

### Important Dates

Submission deadline	July 25, 2008
Acceptance notification	September 7, 2008
Camera-ready deadline	September 21, 2008
Workshop day	October 19, 2008

### Program Co-Chairs

- Tudor Dumitraq, *Carnegie Mellon University, USA* (main contact)
- Danny Dig, *Massachusetts Institute of Technology, USA*
- Iulian Neamtiu, *University of California, Riverside, USA*

### Program Committee

- Ricardo Bianchini, *Rutgers University, USA*
- Gavin Bierman, *Microsoft Research, UK*
- Dilma da Silva, *IBM Research, USA*
- Stéphane Ducasse, *INRIA, France*
- Michael Ernst, *Massachusetts Institute of Technology, USA*
- Ralph Johnson, *University of Illinois at Urbana-Champaign, USA*
- Priya Narasimhan, *Carnegie Mellon University, USA*
- Manuel Oriol, *ETH Zurich, Switzerland*
- Mark E. Segal, *Laboratory for Telecommunications Sciences, USA*
- Peter Sewell, *University of Cambridge, UK*
- Robert Wisniewski, *IBM Research, USA*

### Objectives

The goal of HotSWUp is to identify cutting-edge research ideas for implementing software upgrades. Actively-used software is upgraded regularly to incorporate bug fixes and security patches or to keep up with the evolving requirements. Whether upgrades are applied offline or online, they significantly impact the software's performance and reliability. Recent studies and a large body of anecdotal evidence suggest that, in practice, upgrades are failure-prone, tedious, and expensive.

HotSWUp is an inter-disciplinary workshop, based on synergies among the domains of programming languages (e.g., as reflected at conferences such as OOPSLA or PLDI), software engineering (e.g., as reflected at ICSE or FSE) and systems (e.g., as reflected at SOSP or OSDI). By seeking contributions from both academic researchers and industry practitioners, HotSWUp aims to combine bold, novel ideas, with experience from upgrading real systems.

The topics of interest include, but are not limited to:

- Programming language / runtime system / operating system support for software upgrades.
- Improving the reliability of upgrades (e.g., support for upgrade validation and for rollback after failures).
- Support for system restructuring (e.g., evolving APIs, changes to database schemas).
- Identifying dependencies between components and guaranteeing safe interactions among mixed versions.
- Coordinating and disseminating upgrades in large-scale distributed systems.
- Tools for preparing, testing, and applying software upgrades.
- Human factors in software upgrades (e.g., usability of upgrading tools, common operator mistakes).

### Submission Guidelines

We solicit position papers on software upgrades. Preferably, submissions to HotSWUp should fall into one of the following categories:

- Suggest how a successful approach can be applied in a different context (e.g., static dependency-analysis applied to distributed-system upgrades).
- Refute an old assumption about software upgrades (e.g., by presenting negative results).
- Describe a new problem or propose a novel solution to an old problem.
- Present empirical evidence related to the practical implementation of software upgrades.

Papers must not exceed 5 pages, in the ACM SIGPLAN 10 point format, and must be submitted electronically at <http://www.hotswup.org>. The workshop proceedings will be published in the ACM Digital Library.