

ReproZip

Packing Experiments for Sharing and Publication

Fernando Chirigati, Juliana Freire | NYU-Poly

Dennis Shasha | NYU

Motivation

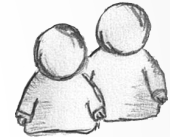
- Published articles are not made reproducible
- Computational reproducibility may be **difficult** to achieve



Author

How to encapsulate my experiment?
Too many dependencies...
Too many files to keep track...
Sigh.

How to compile this program?
How to execute it?
How to *explore* it?
Sigh.



Reviewers
Collaborators

- Some current solutions require the user to adopt a system
 - GenePattern [1], Madagascar [2], Scientific Workflow Systems [3]
- Other solutions rely on capturing information about the computational environment
 - Virtual Machines
 - CDE [4]

ReproZip

- ReproZip is a packaging solution
 - It makes it easier for *authors* to pack experiments and for *reviewers* to verify computational results
- It creates *reproducible packages* from existing experiments on computational environment E
 - No need to port experiments to other system
 - Leverages *provenance* of computational results
- It unpacks an experiment on computational environment E'
- It generates a *workflow specification* that encapsulates the execution of the experiment
 - Eases the verification process
 - Allows users to explore the experiment, while keeping track of provenance

The diagram shows a four-stage process for creating a reproducible package:

- Experiment**: Represented by a terminal icon.
- Provenance Tree**: A tree structure showing the lineage of data and code.
- Workflow**: A directed graph showing the execution flow. Inputs include `input1 (PersistentInputFile)`, `input2 (PersistentInputFile)`, and `input3 (String)`. The central node is `test_exp`. Outputs include `output1 (PersistentOutputFile)`, `stderr (StandardOutput)`, and a `VT` (Virtualization Technology) icon.
- Reproducible Package**: A box icon representing the final package, which includes `files + binaries + workflow`.

```
graph LR; A[Reproducible Package] --> B[Experiment Extraction]; B --> C[files + binaries + workflow]
```

The diagram illustrates a three-step process. It begins with a box labeled "Reproducible Package" containing a closed cardboard box icon. An arrow points to a second box labeled "Experiment Extraction" containing an open cardboard box icon. A second arrow points to a third box containing a folder icon and the text "files + binaries + workflow".

References

1. GenePattern. <http://www.broadinstitute.org/cancer/software/genepattern/>
2. Madagascar. http://www.ahay.org/wiki/Main_Page
3. S. B. Davidson and J. Freire. Provenance and scientific workflows: challenges and opportunities. In SIGMOD, pages 1345-1350, 2008
4. P. Guo. CDE: A Tool for Creating Portable Experimental Software Packages. Computing in Science and Engineering, 14(4):32-35, 2012
5. SystemTap. <http://sourceware.org/systemtap/>
6. MongoDB. <http://www.mongodb.org/>

Thank You!

Fernando Chirigati

fchirigati@nyu.edu

<http://vgc.poly.edu/~fchirigati>