Spoto. *Static Analysis of Android Programs.* In **Information & Software Techonology**, volume 54, number 11, pages 1192–1201, 2012.

12. Đurica Nikolić and Fausto Spoto. *Inferring Complete Initialization of Arrays.* In **Theoretical Computer Science**, volume 484, pages 16–40, 2013.

13. Đurica Nikolić and Fausto Spoto. *Reachability Analysis of Program Variables.* In **ACM Transactions on Programming Languages and Systems**, volume 35, number 4, 2013.

14. Enrico Scapin and Fausto Spoto. *Field-sensitive Unreachability and non-Cyclicity Analysis.* In **Science of Computer Programming**, volume 95, pages 359–375, 2014.

International Conferences

15. Giorgio Levi and Fausto Spoto. *An Experiment in Domain Refinement: Type Domains and Type Representations for Logic Programs.* In Catuscia Palamidessi, Hugh Glaser, and Karl Meinke, editors, proc. of *PLILP/ALP'98*, Principles of Declarative Programming, volume 1490 of **Lecture Notes in Computer Science**, pages 152-169, Pisa, Italy, September 1998.

16. Fausto Spoto and Giorgio Levi. *Abstract Interpretation of Prolog Programs.* In A. M. Haeberer, editor, proc. of *AMAST'98*, the 7th International Conference on Algebraic Methodology and Software Technology, volume 1548 of **Lecture Notes in Computer Science**, pages 455-470, Amazonia, Manaus, Brazil, January 1999.

17. Patricia M. Hill and Fausto Spoto. *Freeness Analysis through Linear Refinement.* In Static Analysis Symposium, *SAS'99*, volume 1694 of **Lecture Notes in Computer Science**, pages 85-100, Venice, Italy, September 1999.

18. Giorgio Levi and Fausto Spoto. *Non Pair-Sharing and Freeness Analysis through Linear Refinement.* In proc. of *PEPM'00*, the ACM SIGPLAN Workshop on Partial Evaluation and Semantics-Based Program Manipulation, pages 52-61, Boston, USA, January 2000.

19. Patricia M. Hill and Fausto Spoto. *Analysis of Downward Closed Properties of Logic Programs.* In *AMAST'00*, Algebraic Methodology and Software Technology, volume 1816 of **Lecture Notes in Computer Science**, pages 181-196, Iowa City, USA, May 2000.

20. Fausto Spoto. *Watchpoint Semantics: A Tool for Compositional and Focussed Static Analyses.* In P. Cousot, editor, *SAS'01*, the Static Analysis Symposium, volume 2126 of **Lecture Notes in Computer Science**, pages 127-145. Paris, France, July 2001.

21. Thomas Jensen and Fausto Spoto. *Class Analysis of Object-Oriented Programs through Abstract Interpretation.* In F. Honsell and M. Miculan, editors, *FOSSACS'01*, Foundations of Software Science and Computation Structure, volume 2030 of **Lecture Notes in Computer Science**, pages 261-275. Genova, Italy, April 2001.

22. Gianluca Amato and Fausto Spoto. *Abstract Compilation for Sharing Analysis.* In H. Kuchen and K. Ueda, editors, proc. of *FLOPS'01*, the Fuji International Symposium on Functional and Logic Programming, volume 2024 of **Lecture Notes in Computer Science**, pages 311-325, Tokyo, Japan, March 2001.

23. Patricia M. Hill and Fausto Spoto. *A Foundation of Escape Analysis.* In H. Kirchner and C. Ringeissen editors, proc. of *AMAST'02*, Algebraic Methodology and Software Technology, volume 2422 of **Lecture Notes in Computer Science**, pages 380-395, St. Gilles les Bains, La Réunion island, France, September 2002.

24. Patricia M. Hill and Fausto Spoto. *A Refinement of the Escape Property.* In A. Cortesi, editor, proc. of *VMCAI'02* workshop on Verification, Model Checking and Abstract Interpretation, volume 2294 of **Lecture Notes in Computer Science**, pages 154-166, Venice, Italy, January 2002.

25. Patricia M. Hill and Fausto Spoto. *Logic Programs as Compact Denotations.* In G. Gupta editor, proc. of *PADL'03*, Practical Aspects of Declarative Languages, volume 2562 of **Lecture Notes in Computer Science**, pages 339-356, New Orleans, Louisiana, USA.

