Everware toolkit

supporting reproducible science and challenge-driven education

Tim Head, Igor Babuschkin[3], Alexander Tiunov[2], Andrey Ustyuzhanin[1,2]

2016-10-11, CHEP

[1]Yandex School of Data Analysis, [2]Higher School of Economics NRU, [3]University of Manchester

Irreproducibility indicators

- ‘Which version of my code I used to generate figure 13?’
- ‘The new student wants to reuse that model I published three years ago but he can’t reproduce the figures’
- ‘I thought I’ve used the same parameters but I’m getting different results…’
- ‘Which dataset did I use to compare algorithms?’
- ‘Why did I do that?!’
- ‘It worked yesterday!!’

Cases in point: Medical science

Amgen (a commercial company) in 2012

- 53 landmark papers in cancer drug development
- Scientific findings confirmed only in 6 (11%) cases

Bayer (a commercial company) in 2011

- 67 projects
- Results confirmed in 20-25% cases

A new study is under way and to be completed in 2017

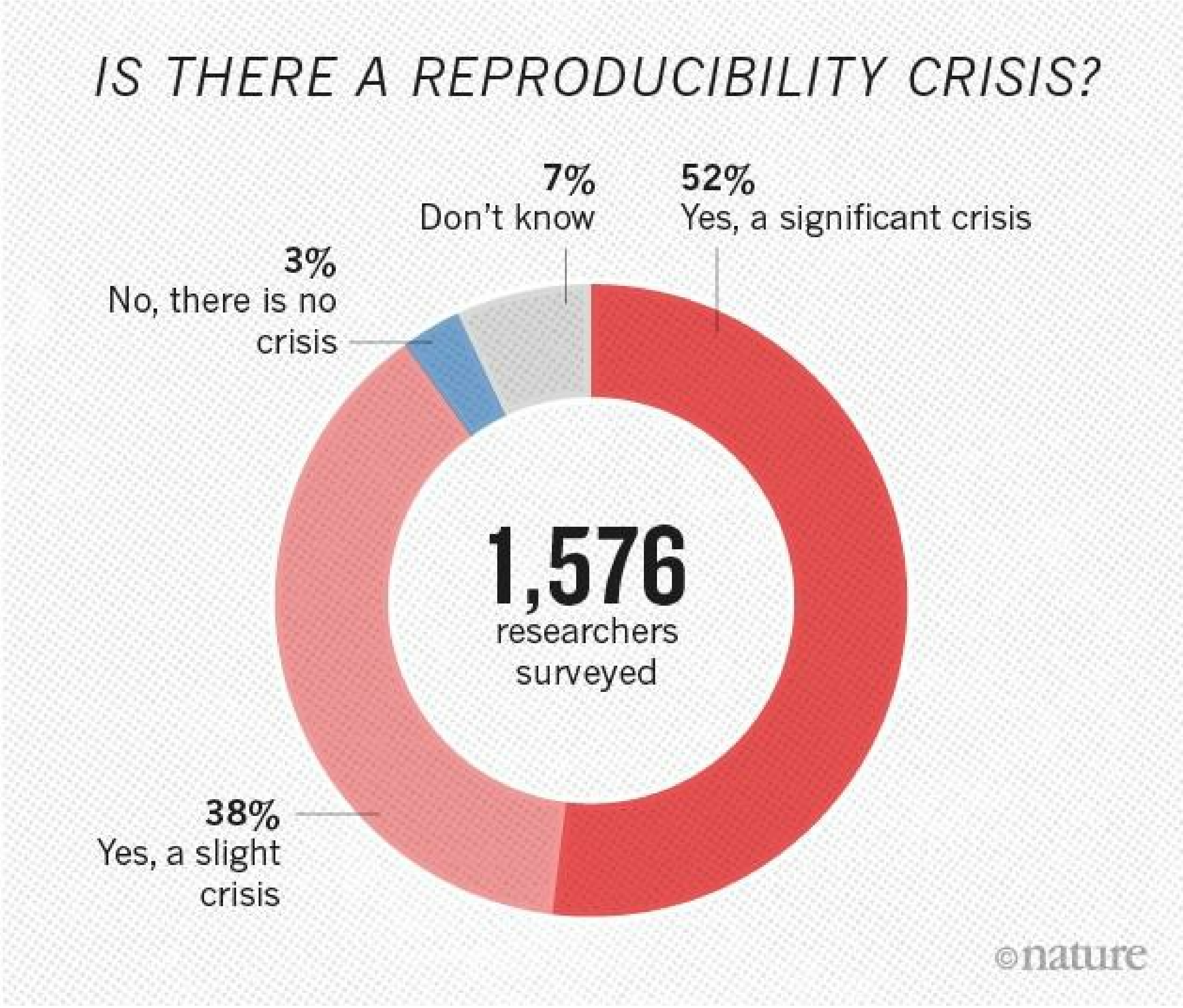
- https://osf.io/e81xl/wiki/home/

http://www.nature.com/nature/journal/v483/n7391/full/483531a.html

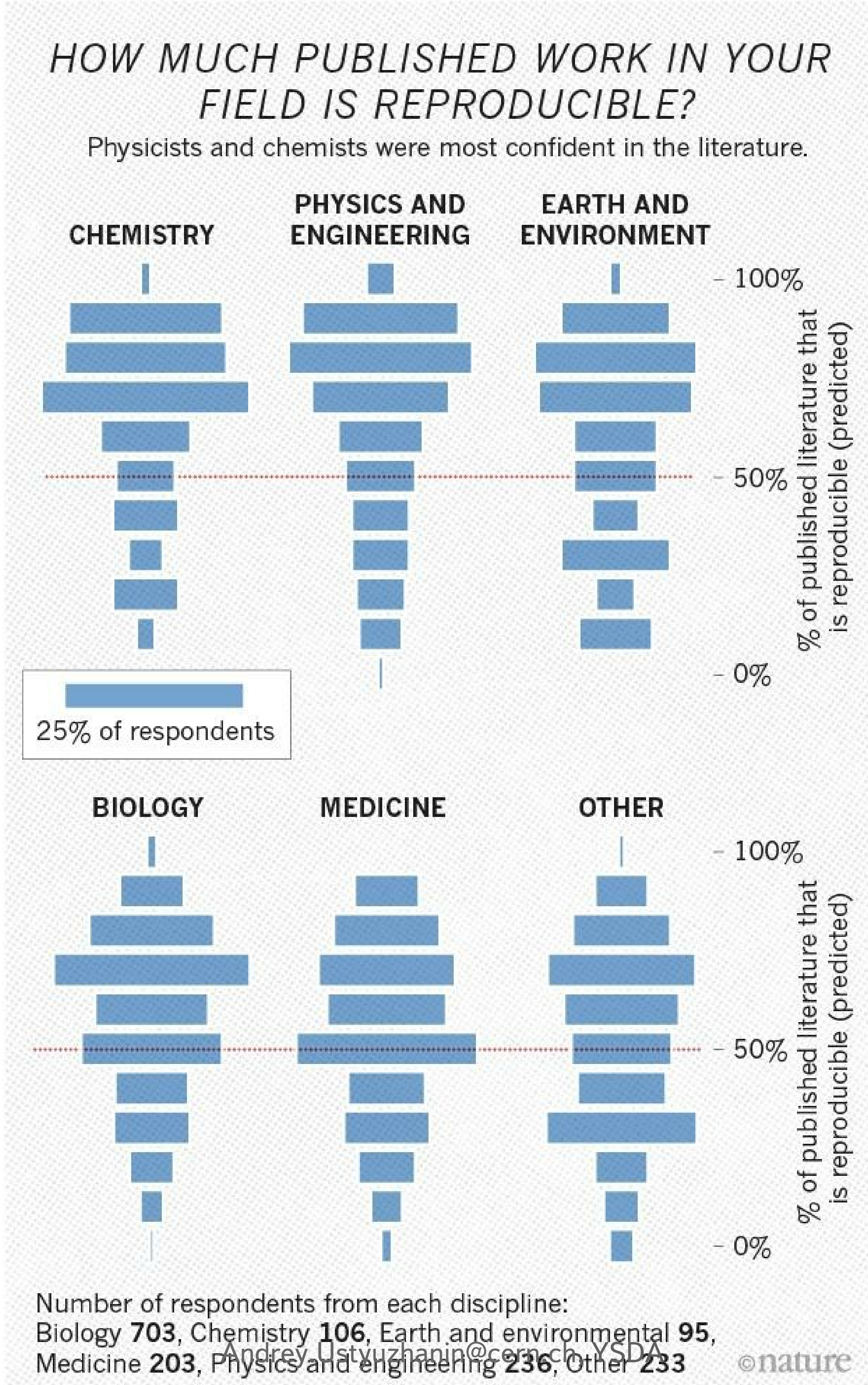
http://www.nature.com/news/cancer-reproducibility-project-scales-back-ambitions-1.18938

