The Story of an Open Science Experiment

Sheeba Samuel

Friedrich Schiller University, Jena, Germany

MPDL Open Science Days

19-20 October 2021

FRIEDRICH-SCHILLER-UNIVERSITÄT JENA

FUSION

MichaelStifelCenterJena
for Data-Driven and Simulation Science

Carl Zeiss Stiftung

Introduction

- **2019-Present**: PostDoc Researcher, University of Jena, Germany
- **2016-2019**: PhD (A Provenance-based Semantic Approach to Support Understandability, Reproducibility, and Reuse of Scientific Experiments)

@sheebasamuel

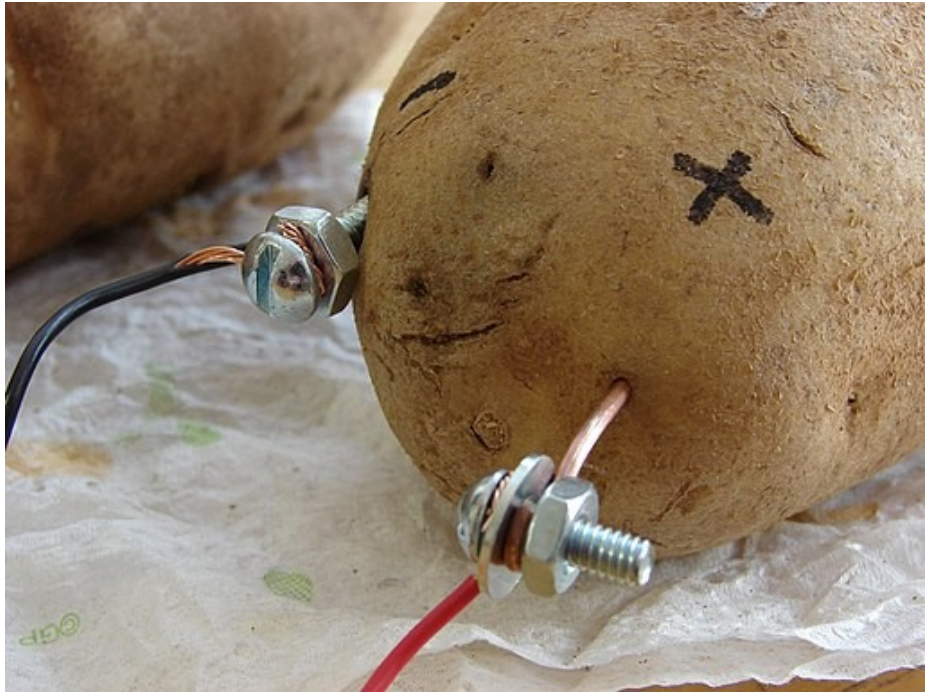
0000-0002-7981-8504

- https://fusion.cs.uni-jena.de/
- https://w3id.org/reproduceme/research
- https://sheeba-samuel.github.io/

Story

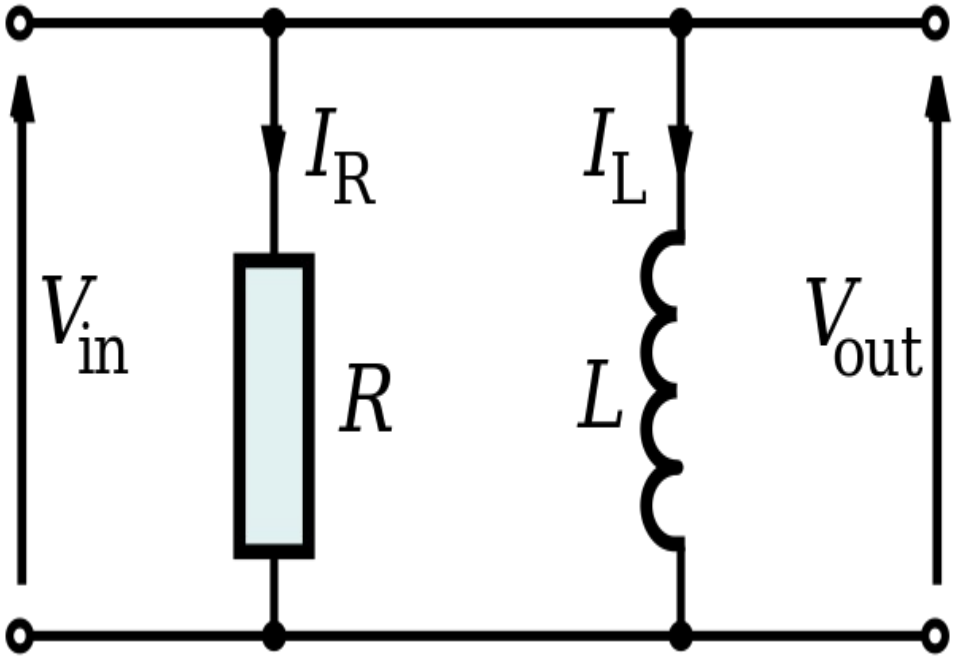
Source: Wikimedia Commons: Snow White and the Seven Dwarfs

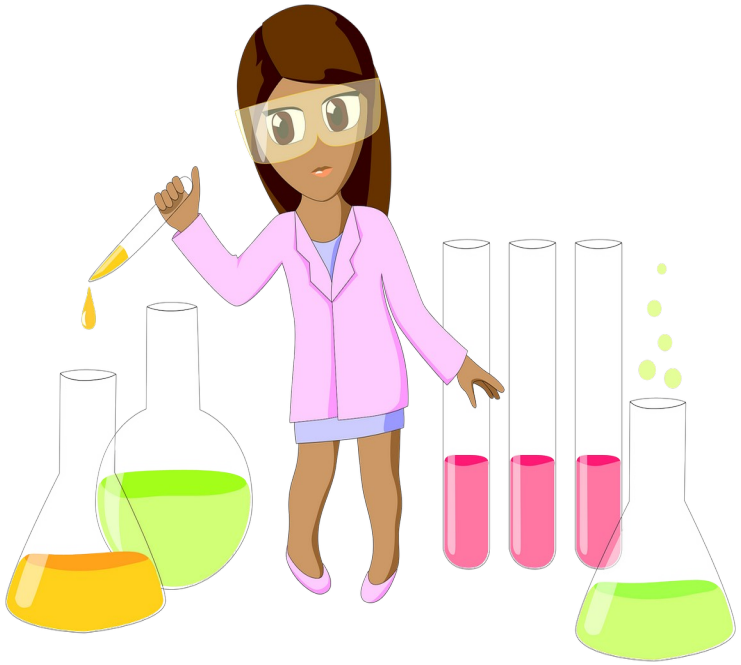
Story of an Experiment



Source: Wikimedia Commons

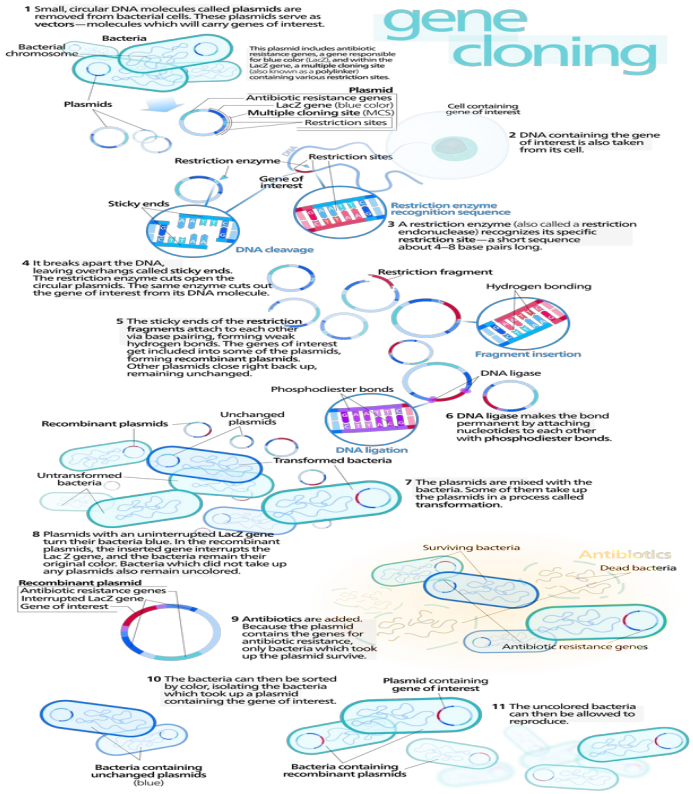
Story of an Experiment



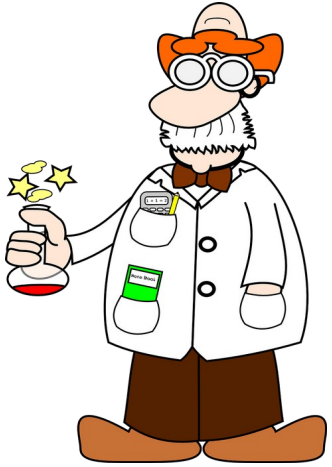


Source: Pixabay

Story of an Experiment



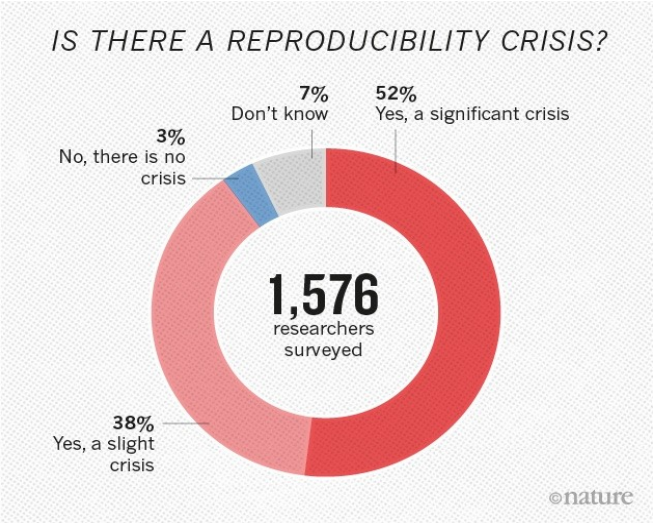
Source: Wikimedia Commons



Source: Pixabay

Introduction

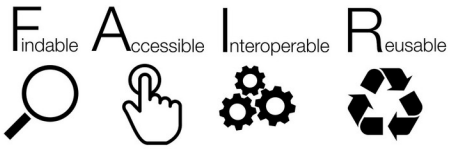
Reproducibility Crisis*



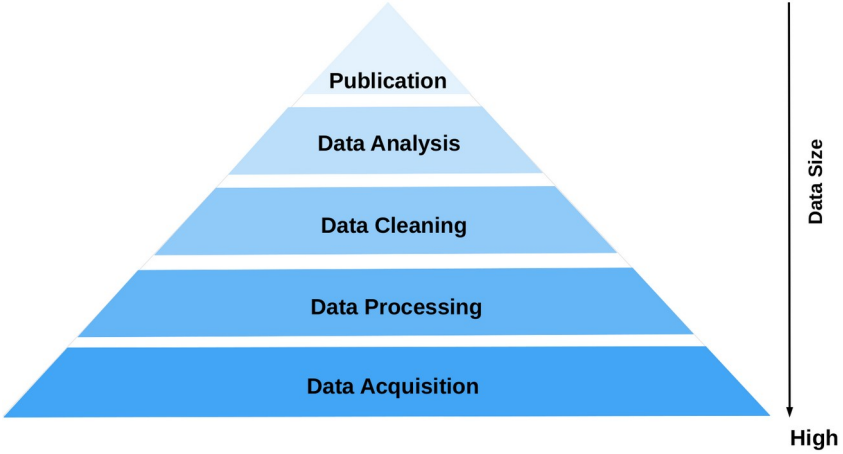
Source: 1,500 scientists lift the lid on reproducibility

Reproducibility Measures

"Authors are required to make materials, data, code, and associated protocols promptly available to readers without undue qualifications." - **natureresearch**



Source: Wikimedia Commons



* [Kaiser 2015, Peng 2015, Begley and Ioannidis 2015, Baker 2016, Hutson 2018]

Insights from Interviews

It is important to document **the individual trials** to understand which **possibilities did not work out**.

Biologist

Methods in the publication are **not sufficient** to fully reproduce an experiment.

Physicist

Lack of awareness of data management process

Ecologist

The documentation of **procedures, devices, and steps** are **vital** for reproducibility.

Biochemist

Since the experimental metadata are stored at different places including **lab books & external hard disks,** it gets **hard to distinguish** good experiments or to find a specific experiment.

Biologist

Simple tools for data management rather than complicated tools which require extra learning.

Ecologist

Code and data which are required to reproduce results need to be **publicly available**. The ability to **visualize and compare** results helps to understand our own and others work.

Computer Scientist

A system providing **a link to the data, metadata, and results** can help in our daily work.

Geologist

Raw and processed data are needed to understand an experiment.

