ReproZip: The Reproducibility Packer

Rémi Rampin[1], **Fernando Chirigati**[1], **Dennis Shasha**[2], **Juliana Freire**[1], **and Vicky Steeves**[3]

1 NYU Tandon School of Engineering **2** Courant Institute of Mathematical Sciences **3** NYU Division of Libraries & Center for Data Science

Summary

ReproZip (Rampin et al. 2014) is a tool aimed at simplifying the process of creating reproducible experiments. After finishing an experiment, writing a website, constructing a database, or creating an interactive environment, users can run ReproZip to create reproducible packages, archival snapshots, and an easy way for reviewers to validate their work.

ReproZip was created to combat the problem of "dependency hell" – the pit of software libraries, inputs, configuration parameters, etc. that comprise everything necessary to run and rerun applications and computational experiments. For researchers to even begin to think about sharing their work reliably and reproducibly, they have to create a compendium of all the steps and dependencies. Doing this manually is not only a huge pain, but also ridiculously difficult and prone to human error, especially if the researcher didn't plan to do this at the beginning.

ReproZip has two steps:

1. The packing step happens in the original environment (currently, only Linux), and generates a compendium of the experiment. ReproZip tracks operating system calls while a project is executing, and creates a package (a `.rpz` file) that contains all the binaries, files, dependencies, and all other necessary information and components for reproduction. These `.rpz` files are much smaller than a virtual machine, and quite easy to share.

2. The unpacking step reproduces the experiment from the `.rpz` file. ReproUnzip offers different unpacking methods, from simply decompressing the files in a directory to starting a full virtual machine, and they can be used interchangeably from the same packed experiment. It is also possible to automatically replace input files and command-line arguments. Reviewers can unpack `.rpz` files on Linux, Windows, and Mac OS X, since ReproUnzip can unpack the experiment in a virtual machine (Vagrant or Docker). This step also has a graphical user interface option for users unfamiliar with the command line.

