Abstract

In the commercial software industry, a new development paradigm has profoundly changed the process by which many software applications are produced. Test driven development (TDD) reverses the traditional role of testing in that developers first write test procedures and only then implement production code that passes those tests. Although somewhat counter-intuitive, this approach has been shown to improve developer productivity and to produce higher-quality interfaces and design.

While TDD is gaining wide acceptance in many professional software communities, numerical algorithms present a number of unique challenges that have cast some doubt on the relevance of the methodology to the development of scientific software. Such concerns range from the need for reliable error estimates for roundoff and truncation errors to the general lack of known analytic solutions for codes designed for fundamental research. Despite these challenges, TDD may be an important element of an overall strategy to improve the reliability and public trust in scientific applications such as climate models that have growing socioeconomic importance.

After a brief introduction to TDD, I will discuss some of the issues encountered in scientific software and argue that many challenges can be resolved by decomposing implementations into small routines while others represent confusion between the role of software testing and scientific validation. I will conclude with a quick discussion of my experience with using TDD with several scientific models.

Bio

Dr. Thomas Clune received his Ph.D. in physics from UC Berkeley for work on nonlinear dynamical systems. As a postdoc at the University of Colorado, he became fascinated with high performance computing during the advent of massively parallel processing. As a parallel applications analyst at NASA Goddard Space Flight Center (GSFC), Dr. Clune worked with a number of complex scientific models that span a wide variety of scientific disciplines. Now, as the Chief of the Software Systems Support Office at GSFC, he investigates the intersection of scientific modeling with modern software engineering practices.