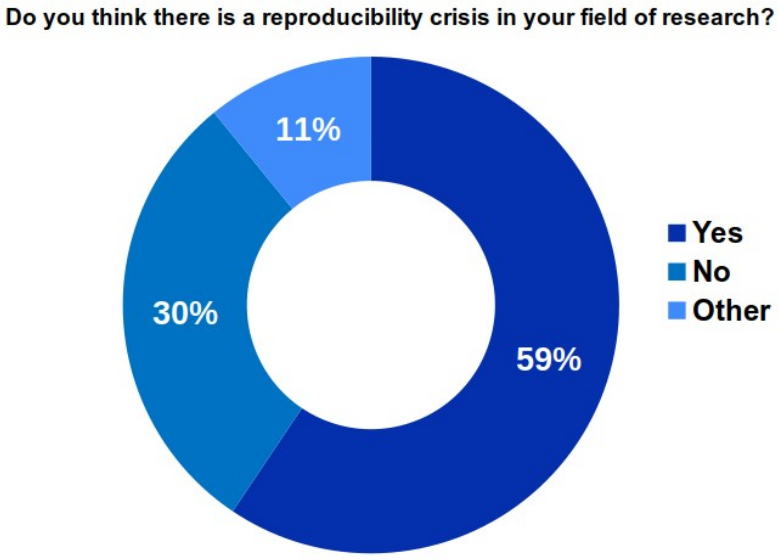
Mathematician

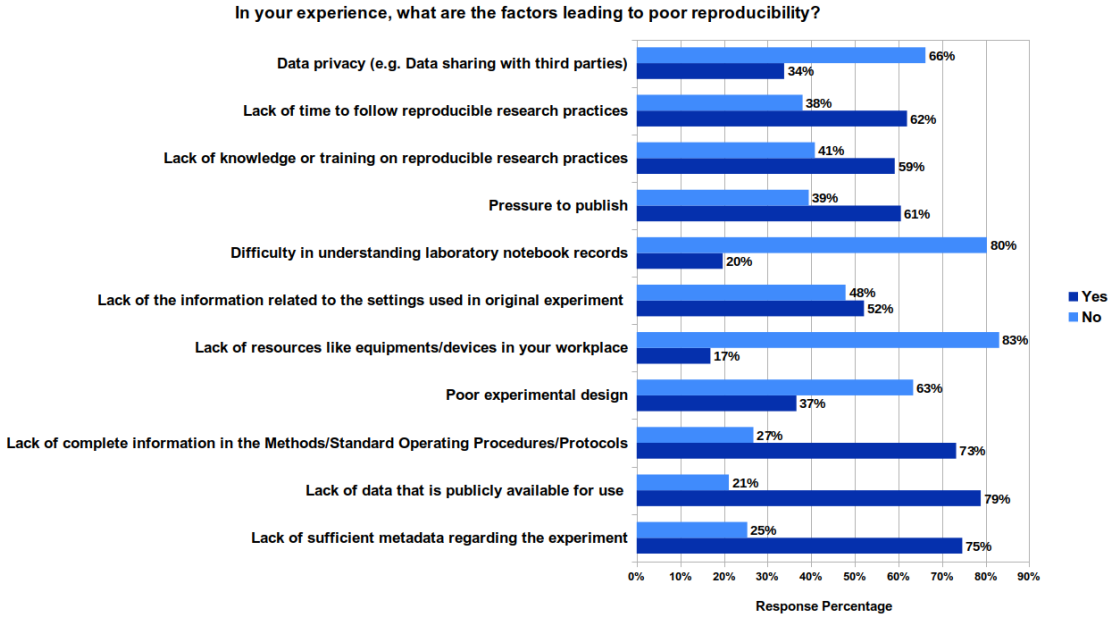
Source: CRC ReceptorLight & "Fostering reproducible science − What data management tools can do and should do for you" workshop (15 June 2017)

Survey on experiments and research practices for reproducibility

Findings

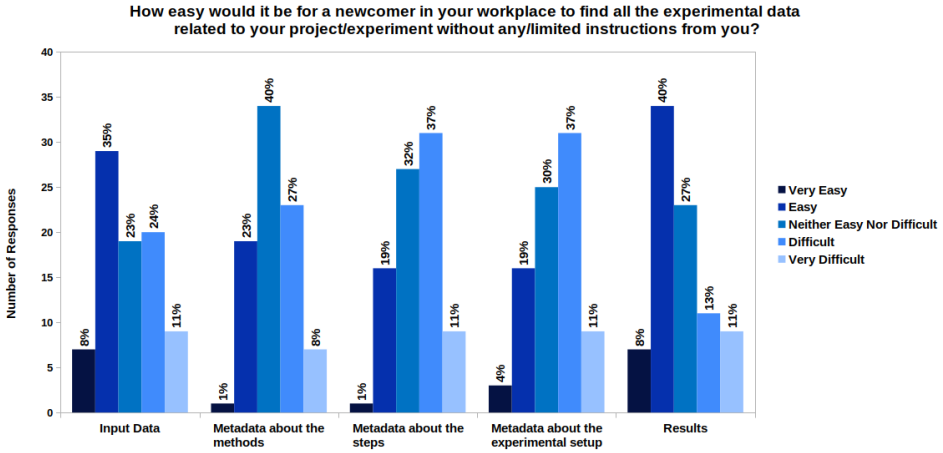
- More than half (59%) of the participants think that there is a reproducibility crisis
- 54% of the participants had trouble reproducing other's published results

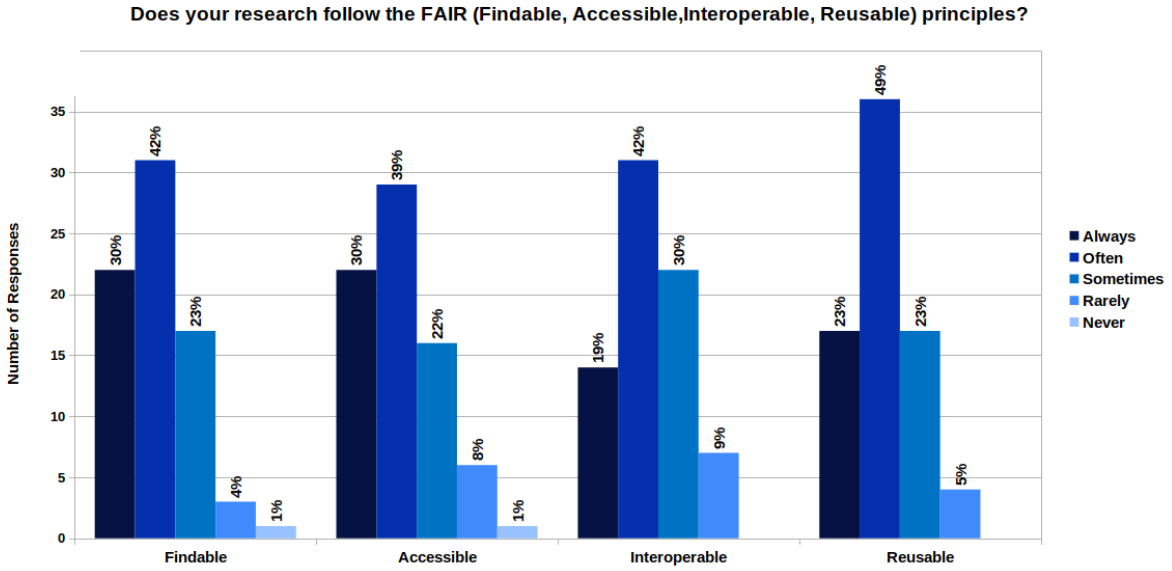




[Understanding experiments and research practices for reproducibility: an exploratory study: Samuel and König-Ries 2021]

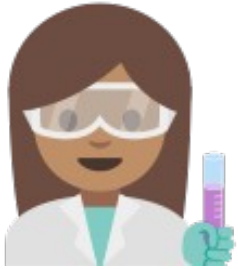
FAIR data principles

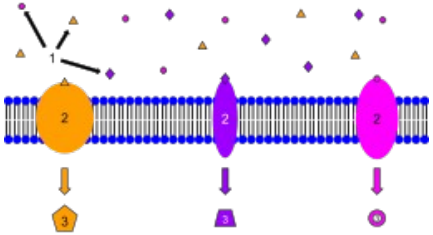


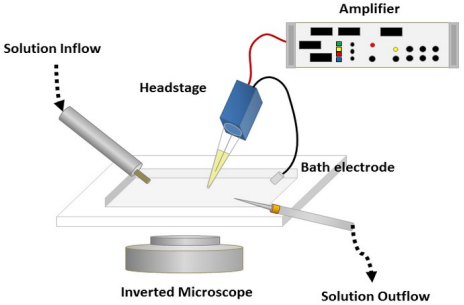


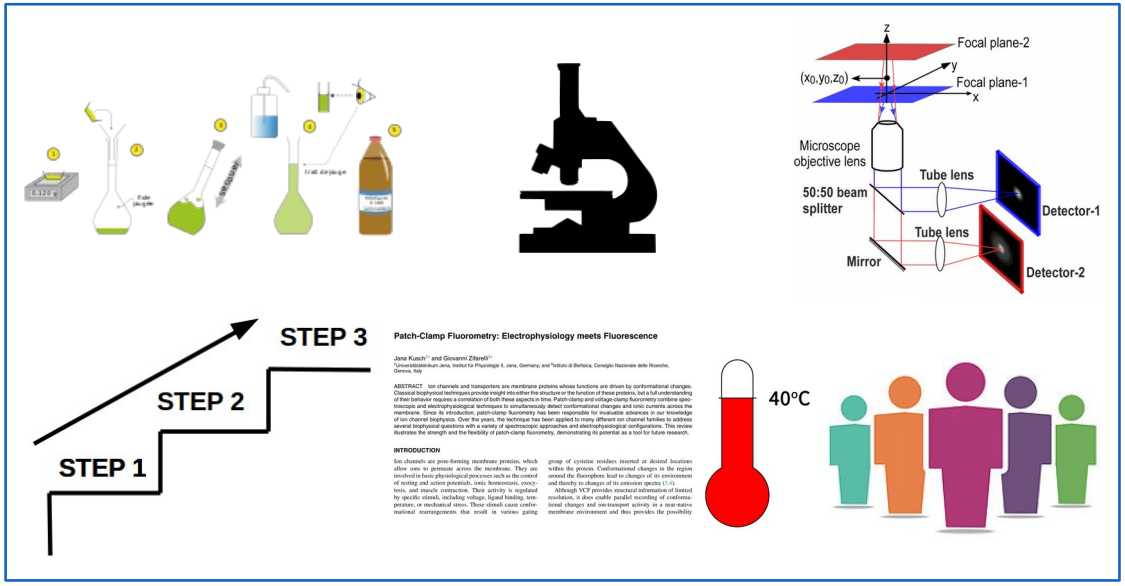
[Understanding experiments and research practices for reproducibility: an exploratory study: Samuel and König-Ries 2021]

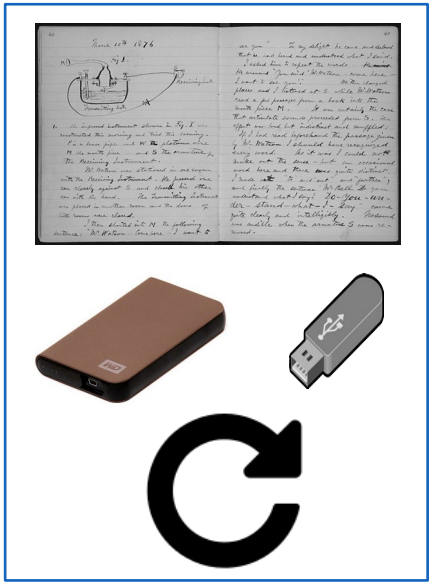
Scientific Experiments





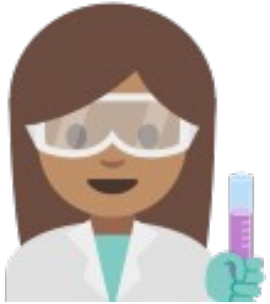


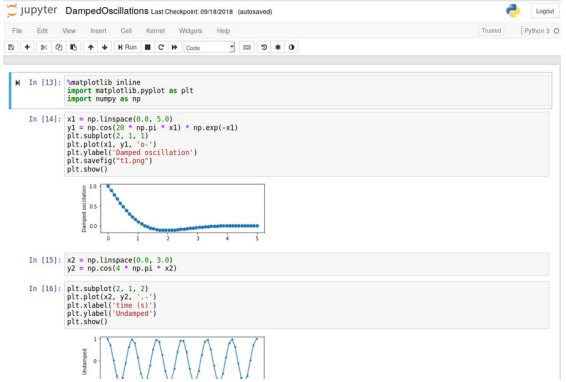




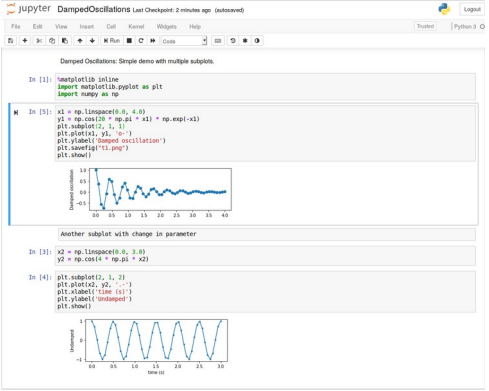
Computational Experiments

- Repeatability



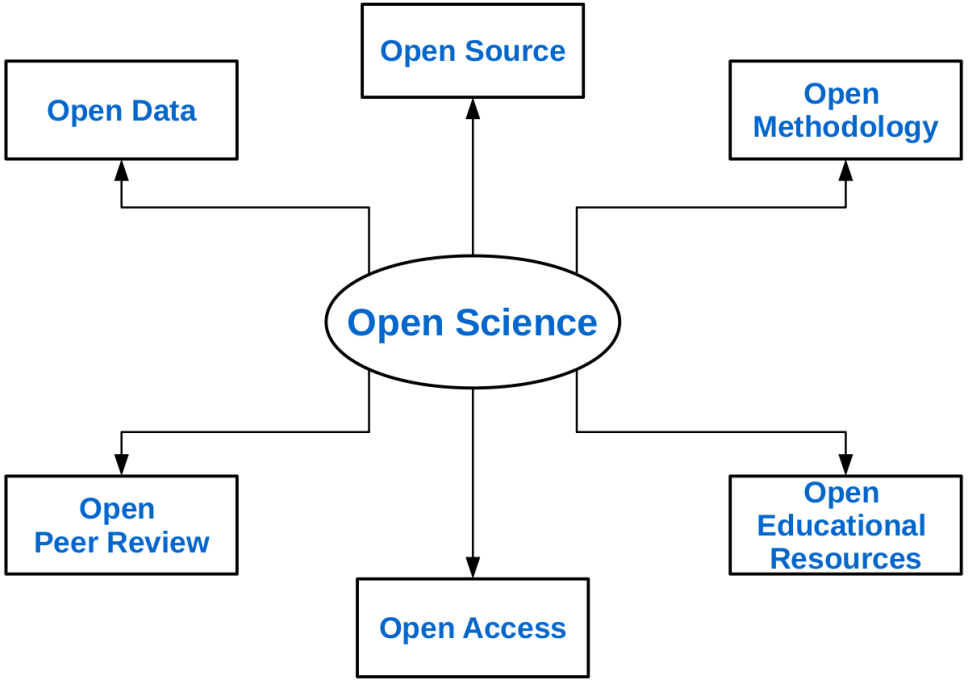


- Reproducibility



Open Science

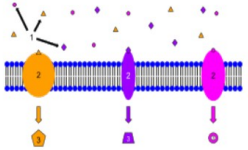
- Open science is the movement to make scientific research available to any member in society.

