Best practices for reproducible research

Vicky Steeves
Librarian for Research Data Management & Reproducibility
New York University Division of Libraries & Center for Data Science

SLIDES: goo.gl/iucFpe

BEFORE WE START!

Please download the following for the hands-on portion of the workshop:

- Vagrant - vagrantup.com/downloads.html
  - This Vagrantfile: github.com/ViDA-NYU/reprozip-examples/blob/master/Vagrantfile
- VirtualBox with extension pack - virtualbox.org/wiki/Downloads
- ReproZip - reprozip.org and in the top right corner, pick your installer

Please make a folder called `reprozip-examples` in your Dekstop or Downloads folder. Put the Vagrantfile you downloaded in that folder.

We'll be using the ReproZip demo VM to practice packing. We'll also use the Vagrant unpacker to practice unpacking ReproZip bundles!

A little bit about me first? And maybe you, too?

- Pronouns: she/hers
- 1st degree was in computer science, then librarianship!
- Openness is life (open source, OER, open access, etc)
- My job is dual appointment between NYU's Center for Data Science and Division of Libraries
- Goal: help folks work reproducibly & then preserve their work

Itinerary for Today

1. Define key terms in data management and reproducibility
2. Understand best practices in data management and reproducibility
3. Go over some tools that help us work towards reproducibility
    a. Especially ReproZip and ReproServer!

Goals for Today

1. You will gain an understanding of some basic concepts in data management & reproducibility
2. You have an idea of some of the tools available to you, to help you work towards research reproducibility
3. You will be able to at the very least know what to look up when Googling

What problem are we trying to solve today?

Research isn't being efficiently managed or made reproducible

Much of the time, the workflow & processes aren't reproducible, the findings (data, code, etc.) aren't managed efficiently, and as a result, we all suffer.

Most Scientific Research Data From the 1990s Is Lost Forever

Article in the Atlantic

A new study has found that as much as **80 percent** of the **raw scientific data** collected by researchers in the early 1990s **is gone forever**, mostly because no one knows where to find it.

You can't have any sort of reproducibility without good **data** and **project management**.

Research Data Management is...

managing the way data is collected, processed, analyzed, preserved, and published for greater reuse by the community and the original researcher

What is Data?

"the recorded factual material commonly accepted in the scientific community as necessary to validate research findings."

High Level Research Data Management

Data Type

-types of data to be generated
-format of data

Roles + Responsibilities

-who is primarily responsible for carrying out the DMP?
-if you know more than 1 person, what are the roles & responsibilities?

Data Storage

-where will you store your data?
-how will it be backed up?

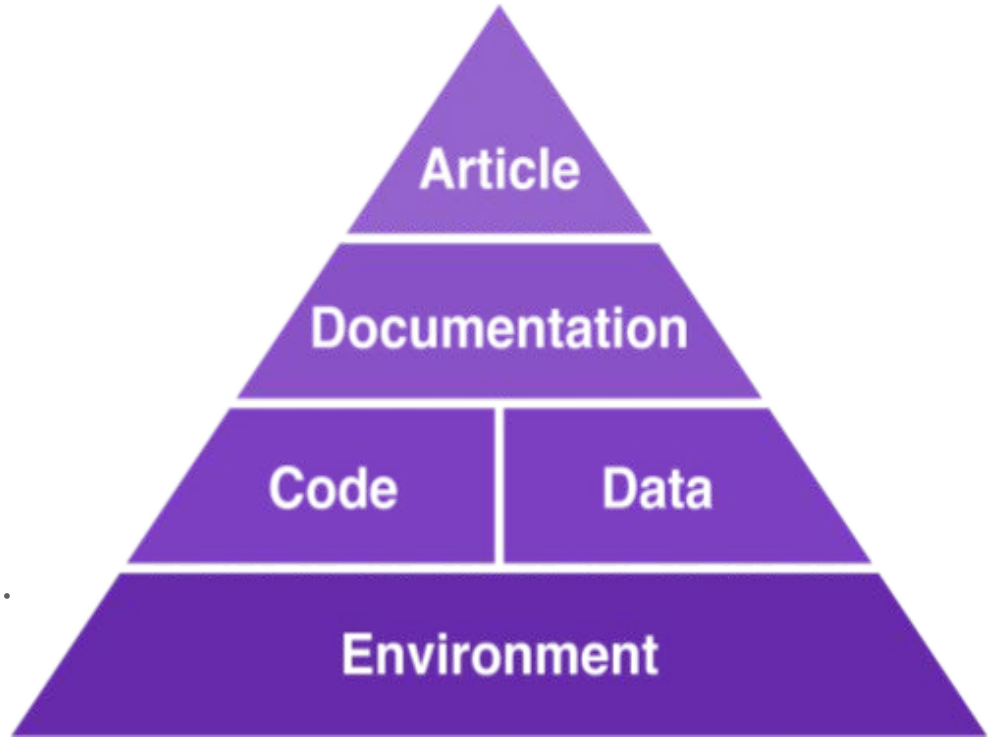
Data Preservation

-how will you preserve your data?
-how will you make your data available to others?

As all things, reproducibility is defined via spectrum

Reviewable Research: Sufficient detail for peer review & assessment.
Replicable Research: Tools are available to duplicate the author's results using their data.
Confirmable Research: Main conclusions can be attained independently without author's software.
Auditable Research: Process & tools archived such that it can be defended later if necessary.
Open/Reproducible Research: Auditable research made openly available

[Stodden et al ICERM report (2013)]



Why Reproducibility?

"If I have seen further, it is by standing on the shoulders of giants." - Sir Isaac Newton

To build on top of previous work – research is incremental!

To verify the correctness of results

To defeat self-deception[1]

To help newcomers

To increase impact, visibility[2] and research quality[3]

1. http://www.nature.com/news/how-scientists-fool-themselves-and-how-they-can-stop-1.18517
2. http://infoscience.epfl.ch/record/136640/files/VandewalleKV09.pdf
3. http://www.nature.com/nature/journal/v483/n7391/full/483531a.html

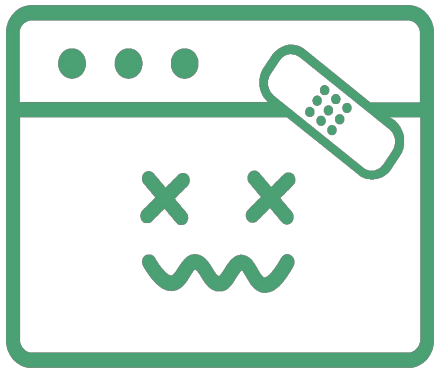
Why Reproducibility?

1. Others can re-use and extend your work more easily!
    a. You can even find interesting collaborations and future research projects out of this.

2. YOU can re-use and extend your work more easily! (sort of selfless…)
    a. Future you is your greatest collaborator.

3. Newbies to the field can more easily learn the methods by reproducing your work!
    a. Your reproducible work is their greatest teacher.

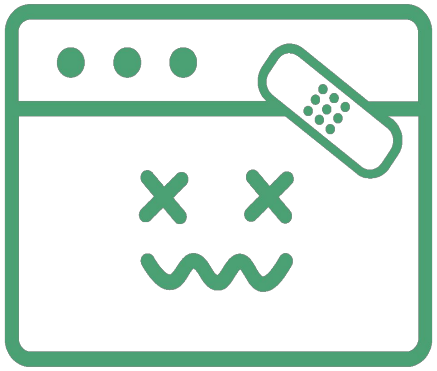
Challenges in Reproducibility



- People make mistakes--and it impacts their research
- It's good to have other people check out your data and analyses--it's like having a copy editor for your data!

- It's *hard* to keep track of what version of what was used
- Software get updates, and these changes can disrupt reproducibility

Let's talk about the human problems first...



- People make mistakes--and it impacts their research
- It's good to have other people check out your data and analyses--it's like having a copy editor for your data!

- It's *hard* to keep track of what version of what was used
- Software get updates, and these changes can disrupt reproducibility

Data doesn't live forever by itself

Current Biology 24, 94–97, January 6, 2014 ©2014 Elsevier Ltd All rights reserved http://dx.doi.org/10.1016/j.cub.2013.11.014

The Availability of Research Data Declines Rapidly with Article Age

Timothy H. Vines,[1,2,*] Arianne Y.K. Albert,[3] Rose L. Andrew,[1] Florence Débarre,[1,4] Dan G. Bock,[1] Michelle T. Franklin,[1,5] Kimberly J. Gilbert,[1] Jean-Sébastien Moore,[1,6] Sébastien Renaut,[1] and Diana J. Rennison[1]
[1]Biodiversity Research Centre, University of British Columbia,

sets (23%) were con
down of the data by
We used logistic r
tionships between t
that at least one e-m

and indeed many studies have found that authors are often unable or unwilling to share their data [8–11]. However, there are no systematic estimates of how the availability of research data changes with time since publication. We therefore requested data sets from a relatively homogenous set of 516 articles published between 2 and 22 years ago, and found that availability of the data was strongly affected by article age. For papers where the authors gave the status of their data, the odds of a data set being extant fell by 17% per year. In addition, the odds that we could find a working e-mail address for the first, last, or corresponding author fell by 7% per year. Our results reinforce the notion that, in the long term, research data cannot be reliably preserved by individual researchers, and further demonstrate the urgent need for policies mandating data sharing via public archives.

http://www.sciencedirect.com/science/article/pii/S0960982213014000

American Economic Review: Papers & Proceedings 100 (May 2010): 573–578
http://www.aeaweb.org/articles.php?doi=10.1257/aer.100.2.573

	B	C	I	J	K	L	M
2			Real GDP growth				
3			Debt/GDP				
4	Country	Coverage	30 or less	30 to 60	60 to 90	90 or above	30 or less
26			3.7	3.0	3.5	1.7	5.5
27	Minimum		1.6	0.3	1.3	-1.8	0.8
28	Maximum		5.4	4.9	10.2	3.6	13.3
29							
30	US	1946-2009	n.a.	3.4	3.3	-2.0	n.a.
31	UK	1946-2009	n.a.	2.4	2.5	2.4	n.a.
32	Sweden	1946-2009	3.6	2.9	2.7	n.a.	6.3
33	Spain	1946-2009	1.5	3.4	4.2	n.a.	9.9
34	Portugal	1952-2009	4.8	2.5	0.3	n.a.	7.9
35	New Zealand	1948-2009	2.5	2.9	3.9	-7.9	2.6
36	Netherlands	1956-2009	4.1	2.7	1.1	n.a.	6.4
37	Norway	1947-2009	3.4	5.1	n.a.	n.a.	5.4
38	Japan	1946-2009	7.0	4.0	1.0	0.7	7.0
39	Italy	1951-2009	5.4	2.1	1.8	1.0	5.6
40	Ireland	1948-2009	4.4	4.5	4.0	2.4	2.9
41	Greece	1970-2009	4.0	0.3	2.7	2.9	13.3
42	Germany	1946-2009	3.9	0.9	n.a.	n.a.	3.2
43	France	1949-2009	4.9	2.7	3.0	n.a.	5.2
44	Finland	1946-2009	3.8	2.4	5.5	n.a.	7.0
45	Denmark	1950-2009	3.5	1.7	2.4	n.a.	5.6
46	Canada	1951-2009	1.9	3.6	4.1	n.a.	2.2
47	Belgium	1947-2009	n.a.	4.2	3.1	2.6	n.a.
48	Austria	1948-2009	5.2	3.3	-3.8	n.a.	5.7
49	Australia	1951-2009	3.2	4.9	4.0	n.a.	5.9
50							
51			4.1	2.8	2.8	=AVERAGE(L30:L44)	

In this pap… g pop-
historical dat… s? Are
search for a s… a man-
public debt …
main result …pirical,
growth and c…torical
mal" debt lev…central
tries with pu…nen M.
of GDP are a…009b).
wise; averag…ifficult
percent lowe…f pub-
between pub…es, and
similar acros…arkets.
economies. T…untries
find no syst…er, the
debt levels a…vations
mies as a gr…, insti-
exceptions in…range-
trast, in emer…
debt levels c…xternal

Our topic …nments
Public debt h…arkets,
recent global financial maelstrom, especially in the epicenter countries. This should not be surprising, given the experience of earlier severe

we find that there exists a significantly more severe threshold for total gross external debt (public and private)—which is almost exclu-

Annals of Science | FEBRUARY 29, 2016 ISSUE

The Stress Test

Rivalries, intrigue, and fraud in the world of stem-cell research.