26. Fausto Spoto and Erik Poll. *Static Analysis for JML's assignable Clauses.* In G. Ghelli editor, proc. of *FOOL-10*, the 10th International Workshop on Foundations of Object-Oriented Languages, New Orleans, Louisiana, USA, January 2003.

27. Samir Genaim and Fausto Spoto, *Information Flow Analysis for Java Bytecode.* In R. Cousot editor, proc. of *VMCAI'05*, the Sixth International Conference on Verification, Model Checking and Abstract Interpretation, volume 3385 of **Lecture Notes in Computer Science**, pages 346-362. Paris, France, January 2005.

28. Fausto Spoto. *Julia: A Generic Static Analyser for the Java Bytecode.* In proc. of *FTfJP'05*, the 7th Workshop on Formal Techniques for Java-like Programs, Glasgow, Scotland, July 2005.

29. Stefano Secci and Fausto Spoto, *Pair-Sharing Analysis of Object-Oriented Programs.* In proc. of *SAS'05*, the 12th International Static Analysis Symposium, volume 3672 of **Lecture Notes in Computer Science**, pages 320-335. London, United Kingdom, September 2005.

30. Fausto Spoto, *Information Flow is Linear Refinement of Constancy.* In proc. of *ICTAC'05*, the International Colloquium on Theoretical Aspects of Computing, volume 3722 of **Lecture Notes in Computer Science**, pages 351–365. Hanoi, Vietnam, October 2005.

31. Fausto Spoto, Patricia M. Hill and Étienne Payet. *Path-Length Analysis for Object-Oriented Programs.* Presented at *EAAI'06*, the 1st International Workshop on Emerging Applications of Abstract Interpretation, March 2006. Vienna, Austria

32. Stefano Rossignoli and Fausto Spoto, *Detecting Non-Cyclicity by Abstract Compilation into Boolean Functions.* In proc. of *VMCAI'05*, the 7th International Conference on Verification, Model Checking and Abstract Interpretation, volume 3855 of **Lecture Notes in Computer Science**, pages 95-110. Charleston, South Carolina, USA, January 2006.

33. Étienne Payet and Fausto Spoto. *Magic-Sets Tranformation for the Analysis of Java Bytecode.* In proc. of *SAS'07*, Static Analysis Symposium, August 2007, Kogens Lyngby, Denmark, volume 4634 of **Lecture Notes in Computer Science**, pages 452-467

34. Samir Genaim and Fausto Spoto. *Constancy Analysis.* In Marieke Huisman, editor, *FTfJP'08*, 10th Workshop on Formal Techniques for Java-like Programs, Paphos, Cyprus, July 2008.

35. Fausto Spoto. *Nullness Analysis in Boolean Form.* In proc. of *SEFM'08*, the 6th IEEE International Conference on Software Engineering and Formal Methods, Cape Town, South Africa, November 2008, pages 21-30, **IEEE Computer Society Press**.

36. Étienne Payet and Fausto Spoto. *Experiments with Non-Termination Analysis for Java Bytecode.* In Samir Genaim and Elvira Albert editors, *Bytecode'09*, the 4th International Workshop on Bytecode Semantics, Verification, Analysis and Transformation, York, UK, March 2009. In **Electronic Notes in Theoretical Computer Science**, 253(5), 83-96.

37. Fausto Spoto, Lunjin Lu and Fred Mesnard. *Using CLP Simplifications to Improve Java Bytecode Termination Analysis*. In Samir Genaim and Elvira Albert editors, *Bytecode'09*, the 4th International Workshop on Bytecode Semantics, Verification, Analysis and Transformation, York, UK, March 2009. In **Electronic Notes in Theoretical Computer Science**, 253(5), 129–144.

38. Fausto Spoto. *The Nullness Analyser of Julia*. In Edmund M. Clarke and Andrei Voronkov editors, *LPAR'10*, the 16th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning, Dakar, Senegal, April 2010. Volume 6355 of **Lecture Notes in Computer Science**, pages 405-424.

39. Fausto Spoto and Michael D. Ernst. *Inference of Field Initialization*. In *ICSE'11*, the 33rd ACM International Conference on Software Engineering, pages 231–240. Honolulu, Hawaii, 2011.