http://www.nature.com/nrd/journal/v10/n9/full/nrd3439-c1.html

Nature's Reproducibility Survey



- Nature: 1,500 scientists lift the lid on reproducibility by Monya Baker


HOW MUCH PUBLISHED WORK IN YOUR FIELD IS REPRODUCIBLE?
Physicists and chemists were most confident in the literature.
CHEMISTRY
PHYSICS AND ENGINEERING
EARTH AND ENVIRONMENT
BIOLOGY
MEDICINE
OTHER
100%
50%
0%
% of published literature that is reproducible (predicted)
25% of respondents
Number of respondents from each discipline:
Biology **703**, Chemistry **106**, Earth and environmental **95**,
Medicine **203**, Physics and engineering **236**, Other **233**
©nature

Rise of challenge-driven education

Learning by solving real-world problems in interdisciplinary & international projects.

- Imagine Cup, http://imaginecup.com/
- Hackathons, e.g., http://webfest.web.cern.ch/
- Open data days, http://opendataday.org/
- Guide to Challenge Driven Education, https://www.kth.se/social/group/guide-to-challenge-d/

Platforms (with plenty of examples):

- Kaggle, https://www.kaggle.com/
- Codalab, https://competitions.codalab.org/
- ...

Rise of challenge-driven education

Learning by solving real-world problems in interdisciplinary & international projects.