The promises of stem-cell research lie at the core of human desires—to understand our origins and to cheat death—and there is a great deal of money and prestige at stake. It is a ruthlessly competitive field, susceptible to fantasy and correspondingly sensitive to bunglers. Human embryonic stem cells were first cultured in 1998; nearly twenty years later, basic assumptions about cell

Its findings implicated both Obokata's sloppy record keeping and her mentors' lax oversight; in some instances, there were no original data to back up her figures and images, and in others no evidence that the experiments had been conducted at all.

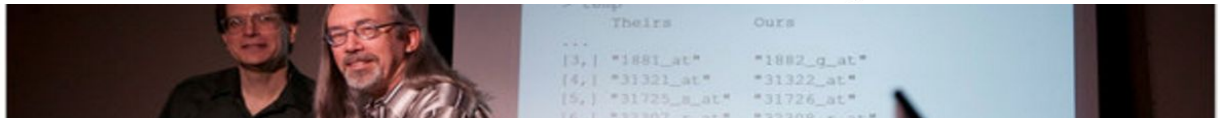
privilege—that discernment can be the difference between brilliance and quackery, and between fame and obscurity.

Five months after publication, both STAP papers were retracted, under intense scrutiny and growing doubt about their validity. By that point, Riken had cited Obokata for research misconduct and charged her mentors with "heavy responsibility"; one of those mentors had implicated her in a fraud; she had been hospitalized for depression; a co-author had suffered a stress-related stroke; and an outside committee had recommended that Riken dismantle

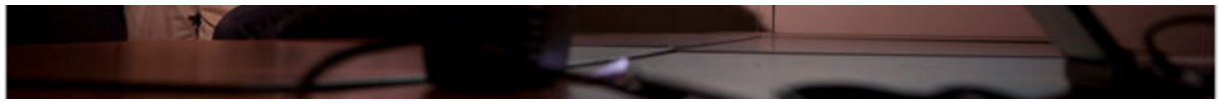
How Bright Promise in Cancer Testing Fell Apart

By GINA KOLATA JULY 7, 2011

But the research at Duke turned out to be wrong. Its gene-based tests proved worthless, and the research behind them was discredited. Ms. Jacobs died a few months after treatment, and her husband and other



Instead, as patients and their doctors try to make critical decisions about serious illnesses, they may be getting worthless information that is based on bad science. The scientific world is concerned enough that two



Doctors say the heart of the problem is the intricacy of the analyses in this emerging field and the difficulty in finding errors. Even well-respected

Email

Share

When Juliet Jacobs found out she had lung cancer, she was terrified, but realized that her hope lay in getting the best treatment medicine could offer. So she got a second opinion, then a third. In February of 2010, she ended up at Duke University, where she entered a research study whose promise seemed stunning.

http://www.nytimes.com/2011/07/08/health/research/08genes.html

We have a culture barrier

Workload & Time Challenges
It is a time commitment to get data and code ready to share, and to share it

Normative Dissonance[1]
Espoused values don't always match practice

Otherwise known as…

the Incentive Problem
Reproducibility takes time, and is not always valued by the academic reward structure

"Insufficient time is the main reason why scientists do not make their data and experiment available and reproducible."
Carol Tenopir, Beyond the PDF2 Conference

"77% claim that they do not have time to document and clean up the code."
Victoria Stodden, Survey of the Machine Learning Community – NIPS 2010

Project Structure

Put each project in its own directory, which is named after the project.

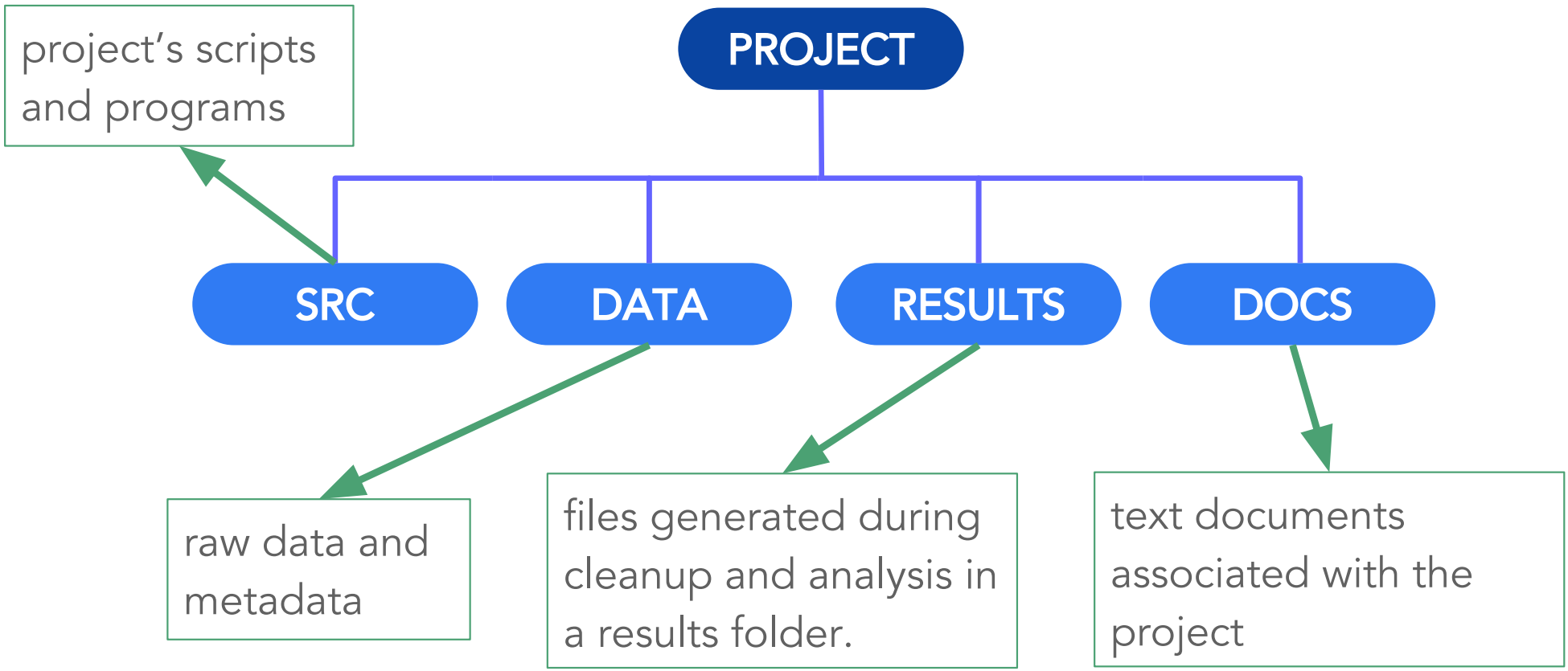
Put text documents associated with the project in the `doc` folder.

Put raw data and metadata in the `data` folder, and files generated during cleanup and analysis in a `results` folder.

Put source for the project's scripts and programs in the `src` folder.

Name all files to reflect their content or function, with NO special characters (!@#$%^*) or spaces! Use underscores or dashes, A-Z, and numbers

Project Structure



Best Practices: Documentation

Methodology

We are collecting data from 30 women ages 18-25 about their sexual histories through individual interviews.

We will analyze this data using XYZ software and XYZ analytical framework.

take note of changes to this as the project continues

Data Collection

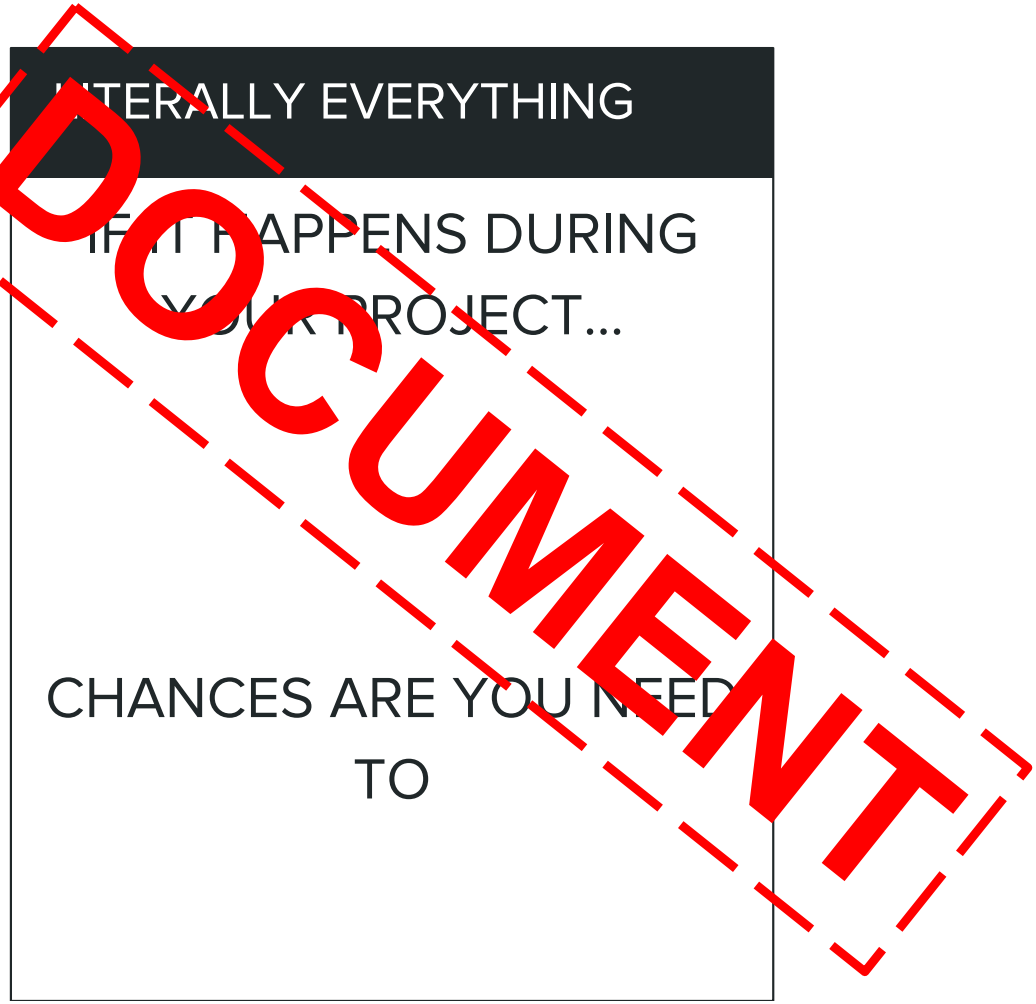
We will use the Open Science Framework to document our data collection process.

"Subject CYZ was intereviewed in my office at 70 Washington Sq South from 1-3pm. The recording file is located in 2015/PROJ/INTERVIEWS"

Variables Names

Variable Name: **employ_lev**
Description: A derived variable based on the percentage of a given economic development area employed in full time work. Expressed as the value of the variable **employ** divided by the number of work-eligible adults resident in that district as listed in the 1980 census.

Best Practices: Documentation



Literate programming as RDM - IPYNB

Web Application

- in browser editing for code with auto-syntax highlighting, indentation, tab completion/introspection
- in browser code execution, with results attached to the code that generated them
- easily include math markdown using LaTeX
- display results of computation in rich media (LaTeX, HTML, SVG, etc.)