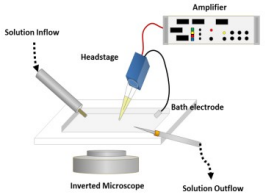


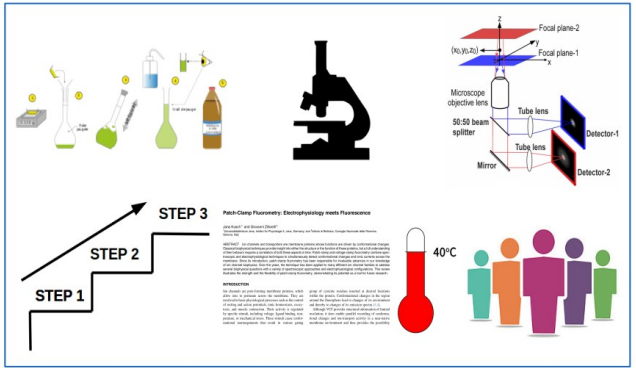
The Open Traits Network: Using Open Science principles to accelerate trait-based science across the Tree of Life, Gallagher et al. 2020

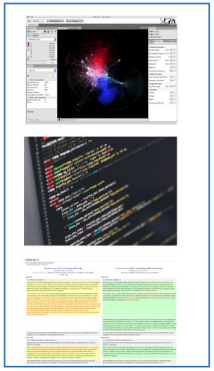
Open Reproducible Scientific Experiments

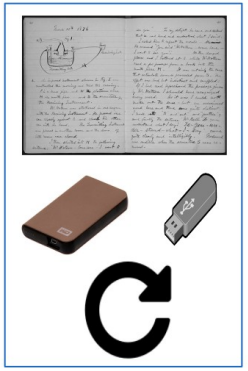
- Open Data
- Open Source
- Open Publications
- Open Experiment Materials
- Open Hardware
- Open Instruments
- Open Notebook Science
- Open Methodology
- Open Steps
- Open Data Standards
- Open Workflows
- Open File Formats





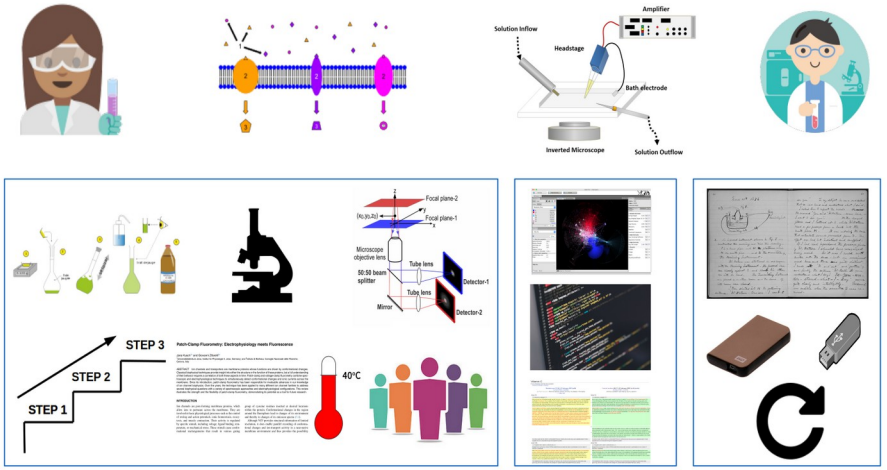






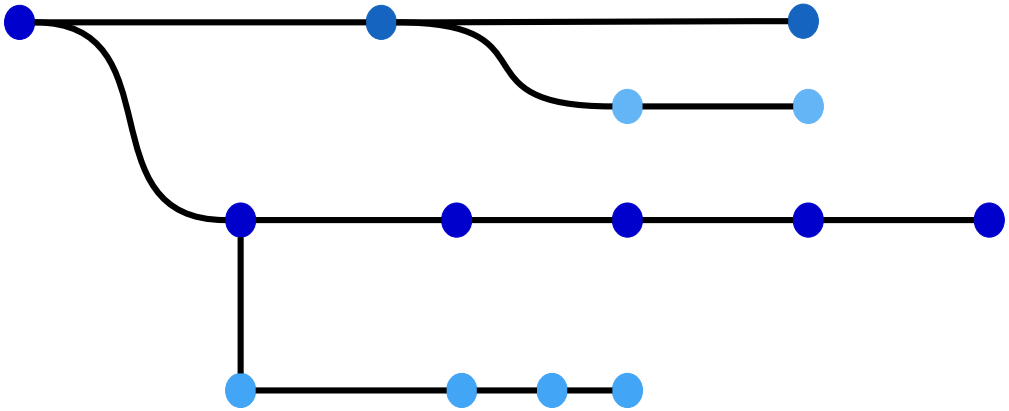
Challenges: Reproducibility

- Proprietary File Formats
- Availability of hardware
- Proprietary Data Standards
- Unavailability of Experiment Materials
- Not enough description in methodology
- Missing steps



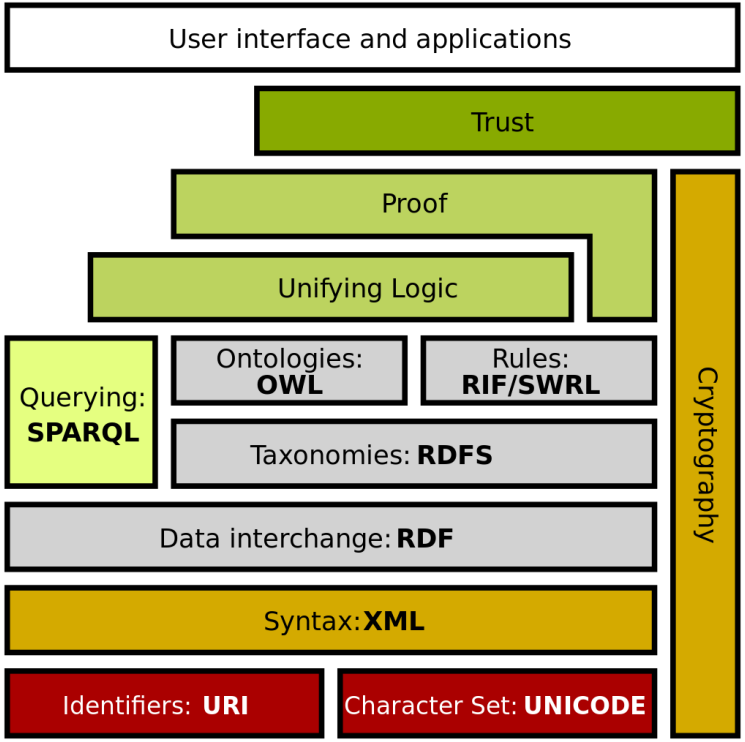
Provenance

The source or origin of an object; its history



Semantic Web

Machine Understandable



Source:Wikimedia Commons

Open Science Contributions for Supporting Reproducibility

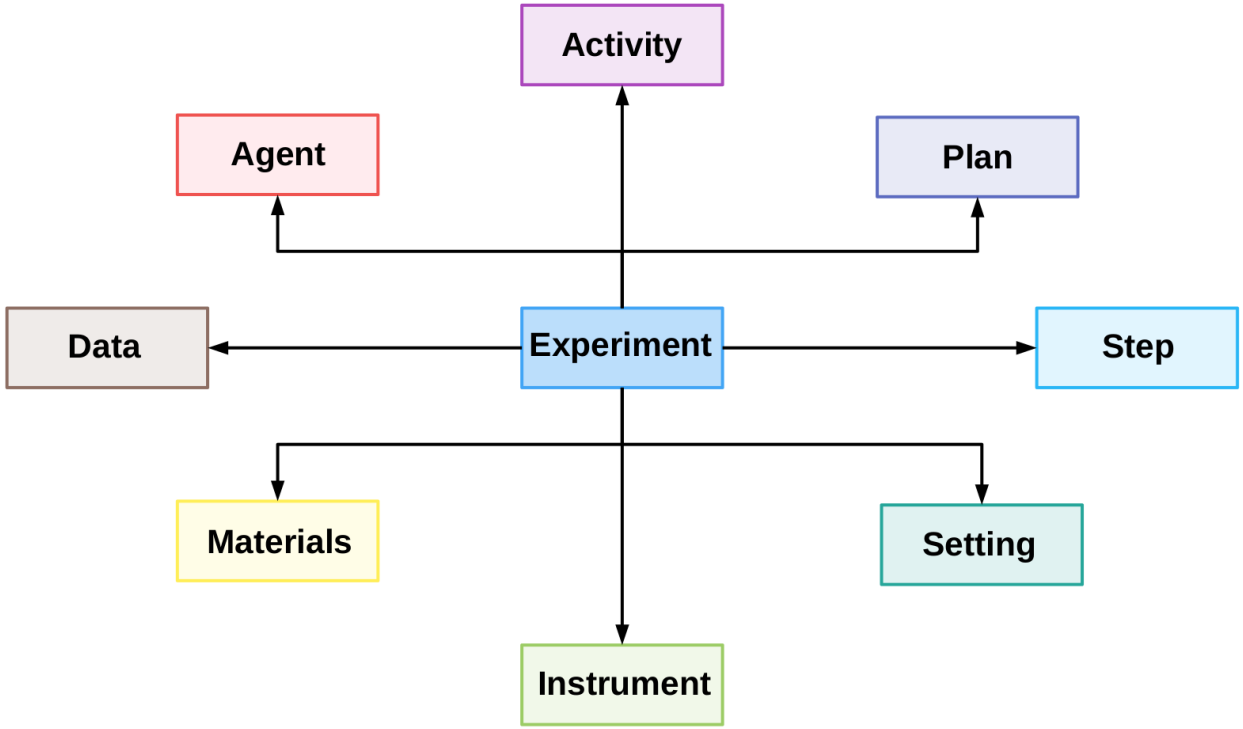
1	Semantic Description of Scientific Experiments	➢ **The REPRODUCE-ME Data Model & Ontology** Semantic Description of Scientific experiments including Scripts and Computational Notebooks ➢ **The ReproduceMeON** An Ontology Network for Reproducibility of Scientific Studies
2	Support of computational reproducibility	➢ **ProvBook** Provenance capture, visualize, represent and difference of results ➢ **MLProvLab** Provenance capture, visualize, represent and difference of ML notebooks ➢ **ReproduceMeGit** A tool for analyzing the reproducibility of Jupyter Notebooks
3	End-to-end provenance management of scientific experiments	➢ **CAESAR** Support of a collaborative environment with visualization of the complete path of a scientific experiment

The REPRODUCE-ME Data Model & Ontology

The REPRODUCE-ME Data Model

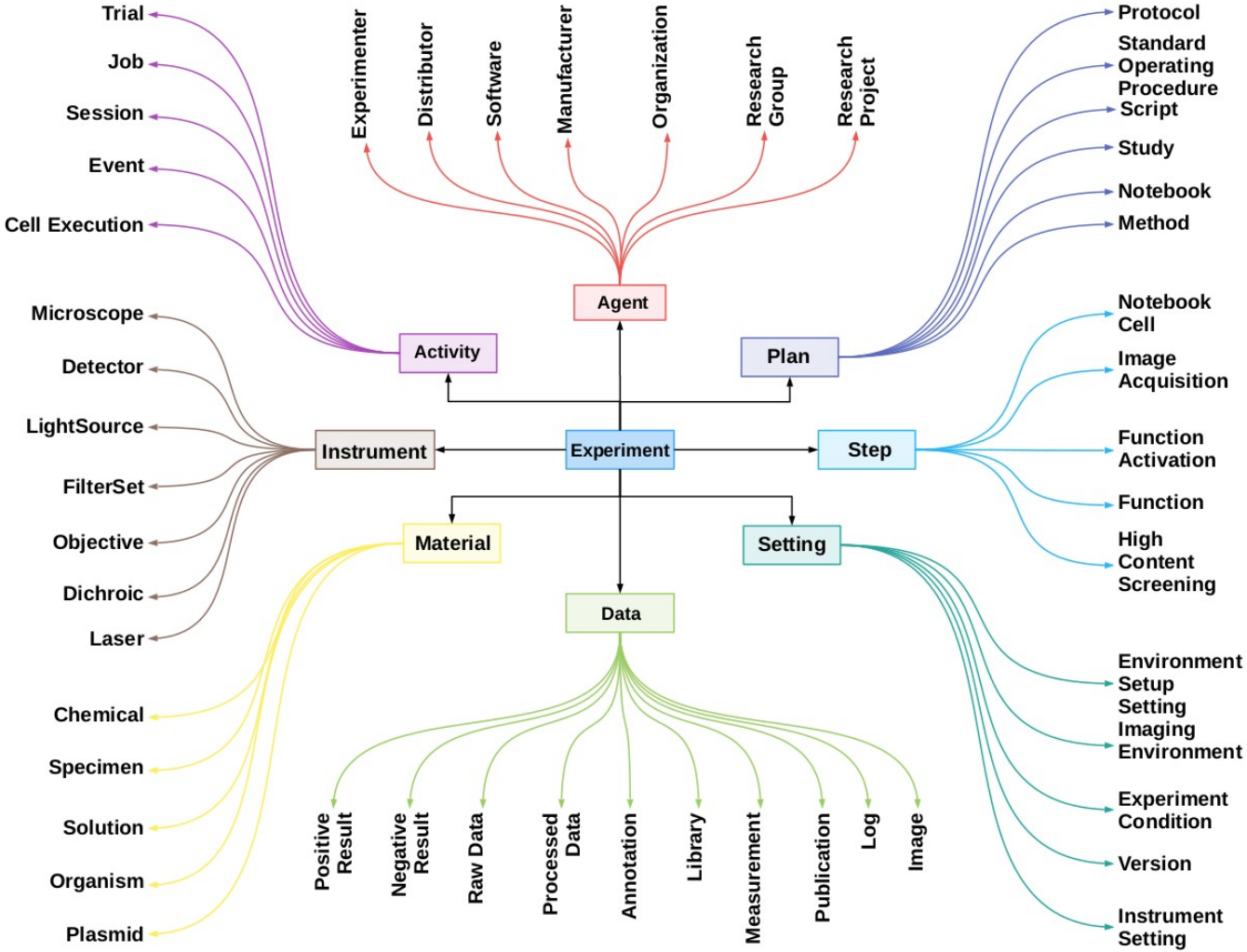
Experiment is defined as an n-tuple

E = (Data, Agent, Activity, Plan, Step, Setting, Instrument, Material)