- Imagine Cup, http://imaginecup.com/
- Hackathons, e.g., http://webfest.web.cern.ch/
- Open data days, http://opendataday.org/
- Guide to Challenge Driven Education, https://www.kth.se/social/group/guide-to-challenge-d/

Platforms (with plenty of examples):

- Kaggle, https://www.kaggle.com/
- Codalab, https://competitions.codalab.org/
- ...

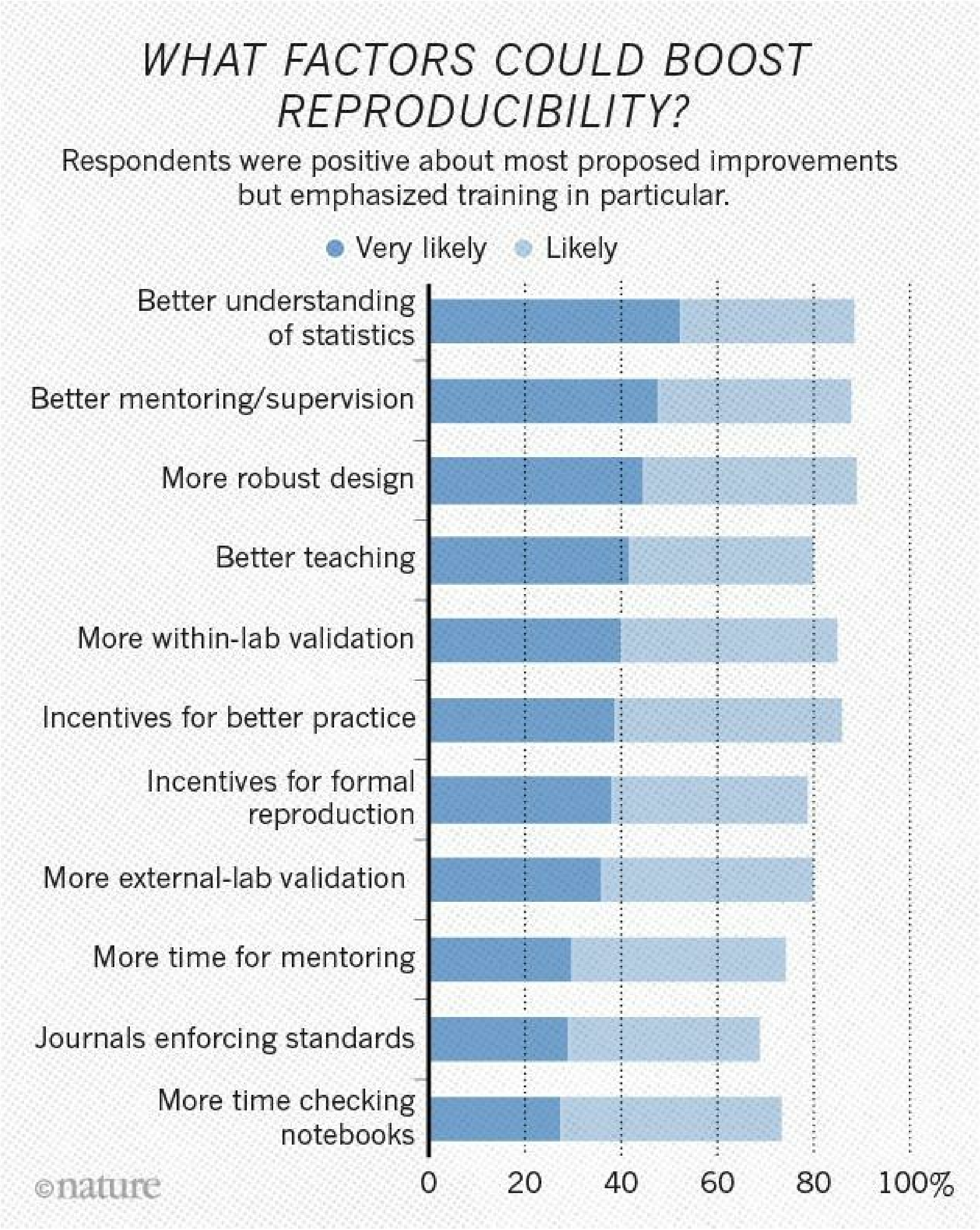
Complication and boost factors are similar to research reproducibility.

Computational experiment

***Computational experiment* is a significant part of the experiment, that starts after the data is collected.**

Possible effects (see previous slide):

- Practical
  - better mentoring / supervision
  - more within-lab validation
  - simplified external-lab validation
  - incentive for better practice
  - robust design
- Educational
  - wider access to the best practices
  - better teaching