Notebook

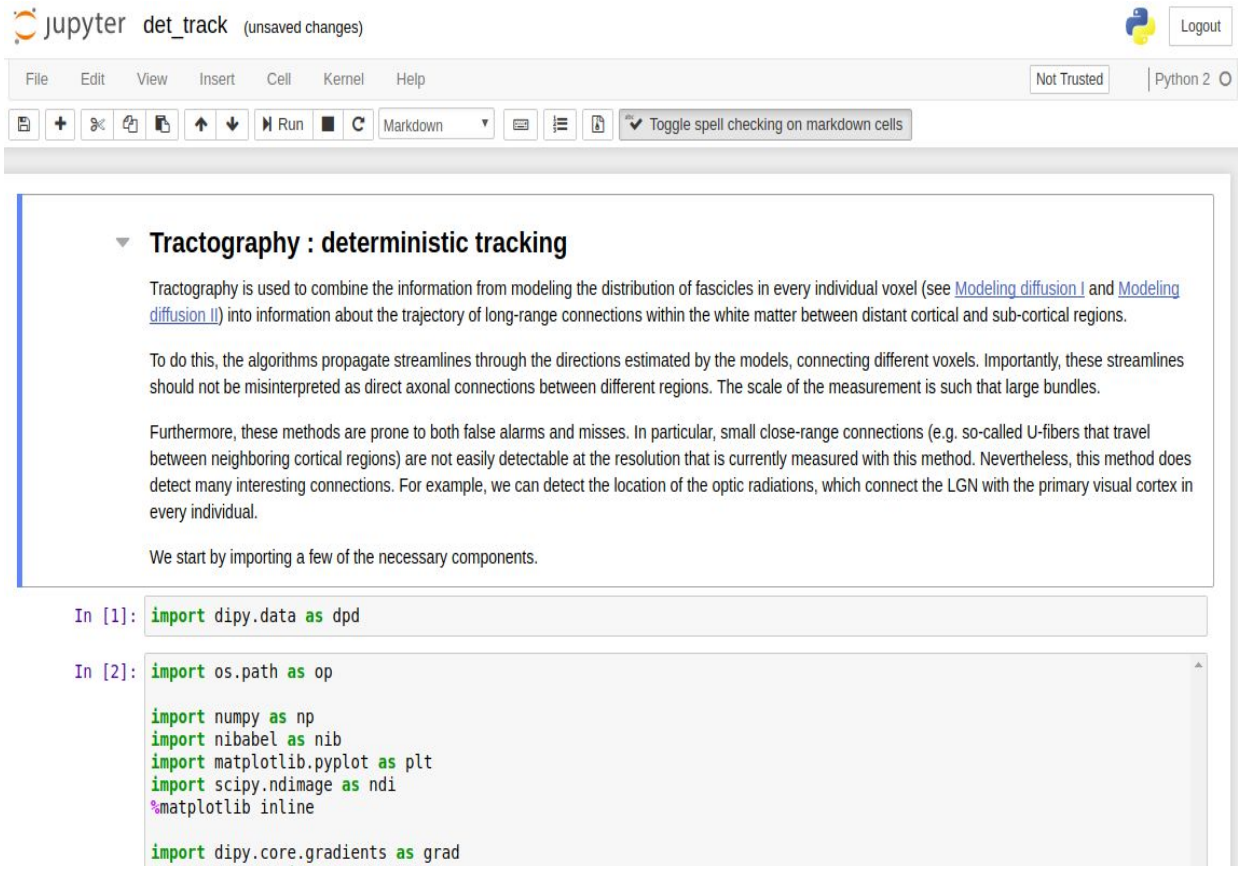
- a complete computation record of a session, interleaving executable code with text, maths, and rich representations of objects
- can export to LaTeX, PDF, slideshows, etc. or available from a public URL that renders it as a static webpage

Jupyter Notebooks - code + documentation

Jupyter notebooks lets you interweave your analysis with some documentation!

This creates what some call an "executable paper"

However, the same problems occur where computing environments differ, so you have to take extra steps to make these reproducible



https://github.com/arokem/visual-white-matter

Jupyter Notebooks + Binder for Reproducibility

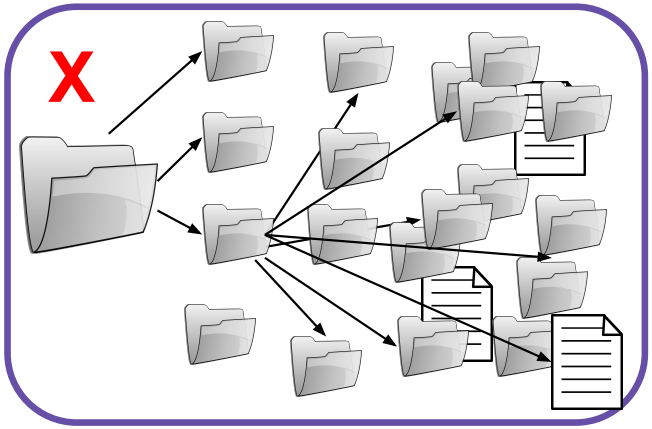
Have a GitHub repository full of Jupyter notebooks? With Binder, open those notebooks in an executable environment, making your code immediately reproducible by anyone, anywhere.

1. Enter your repository information (a URL to a GitHub repo with Jupyter Notebooks)
2. Binder builds a Docker image of your repository using a requirements.txt file from the repository.
3. Interact with your notebooks in a live environment! JupyterHub server will host your repository's contents. We offer you a reusable link and badge to your live repository that you can easily share with others.

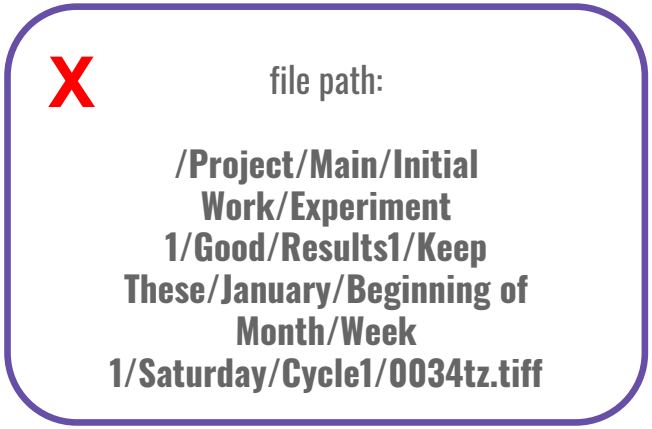
E.g.: https://mybinder.org/v2/gh/TiesdeKok/LearnPythonforResearch/master

Best Practices: Documenting Files

AVOID TANGLED FOLDER NESTS



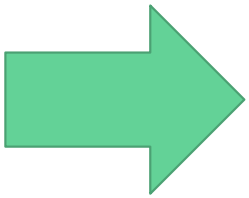
X

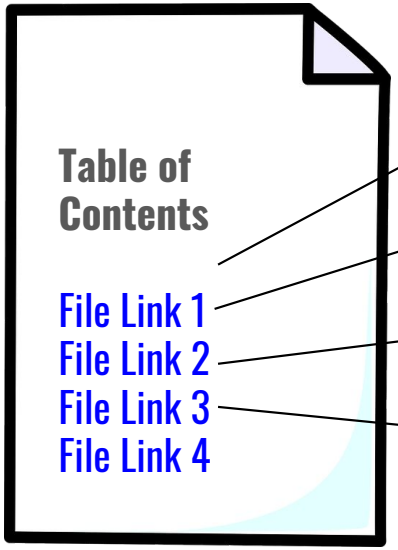


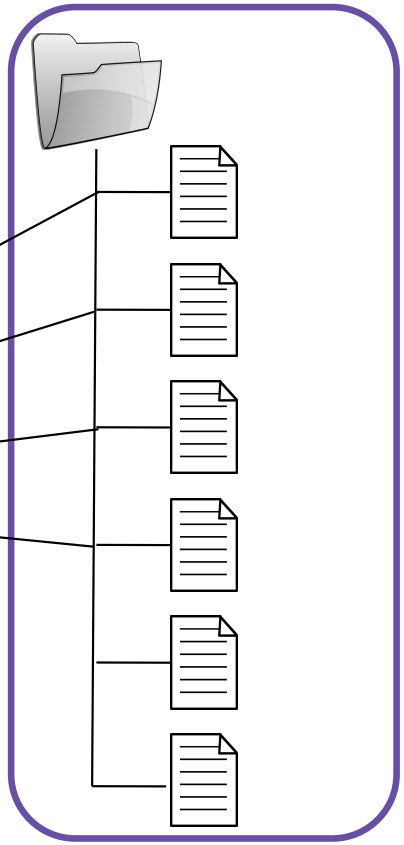
X

file path:

/Project/Main/Initial Work/Experiment 1/Good/Results1/Keep These/January/Beginning of Month/Week 1/Saturday/Cycle1/0034tz.tiff



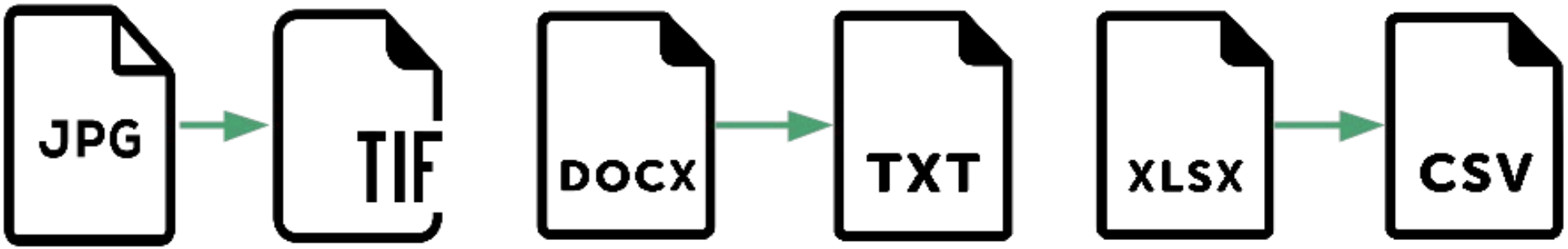




Best Practices: Open, Interoperable File Formats

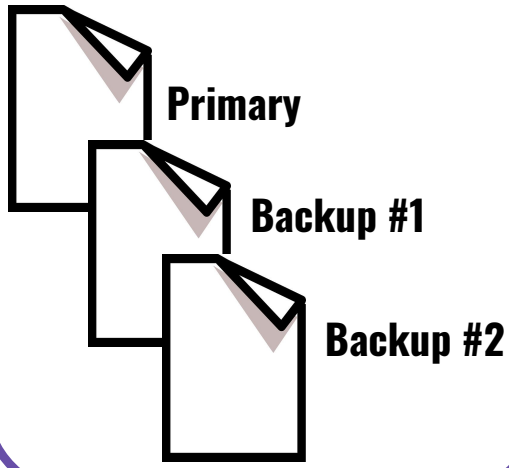
Put your data into an archival format → this should be open + accessible as well as software agnostic.

Someone shouldn't have to pay lots of money to buy a software to use your data. Pick a format that is open, well-documented, and can be used by lots of different tools!



Best Practices: Short-Term Storage

Keep 3 copies of any important file



Store files on 2 different media types

Secure Server

External HD

OR

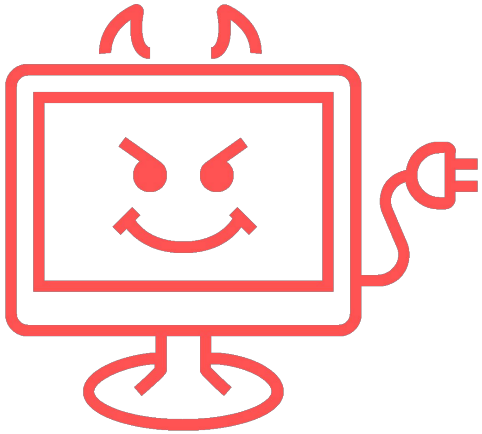
Secure Cloud

Keep at least 1 copy offsite

Let's talk about the tech problems now

- People make mistakes--and it impacts their research
- It's good to have other people check out your data and analyses--it's like having a copy editor for your data!

