Towards Open, Transparent, and Reproducible Data-Driven Science with Whole Tale

Bertram Ludäscher, Adam Brinckman, Mary Terese Campbell, Kyle Chard, Niall Gaffney, Mihael Hategan, Matthew B. Jones, Kacper Kowalik, Michael Lambert, Bryce Mecum, Jarek Nabrzyski, Damian Perez, Victoria Stodden, Ian Taylor, Thomas Thelen, Matthew Turk, Craig Willis, Sebastian Wyngaard

Scientific results are communicated through research publications that increasingly include references to the data and computational methods used in the publication. Driven by the desire for open science and improved computational reproducibility, there is a growing trend towards sharing the data used in a paper along with the associated analysis code, e.g., by referencing a public git repository that contains the analysis scripts used. In this way, the research community is exposing details of the computational methods used, creating ever more transparent scientific studies, while at the same time allowing enhancements to and new uses of the methods and tools being shared. However, considerable challenges still remain for readers of a research paper to scrutinize or leverage research results, even if all data and code is shared. For example, it is non-trivial to install the required combination of software tools (which version of Python or R should I use?) and software libraries (which version of NumPy, SciPy, NLTK, etc. for Python; or dplyr, tidyr, ggplot2, plotly, etc. for R is needed) on a user's computer. After all, most potential users already have pre-existing software packages installed on their computers and consequently have their installation paths, environment variables etc. set to values that may conflict with the requirements of the software used in a published research study. Analytical scripts are also typically written to work against a particular system configuration, including the location of executables, file paths to data and configuration files, and other local details.

One approach to address these challenges is to move towards "living papers": These add to the scholarly narrative (i) data, (ii) code, and (iii) a "virtual computer" that allows readers and potential users to easily execute the associated code and methods in a self-contained containerized environment on a virtual machine running in a computational cloud.

Whole Tale[1] is an open-source, community-driven project, funded initially by the U.S. National Science Foundation for 5 years to develop the software building blocks and tools that allow researchers to easily develop and share *tales*, a form of living papers that can encompass a science narrative, all relevant data (directly or by reference), analysis code, and the actual execution environment, running on a virtual machine (VM) in the cloud. A tale may also include a *workflow specification* and *provenance* information that links derived output data via intermediate data and computational steps back to the inputs of the study/experiment to reveal relevant dependencies. Thus, tales are shareable, reproducible, and preservable research objects, along with a complete execution environment required for community reuse and preservation. The modular software architecture leverages and contributes back to community code, and includes containerized VMs running in cloud environments and configured through Dockerfiles. Similar approaches can be found in other projects, both commercial (CodeOcean) and open (myBinder.org). We are in discussions with these groups to employ compatible formats and standards for tale-like research objects. We believe that projects like *Whole Tale*, that aim to capture and preserve a scientist's journey towards discovery will benefit from joining forces when adopting or developing open community standards for executable research objects.

Keywords: computational reproducibility; living papers; provenance; cloud computing

References

- Whole Tale Open Source Repositories: https://github.com/whole-tale
- Whole Tale Dashboard: https://dashboard.wholetale.org

[1] https://wholetale.org/