High Energy Physics

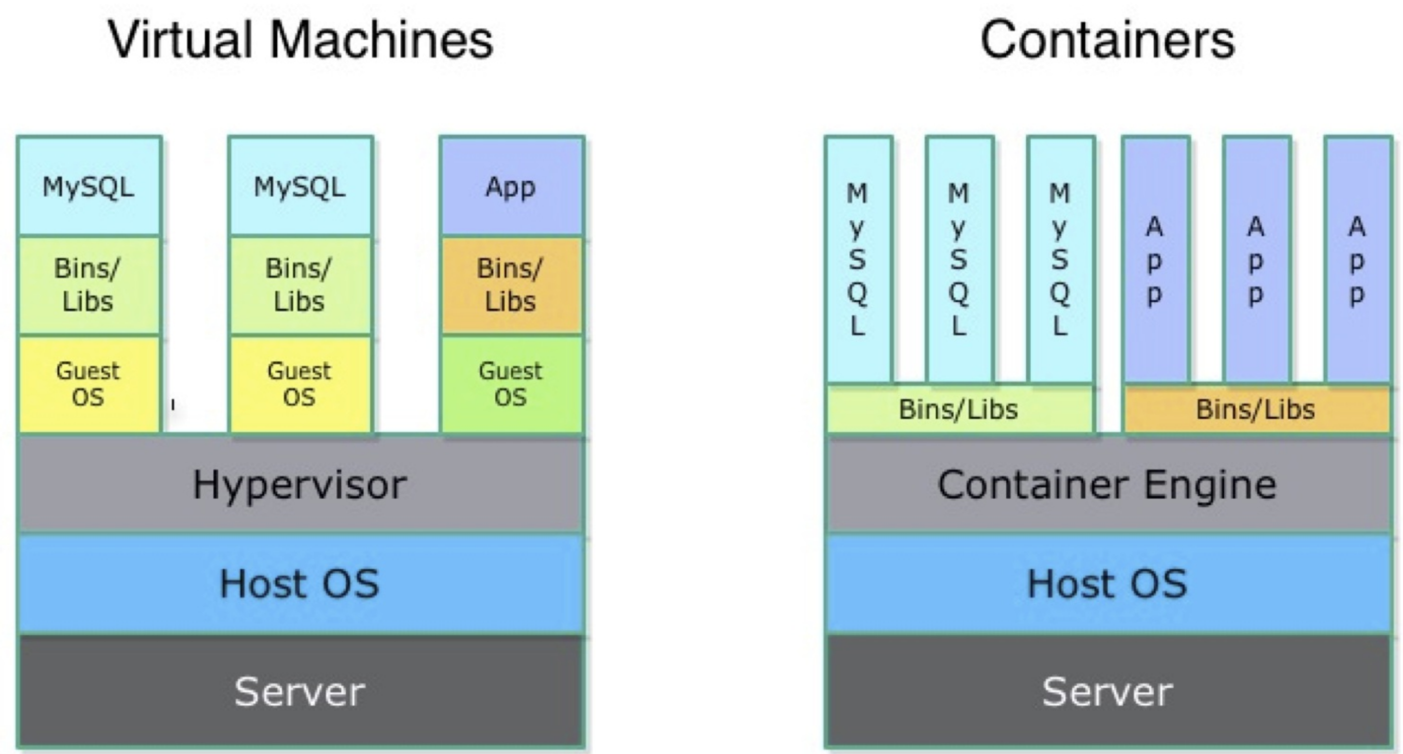
- **data** storage
  - shared storage (XROOTD, AFS, EOS, CERNBOX, ...)
- standardized **environment**
  - software: ROOT, minuit, experiments software stacks , ...
  - computational cluster (e.g. `lxplus`)
- **code** versioning repository (gitlab)
- advanced analysis approaches
  - blind analysis
  - reviews, cross-checks within group, inter-group collaboration
- collaborative culture
  - q&a groups, experts
  - publishing workflow

Reproducible computational study key components

- Basic assumptions (vocabulary)
- Data
- Environment + Resources (CPU/GPU)
- Code/scripts
- Workflow
- Automated intermediate results checks
- Final results (datasets, publications)

Key missing part: environment version control

- language and OS agnostic,
- capture and restore environment configuration,
- run configurations