- It's *hard* to keep track of what version of what was used
- Software get updates, and these changes can disrupt reproducibility



We have an infrastructure problem

Technical Obsolescence

Technology changes affect the reproducibility

Otherwise known as…

the Pipeline Problem

Reproducibility requires skills that are often not included in most curriculums!

"It would require huge amount of effort to make our code work with the latest versions of these tools." Collberg et al., Repeatability and Benefaction in Computer Systems Research, University of Arizona TR 14-04

[1] https://www.ncbi.nlm.nih.gov/pubmed/19385804

Even if runnable, results may differ...

The Effects of FreeSurfer Version, Workstation Type, and Macintosh Operating System Version on Anatomical Volume and Cortical Thickness Measurements

We investigated the effects of data processing variables such as FreeSurfer version (v4.3.1, v4.5.0, and v5.0.0), workstation (Macintosh and Hewlett-Packard), and Macintosh operating system version (OSX 10.5 and OSX 10.6). **Significant differences** were revealed between **FreeSurfer version v5.0.0 and the two earlier versions**. [...] About a factor two smaller differences were detected between **Macintosh and Hewlett-Packard workstations** and between **OSX 10.5 and OSX 10.6**

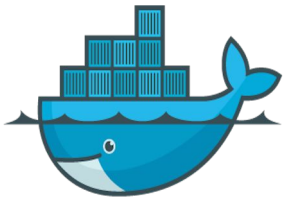
Infrastructure for reproducibility

Sadly, we know that you can work as reproducibly as possible, be AMAZING at documentation, and still fall short of 100% reproducibility. There are a few tools to actually help with that!

1. **Containers**: lightweight virtual operating system
   a. Singularity
   b. Docker
   c. o2R ... etc.

2. **Packaging Systems**: auto-capture of dependencies & source code used at time of running
   a. ReproZip



Containers - Singularity

Starting a Singularity container "swaps" out the host operating system environment for one the user controls -- instantly virtualize the operating system, without having root access, and allow you to run that application in its native environment!

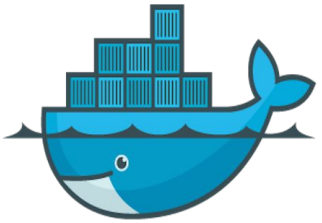
Really good for HPC environments -- made for HPC in fact

Linux only

You enter and work within the Singularity container, and the image grows and shrinks in real time. If you want to copy a container, you copy the image.

All metadata operations within the container occur within the container image (and not on the metadata server!) -- makes this good for parallel computing

Containers - Docker



Docker is “an open source project to pack, ship and run any application as a lightweight container.” -- idea is to provide a comprehensive abstraction layer that allows developers to “containerize” or “package” any application and have it run on any infrastructure.
Made by a for-profit company, partly open source

Available on Windows (not home edition), Linux, and Mac, but only virtualizes Linux.

Basically, you start a container with X operating system, can install whatever software, and ship it together

Good intro:
https://nuest.github.io/docker-reproducible-research/

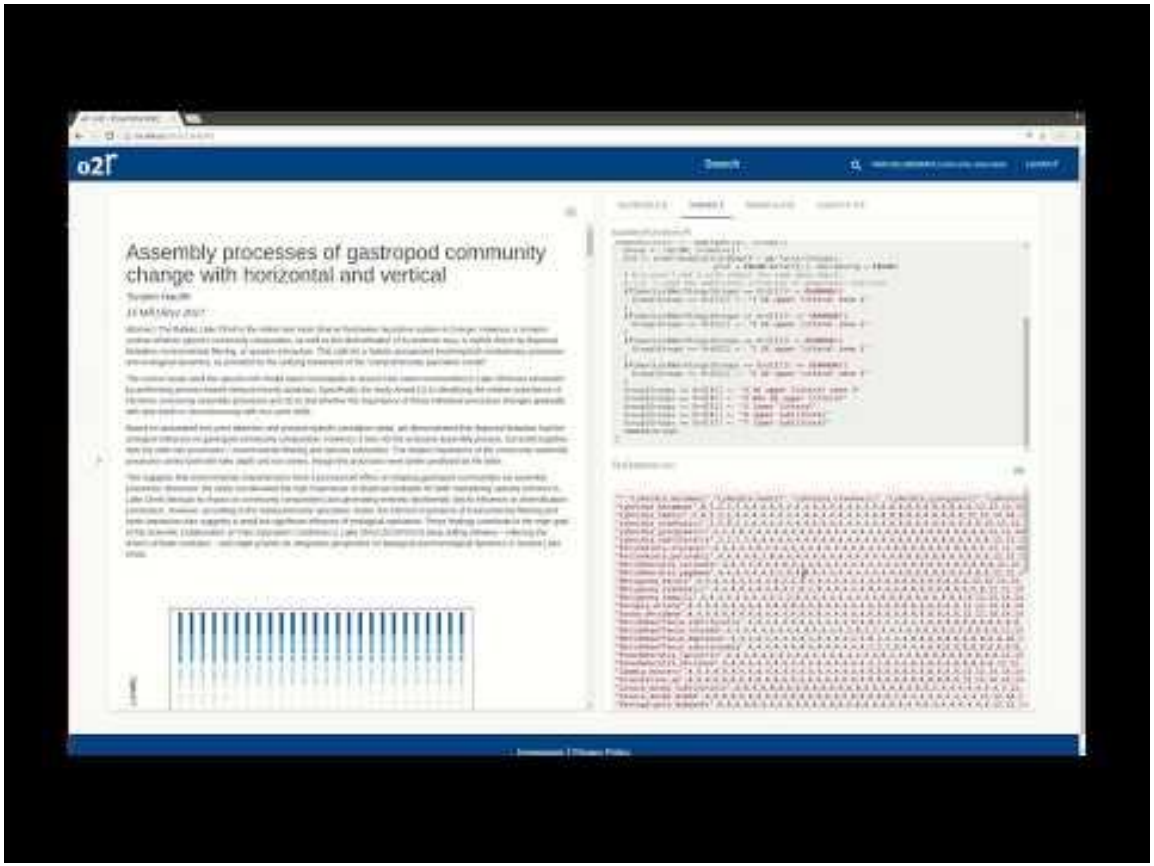
o2r - o2r.info

Project out of Uni of Muenster

Executable compedia -- basically, making research papers executable.

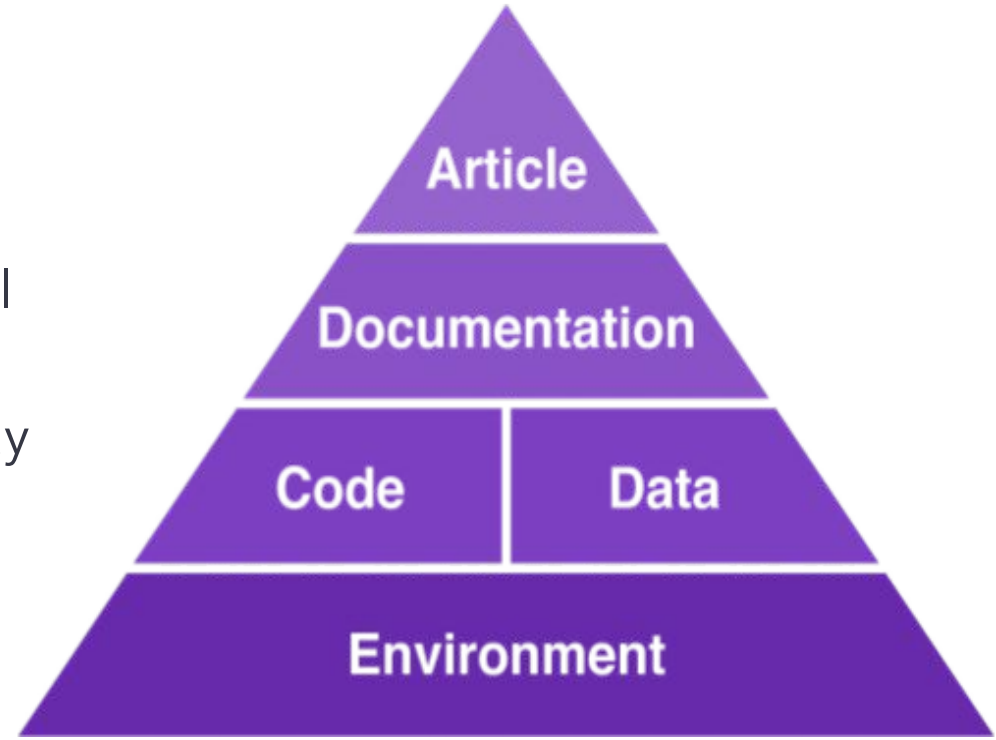
Give them a R workspace and R markdown file, get a nice executable compedia!

Leverages Docker on the backend



To Recap:

- Good project/data/code management enables reproducibility but doesn't guarantee it!
- Reproducibility is a spectrum! Work small and ramp up to full computational reproducibility!
    - Once we come up with a better way to share/access big data, it'll be easier for y'all
- T



The goal is to encourage the community to consciously choose open tools to increase interoperability & sustainability of their research.

Questions before 15min break?

Please run

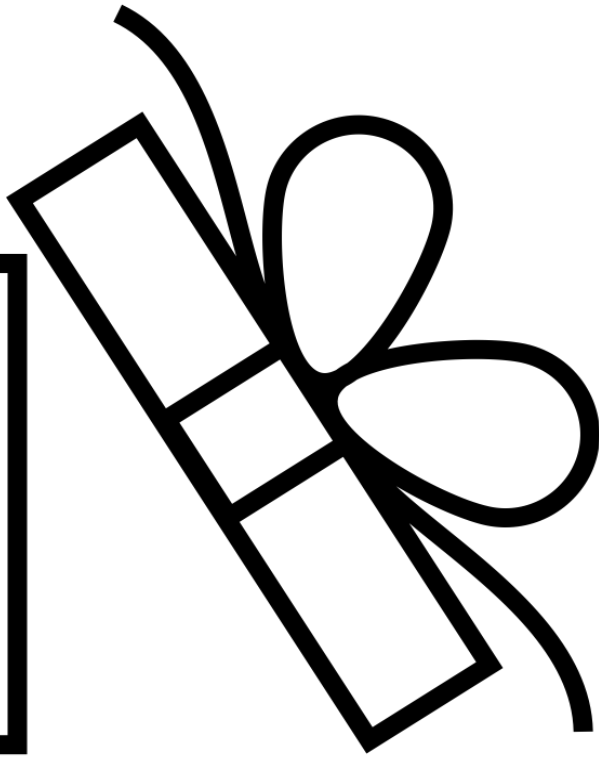
`vagrant up prebuilt`

on the terminal in the reprozip-examples folder

ReproZip

The Reproducibility Packer!

open, unpack, and reproduce anywhere, anytime!



ReproZip - reprozip.org

Open source tool that automatically captures provenance of research and packs all the necessary files, library dependencies, and variables to reproduce the results.

Anyone can then unpack and reproduce the research without having to install any additional software!

At the end of your research process, once you know everything works, you run

```
reprozip trace <command>
```

Once it's done, you finish up with:

```
reprozip pack <package-name.rpz>
```

You then upload that file to the OSF, and simply point people to it if they want to reproduce your work!

ReproZip can pack:

Data analysis scripts / software (any language, you name it!)