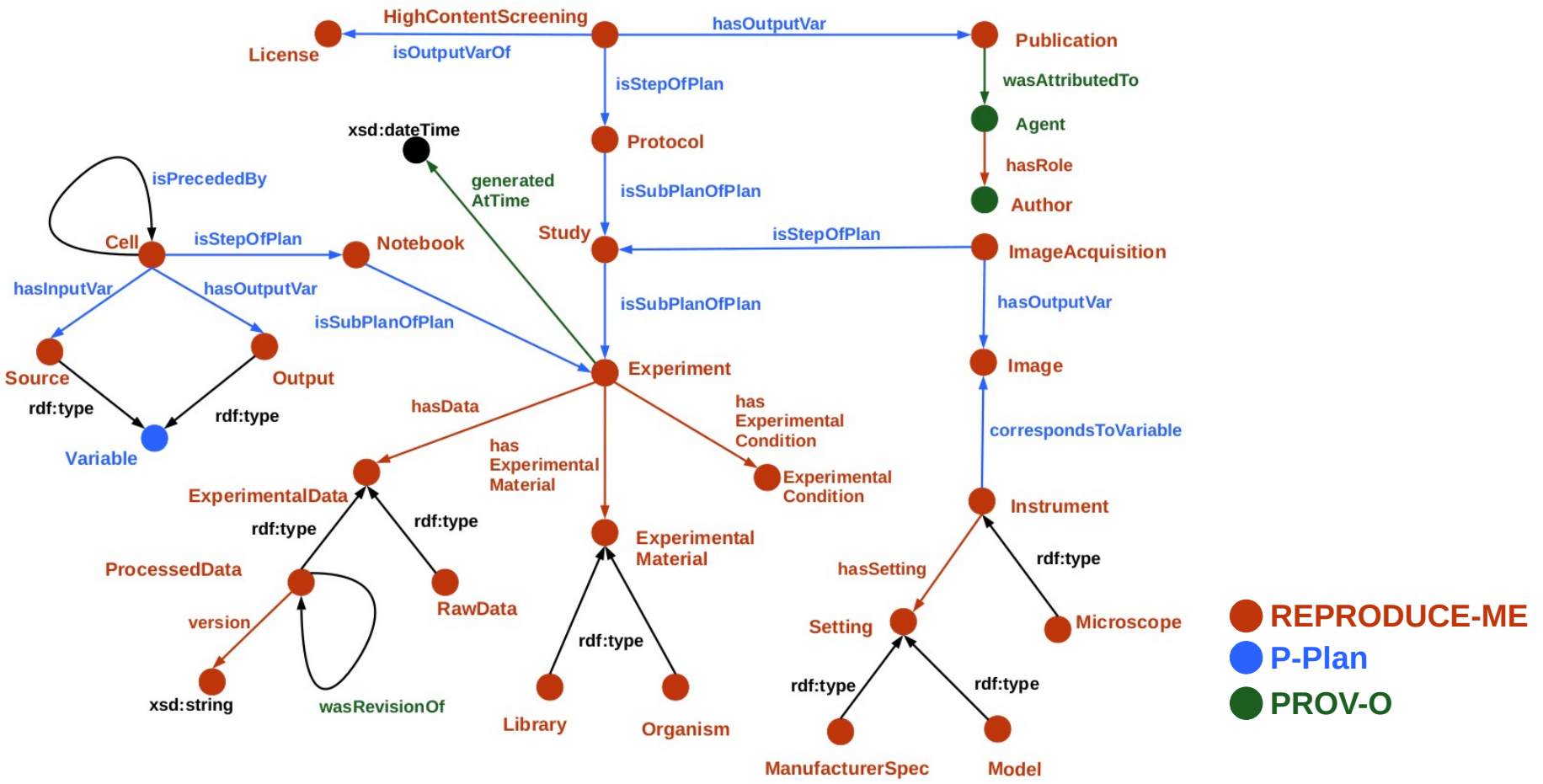


[Samuel, 2019]

The REPRODUCE-ME Data Model



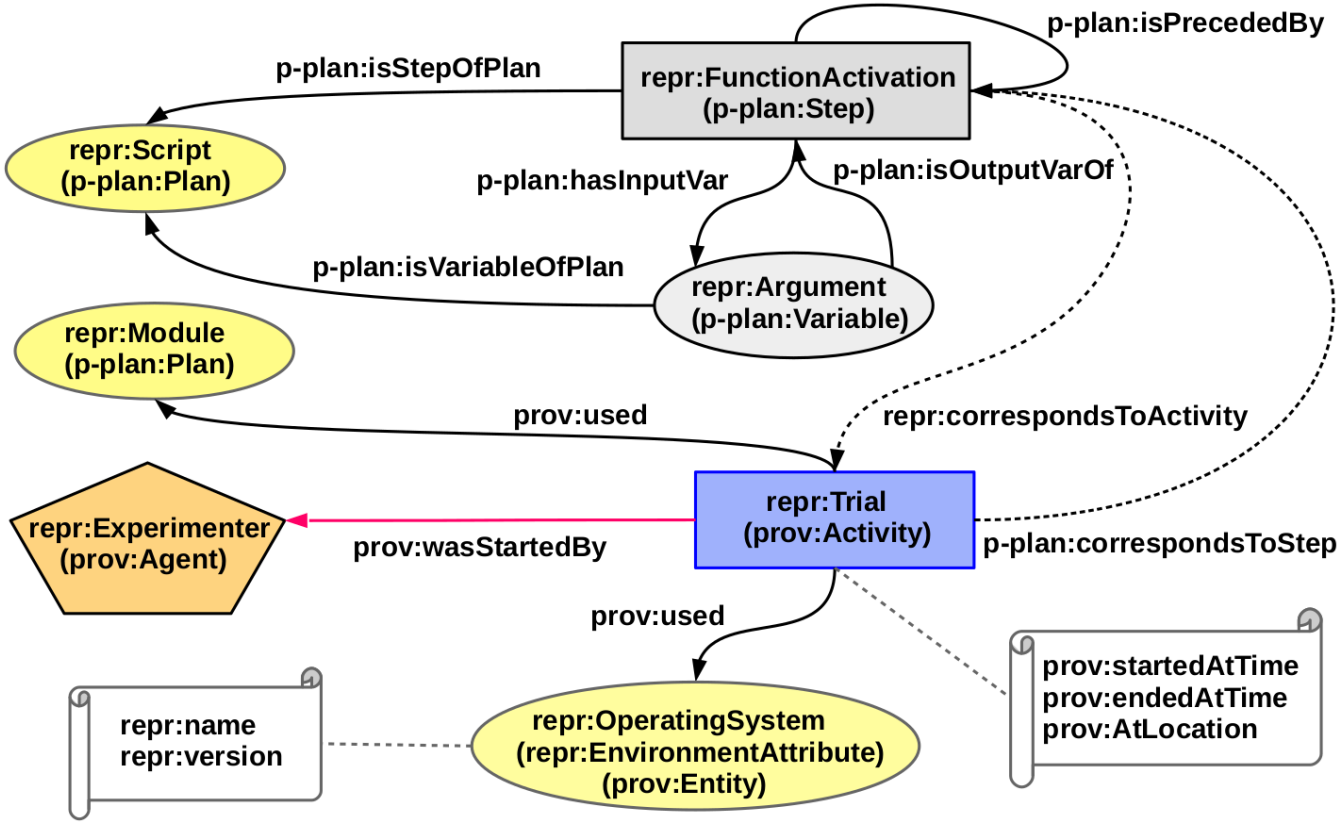
The REPRODUCE-ME Ontology



[Samuel and König-Ries 2017, Samuel and König-Ries 2018a, Samuel et al. 2018]

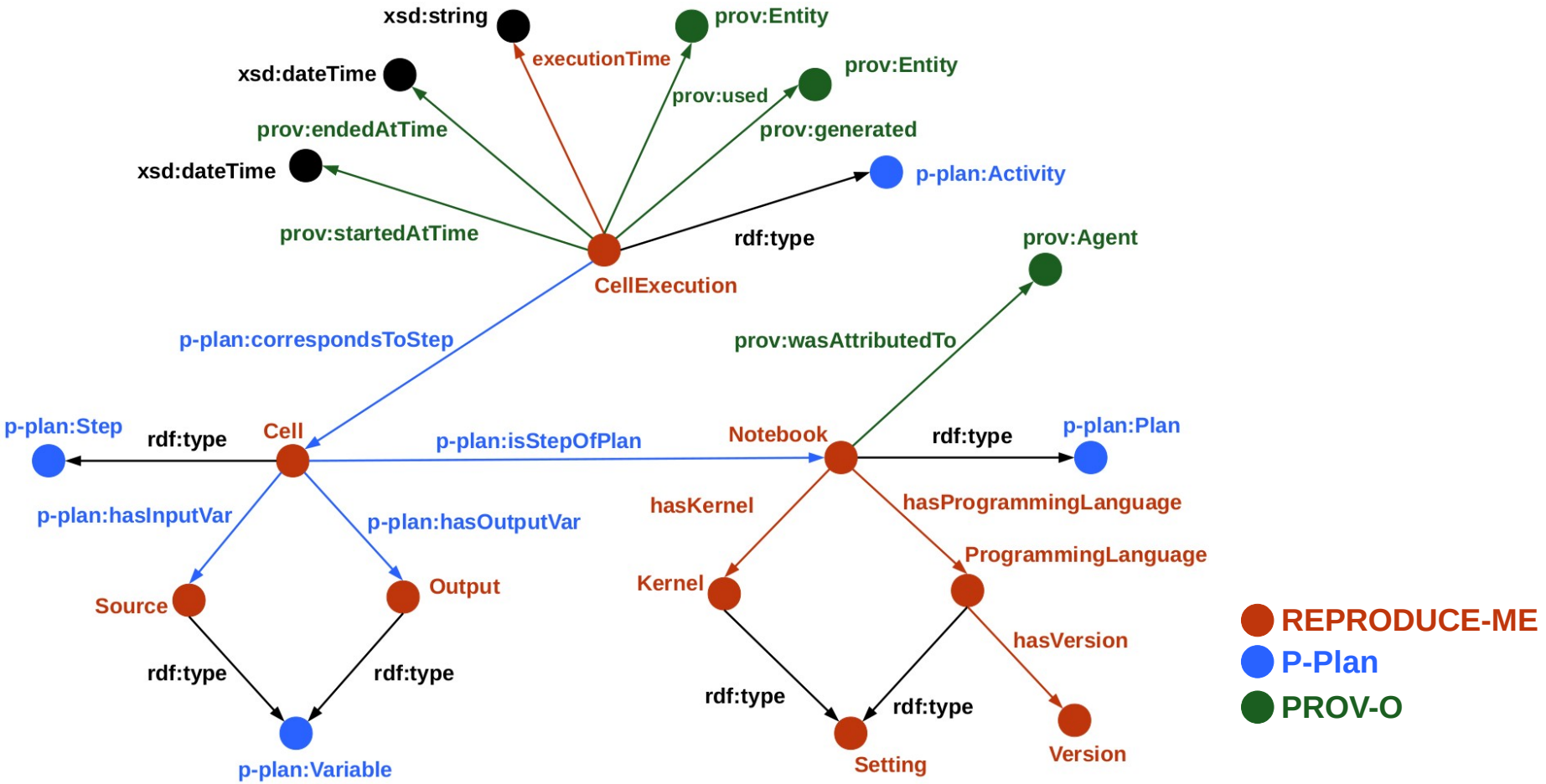
Script Provenance

Ontology term	Remarks
repr:Script	*p-plan:Plan*
repr:Function	*p-plan:Plan*
repr:Module	*p-plan:Plan*
repr:Version	*repr:Setting*
repr:Argument	*p-plan:Variable*
repr:Input	*p-plan:Variable*
repr:Output	*p-plan:Variable*
repr:FunctionActivation	*p-plan:Step*
repr:Trial	*p-plan:Activity*
repr:OperatingSystem	*p-plan:Setting*
repr:Author	*prov:Person*
prov:startedAtTime	Data property
prov:endedAtTime	Data property
p-plan:isPrecededBy	Object property