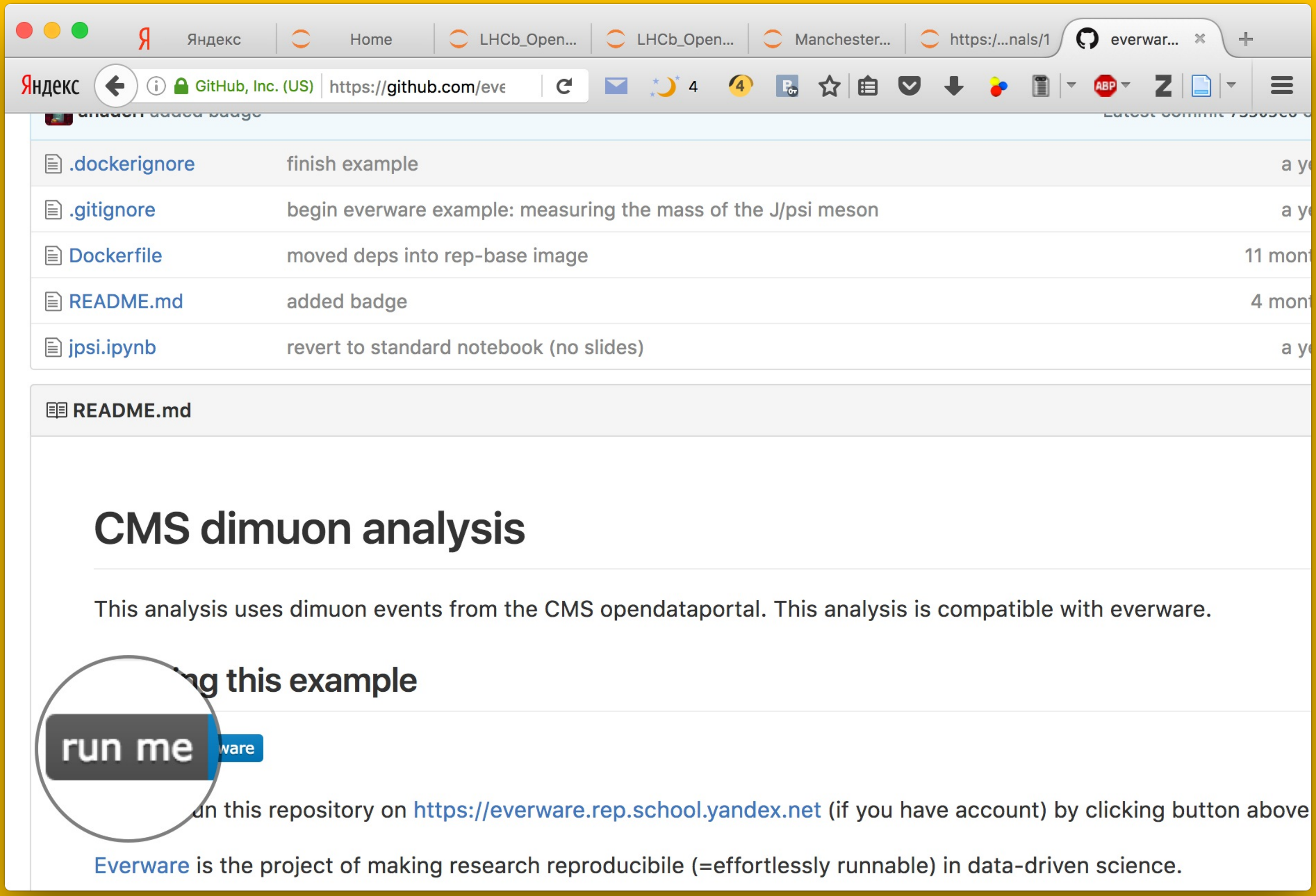
would enable:

- workflow automation
- automated results re-validation
- archiving data analysis along with containers/VMs

Example

Running https://github.com/everware/everware-dimuon-example

Sorry, printed version doesn't support animation.



How it works

- **resources**: wherever *everware* is installed (Yandex)
- **data**: CERNBOX

How it works

- **resources**: wherever *everware* is installed (Yandex)

- **data**: CERNBOX

- **environment** management:
  - conda or virtualenv
  - docker

How it works

- **resources**: wherever *everware* is installed (Yandex)
- **data**: CERNBOX
- **environment** management:
  - conda or virtualenv
  - docker
- github: analysis **code** versioning

How it works

- **resources**: wherever *everware* is installed (Yandex)
- **data**: CERNBOX
- **environment** management:
  - conda or virtualenv
  - docker
- github: analysis **code** versioning
- Jupyter(Hub): runs the code interactively (a-la **workflow**)

How it works

- **resources**: wherever *everware* is installed (Yandex)
- **data**: CERNBOX
- **environment** management:
  - conda or virtualenv
  - docker
- github: analysis **code** versioning
- Jupyter(Hub): runs the code interactively (a-la **workflow**)
- continuous integration: intermediate **results checks** & report

How it works

- **resources**: wherever *everware* is installed (Yandex)
- **data**: CERNBOX
- **environment** management:
  - conda or virtualenv
  - docker
- github: analysis **code** versioning
- Jupyter(Hub): runs the code interactively (a-la **workflow**)
- continuous integration: intermediate **results checks** & report
- **everware**: to rule them all (just a bunch of wrappers!)

Everware is ...

... about re-usable science, it allows people to jump right into your research code. Lets you launch *Jupyter* notebooks from a git repository with a click of a button.

- https://github.com/everware
- https://everware.rep.school.yandex.net (Yandex instance)

Examples:

- algorithm meta-analysis, https://github.com/openml/study_example
- gravitational waves, https://github.com/anaderi/GW150914
- COMET, https://github.com/yandexdataschool/comet-example-ci

Everware is ...

... about re-usable science, it allows people to jump right into your research code. Lets you launch *Jupyter* notebooks from a git repository with a click of a button.

- https://github.com/everware
- https://everware.rep.school.yandex.net (Yandex instance)

Examples:

- algorithm meta-analysis, https://github.com/openml/study_example
- gravitational waves, https://github.com/anaderi/GW150914
- COMET, https://github.com/yandexdataschool/comet-example-ci

Think of transition from procedural coding approach to object-oriented.