Graphical tools

Interactive tools

Client-server applications (including databases)

Jupyter notebooks

MPI experiments (setting up the experiment can be involved but…)

… and much more!

Current Use Cases:

Academic Use Cases

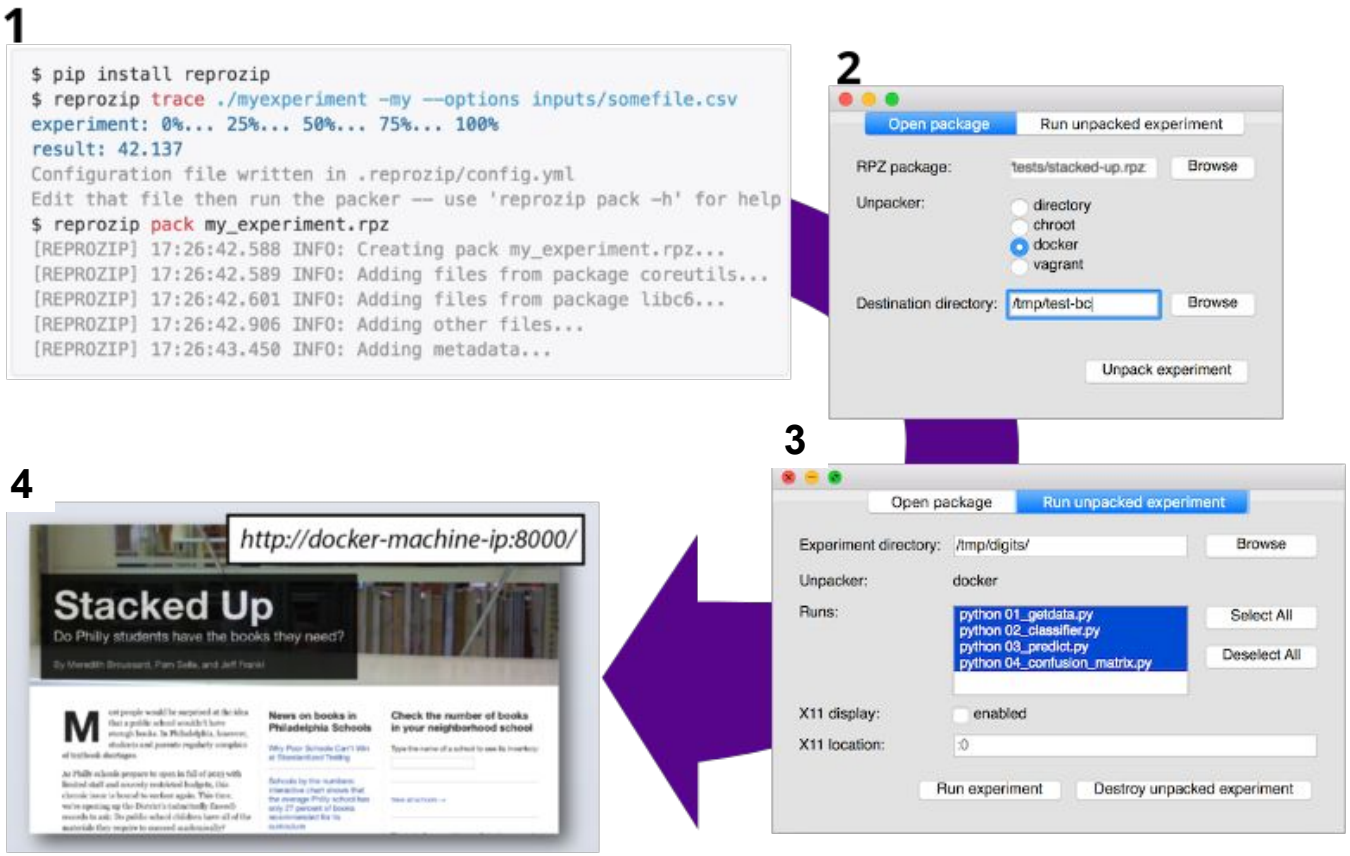
- Recommended by the Information Systems Journal, Reproducibility Section
- Recommended by the ACM SIGMOD Reproducibility Review
- Listed on the ACM Artifact Evaluation Process Guidelines

Outside Project Integration

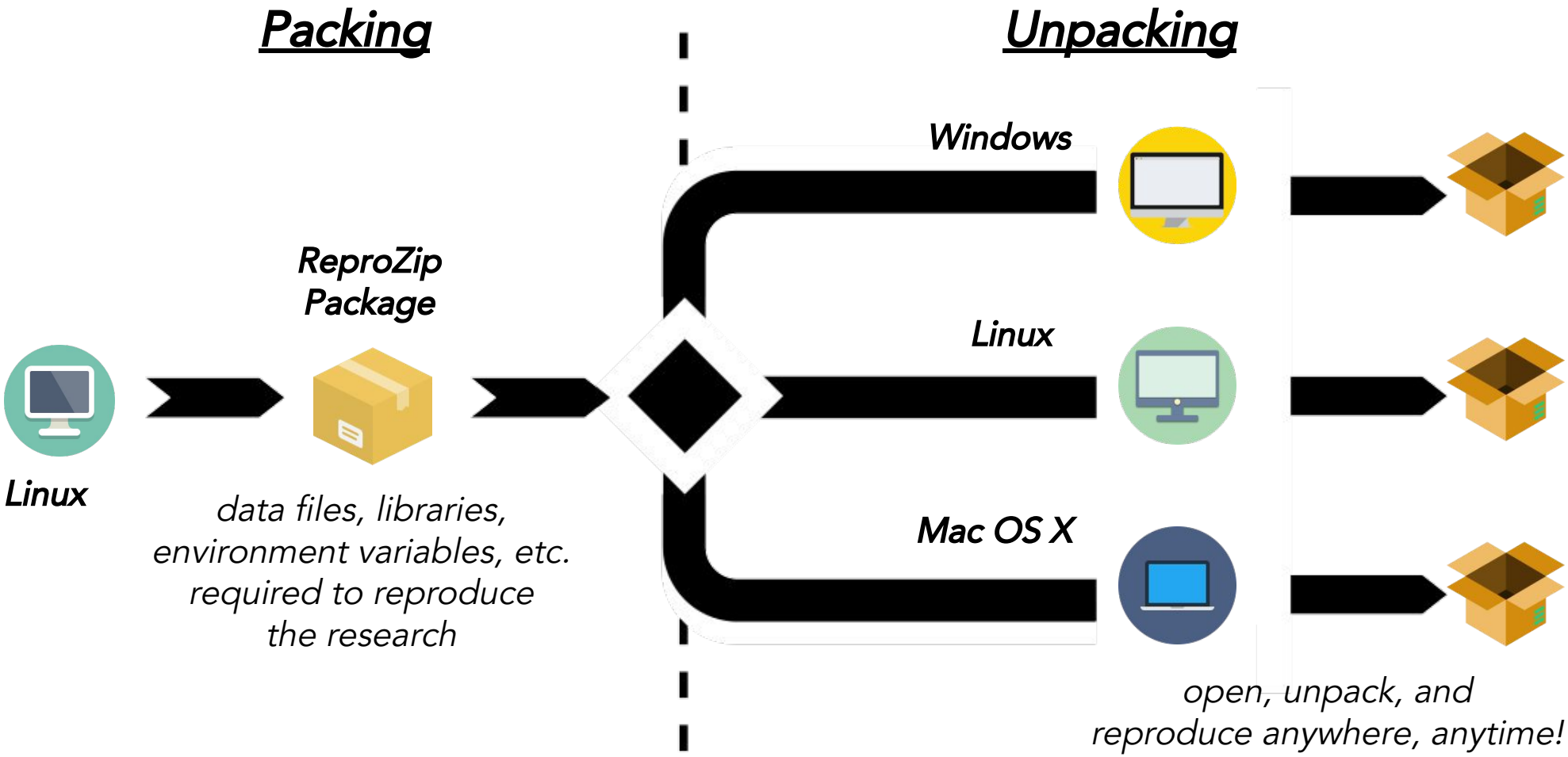
- Integrated as a component of CoRR
- Archiving data journalism apps, e.g.: Stacked Up
- Used by neurodocker to build minimal Docker images

… and many more!

ReproZip: Packing & reproducing research in 4 steps!



ReproZip Workflow

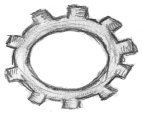


Packing Research

Computational Environment ***E*** (Linux)

Data Analysis (e.g.: Python, R, Julia)

Executing



reprozip

Tracing

Experiment Provenance

Data

Input files, output files, parameters …

Workflow

Executable programs and steps

Environment

Environment variables, dependencies, …

Configuration File

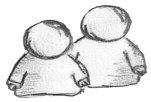
Creating Configuration

Configuring

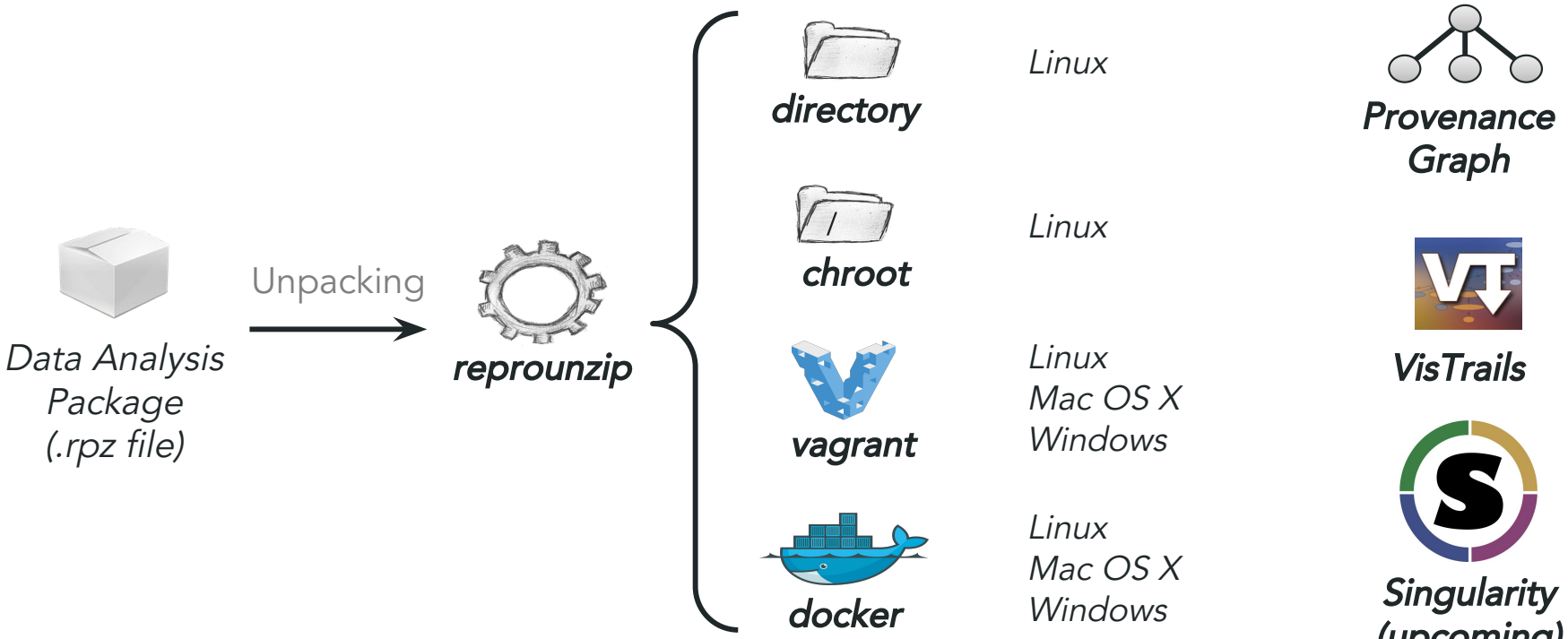
Packing

Data Analysis Package (.rpz file)

