SE-CSE 2008: The First International Workshop on Software Engineering for Computational Science and Engineering

Jeffrey C. Carver
Mississippi State University
329 Butler Hall, Mississippi State, MS 39759 USA
+1-662-325-0004
carver@cse.msstate.edu

ABSTRACT
Computational Science and Engineering (CS&E) software supports a wide variety of domains including nuclear physics, crash simulation, satellite data processing, fluid dynamics, climate modeling, bioinformatics, and financial modeling. The recent increase in the importance of this type of software motivates the need to better understand how it is developed. This movement creates an opportunity for the software engineering community to apply our techniques and knowledge to a new and important application domain. Furthermore, the design, implementation, and maintenance used in CS&E software systems can be significantly different from that used in systems more typically studied by the software engineering community. This workshop brings together researchers from the software engineering community with researchers and practitioners from the CS&E community. The workshop allows participants to share perspectives and present findings from research and practice that are relevant to the CS&E application development. A significant portion of the workshop is devoted to discussion of the position papers with the goal of generating a research agenda to improve tools, techniques, and experimental methods for CS&E software engineering in the future.

Categories and Subject Descriptors
D.2.0 [**Software Engineering:General**]; D.2.9 [**Software Engineering: Management**], J.2 [**Computer Applications: Physical Sciences and Engineering**]

General Terms
Management, Design, Economics, Human Factors

Keywords
Computational Science, Computational Engineering, Software Process

1. INTRODUCTION
An important type of software that has received little attention from software engineering researchers is software developed for computational science and engineering (CS&E) applications. This software is vital for the study of many important topics from diverse application domains. A list of the top 500 supercomputers (http://www.top500.org), for which many CS&E applications are written, provides an example of the diversity of government, scientific, and commercial organizations that use CS&E and highlights its growing prevalence and impact on modern society. As an example of the importance and diversity of the types of problems addressed through CS&E, a recent article in the Computing Research News listed topics being addressed with CS&E at Los Alamos National Laboratory, including designing and maintain nuclear weapons, simulation of the public infrastructure, climate change, HIV vaccines, defense against radiological attacks, and astrophysics. In addition to these topics, other institutions are using CS&E to study problems related to crash simulation, satellite data processing, bioinformatics, and financial modeling. Finally, while much of the CS&E software is developed for supercomputers, there are important CS&E applications that are created for stand alone computers. Because many of these domains are complex and involve advanced scientific or engineering concepts, much of the CS&E software is written by domain experts rather than by software engineers.

Furthermore, the design, implementation, development, and maintenance of CS&E applications can differ in significant ways from the systems and development processes more typically studied by the software engineering community:

- The requirements often include conformance to sophisticated mathematical models. Therefore, the requirements may take the form of an executable model in a system such as Matlab, with the implementation involving porting to proper platform.
- Often these projects are exploring unknown science, which makes it difficult to determine a concrete set of requirements *a priori*.
- The software development process, or "workflow" for CS&E application development may differ profoundly from traditional software engineering processes. For example, one scientific computing workflow, dubbed the "lone researcher", involves a single scientist developing a system to test a hypothesis. Once the system runs correctly and returns its results, the scientist has no further need of the system. This approach contrasts with more typical software engineering lifecycle models, in which the useful life of the software is expected to begin, not end, after the first correct execution.

- CS&E applications often require more computing resources than are available on a typical workstation. Existing solutions for providing more computational resources (e.g., clusters, supercomputers, grids) can be difficult to use, resulting in additional software engineering challenges.
- "Usability" in the context of CS&E application development may revolve around optimization to the machine architecture so that computations complete in a reasonable amount of time. The effort and resources involved in such optimization may exceed initial development of the algorithm.

2. WORKSHOP GOALS

The goal of this workshop is to facilitate the collaboration between software engineering researchers and CS&E researchers/practitioners. This workshop provides a forum for each side to present issues of relevance to CS&E software development. The workshop will be organized to allow participants to share perspectives and present findings from research and practice that are relevant to the CS&E application development. A significant portion of the workshop is devoted to discussion of the position papers with the goal of generating a research agenda to improve tools, techniques, and experimental methods for CS&E software engineering in the future. By bringing together these different groups our goal is to support the building of a common understanding of the issues involved in this complex process. An output of the workshop will be a report summarizing the conclusions agreed upon during the workshop.

The position papers submitted by the participants address issues including but not limited to:

- Case studies of software development processes (workflows) used in CS&E applications.
- Measures of software development productivity appropriate to CS&E applications.
- Activity and Purpose-based benchmarks for evaluating existing or proposed CS&E architectures
- Software engineering metrics and tool support for CS&E applications.
- The design of empirical studies to better understand the environment, tools, languages, and processes used in CS&E application development and how they might be improved.

The position papers allow a broad range of participants from both software engineering and CS&E communities to be represented at this workshop. Segments of the CS&E community who have not participated in previous workshops have been specifically targeted and invited to submit papers, for example bioinformatics and scientists working in non-supercomputer domains. This workshop was advertised on various mailing lists and contacts in both communities to attract a diverse, interdisciplinary group of participants to maximize the benefits to both communities.

3. RELEVANCE OF THE WORKSHOP

The SE-CSE workshop represents the kind of interdisciplinary work that is critical to the continued vitality of software engineering as a discipline. This workshop builds on the success of three previous ICSE workshops. While, the earlier workshops focused on Software Engineering for High Performance Computing Applications (SE-HPC), this workshop seeks to change the audience by shifting the focus to the broader CS&E community rather than only the HPC community. The first SE-HPC workshop, held at ICSE-2004 in Edinburgh, was a great success and participants from both the SE and HPC communities indicated an interest in continuing dialogue and more collaboration. The initial workshop was followed up with an equally successful workshop held at ICSE-2005 in St. Louis. The third workshop in the series was held at ICSE 2007 in Minneapolis. This workshop had a nice cross-section of participation from the SE and HPC application community. It laid the groundwork for future workshops to continue the dialogue between the communities. With this edition of the workshop being held in Europe, we expect to increase the participation of researchers that were not able to attend previous workshops.

4. WORKSHOP PLAN

This workshop will follow a similar organization to the successful SE-HPC 2007 workshop from which it has evolved. A portion of the day (approximately half) is devoted to short presentations of position papers and invited keynote speakers. The position papers and invited speakers are grouped into topically coherent sessions. After each session, time is allowed to discuss the material that has been presented.

The second portion of the workshop is devoted entirely to discussion. Using the presentations as a framework, the participants of the workshop are divided into smaller break-out groups to discuss specific topics relevant to the earlier presentations. The break-out sessions are organized either by topic of discussion or by background (software engineering vs. CS&E). At the conclusion of the break-out sessions, each group reports back to the larger group. The goal of these reports and associated discussion is to arrive at an agreed upon agenda of important research problems that must be addressed in the area of software engineering for CS&E.