

The IFIP Working Conference on Verified Software: Theories, Tools, Experiments

Tony Hoare, Jayadev Misra, and N. Shankar

October 20, 2005

The IFIP Working Conference on Verified Software: Theories, Tools, and Experiments (VSTTE; <http://vstte.inf.ethz.ch>) was held at Zurich, Switzerland, during Oct 10–13, 2005. The goal of the conference was to establish a concrete plan of action for taking on Professor Tony Hoare’s proposal of verified software as a Grand Challenge for research in computing. The conference was hosted and organized by Professor Bertrand Meyer, Chair of Software Engineering, ETH Zurich, as part of the 150th anniversary celebration of this august institution, and cosponsored by IFIP Working Group 2.3. Financial support for the meeting came from ETH Zurich, the National Coordination Office for Networking and Information Technology Research through the National Science Foundation of USA, Microsoft Research (Cambridge), and the Engineering and Physical Research Council of UK.

The meeting included

- A plenary address by Professor Tony Hoare outlining the challenge.
- Eleven keynote talks by Amir Pnueli (NYU/Weizmann Institute), Wolfgang Paul (Saarbruecken), Thomas Ball (Microsoft Research), Anthony Hall (independent consultant), Tom Reps (Wisconsin), Greg Nelson (Shakti Systems), John Rushby (SRI International), Patrick Cousot (ENS Paris), Gerard Holzmann (NASA/JPL), Mathai Joseph (TRDDC/TCS), and J Strother Moore (Texas).
- Five panel discussion sessions:
 1. Challenge codes
 2. Programming Methodology
 3. Proof and Analysis Tools
 4. Dependability and Verification
 5. Milestones and Roadmap

- There were twenty seven contributed presentations covering a wide range of topics of relevance to the Verified Software challenge. These presentations were selected from seventy two submissions of position papers.
- Participation at the Working Conference was by invitation. There were over ninety participants at the workshop including the authors/coauthors of accepted position papers.

The VSTTE Working Conference yielded a good consensus on the shape of the Grand Challenge and the nature of the theories, tools, and experiments needed to achieve success in the envisioned fifteen-year time-frame. A number of relevant and interesting challenges have been identified, on varying scales and different degrees of difficulty. The more straightforward ones could be tackled immediately by existing tools, and might serve as useful pilot projects before the main work of the Challenge starts. Examples include a lightweight file system, embedded system controllers, and medical devices. The Grand Challenge activity will be centered around a digital repository containing a suite of tools, reference specifications and semantics, and challenge codes with specifications and annotations. Another focus will be the design of a formal tool bus to mediate the coarse-grained interaction between a variety of syntactic and semantic tools to support verification activity. The conference ended by setting up a number of specialist panels that will draft ideas for concrete projects and research directions which will lead to ultimate success. Panel topics include

1. Integral Verification Systems
2. Digital Repository and Challenge Codes
3. Theory
4. Correctness by Construction
5. System Certification
6. Tool Integration and Interoperability

An Advisory Board will supervise the activities of the topical panels. Attendants at the conference are encouraged to organise sessions and workshops on verified software on the next occasion that they attend a conference in one of the recognised national or international series. Such workshops encourage members of the individual communities to direct and extend and apply their distinctive research skills towards the goals of the project.

With the success of the VSTTE Working Conference, we are optimistic that the Verified Software grand challenge will stimulate advances in software specification, software design methods, programming language design and semantics, and program analysis and verification technology that can have a dramatic impact on the engineering of reliable software with reduced lifetime costs.