Unpacking Research

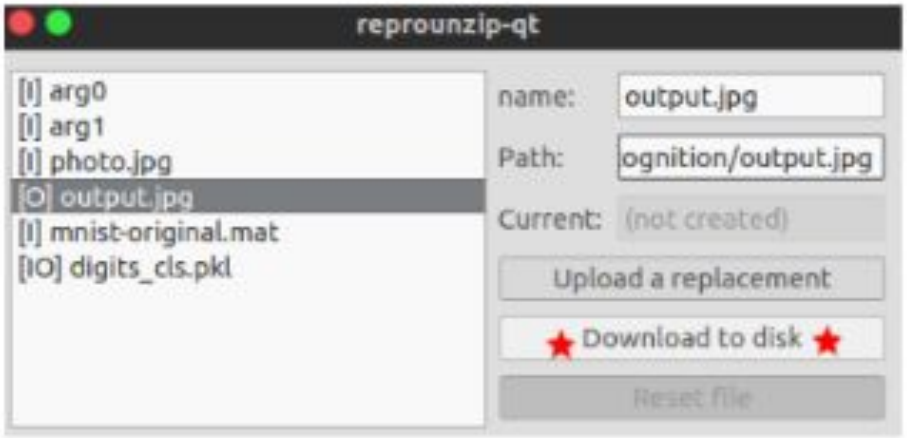


Computational Environment ***E'*** (potentially different than ***E***)

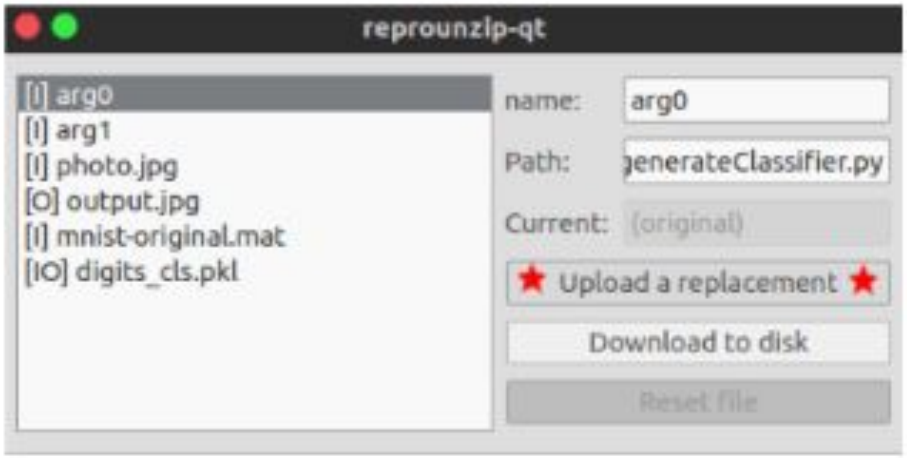


Extending the original work is also simple!

Download Ouput



Upload New Inputs



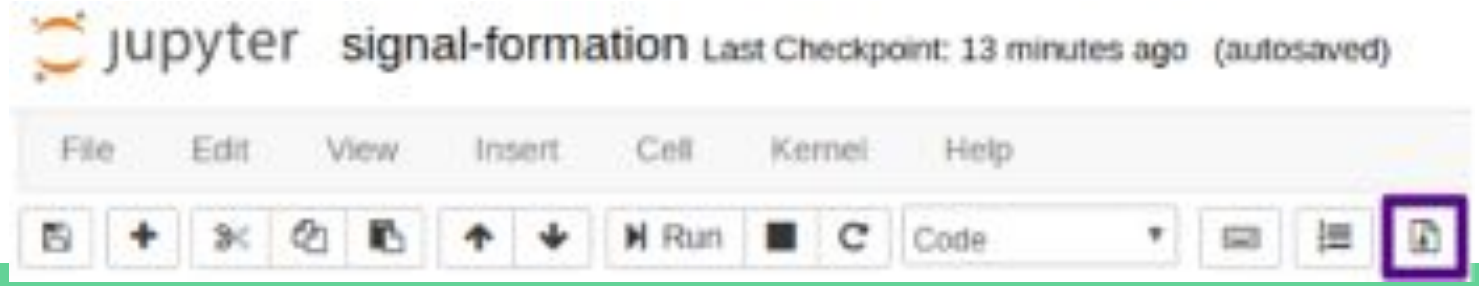
You can also use the CLI to do this if your data is remote or not easily uploaded via a GUI!

NEW: Jupyter notebook plugin!

You can read more in the docs. To install, on the terminal:

```
$ pip install reprozip-jupyter ## or conda install
$ jupyter nbextension install --py reprozip_jupyter --user
$ jupyter nbextension enable --py reprozip_jupyter --user
$ jupyter serverextension enable --py reprozip_jupyter
--user
```

Then you should see a little icon when you next start up your jupyter notebook. If you click that, ReproZip will trace and pack your notebook!



A note on MPI

It's possible to pack with ReproZip on MPI! You need some qsub/pbs/slurm magic to record all the processes, so it's not quite as easy as `reprozip trace && reprozip pack`, but it is do-able!

You would need a script that does `reprozip trace <something>` and you'd submit that script to the batch system. You can then merge all the traces from all the machines and pack them!

We are in the process of finishing the new Singularity unpacker for ReproZip, which will let you unpack and reproduce research captured on MPI much easier!

EXAMPLE 1: Image Analysis & Jupyter Notebooks

Brain segmentation with median_otsu

We show how to extract brain information and mask from a b0 image using dipy's segment.mask module.

First import the necessary modules:

```
import numpy as np
import nibabel as nib
```

Download and read the data for this tutorial.

The scil_b0 dataset contains different data from different companies and models. For this example, the data comes from a 1.5 tesla Siemens MRI.

```
from dipy.data.fetcher import fetch_scil_b0, read_siemens_scil_b0
fetch_scil_b0()
img = read_siemens_scil_b0()
data = np.squeeze(img.get_data())
```

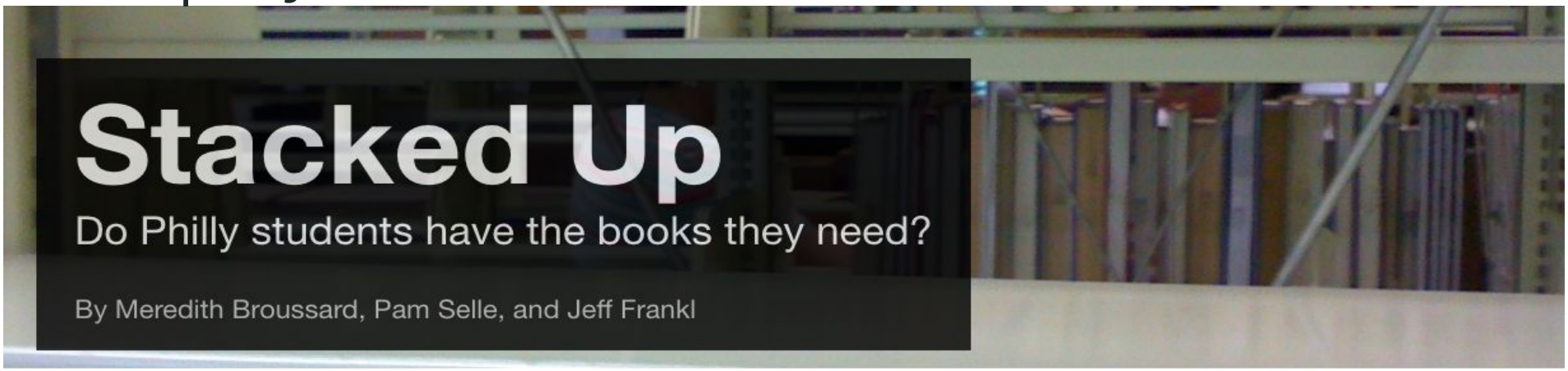
`img` contains a nibabel Nifti1Image object. Data is the actual brain data as a numpy ndarray.

Segment the brain using dipy's mask module.

`median_otsu` returns the segmented brain data and a binary mask of the brain. It is possible to fine tune the parameters of `median_otsu` (`median_radius` and `num_pass`) if extraction yields incorrect results but the default parameters work well on most volumes. For this example, we used

Original Experiment: http://nipy.org/dipy/examples_built/brain_extraction_dwi.html | 2GB
ReproZip Package: brain-segmentation.rpz | 47 MB

EXAMPLE 2: Packing Research App & Unpacking it to deploy on AWS!



Most people would be surprised at the idea that a public school wouldn't have enough books. In Philadelphia, however, students and parents regularly complain of textbook shortages.

As Philly schools prepare to open in fall of 2013 with limited staff and severely restricted budgets, this

News on books in Philadelphia Schools

Why Poor Schools Can't Win at Standardized Testing

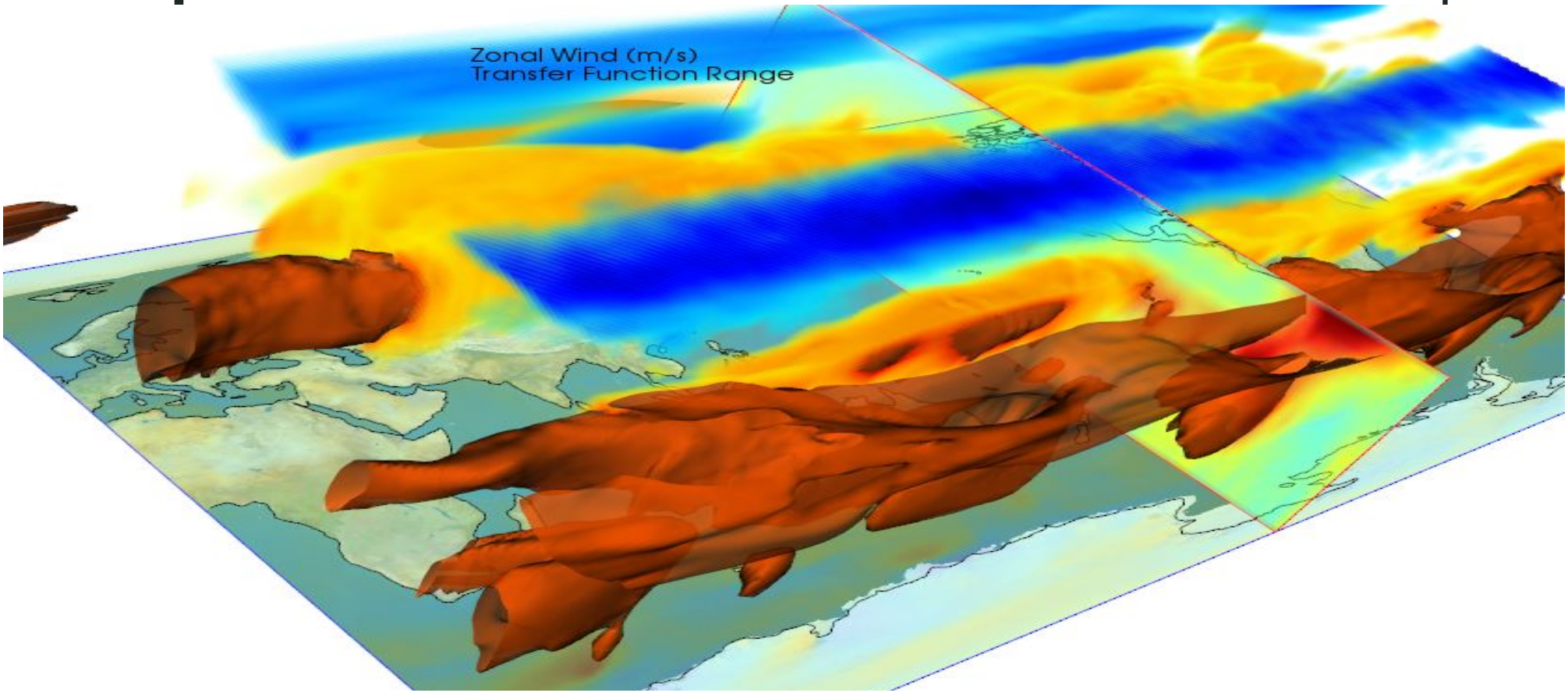
Schools by the numbers: interactive chart shows that

Check the number of books in your neighborhood school

Type the name of a school to see its inventory:

Original Experiment: https://github.com/merbroussard/sdp_curricula
ReproZip Package: stacked-up.rpz

Example 3: 3D Visualization of Wind on Map



Original Experiment: https://uvcdat.llnl.gov/examples/vcs3D_multiplot.html
ReproZip Package: https://osf.io/93rvc

HANDS-ON PORTION

You should be logged into the vagrant VM now -- if not please run the following from the `reprozip-examples` folder on the terminal:

```
vagrant ssh prebuilt
```

Let's try packing!