Overview of the ontology terms to model script provenance

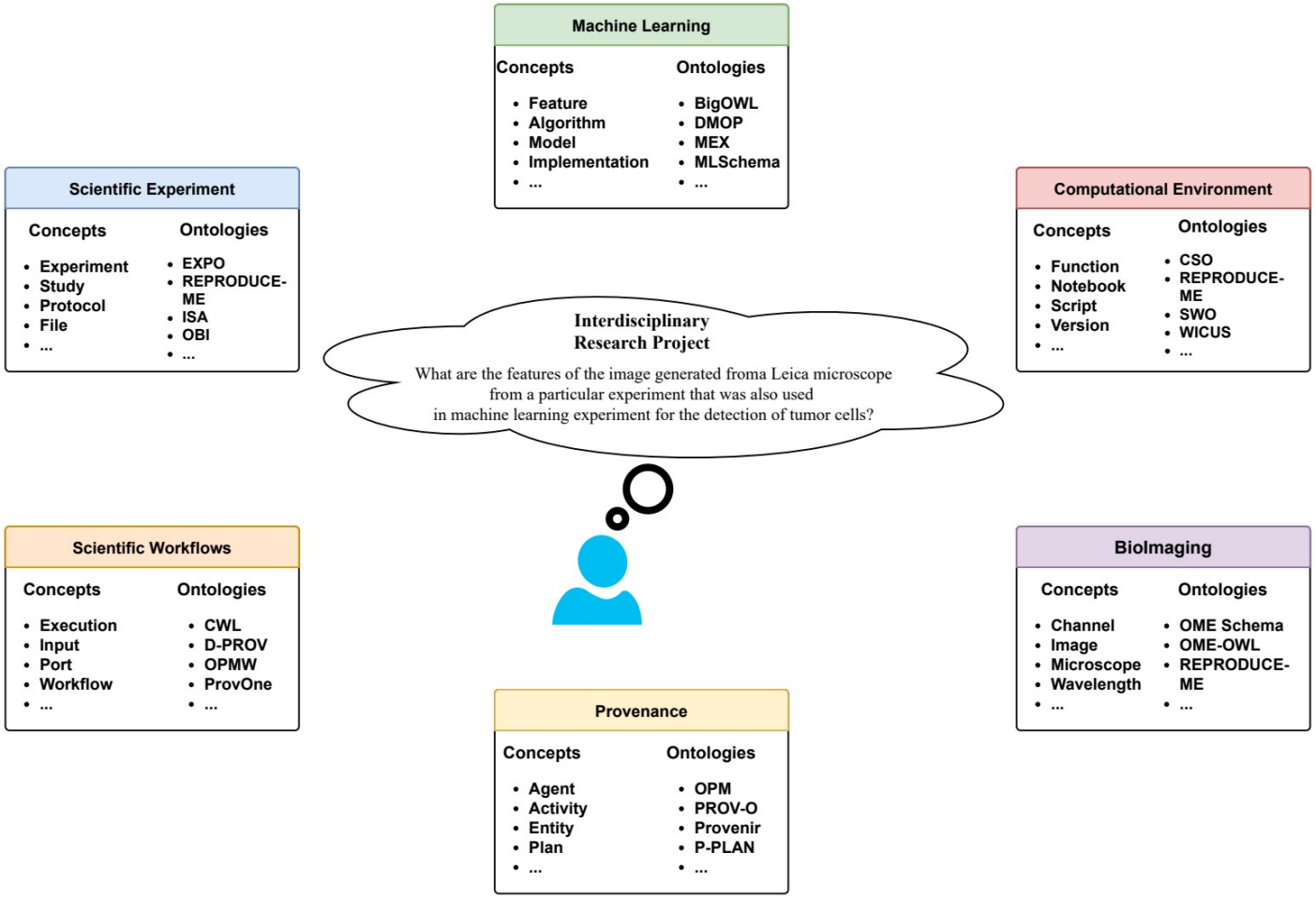
Computational Provenance



[Samuel and König-Ries 2018a, Samuel and König-Ries 2018b]

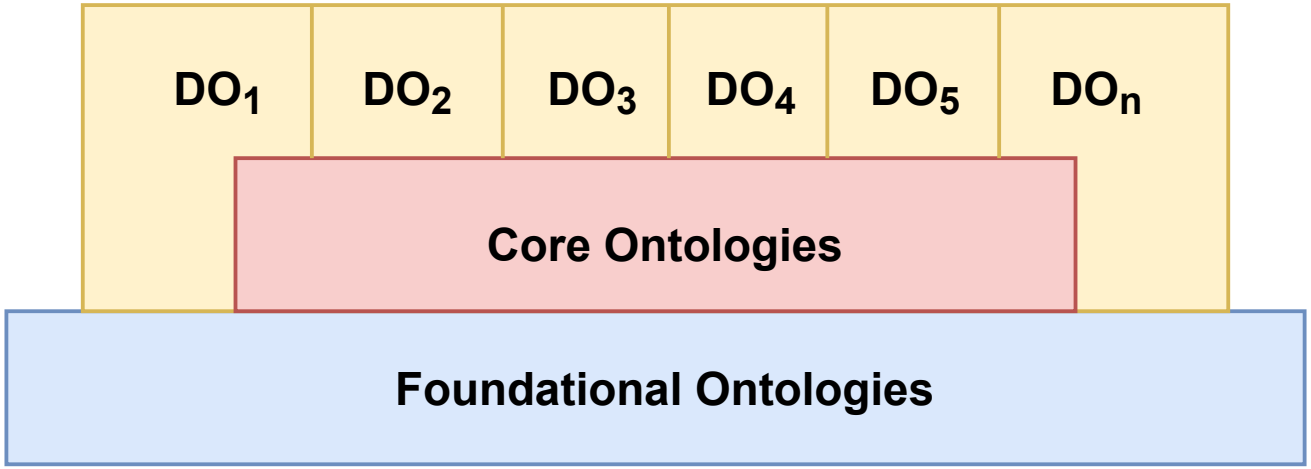
The ReproduceMe Ontology Network (ReproduceMeON)

Reproducibility related area assimilation



ReproduceMeON

- A novel approach that brings together knowledge from several domains, such as ML, provenance, and scientific computing, based on the three-layered architecture



Towards an Ontology Network for the reproducibility of scientific studies., Samuel et al. 2021

Computational Reproducibility

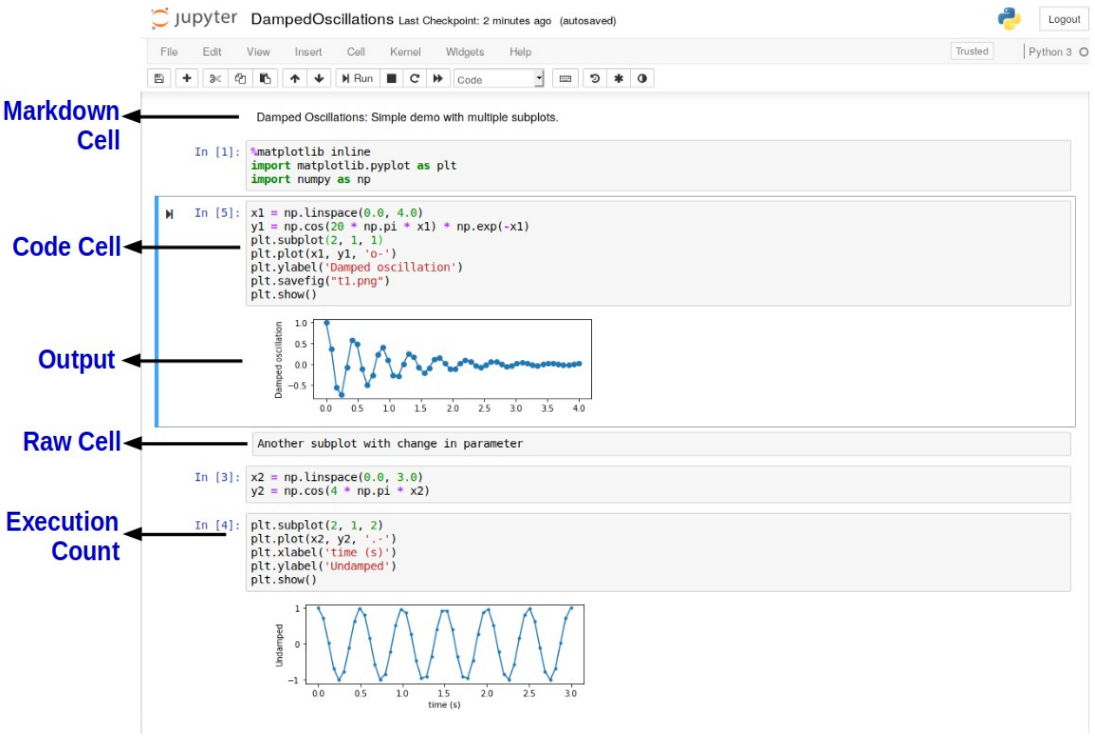
Computational Notebooks

- Share code along with documentation
  - Project Jupyter
  - RStudio

Jupyter Notebook Facts

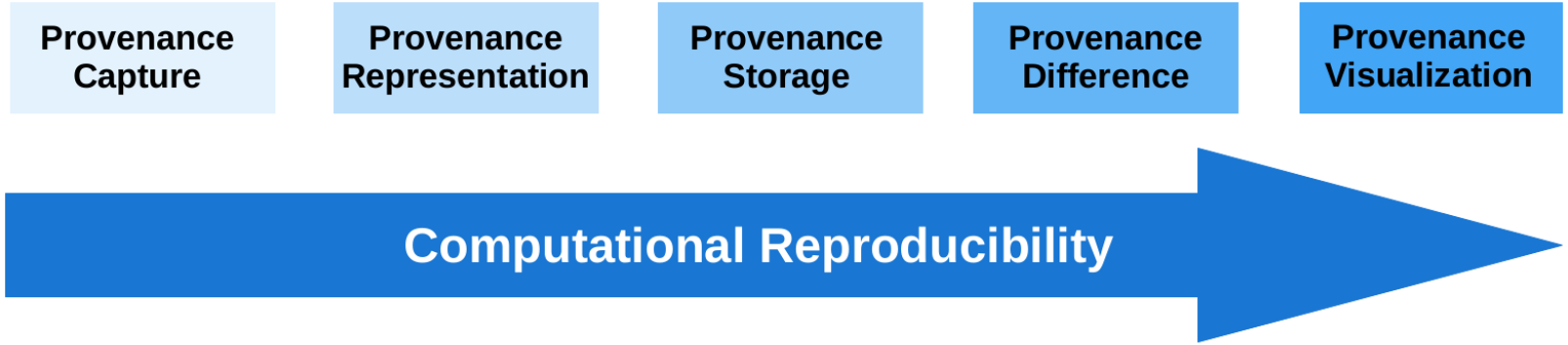
- Formerly known as *IPython* Notebook
- *1.7* million Jupyter notebooks on Github
- Millions of *users*
- Different *computational kernels* including Python, R, and MATLAB
- Export in different *formats* like HTML, LaTeX, PDF

Structure of a Jupyter Notebook



Computational Reproducibility

- Provenance support is limited [Rule et al., 2018, Pimentel et al., 2019]
  - Tracking provenance when the cells are over-written and re-run
  - Track how exactly a final result has been achieved
  - Track of the experiments that have been attempted
- "Record all intermediate results in a standardized format"
  - One of the ten simple rules for computational reproducible research [Sandve et al., 2013]



The key components for the end-to-end provenance management for computational reproducibility

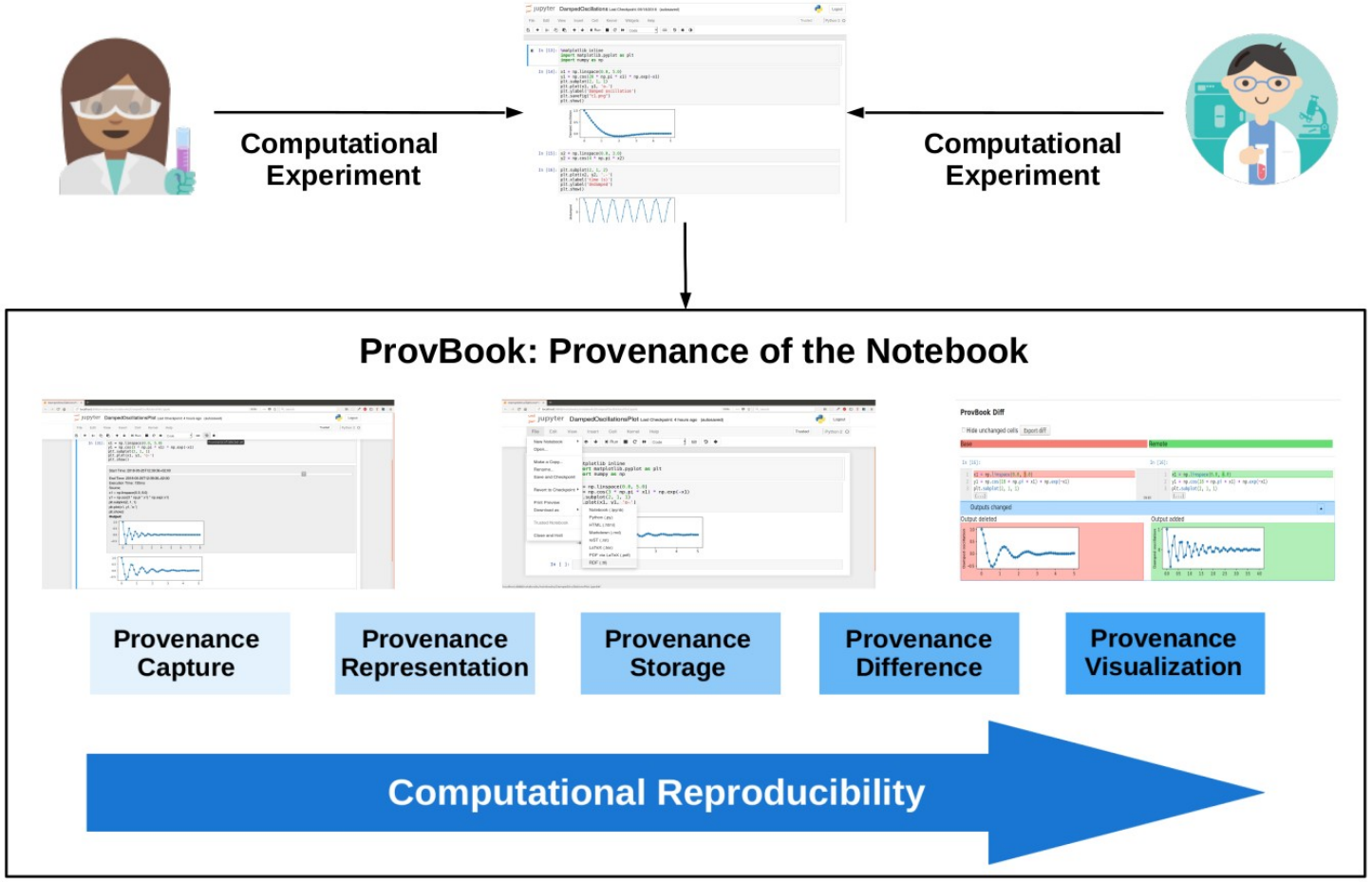
Design Goals

- Support the provenance lifecycle
  - Tracking
  - Storage
  - Querying
  - Compare
  - Visualization