Everware toolkit

- extension for *JupyterHub*:
  - spawner for building and running custom *docker* images
- integrated with:
  - dockerhub
  - github (for authentication and repository interaction)
- similar to *mybinder.org* but with focus on scientific research
- Research guidelines

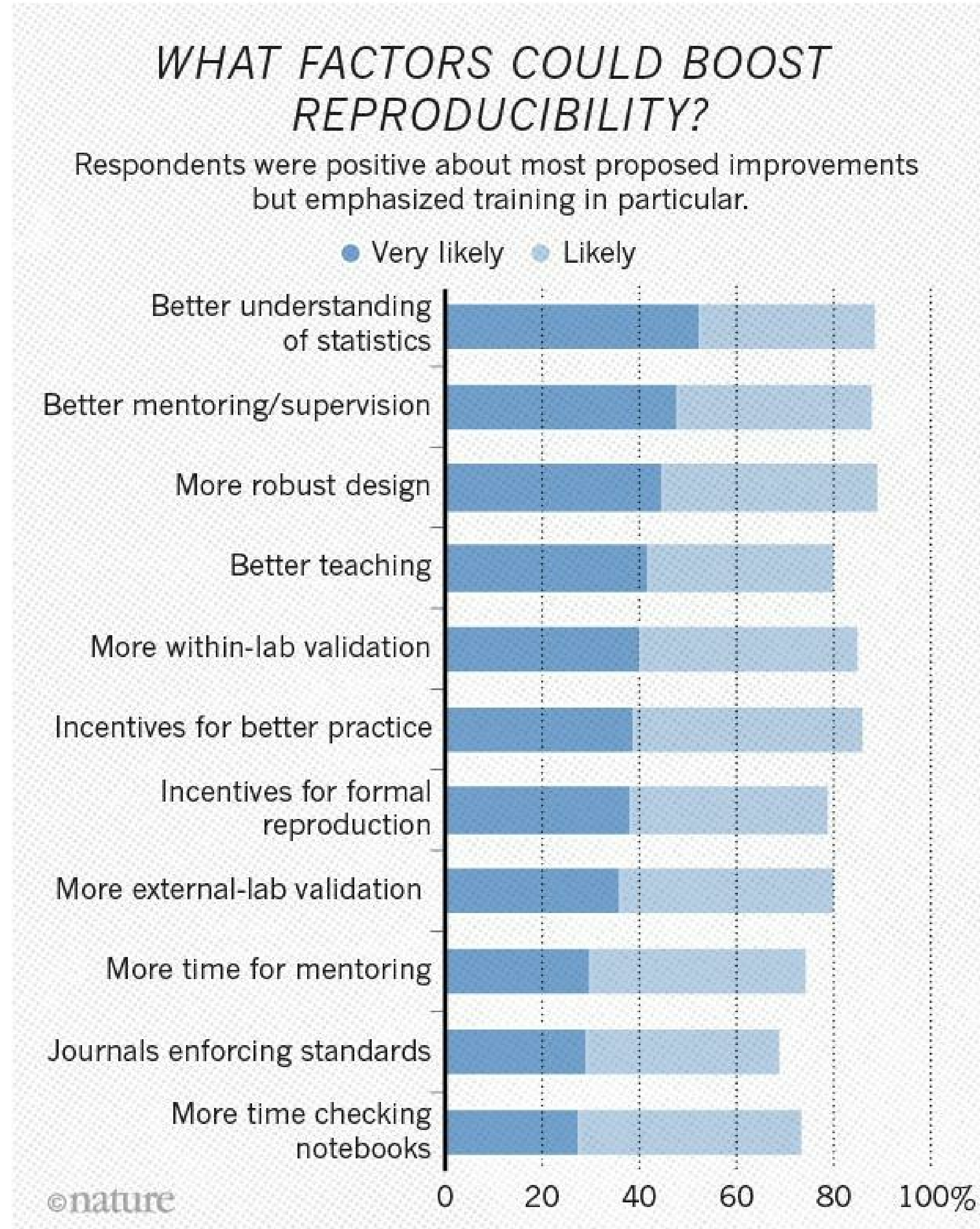
Pros & cons

Pros

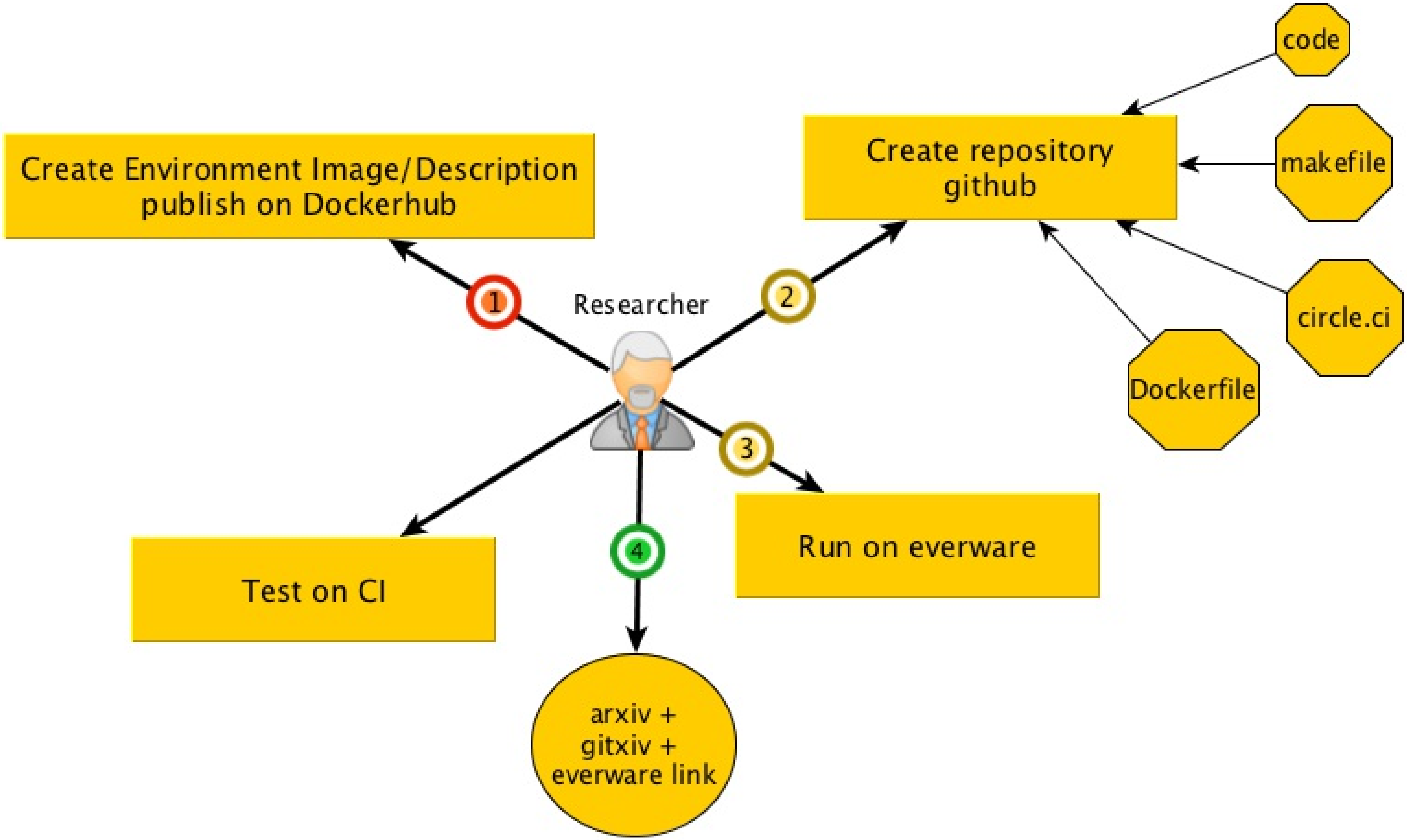
- easier supervision / mentoring
- easier within-lab validation