40. Etienne Payet and Fausto Spoto. *Static Analysis of Android Programs*. In *CADE'11*, the 23rd International Conference on Automated Deduction. Volume 6803 of **Lecture Notes in Computer Science**, pages 439–445. Wroclaw, Poland, 2011.

41. Đurica Nikolić and Fausto Spoto. *Automaton-based Array Initialization Analysis*. In *LATA'12*, the 6th International Conference on Language and Automata Theory and Applications. Volume 7183 of **Lecture Notes in Computer Science**, pages 420–432. A Coruña, Spain. 2012.

42. Đurica Nikolić and Fausto Spoto. *Definite Expression Aliasing Analysis of Program Variables*. In *ICTAC'12*, the 9th International Colloquium on Theoretical Aspects of Computing, volume 7521 of **Lecture Notes in Computer Science**, pages 74–89. Bangalore, India. 2012.

43. Đurica Nikolić and Fausto Spoto. *Reachability Analysis of Program Variables*. In *IJCAR'12*, the 6th International Joint Conference on Automated Reasoning. Volume 7364 of **Lecture Notes in Computer Science**, pages 423–438. Manchester, UK. 2012.

44. Alberto Lovato, Damiano Macedonio and Fausto Spoto. *A Thread-Safe Library for Binary Decision Diagrams*. In *SEFM'14*, the 12th International Conference on Software Engineering and Formal Methods. Volume 8702 of **Lecture Notes in Computer Science**, pages 35–49. Grenoble, France, 2014.

45. Etienne Payet and Fausto Spoto. *An Operational Semantics for Android Activities*. In *PEPM'14*, the ACM SIGPLAN 2014 workshop on Partial Evaluation and Program Manipulation, pages 121–132. San Diego, CA, 2014.

46. Michael D. Ernst, Alberto Lovato, Damiano Macedonio, Ciprian Spiridon and Fausto Spoto. *Boolean Formulas for the Static Identification of Injection Attacks in Java*. In *LPAR'15*, the 20th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning. Suva, Fiji, November 2015. Volume 9450 of **Lecture Notes in Computer Science**, pages 130-145.

47. Michael D. Ernst, Alberto Lovato, Damiano Macedonio, Fausto Spoto, Javier Thaine. *Locking Discipline Inference and Checking*. In *ICSE'16*, the 38th International Conference on Software Engineering. Austin, TX, USA, May 2016, pages 1133-1144. ACM.

48. Michael D. Ernst, Damiano Macedonio, Massimo Merro, Fausto Spoto. *Semantics for Locking Specifications*. In *NFM'16*, the 8th International NASA Symposium on Formal Methods. Minneapolis, MN, USA, June 2016. Volume 9690 of **Lecture Notes in Computer Science**, pages 355-372.

49. Fausto Spoto. *The Julia Static Analyzer for Java*. In *SAS'16*, the 23rd International Symposium on Static Analysis. Edinburgh, UK, September 2016. Volume 9837 of **Lecture Notes in Computer Science**, pages 39-57.

50. Pietro Ferrara, Elisa Burata, Fausto Spoto. *Security Analysis pf the OWASP Benchmark with Julia*. In 1st Italian Conference on Cybersecurity (ITASEC'17), Venice, Italy, January 17-20, 2017. Volume 1816 of CEUR Workshop Proceedings, pages 242–247.

Editing

51. Fausto Spoto (ed.). *Proceedings of the First Workshop on Bytecode Semantics, Verification, Analysis and Transformation* (Bytecode'05). Volume 141, Issue 1 of **Electronic Notes in Theoretical Computer Science**, pages 1-274 (5 December 2005). Edinburgh, UK, April 2005.

52. Marieke Huisman and Fausto Spoto (eds). *Proceedings of the Second Workshop on Bytecode Semantics, Verification, Analysis and Transformation* (Bytecode'07). Volume 190, Issue 1 of **Electronic Notes in Theoretical Computer Science**, pages 1-160 (31 July 2007). Braga, Portugal, March 2007.

53. Fausto Spoto (ed). *Proceedings of the Software Verification track* of the 23rd Annual ACM Symposium on Applied Computing (SV'08). Pages 326-394. **ACM Press**. Fortaleza, Brazil, March 2008.