Design Goals

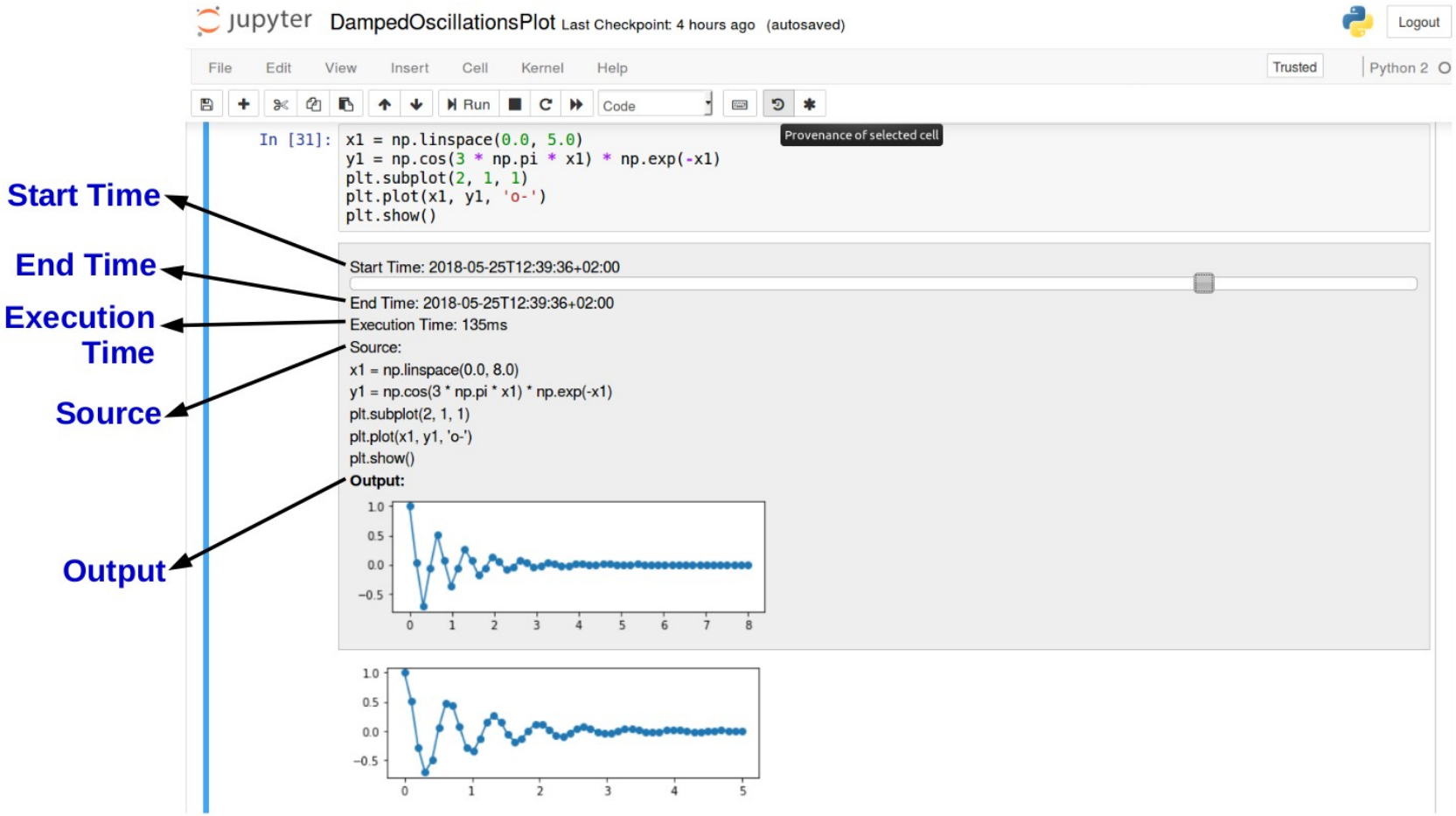
- Support
  - Reproducibility
  - Collaboration
  - Semantic annotation and interoperability
  - Exporting provenance in different formats
  - Extensibility
- Ease of use

ProvBook: Provenance of the Notebook



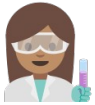
ProvBook: Provenance-based Semantic Enrichment of Interactive Notebooks for Reproducibility, Samuel et al., 2018

ProvBook: Capture & Visualization

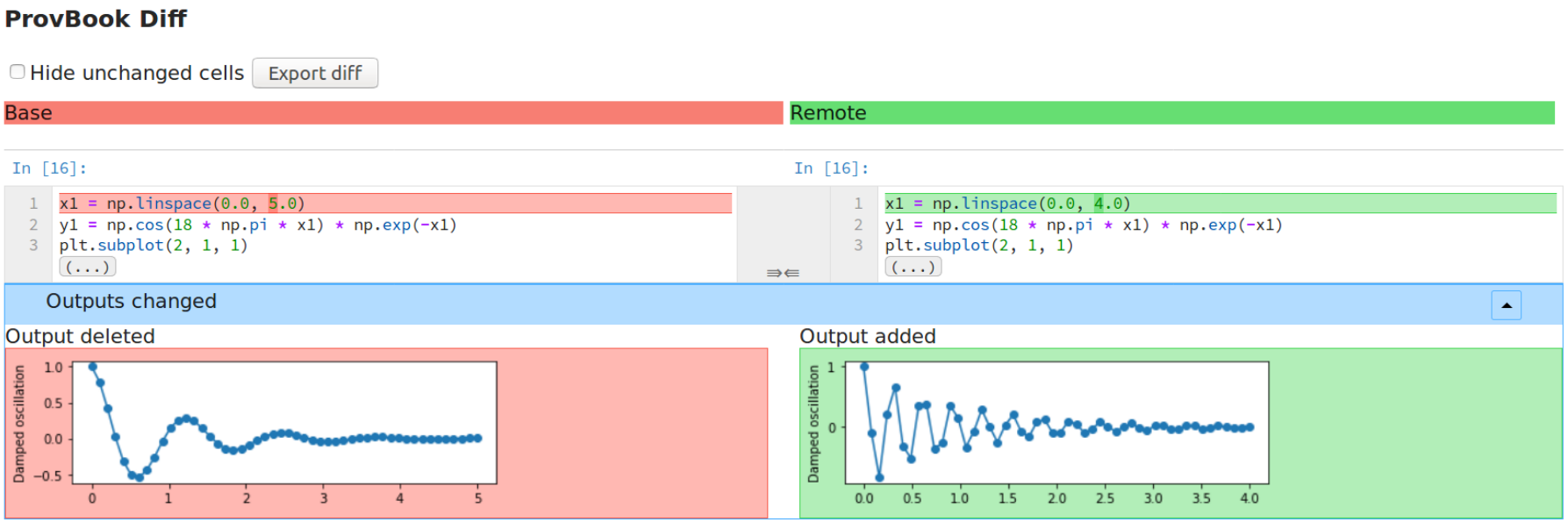


ProvBook: Difference

- A provenance difference module to compare the different executions of a notebook
- Comparison of the input and the output

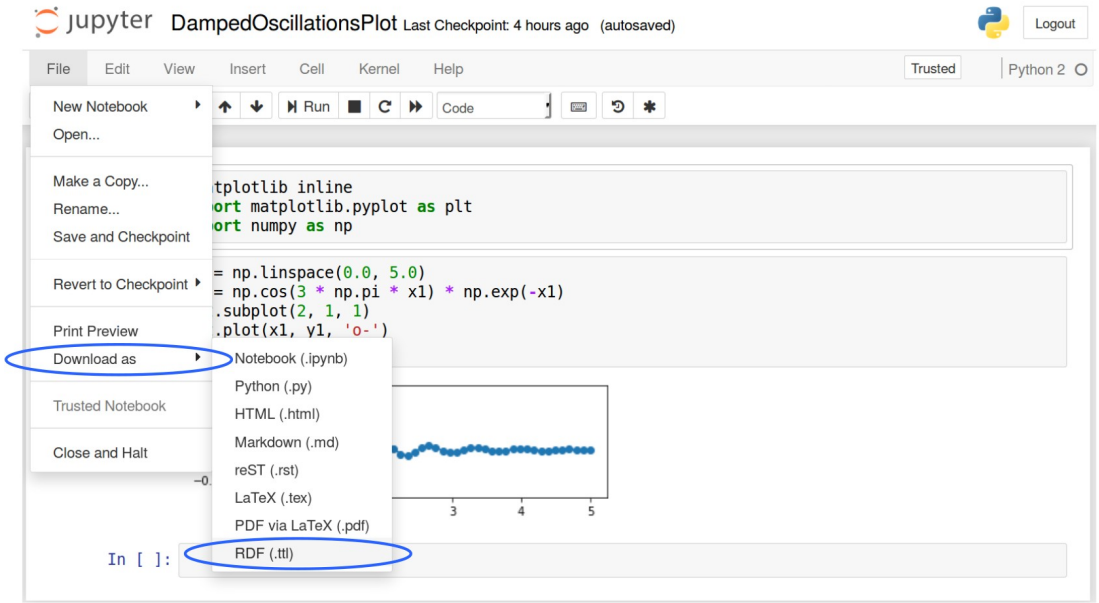


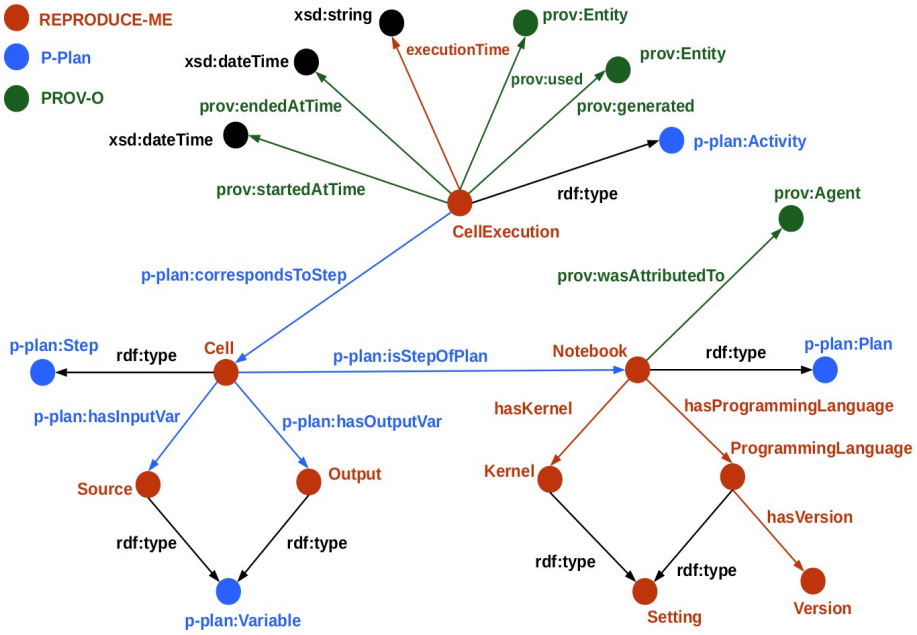
- Starting time to differentiate between two executions
- Extends the nbdime library from the Project Jupyter



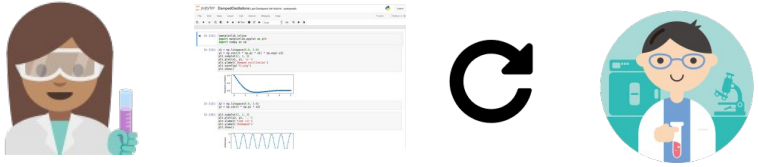
ProvBook: Semantic Representation

- Jupyter Notebook and its provenance described using the REPRODUCE-ME ontology
- **New extension**: Export in RDF from the user interface or command line

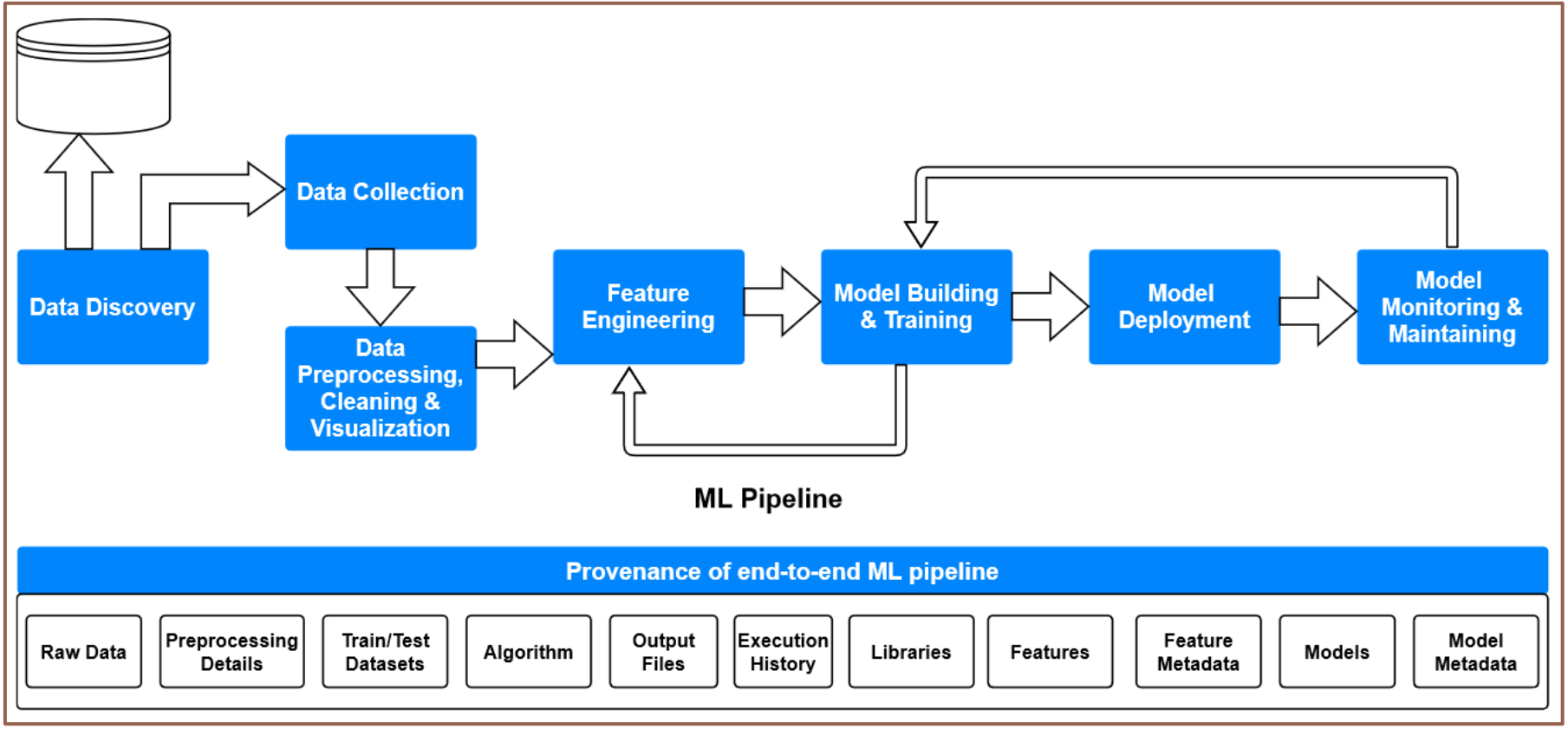




[Samuel and König-Ries 2018a]