Please run: `vagrant ssh prebuilt` in the same directory where you have the Vagrantfile

We are going to go through packing some examples in the demo VM!

You can read more about the examples we are going to walk through on examples.reprozip.org

In case it all fails: nyu-dataservices.github.io/Reproducibility-in-Research/#/22

Bechdel-Test

This example is an attempt to replicate the findings from an article in FiveThirtyEight that examines gender bias in the movie business using the Bechdel test: a movie passes the Bechdel test if there are (1) two named women in it, (2) who talk to each other, (3) about something besides a man. This example is based on an excellent blog post by Brian Keegan, who strongly advocates for reproducibility in data journalism.

1. Data Collection: the datasets used by this example are collected from the Web. Four datasets are needed: revenue data from movies, inflation data, Bechdel test data, and data from IMDB.
2. Data Analysis: the datasets collected in the first step are joined and analysed, resulting in a number of different plots.

Let's reproduce it with ReproZip!

1. `workon bechedl-test`
2. `cd reprozip-examples/bechdel-test`
3. `reprozip trace python bechdel.py`
4. Get the output: `cp median_budget.png /vagrant/`
5. `reprozip pack bechdel-full.rpz`
6. `deactivate`

Irish-Schools

This example is an attempt to replicate the findings from Nick Wolf's National School System and the Irish Language Heaney Lecture 2015. The materials are from his lecture given as part of the Heaney Lecture Series at St. Patrick's College, Drumcondra, Ireland.

The original experiment uses this R script to create plots from the extracted Irish census data.

Let's reproduce it with ReproZip!

1. `cd reprozip-examples/irish-schools/`
2. `reprozip trace Rscript NationalSchools_Wolf_2016.R`
3. Get the output: `cp Rplots-1.png /vagrant/`
4. `reprozip pack national-schools.rpz`
5. `deactivate`

Digits-SKLearn-OpenCV

This example creates a SVM classifier for the digits dataset using scikit-learn, and predicts the values of handwritten digits of an image. The original experiment as two steps:

1. Classification: the SVM classifier is created.
2. Prediction: the values of hand-written digits from the input file are predicted and recognized (output.jpg).

Let's reproduce it with ReproZip!

1. `workon digits-sklearn-opencv`
2. `cd reprozip-examples/digits-sklearn-opencv/`
3. `reprozip trace python generateClassifier.py`
4. `reprozip trace --continue python performRecognition.py`
5. Get the output: `cp output.jpg /vagrant/`
6. `reprozip pack digitRecognition.rpz`
7. `deactivate`

Let's try UNpacking!

Double check you have ReproZip installed: reprozip.org and in the top right corner, pick your OS to get the right installer

We're going to use the Vagrant unpacker (because you already have it installed!), but you can use these examples with any unpacker that you want!

After we do some more stuff locally, I'll show you how to unpack these ReproZip packages in-browser!

Bechdel-Test

GUI Instructions:

1. Double click on `bechdel-full.rpz`
2. Select Vagrant as the unpacker via the radio buttons
3. Click 'Unpack Experiment'
4. Let it run! It will redirect you to another tab to run the script
5. Make sure the scripts are highlighted
6. Click run & watch it go!

On the terminal:

1. **Open up a new terminal window**
2. **Navigate to the `reprozip-examples` folder in your Desktop or Downloads**
3. `reprounzip vagrant setup bechdel-full.rpz bechdel/`
4. `reprounzip vagrant run bechdel/ collectdata`
5. `reprounzip vagrant run bechdel/ plotresults`
6. `reprounzip vagrant download bechdel/ --all`

Irish-Schools

GUI Instructions:

1. Double click on `national-schools.rpz`
2. Select Vagrant as the unpacker via the radio buttons
3. Click 'Unpack Experiment'
4. Let it run! It will redirect you to another tab to run the script
5. Make sure the R script is highlighted
6. Click run & watch it go!

On the terminal:

1. **Open up a new terminal window**
2. **Navigate to the `reprozip-examples` folder in your Desktop or Downloads**
3. `reprounzip vagrant setup national-schools.rpz national-schools/`
4. `reprounzip vagrant run national-schools/`
5. `reprounzip vagrant download national-schools/ --all`

Digits-SKLearn-OpenCV

GUI Instructions:

1. Double click on **`digitRecognition.rpz`**
2. Select Vagrant as the unpacker via the radio buttons
3. Click 'Unpack Experiment'
4. Let it run! It will redirect you to another tab to run the script
5. Make sure the scripts are highlighted
6. Click run & watch it go!

From the command line:

1. **Open up a new terminal window**
2. **Navigate to the `reprozip-examples` folder in your Desktop or Downloads**
3. **`reprounzip vagrant setup digitRecognition.rpz digit-recognition/`**
4. **`reprounzip vagrant run digit-recognition/ classification`**
5. **`reprounzip vagrant run digit-recognition/ prediction`**

Digits-SKLearn-OpenCV - Down/uploading data

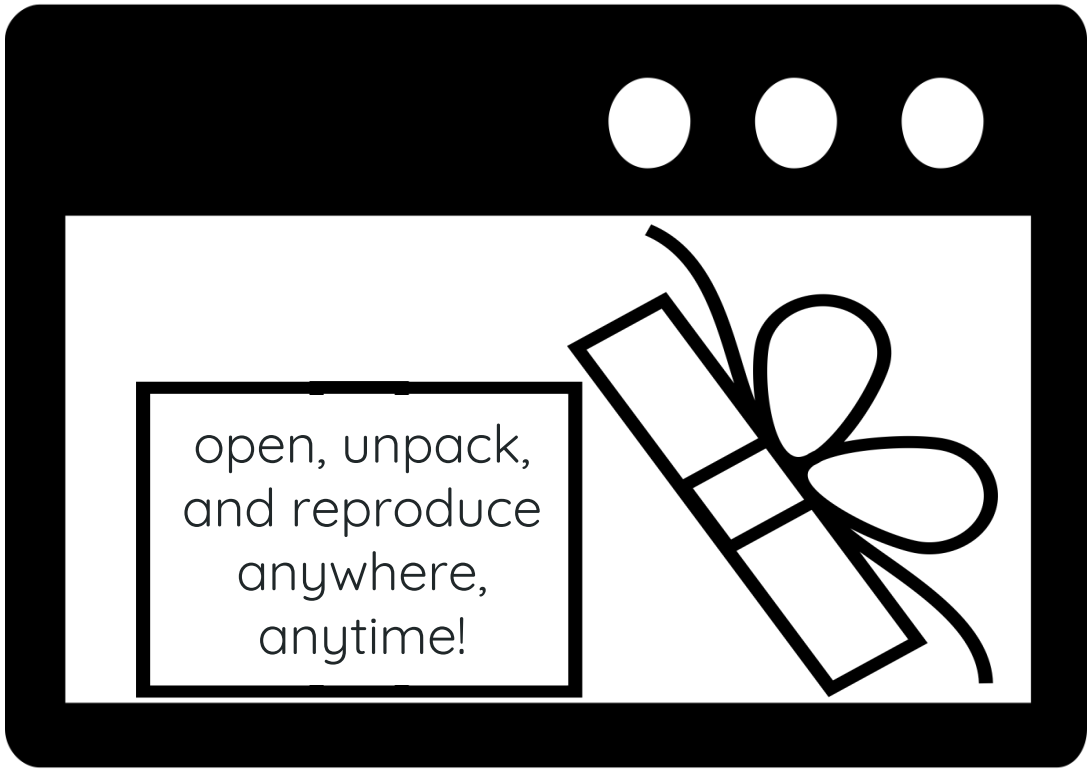
Let's see the output again using ReproUnzip:

```
reprounzip vagrant download digit-recognition/ output.jpg
```

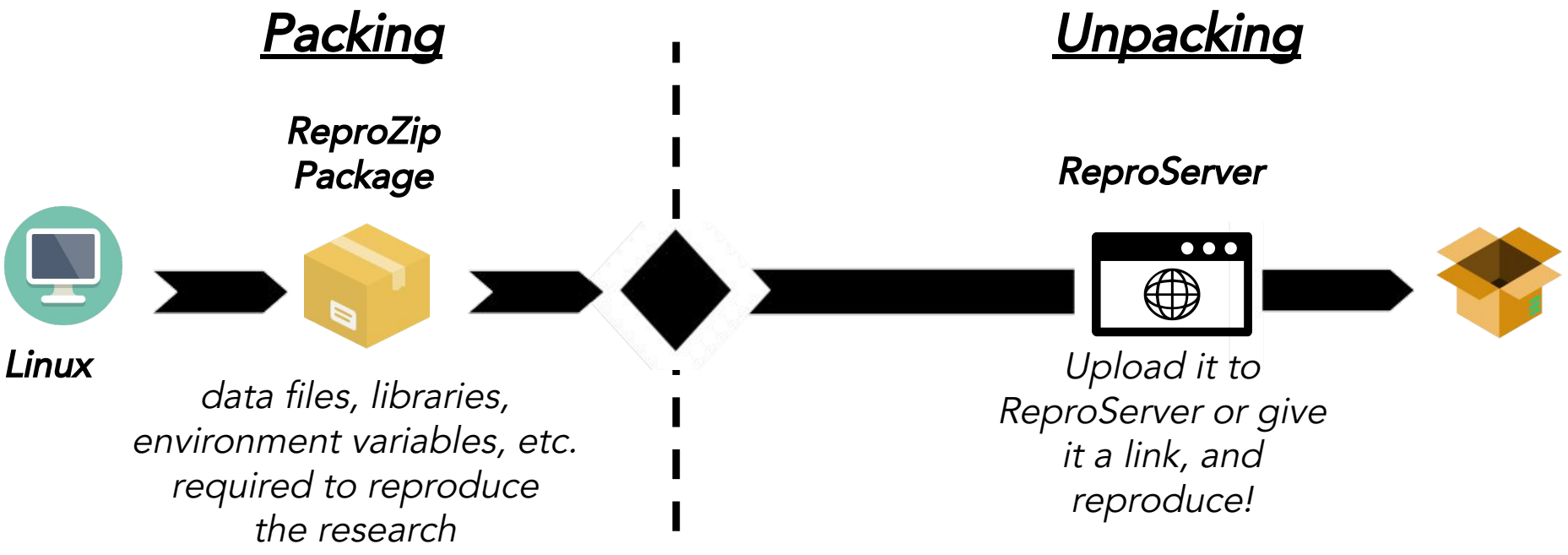
Let's try uploading some new data to use with the same workflow!

