

Chapter 15

Development of a Mesh Generation Code with a Graphical Front-End: A Case Study

Jeffrey Carver
University of Alabama, USA

ABSTRACT

Scientists and engineers are increasingly developing software to enable them to do their work. A number of characteristics differentiate the software development environment in which a scientist or engineer works from the development environment in which a more traditional business/IT software developer works. This paper describes a case study, specifically about the development of a mesh-generation code. The goal of this case study was to understand the process for developing the code and identify some lessons learned that can be of use to other similar teams. Specifically, the paper reports on lessons learned concerning: requirements evolution, programming language choice, methods of communication among teammates, and code structure.

INTRODUCTION AND BACKGROUND

The development of software applications for use in Computational Science and Engineering (CSE) is an important and growing segment of the soft-

ware development landscape. The development of CSE software differs from the development of the typical types of software that software engineering researchers study (Carver, Kendall, Squires, & Post, 2007; Carver, 2009; Kelly, 2007; Segal, 2008; Wilson, 2006). These differences mean that

DOI: 10.4018/978-1-4666-2059-9.ch015

many existing software engineering practices are not directly applicable within the CSE domain. Some of these characteristics are:

- The requirements discovery and gathering process is often difficult due to the large number of unknowns at the beginning of a project;
- Achieving the correct scientific or engineering output is more important to the developers than using correct or appropriate software engineering techniques; and
- Even when they are aware of traditional software engineering processes, which is not common, developers tend to be averse to using those processes.

Most software engineering research has been conducted in a business-oriented environment rather than a research-oriented environment. It is not clear whether the results of this research and the lessons learned about the usefulness of various techniques will translate to the CSE environment. To better understand the impact of the CSE environment and to document successful CSE software development practices, the author, along with other colleagues, as part of the DARPA High Productivity Computing Systems (HPCS) project (<http://www.highproductivity.org>), have conducted a series of case studies of successful CSE projects. To protect the anonymity of the projects, we assign each one a bird pseudonym. This paper describes the seventh case study in the series after Falcon (Post, Kendall, & Whitney, 2005), Hawk (Kendall et al., 2005), Condor (Kendall, Mark, Post, Squires, & Halverson, 2005), Eagle (Kendall, Post, Squires, & Carver, 2006), Nene (Kendall, Post, & Mark, January, 2007), and Osprey (Kendall et al., 2008) which will be referred to as the HARRIER project.

The common objectives for all of the case studies in the series are to (Carver et al., 2007):

- provide feedback to software vendors about issues that should be addressed to improve productivity;
- develop a body of case studies of CSE projects that can serve as a guide to the community; and
- document lessons learned about CSE software development to provide a resource for other similar teams.

The case study approach, as defined by Yin, is a well-understood method for gathering this type of information (Yin, 2002). The case studies, including the one reported in this paper, all followed a similar format that has proved successful. To provide context for this paper, a brief summary of the methodology is provided. The complete methodology is described in a previous publication (Carver et al., 2007). After identifying a suitable project, members of the code team complete a survey to provide background information about the project. Using this information, the researcher develops a set of questions that will be used to explore interesting topics further during an on-site interview. The researcher conducts the on-site interview with the team to gather more detailed information about the software development processes. Finally, using information from the survey and interview, the researcher writes a report that is circulated with the development team to ensure its accuracy.

Specifically for HARRIER, the project was identified through collaborations of the author. After identifying the project, the project manager completed the background survey. Based on the responses, the on-site interview guide was developed to elicit more details about some important topics. Table 1 lists the topics covered by the pre-interview survey and the on-site interview. The on-site interview lasted approximately two hours. The interviewees included: The project manager, all members of the core development team, and

13 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/development-mesh-generation-code-graphical/69623

Related Content

A Study on the Influence of Technology Products Introduced Into Green Hotels

Chih-Hung Pai, Yunfeng Shang, Long Wang and Yin Zhang (2023). *Journal of Organizational and End User Computing* (pp. 1-20).

www.irma-international.org/article/a-study-on-the-influence-of-technology-products-introduced-into-green-hotels/325649

Robust Security With Strong Authentication in Mobile Cloud Computing Based on Trefoil Congruity Framework

Jerald Nirmal Kumar S., Ravimaran S. and Sathish A. (2021). *Journal of Organizational and End User Computing* (pp. 1-28).

www.irma-international.org/article/robust-security-strong-authentication-mobile/278403

Case Study of a Patient Data Management System: A Complex Implementation in an Intensive Care Unit

Nathalie Mitev and Sharon Kerkham (2002). *Advanced Topics in End User Computing, Volume 1* (pp. 19-35).

www.irma-international.org/chapter/case-study-patient-data-management/4422

Thinking Outside of the Ballot Box: Examining Public Trust in E-Voting Technology

Susan K. Lippert and Ekundayo B. Ojumu (2010). *Computational Advancements in End-User Technologies: Emerging Models and Frameworks* (pp. 206-224).

www.irma-international.org/chapter/thinking-outside-ballot-box/38094

Adaptive Virtual Reality Museums on the Web

George Lepouras and Costas Vassilakis (2008). *End-User Computing: Concepts, Methodologies, Tools, and Applications* (pp. 731-744).

www.irma-international.org/chapter/adaptive-virtual-reality-museums-web/18218