Geoscientific
Model Development
Discussions

Interactive comment on "A multi-isotope model for simulating soil organic carbon cycling on an eroding landscape (WATEM_C v1.0)" *by* Zhengang Wang and Kristof Van Oost

David Ham

david.ham@imperial.ac.uk

This is an executive editor comment highlighting the ways in which this manuscript is not currently compliant with GMD policy on code and data availability. The issues here must be addressed before a revised manuscript can be accepted for publication:

1. Github URL. Github is an excellent development platform, but it lacks the features required of an archive. GitHub themselves tell authors to use Zenodo for this purpose. The authors should follow the procedure detailed there to archive the exact version of the software used to create the results presented: https://guides.github.com/activities/citable-code/. The resulting Zenodo repositories present the

correct bibliography entries to use.

2. No data identified. The datasets used to conduct the evaluation experiments presented must be cited from the code and data availability section with enough precision to allow a reader to reproduce the work in the manuscript.

3. No configuration, run, or data processing scripts. The configuration files, run scripts and any data processing or analysis scripts used to produce the results presented in the manuscript need to be publicly and persistently archived, and cited from the code and data availability section. As a guide, every file the user would need to reproduce the manuscript should accessible.

Further details on code and data availability requirements are in the GMD model code and data policy: https://www.geoscientific-model-development.net/about/code_and_data_policy.html. The reasons for the policy and more detail are provided in this editorial: https://doi.org/10.5194/gmd-12-2215-2019.

Use of Github

In addition to the policy compliance issues raised above, I should point out that the authors are currently not really using GitHub in the correct way. Uploading a zip file of the author's installation basically defeats the whole point of revision control. Instead, the git repository should contain the source files and build scripts for the model directly (not in a zip file) as well as the source files for the model documentation and verification tests. Small data sets used for verification could be included, but no other binary files. In particular, including the compiled windows binaries as the authors do makes life difficult for users of their code who will encounter constant conflicts with their own binaries every time they pull updates. If distributing binaries is desirable then this should be accomplished via the appropriate mechanism. See: https://help.github.com/en/github/administering-a-repository/about-releases

C3