Improving scientific software development

Luke Nguyen-Hoan

1 Introduction

Scientific software is an area which has recently gained attention in software engineering community. There exist many opportunities to improve the manner in which scientific software is developed, through both the application of existing software engineering methodologies, as well as the development of new methodologies specific to the scientific software development domain. The proposed research aims to identify areas where improvement can be made and determine, demonstrate and characterise methodologies which result in improvements in the particular domain of ecological simulation modelling.

2 Scientific software today

Scientific software has been paid increasing attention by the software research community, for example in the special issue of IEEE Software (Volume 25, Issue 4, July-August 2008) and the First International Workshop on Software Engineering for Computational Science and Engineering. The research concentrates on three main areas:

2.1 Characteristics

Figure 1 shows a large spectrum of characteristics exist within scientific software applications, as identified by Sanders in her work on the development and use of scientific software.²

2.2 Development processes

Improving software engineering development processes onto the scientific software domain has proven not to be a success in various field studies³. The sort of clash that occurs in these situations is illustrated in Figure 2 between a software engineer who expects the up-front specification of requirements and a scientist who expects the requirements to emerge as more information and experience is acquired.

2.3 Quality

The quality of scientific software is an area which can be improved. For scientific software, “validation testing is used almost exclusively.² Neither inspections, which have been shown to be the most effective quality assessment technique⁴, or formal methods, which are mathematically rigorous⁵, are widely used in scientific software development, or indeed in software development of any kind. Future work to identify quality assessment methods appropriate for scientific software has already been identified by other researchers.

2.4 People

Broadly, there are two categories of people who develop scientific software. Software professionals, and professional end user developers⁶ who are who are who have no formal training in software engineering. These professional end user developers tend to exist in an unstable community, having learnt their programming skills by “hands on experience”, or “trial and error”.⁷, ⁸ These professional end user developers are widely used in the scientific software domain. The appropriate tailoring of existing software practices to the target user is a key area for improvement.

3 Current solutions

The appropriate tailoring of existing software practices to the target domain is advocated and examples of the successful adaptation of software engineering methodologies have been shown by Ackroyd et al⁹, ¹⁰, ¹¹, Matthews et al¹², and Macaulay et al¹³. However, these studies have been limited to specific case studies and there have been no attempts to apply these proposed solutions more widely.

4 Proposed work

This research will concentrate on the ecological simulation modelling domain. This domain has been scoped for two major reasons:

1. scientific software as a whole is an extremely broad field, as can be seen from existing research; and
2. ease of collaboration with ecological simulation modellers from the Fenner School of Environment and Society at the Australian National University.

4.0.1 Research questions

1. What kind of testing, veriﬁcation, or other conﬁdence-building activities are commonly performed in the development of software that is intended for use in research? Are there different approaches to testing software for research compared to software for other purposes? Conversely, are there any software testing approaches that are not used in research but that should be?

2. What is the opinion of scientists on the quality of scientiﬁc software, as deﬁned by scientists, and what is the opinion of software engineers on the quality of scientiﬁc software, as deﬁned by software engineers? How are these opinions different, and what factors contribute to these differences?

3. What current methodologies of the software engineering domain are being used in the development of scientiﬁc software? What beneﬁts and limitations do these methodologies have in the context of scientiﬁc software development?

4. How can existing methodologies of the software engineering domain be adapted to improve the quality of scientiﬁc software development? What are the challenges and opportunities in adapting these methodologies for scientiﬁc software development?

5. What new methodologies can be developed which can improve the quality of scientiﬁc software development? What are the key factors to consider when developing new methodologies for scientiﬁc software development?

5 Expected outcomes

This research seeks to answer the research questions posed earlier. Given these answers, the following outcomes are expected:

1. Describe how the quality of scientiﬁc research is affected by the quality of scientiﬁc software produced within the ecological simulation modelling domain

2. Demonstrate some improvement in the quality of scientiﬁc software in the ecological simulation modelling domain through the use of existing and/or new software engineering knowledge

3. Define the characteristics of these improvements which make them successful

4. Show that the improvements made to the scientiﬁc software also improve the quality of scientiﬁc research in general