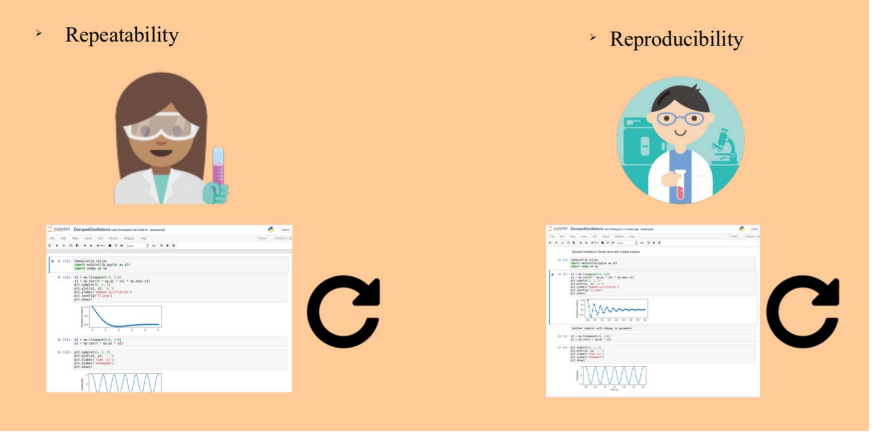


Provenance of ML Pipelines



MLProvLab

- Provenance management of end-to-end ML pipelines in a notebook environment
- An extension to JupyterLab
- This tool tracks, compares, manages, and visualizes provenance information
- Track, at runtime, the datasets, variables, libraries, and functions used in the notebook and their dependencies between cells.



Provenance Capture

Dependencies betweeen cells

Imported and Used Modules

Execution Information

```
[1]: import numpy as np
     from pandas import read_csv
     from sklearn.model_selection import train_test_split
     from sklearn.metrics import classification_report
     from sklearn.metrics import confusion_matrix
     from sklearn.metrics import accuracy_score
     from sklearn.svm import SVC

[2]: # Load dataset
     url = "https://raw.githubusercontent.com/jbrownlee/Datasets/master/iris.csv"
     names = ['sepal-length', 'sepal-width', 'petal-length', 'petal-width', 'class']
     dataset = read_csv(url, names=names)
     print(dataset)
```

Variable, Class, and Function Definitions

Code

Outputs

Function and Object Calls

Program Structure (Loops, Conditions)

Modules and Libraries

evaluation_notebook.ipynb

Download MLProvLab

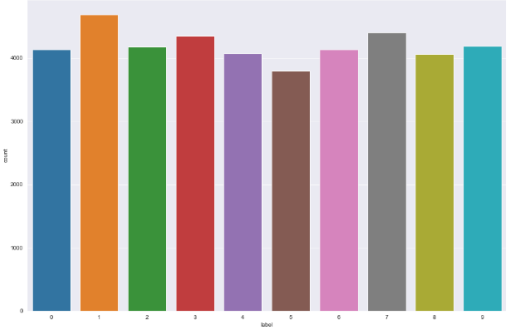
[5]:
```
X=df.drop("label",axis=1).values
y=df["label"].values
```

[6]:
```
print(X.shape)
print(y.shape)
print(test.shape)
```

```
(42000, 784)
(42000,)
(28000, 784)
```

[7]:
```
import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(15,10))
sns.set_style("darkgrid")
sns.countplot(x="label",data=df)
```

[7]:
```
<AxesSubplot:xlabel='label', ylabel='count'>
```



[8]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(
print(X_train.shape)
print(X_test.shape)
print(y_train.shape)
print(y_test.shape)
```

