Geosci. Model Dev. Discuss., 5, C138–C140, 2012
www.geosci-model-dev-discuss.net/5/C138/2012/

Geoscientific
Model Development
Discussions

Interactive comment on "Assessing climate model software quality: a defect density analysis of three models" *by* J. Pipitone and S. Easterbrook

T. Clune (Referee)

thomas.l.clune@nasa.gov

Received and published: 9 April 2012

This paper is of outstanding quality and I highly recommend it to anyone interested in software quality as it applies to scientific computing in general and climate modeling in particular.

The authors have provided a very broad summary of the various definitions of software quality as well as an in depth exploration of how software quality is regarded within the scientific computing community. The paper should serve very well as a starting point for scientists that are interested in this topic.

The primary novelty of this paper is that it attempts to quantitatively measure a quantity (software quality) that is generally regarded by the climate modeling community as

being either lacking sufficient hard-data or is too vague to be discussed in a useful manner. The authors make a convincing argument that defects reported in commit logs are a proxy for something instrinsic that is plausibly correlated with quality.

Although only a first step, this paper successfully brings this very important discussion out from the shadows where hopefully more climate modelers and software engineers can engage climate model quality in a constructive manner.

The authors have been painstakingly clear not just in their methodology, but in particular in acknowledging the substantial weaknesses and limitations of their methodology: sampling bias, reliance on repository log messages, measurement of code size, etc. The number and quality of references in the paper is superb.

The authors provide several alternative explanations for the somewhat surprising conclusion that climate models have impressively low defect rates. Each of the five hypotheses could prove to be a profitable direction for further investigation. However, I do believe that there is at least one additional hypotheses that could/should be considered:

Hypothesis-6: Domination by caution. Fear of introducting defects may be driving climate developers to proceed with such caution as to significantly slow development. Slow development then artificially reduces the rate of introduction of defects. In theory one might expect this to be reflected in a low rate of change for SLOC, but other development practices on the part of scientists (cut-and-paste, lack of granularity) could mean that conceptually small changes result in disproportionate source code changes.

My largest concern for this paper is that it strongly equates "quality" with "correctness". To be fair the paper is very open about this debatable association and even provides significat discussion of other definitions/metrics of software quality. I think we must learn to ask why we care about quality in the first place. One important answer is that we care about quality because it reflects the scientific accuracy of our models, and indeed this paper is largely focused on correctness for that reason. However, we also care about quality because it impacts _productivity_: i.e. how expensive is an improvement to the fidelity of a model in terms of developer time (and/or possibly computing resources)? It is important for readers to be aware that the paper does nothing to address that very important consequence of quality. My experience suggests that climate models (and generally most scientific software) have very low quality by that metric.

Again, the authors have been exceptionally clear that they are focused on correctness, and my concern should thus not be considered a substantial weakness for this paper, and instead as a driver for additional lines of research. However, it is worth pointing out that the two notions of quality do converge when considering hypothesis-6 above. I.e. low software quality from the vantage point of productivity my ironically support high software quality from the vantage point of defect rates.

Interactive comment on Geosci. Model Dev. Discuss., 5, 347, 2012.