- wider access to the best practices
- simplified cross-lab validation
- good incentive for formal reproduction
- *good thing for industry career track development*

Cons

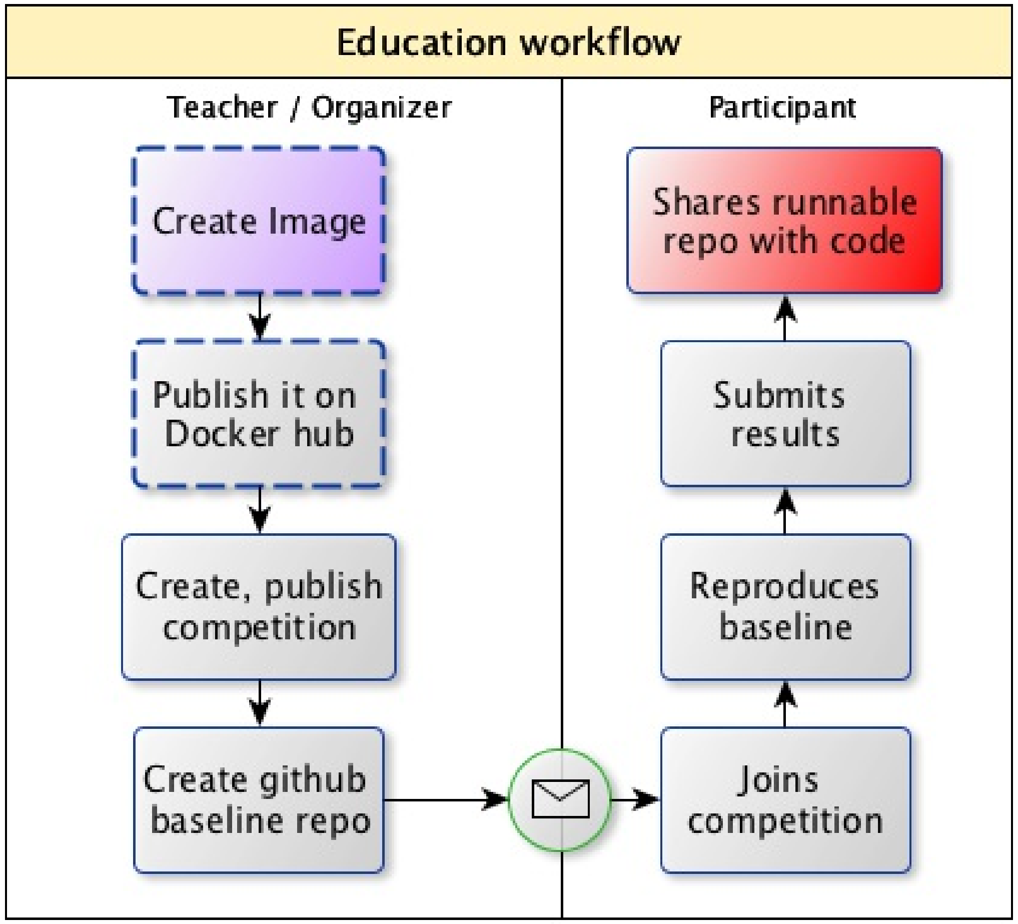
- learning a bit of (open-sourced) technology
- re-organize internal research process
- inner barrier for openness
- higher incentive for mindless *borrowing*
- divergence / potential learning curves (promotes users to create unique environments)