```
(39900, 784)
(2100, 784)
```

Provenance: evaluation_note

Options | Export | Environment info | Import info | Code info | General info | Help

Info about imports and modules in epoch **4**

Imports from module **keras.preprocessing.image**:

Import **ImageDataGenerator** was used **1** times

Imports from module **keras.utils.np_utils**:

Import **to_categorical** was used **1** times

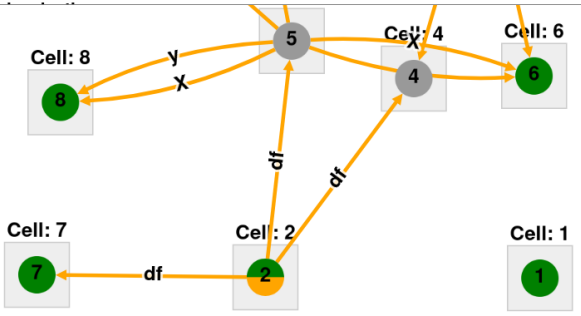
Import **matplotlib.pyplot** with alias **plt** was used **3** times

Import **numpy** version 1.19.5 with alias **np** was used **0** times

Import **os** was used **2** times

Import **pandas** version 1.2.3 with alias **pd** was used **3** times

Import **seaborn** version 0.11.1 with alias **sns** was used **2** times



Epoch **4/4** Thu, 24 Jun 2021 10:19:30 GMT

Execution **19/19** Thu, 24 Jun 2021 10:26:21 GMT

Datasets

Provenance: evaluation_note

Options | Export | Environment info | Import info | Code info | General info | Help

Source **D:/Projects/mnist-evaluation/data** was first used in execution **1** in variable

Source **D:/Projects/mnist-evaluation/data/train.csv** was first used in execution **2** in variable **df**

Source **D:/Projects/mnist-evaluation/data/test.csv** was first used in execution **3** in variable **test**

Execution Environment

Provenance: evaluation_not

Options | Export | Environment info | Import info | Code info | General info | Help

Environment information of epoch 3

Language: python

Version: 3.9.2

Mimetype: text/x-python

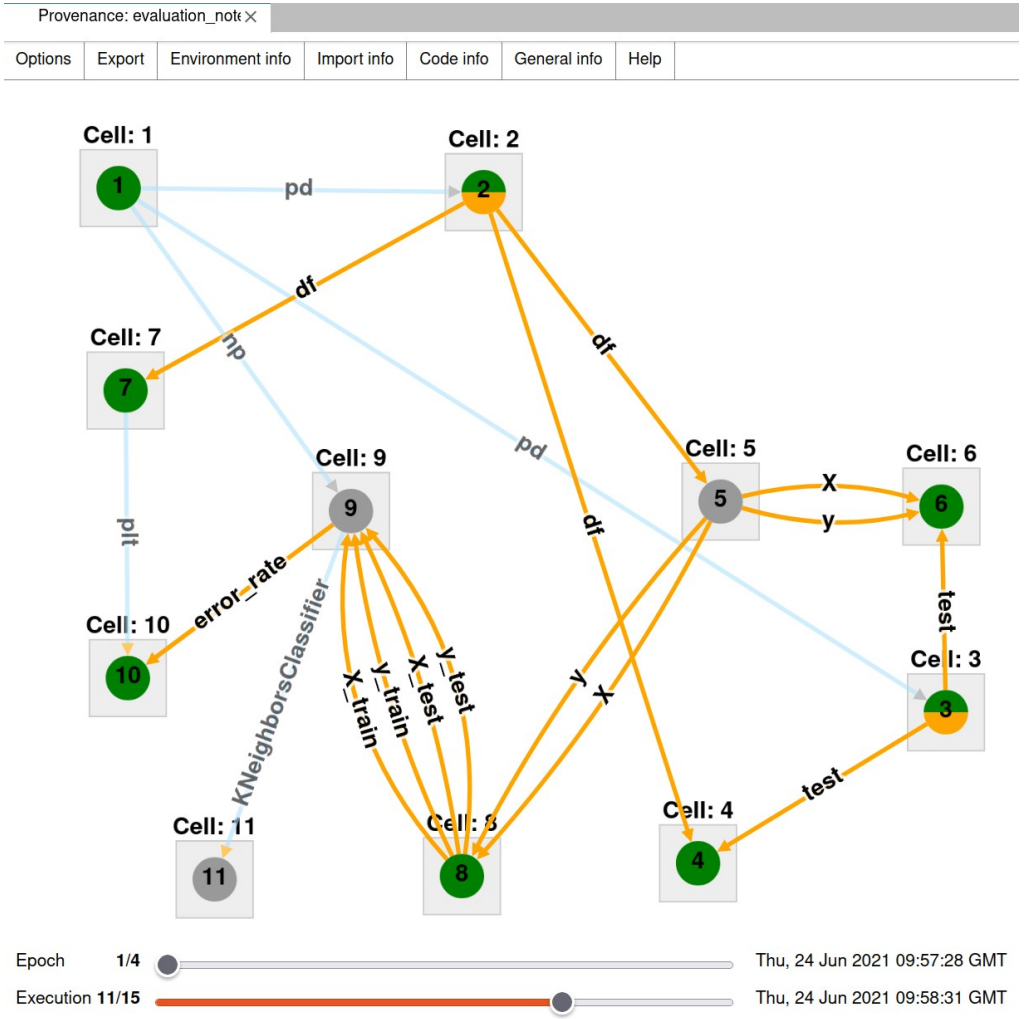
Kernel start time: Thu, 24 Jun 2021 10:17:35 GMT

Kernel implementation: ipython

Kernel version: 7.22.0

User agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:89.0) Gecko/20100101 Firefox/89.0

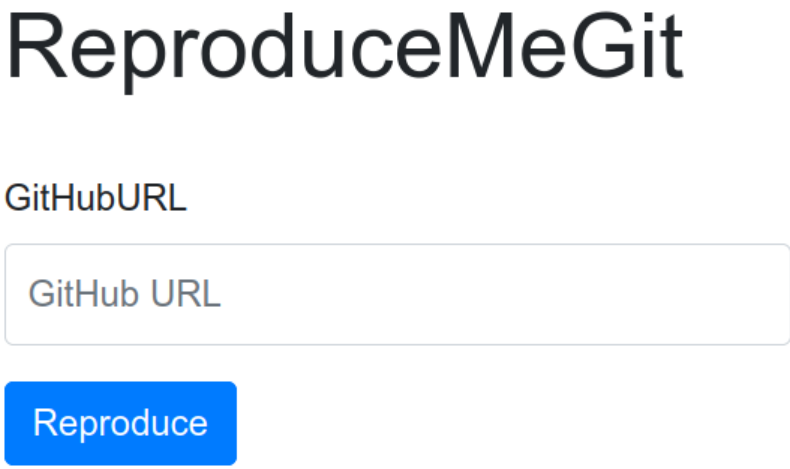
MLProvLab: Provenance Visualization



ReproduceMeGit

ReproduceMeGit

- A visualization tool for analyzing the reproducibility of Jupyter Notebooks.
- Goals:
  - Help repository users and owners to reproduce, directly analyze and assess the reproducibility of notebooks
  - Get information on notebooks
    - that were successfully reproducible
    - that resulted in exceptions during runs
  - Analyze the notebooks:
    - the difference in the results from the original notebooks
    - provenance history of runs



An Overview

Reproducibility Study

Notebooks (un-)successfully finishing the executions

Notebooks with same or different results compared to the original.

Exceptions occurred in the runs
ImportError, ModuleNotFoundError
FileNotFoundError, IOError, SyntaxError

Provenance History in RDF using REPRODUCE-ME ontology

Direct access to Binder and ProvBook

ReproduceMeGit

Structure & Usage

Repository Overview

Notebook Overview

Cells Overview

Modules Overview

Distribution of programming Languages and the versions used

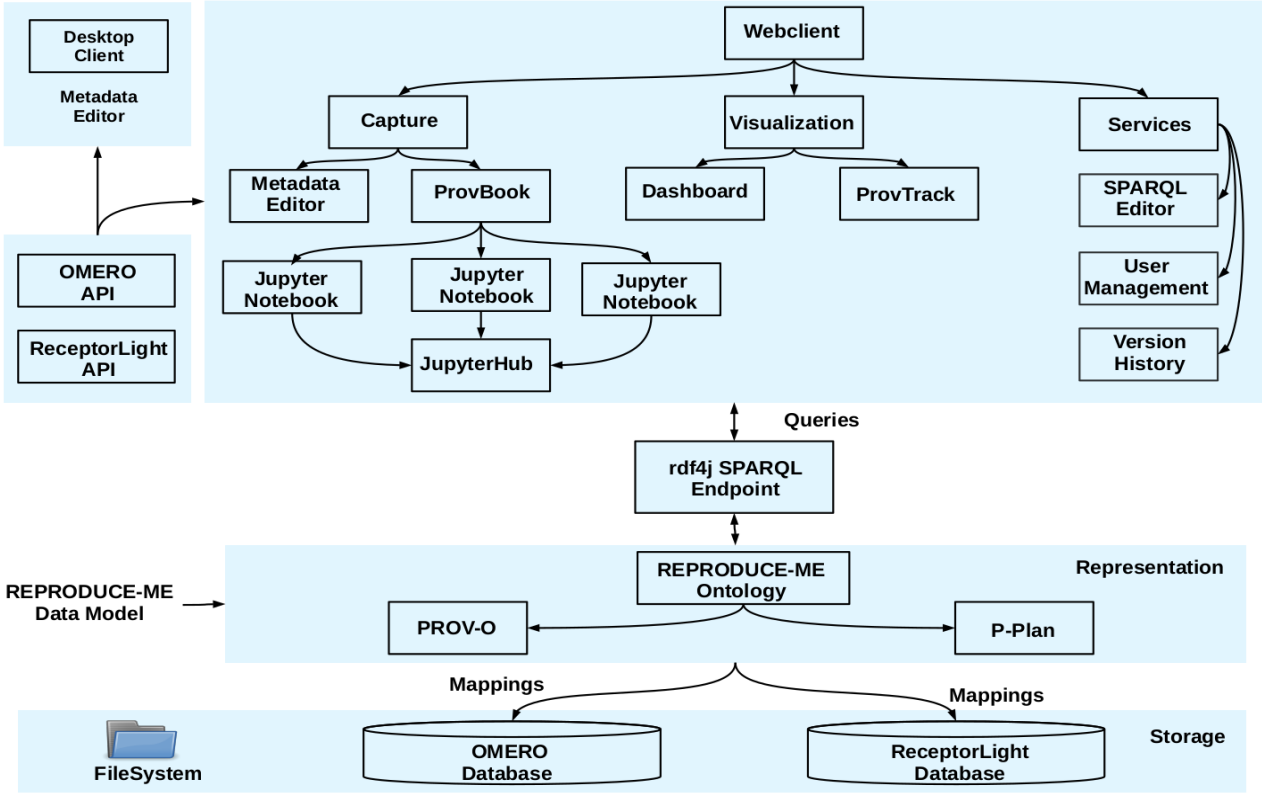
CAESAR

A Collaborative Environment for Scientific Analysis with Reproducibility

CAESAR

CollAborative Environment for Scientific Analysis with Reproducibility

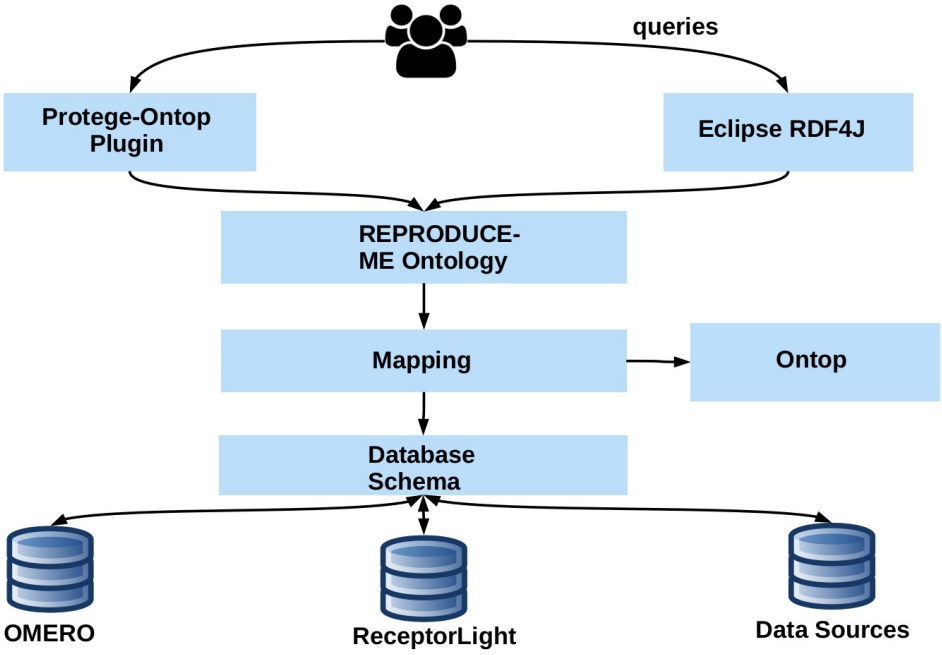
- Extends OMERO
- Experimental data management with images.
- Form-based provenance capture system
- Link Experiments with its associated variables
- User and Group management
- Suggestions on other user's experimental data.
- Version history of an experiment
- Reuse of experiments



[Samuel et al. 2017, Samuel et al., 2018]

CAESAR: Provenance Representation

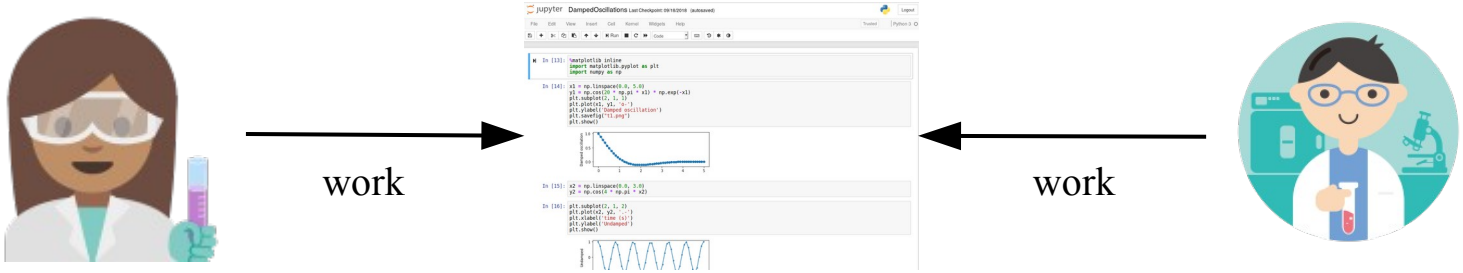
- Ontology-based data access
  - Data Sources
  - Ontologies
  - Federation
  - Mappings
- Around 800 mappings to create the virtual RDF graph



[Samuel and König-Ries 2017]

CAESAR: Computational Reproducibility

- Integration of JupyterHub (https://jupyter.org/hub)
  - A distributed, collaborative and multi-user research environment
  - Directly work with the images and other datasets linked to an experiment in CAESAR
- Integration of ProvBook
  - The Notebook is linked to the experiment that used them
- Create a knowledge graph of the provenance of experiments with their computational and non-computational steps.



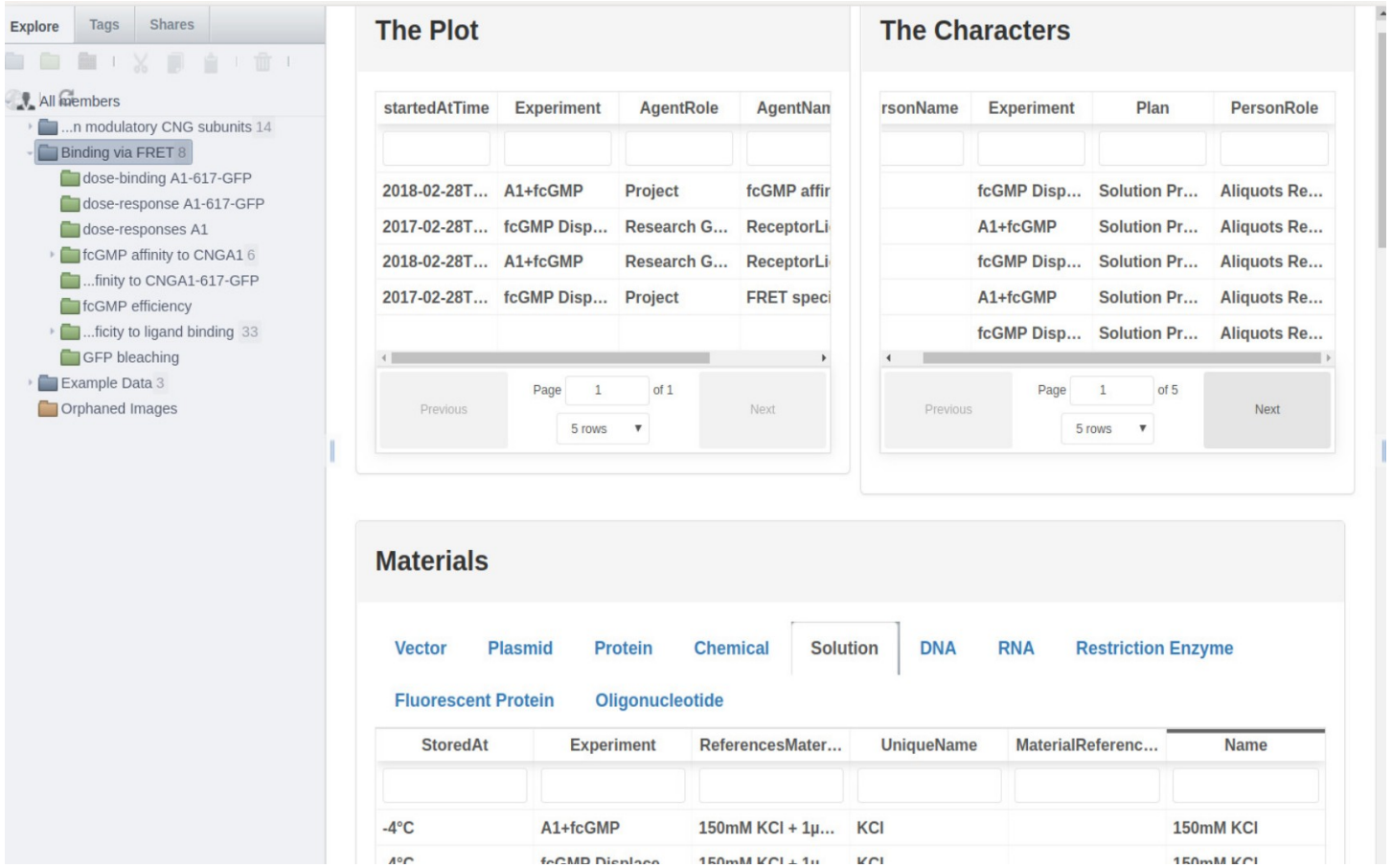
Project Dashboard: Provenance Visualization

Features

- Visualization of provenance data at the project level
- A panel for each component of a story

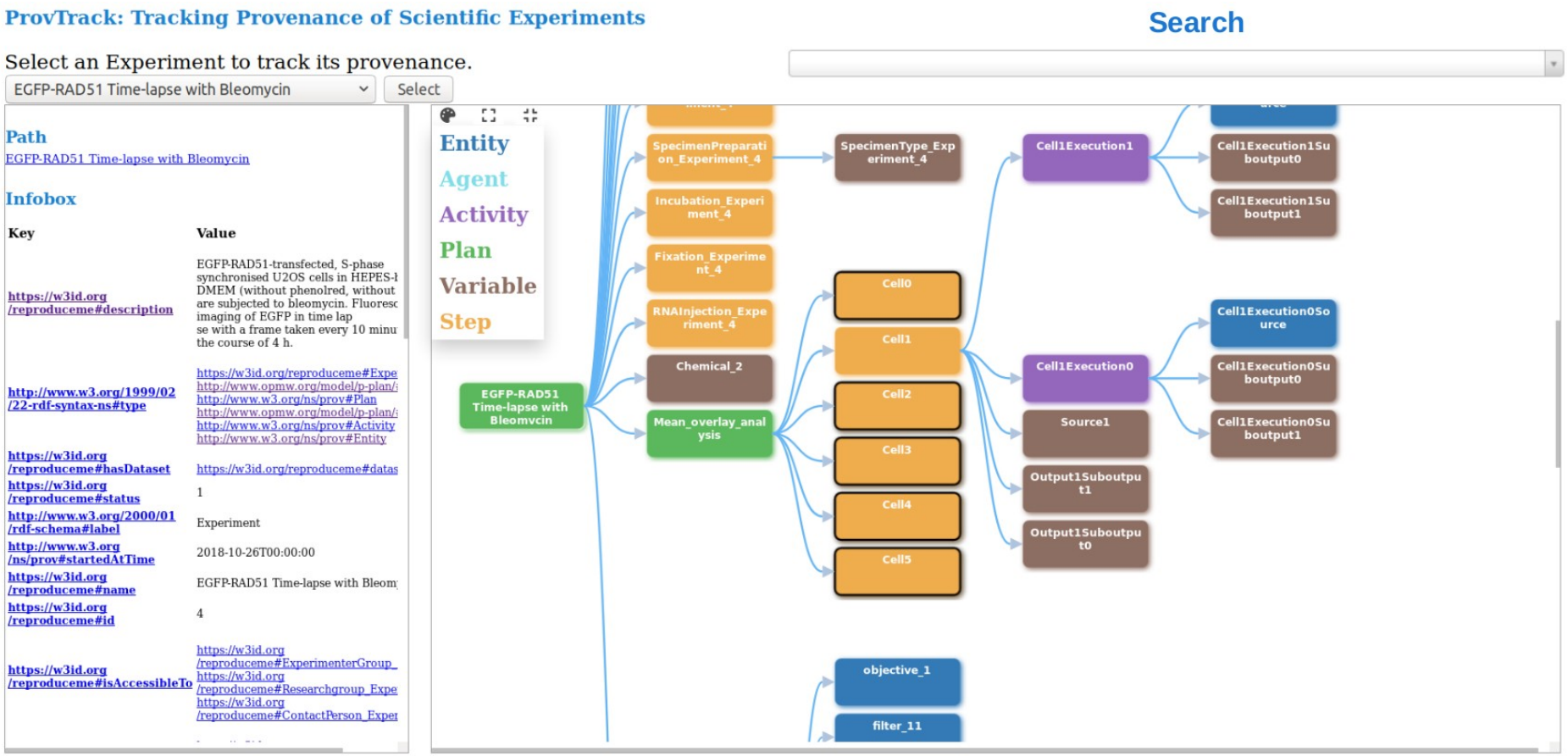
Panels

- External Resources
- Steps
- Devices
- Settings
- Jupyter Notebooks
- Results



[Samuel et al., 2018]

ProvTrack: Provenance Visualization



InfoBox

Provenance Graph

Conclusion

- Reproducibility: an important concern and needs much attention
- Open Science for Reproducibility
- Different tools to support reproducibility with provenance and semantic web

Thank you for your attention

@sheebasamuel

0000-0002-7981-8504

- https://fusion.cs.uni-jena.de/
- https://w3id.org/reproduceme/research
- https://sheeba-samuel.github.io/

FRIEDRICH-SCHILLER-UNIVERSITÄT JENA

FUSION

MichaelStifelCenterJena
for Data-Driven and Simulation Science

Carl Zeiss Stiftung