Reproducible modelling: Why is it so hard?

D. Holzworth and **N. Huth**

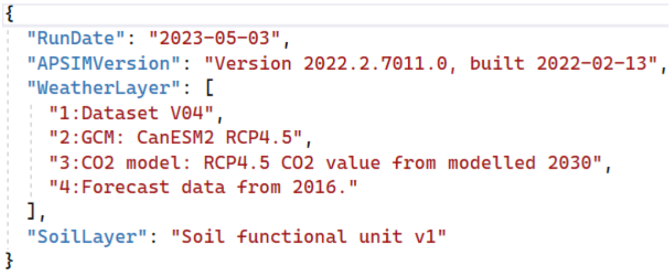
CSIRO Agriculture and Food, Australia
Email: dean.holzworth@csiro.au

Abstract: Modelling at scale involves creating workflows that connect data to tools, utilities, and models. Often this is a manual process (e.g. scripts with no automation) that evolves over time. Unless there is clear, detailed documentation, that is accessible, it can be very difficult to reproduce simulation results at some point in the future. Journal paper descriptions of simulation results are often not reproducible!

The software development industry created Docker images to very clearly define an execution environment that is reproducible. The docker user creates a simple text-based recipe (dockerfile) that installs the software application (model) and its dependencies into an image that can be executed repeatedly. If the image is pushed to a docker repository (e.g. DockerHub) then it will be accessible by others. This solves part of the reproducibility problem by encapsulating the execution environment into a sharable image. It doesn't solve the problem of identifying the model input data.

To run crop models at scale across the cropping region of Australia we created a workflow (e.g. scripts) that connect gridded future climate, soil and crop management data to APSIM (Holzworth et al., 2018) and runs various post-simulation tools to create data cubes with visualisations (e.g. maps, PDF reports). A dockerfile was created to install APSIM, python and various packages into a Docker image. This provided a reproducible runtime environment to run the workflow. To identify the input data used, we developed a simple data provenance system to capture metadata from various sources. This contained the version of APSIM, the version as reported by the future climate API and the version of the soil data. This was stored in a simple JSON file in the same directory as the model outputs (Figure 1). Finally, all files were stored in a GIT repository.

By combining the use of Docker, a GIT repository, and a simple data provenance system, we have gone a long way towards a reproducible simulation workflow. Issues remain though, in particular the need for any external data sources (e.g. the future climate API) to exist into the future. Incorporating this data into the docker image wasn't feasible due to the size of the data (1Tb).



```
{
  "RunDate": "2023-05-03",
  "APSIMVersion": "Version 2022.2.7011.0, built 2022-02-13",
  "WeatherLayer": [
    "1:Dataset V04",
    "2:GCM: CanESM2 RCP4.5",
    "3:CO2 model: RCP4.5 CO2 value from modelled 2030",
    "4:Forecast data from 2016."
  ],
  "SoilLayer": "Soil functional unit v1"
}
```

Figure 1. Example of a metadata file captured for a gridded APSIM data cube

This approach can be used for all modelling studies regardless of their size. We would like to see software / simulation journals and journal paper authors take science reproducibility more seriously. This conference paper shows a simple methodology that progresses towards this goal.

REFERENCES

Holzworth, D., Huth, N.I., Fainges, J., Brown, H., Zurcher, E., Cichota, R., Verrall, S., Herrmann, N.I., Zheng, B., Snow, V., 2018. APSIM Next Generation: Overcoming challenges in modernising a farming systems model. Environmental Modelling & Software 103, 43–51. https://doi.org/10.1016/j.envsoft.2018.02.002

Keywords: *Reproducible modelling, workflows, Docker, data provenance, GIT*