We have extensive documentation (Rémi Rampin, n.d.), a website visually explaining ReproZip (V. S. Fernando Chirigati Rémi Rampin 2016), and available examples (Vicky Steeves 2016) with explicit instructions on how to reproduce a subset of use cases. This repository of examples provides documentation, `.rpz` files, and also a Vagrantfile that automatically configures a machine with six of the eleven case studies in the examples repository. Users can also watch a demo video (V. S. Fernando Chirigati 2015) to better understand the execution of ReproZip. The most recent paper (Chirigati et al. 2016) on ReproZip was published this year in the proceedings of SIGMOD.
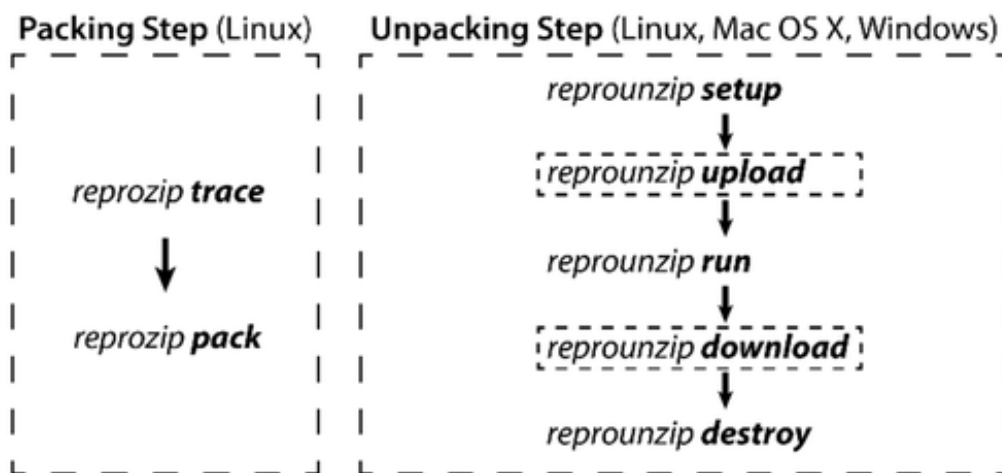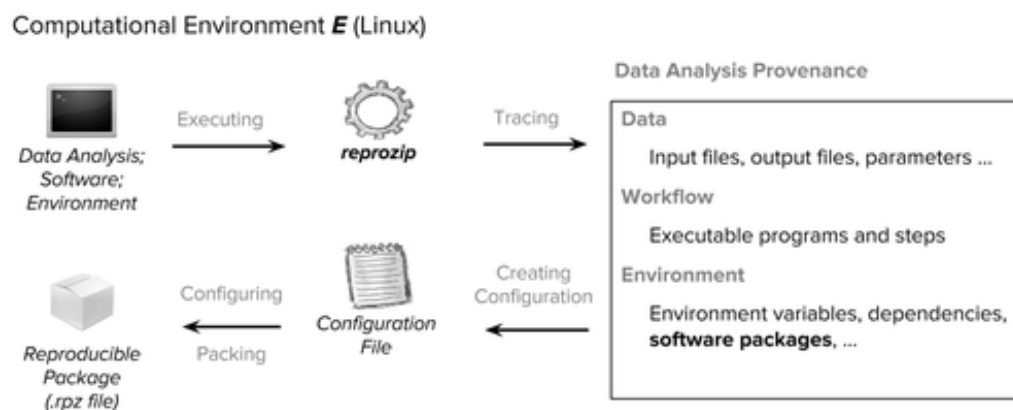
Figure 1: ReproZip Commands
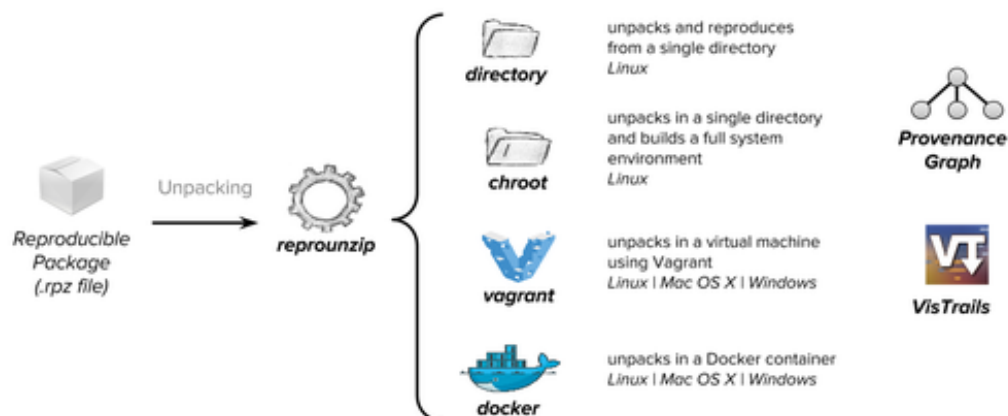

Figure 2: Step 1. Packing

Figure 3: Step 2. Unpacking

References

Chirigati, Fernando, Rémi Rampin, Dennis Shasha, and Juliana Freire. 2016. "ReproZip: Computational Reproducibility with Ease." In *Proceedings of the 2016 International Conference on Management of Data*, 2085–8. SIGMOD '16. New York, NY, USA: ACM. doi:10.1145/2882903.2899401.

Fernando Chirigati, Vicky Steeves. 2015. "Packing and Unpacking Experiments with Reprozip." Youtube. https://www.youtube.com/watch?v=-zLPuwCHXo0.

Fernando Chirigati, Vicky Steeves, Rémi Rampin. 2016. "ReproZip Website." https://vida-nyu.github.io/reprozip/.

Rampin, Rémi, Fernando Chrigiati, Vicky Steeves, Dennis Shasha, and Juliana Freire. 2014. "ReproZip: The Reproducibility Packer." doi:https://doi.org/10.5281/zenodo.159604.

Rémi Rampin, Vicky Steeves, Fernando Chrigiati. n.d. "ReproZip Documentation." https://reprozip.readthedocs.io/en/1.0.x/.

Vicky Steeves, Fernando Chirigati, Rémi Rampin. 2016. "ReproZip Examples Repository." doi:http://doi.org/10.17605/OSF.IO/JB2UV.