Organisation of Scientific Events

- Organizing committee member of *SAS'98* (Static Analysis Sysposium) and *PLILP/ALP'98* (Programming Languages, Implementations, Logics and Programs/Algebraic and Logic Programming), Pisa, Italy, 1998
- Organizing committee chair of *AMiLP'03* (Algebraic Methods in Language Processing), Verona, Italy, 2003
- Program committee member of *WLPE'03* (Workshop on Logic Programming Environments), Mumbai, India, 2003
- Workshops coordinator of *ICLP'04* (International Conference on Logic Programming), Saint-Malo, France, 2004
- Workshops coordinator and local organisation chair of *LOPSTR'04+PEPM'04+PPDP'04+SAS'04* (International Symposium on Logic-based Program Synthesis and Transformation, ACM SIGPLAN 2004 Symposium on Partial Evaluation and Semantics Based Program Manipulation, ACM-SIGPLAN International Conference on Principles and Practice of Declarative Programming and Static Analysis Symposium), Verona, 2004
- Programme committee member of *CILC'04* (Convegno Italiano di Logica Computazionale), Parma, 2004
- Workshop chair of *Bytecode'05*, First Workshop on Bytecode Semantics, Verification, Analysis and Transformation, Edinburgh, Scotland, UK, 2005.
- Programme committee member of *CILC'05* (Convegno Italiano di Logica Computazionale), Rome, 2005
- Programme committee member of the software verification track of SAC'06 (ACM Symposium on Applied Computing), Dijon, Francia, 2006

- Workshop co-chair of *Bytecode'07*, Second Workshop on Bytecode Semantics, Verification, Analysis and Transformation, Braga, Portogallo, 2007
- Programme co-chair of the software verification track of SAC'07 (ACM Symposium on Applied Computing), Seoul, Corea del Sud, 2007
- Programme chair of the software verification track of SAC'08 (ACM Symposium on Applied Computing), Fortaleza, Brazil, 2008.
- Programme Committee member of *Bytecode'08*, *Third Workshop on Bytecode Semantics, Verification, Analysis and Transformation*, Budapest, Hungary, 2008.
- Programme committee member of *Bytecode'09*, *Fourth Workshop on Bytecode Semantics, Verification, Analysis and Transformation*, York, UK, 2009.
- Programme committee member of *Bytecode'10*, *Fifth Workshop on Bytecode Semantics, Verification, Analysis and Transformation*, Cyprus, 2010.
- Programme committee member of *Bytecode'12*, *Seventh Workshop on Bytecode Semantics, Verification, Analysis and Transformation*, Tallinn, Estonia, 2012.
- Programme committee member of *TAPAS'12*, *Tools for Automatic Program Analysis*, Deauville, France, 2012.
- Programme committee member of *Bytecode'13*, *Eighth Workshop on Bytecode Semantics, Verification, Analysis and Transformation*, Rome, Italy, 2013.
- Programme committee member of *LOPSTR'14*, *24th International Symposium on Logic-Based Program Synthesis and Transformation*, Canterbury, UK, 2014.
- Programme committee member of *SAS'17*, *24th International Symposium on Static Analysis*, New York City, NY, USA, 2017.

Reviewing of Scientific Papers

Beyond the workshops above where he has been programme committee member or chair, he has been contacted to review papers for the following international journals: *Information and Computation*, *Journal of Systems and Software*, *Theoretical Computer Science*, *ACM Transactions on Computational Logic*, *ACM Transactions on Programming Languages and Systems*, *Theory and Practice of Logic Programming*, *Journal of Computer Science and Technology*, *Software Practice and Experience*; and for the following international conferences: *Algebraic Methodology And Software Technology*, *European Symposium on Programming*, *International Colloquium on Automata, Languages and Programming*, *International Conference on Logic Programming*, *International Symposium on Logic-based Program Synthesis and Transformation*, *International Conference on Embedded Software*, *Mathematical Foundations of Programming Semantics*, *Principles and Practice of Declarative Programming*, *Workshop on Quantitative Aspects of Programming Languages*, *Static Analysis Symposium*, *Journal of Logical and Algebraic Methods in Programming* and *Verification, Model Checking and Abstract Interpretation*. All reviews have always been completed on the requested time.

Verona
October 10, 2017

Fausto Spoto