SE-CSE 2009: The Second International Workshop on Software Engineering for Computational Science and Engineering

Jeffrey C. Carver
University of Alabama
carver@cs.ua.edu

Abstract

This workshop is concerned with the development of Computational Science & Engineering (CS&E) software. This software includes: 1) Scientific software applications, where the focus is on directly solving scientific problems, including, but not limited to, large parallel models/simulations of the physical world (high performance computing systems); and 2) Applications that support scientific endeavors, including, but not limited to, systems for managing and/or manipulating large amounts of data. Despite its importance in our everyday lives, the development of CS&E software has historically attracted little attention from the software engineering community. Due to significant differences in the development context, CS&E software development needs to be studied in its own right. This workshop will devote approximately equal time to presentation of position papers and to discussing topics that arise out of those presentations.

1. Introduction

This workshop is focused on the development of various types of software applications, including:

- Scientific software applications, where the goal is directly solving scientific problems. These applications include, but are not limited to, large parallel models/simulations of the physical world using high performance computing systems.
- Applications that support scientific endeavors. Such applications include, but are not limited to, systems for managing, manipulating or visualizing large data sets.

A particular software application might fit into both categories (for example, a weather forecasting system might both run climatology models and produce visualizations of big data sets) or just one (for example, nuclear simulations fit into the first category and laboratory information management software into the second). For brevity, we refer to both categories under the umbrella title of *Computational Science and Engineering* (CS&E).

Despite its importance in our everyday lives, the development of CS&E software has historically attracted little attention from the software engineering community. Indeed, the development of CS&E software is significantly different than the development of business information systems, from which many of the software engineering best practices, tools and techniques have been drawn. These differences include:

- CS&E projects are often exploring unknown science, making it difficult to determine a concrete set of requirements *a priori*.
- For the same reason, a test oracle may not exist (for example, the physical data needed to validate a simulation result may not exist). The lack of an oracle clearly poses challenges to the development of a testing strategy.
- The development process for CS&E applications 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.

- CS&E developers may have no formal knowledge of software engineering tools and techniques and may be developing software in a very isolated fashion. For example, it is common for a single scientist in a lab to take on the (formal or informal) role of software developer and to have to rely solely on web resources to acquire the relevant development knowledge.

Recent endeavors to bring the software engineering and CS&E communities together include two special issues of IEEE Software (July/August 2008 and January 2009) and this current ICSE workshop series. The 2008 edition of the workshop (http://www.ua.edu/~SECSE08) brought together computational scientists, software engineering researchers and software developers to explore issues such as:

- The characteristics of CS&E software development that distinguish it from general business software development;
- The different contexts in which CS&E developments take place;
- The quality goals of CS&E software;
- How the perceived chasm between the CS&E and software engineering communities might be bridged.

This 2009 workshop will build on the results of the previous workshops.

2. Workshop Goals

The goal of this workshop is to facilitate the collaboration between software engineering researchers and CS&E researchers and practitioners. This workshop provides a venue for those parties to meet and discuss issues that are relevant to everyone. The workshop is devoted to 1) discussion of the submitted position papers and 2) continuing the discussions that began during previous workshops (as described in Section 1).

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

- Case studies of software development processes used in CS&E applications;
- Educating CS&E developers on appropriate software development practices;
- Lessons learned from the development of CS&E applications;
- Software engineering metrics and tool support for CS&E applications;
- The use of empirical studies to better understand the environment, tools, languages, and processes used in CS&E application development and how they might be improved;

By soliciting position papers from a researchers and practitioners with a broad range of expertise, this workshop will provide a broad base for discussion of the important issues facing CS&E software development.

3. Relevance of the Workshop

The SE-CSE workshop series represents the kind of interdisciplinary work that is important for maintaining the relevancy of software engineering as a discipline. This workshop builds on the success of four previous ICSE workshops. The first three focused specifically on software engineering as it relates to high-performance computing applications. The previous workshop and this workshop focus more broadly on computational science and engineering whether or not a high-performance computer is involved.

4. Workshop Plan

This workshop follows a similar organization that was successful in the previous editions of the workshops. Approximately half of the day is devoted to the discussion of the accepted position papers. These presentations are highly interactive and geared towards generating topics that require further discussion. Then, in the second half of the workshop, those topics that the attendees find to be important will be discussed in more detail.

5. Organization

The workshop website, on which the position papers and presentations will be posted, can be accessed at http://www.cs.ua.edu/~SECSE09/. The organizing committee for the workshop includes:

- Jeffrey Carver, *University of Alabama*, USA (chair of the organizing committee)
- Steve Easterbrook, *University of Toronto*, Canada
- Tom Epperly, *Lawrence Livermore National Laboratory*, USA
- Michael Heroux, *Sandia National Laboratories*, USA
- Lorin Hochstein, *USC-ISI*, USA
- Diane Kelly, *Royal Military College of Canada*
- Chris Morris, *Daresbury Laboratory*, UK
- Judith Segal, *The Open University*, UK
- Greg Wilson, *University of Toronto*, Canada