1. Download the new input to the `digit-recognition/` folder: https://github.com/ViDA-NYU/reprozip-examples/blob/master/digits-sklearn-opencv/photo_2.jpg
2. Run: `reprounzip vagrant upload digit-recognition/ photo_2.jpg:photo.jpg`
3. Run: `reprounzip vagrant run digit-recognition/ prediction`
4. And see your new output image!

ReproServer, reproducibility in-browser!

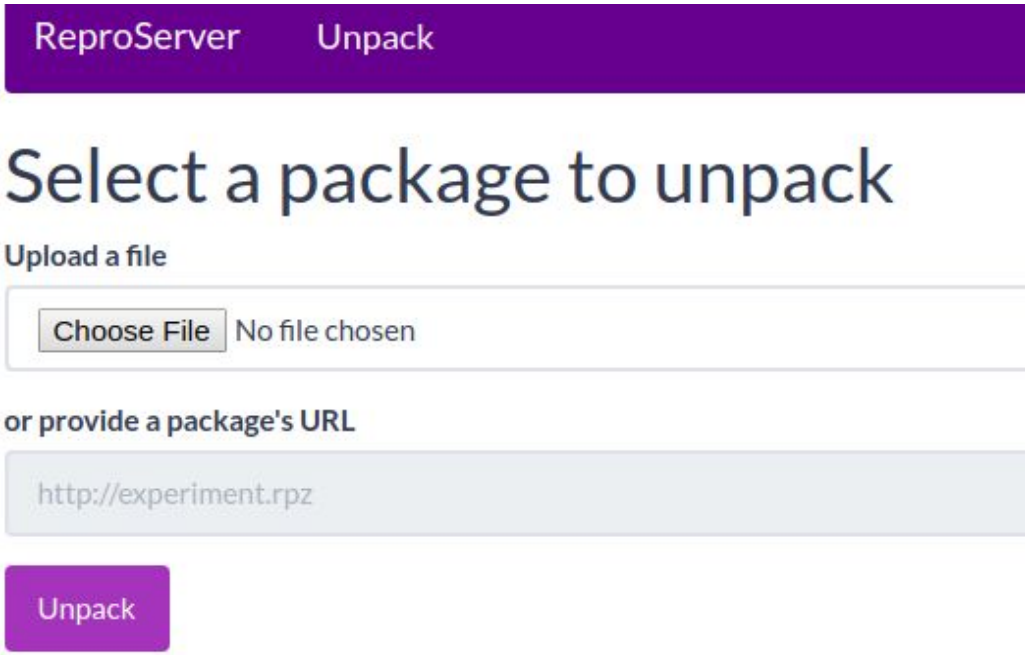


ReproZip + ReproServer = Easy Reproducibility!

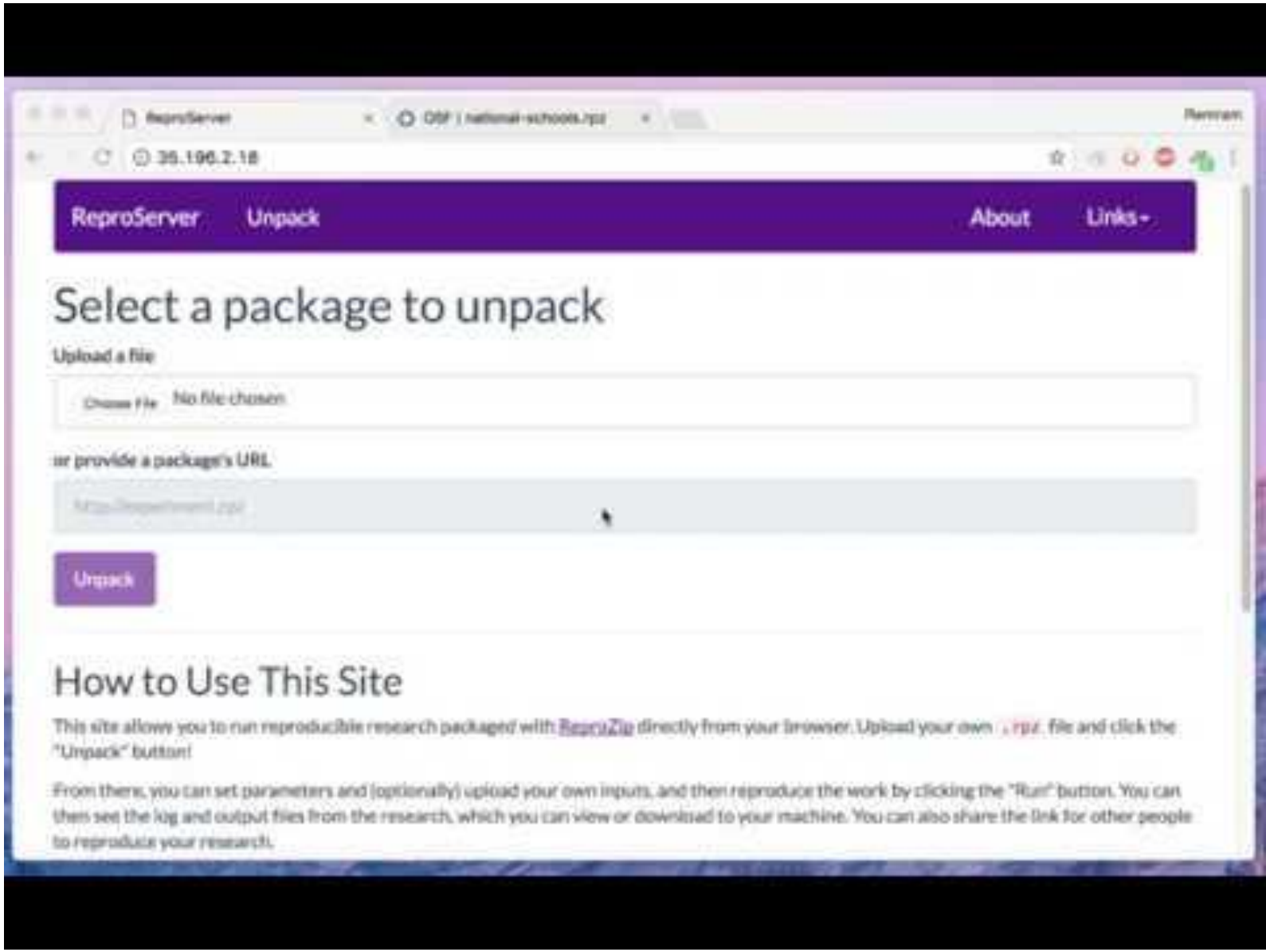


ReproServer

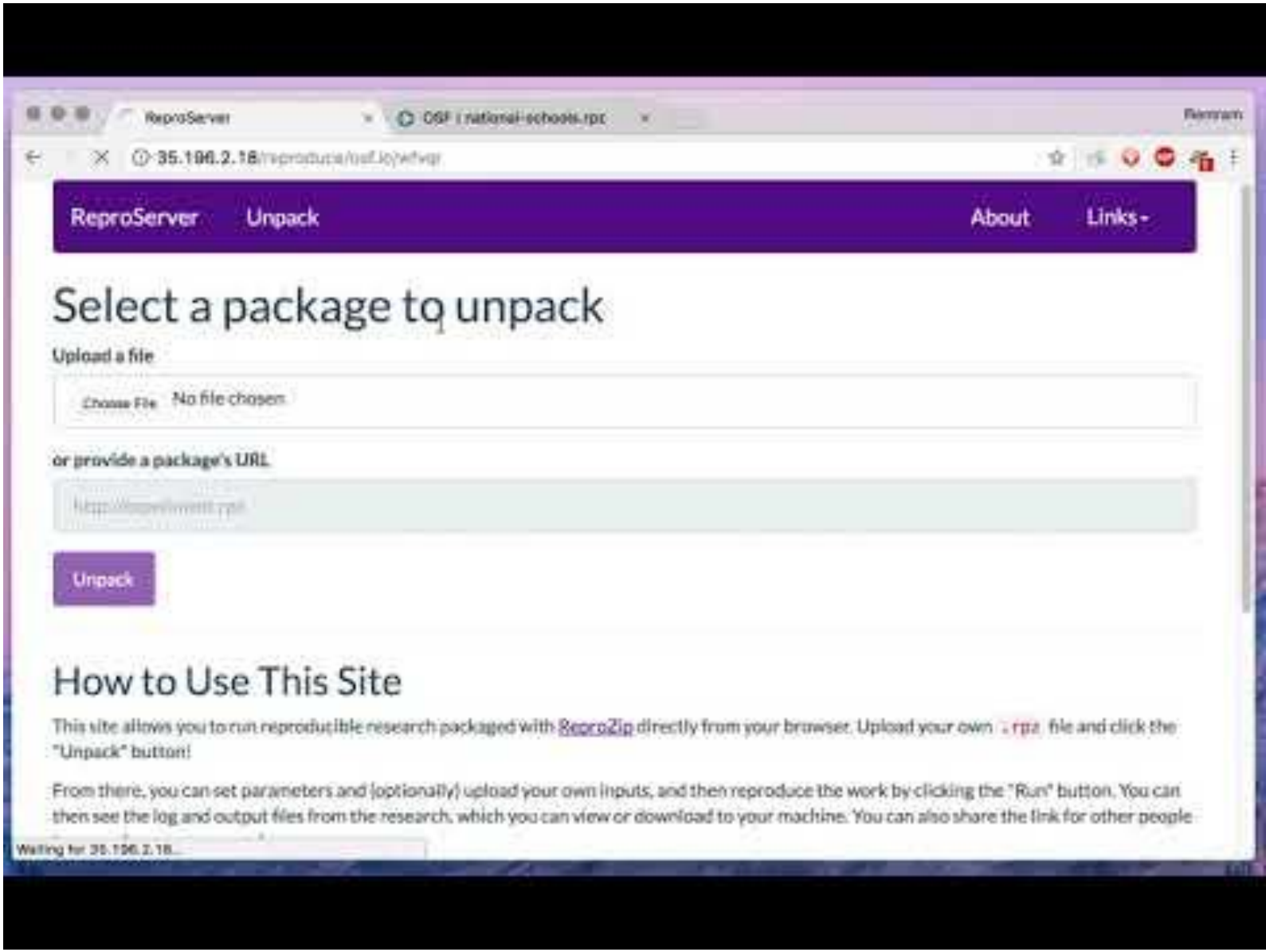
- Runs ReproZip packages **in the browser**, no local software needed
- Allows **changing** input data, configuration, command-lines
- Gives you **a URL to include in papers** to reproduce your experiment
- Offloads archiving responsibility to people who are good at it (repositories)
- **No lock-in**: build on your laptop, pack automatically, reproduce anywhere



Unpacking ML Scripts with ReproServer



Unpacking
R Plots
with
ReproServer



Other Resources for ReproZip & ReproServer

ReproZip Website:
reprozip.org

ReproZip Examples:
examples.reprozip.org

ReproZip GitHub:
github.com/ViDA-NYU/reprozip

ReproServer GitHub:
github.com/ViDA-NYU/reproserver

YouTube Demos:

- ReproZip Demo:
  goo.gl/o1Hqrx
- Website packing:
  goo.gl/yMEOZJ
- Jupyter notebook:
  goo.gl/NvMHnw
- ReproServer:
  goo.gl/Wk7Xnz
- ReproServer OSF integration:
  goo.gl/XfF78z

Summary

- You can work reproducibly in many ways -- your work can be somewhat reproducible, fully reproducible, or not at all.
  - Introduce reproducible workflows in small bits, get comfortable, and expand!
- Open formats & open tools enable more reproducibility. Use them!
- Think about research holistically

Thank You & Contact

Thanks to Daina & CfA for the invitation to come speak with you all today!

Thanks to Remi Rampin (main dev), Fernando Chirigati (team member) , & Juliana Freire, (P.I.)!

Email me: vicky.steeves@nyu.edu
Tweet me: @VickySteeves
Toot me: Vicky.Steeves@octodon.social