Basic research workflow with everware



Education workflow with everware



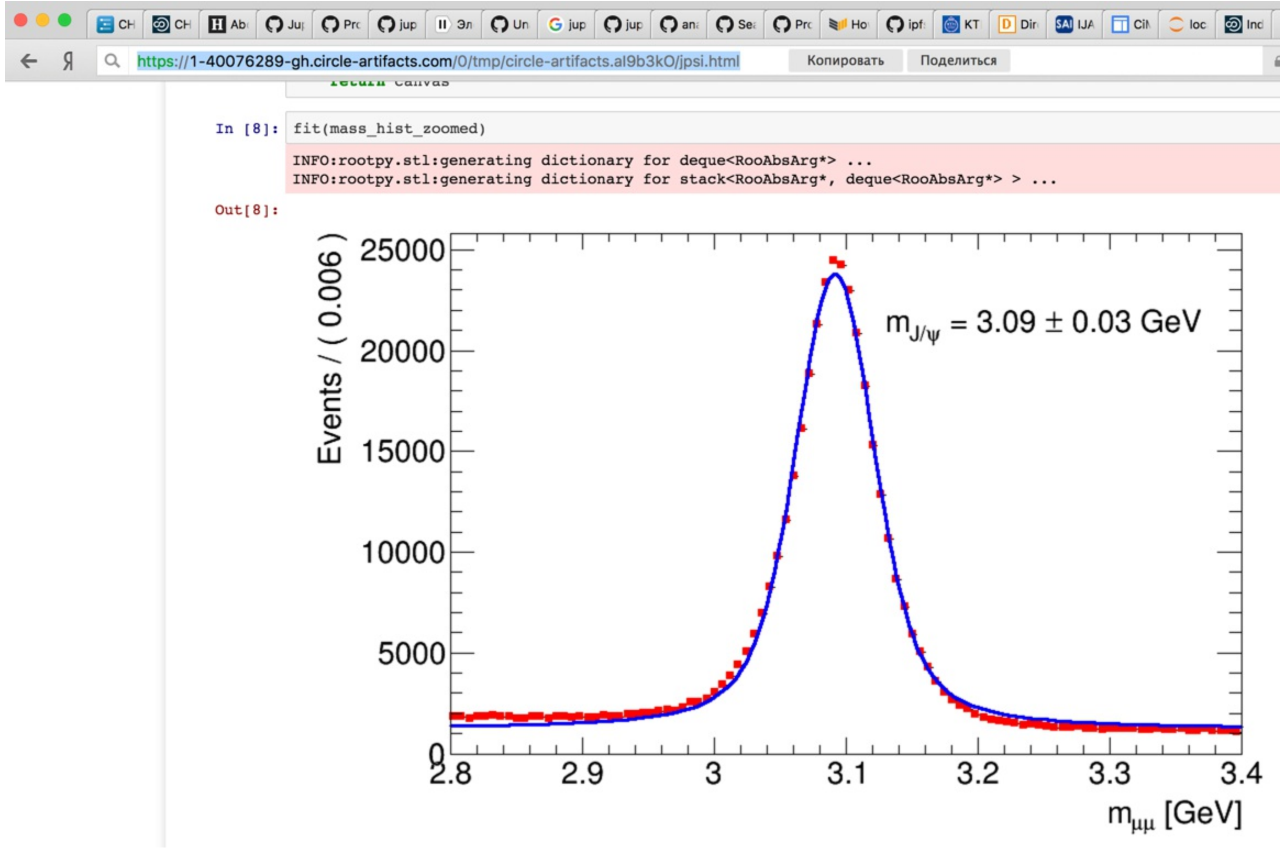
Tested on (some examples):

- Python course at YSDA 2015
- Machine Learning in High Energy Physics summer school 2016
- YSDA course on Machine learning at Imperial College London 2016
- Kaggle competitions 2016
- Machine learning course at University of Eindhoven
- LHCb open data masterclass

Bonus: automatic results checking

- Continuous integration
  - add `circle.yml`
  - enable repository checking at https://circleci.com
  - add badge

- monitor status by email/slack/telegram/...
- automatically generate research artefacts - dashboard of the experiment



https://1-40076289-gh.circle-artifacts.com/0/tmp/circle-artifacts.al9b3kO/jpsi.html

Roadmap

- Integrate with data sharing resources (zotero, figshare, etc)
- Automatic capture of environment (integrate with repro-zip)
- Integration with publishing resources (gitxiv, re-science, openml)
- Bring your own resources computational model
- Computations based on models other than Jupyter

Envoi

- Reproducibility is not easy;
  - ...but is not that scary,
  - ...with a bit of openness,
  - and technology.
- everware *works* for research and education (no people were harmed during testing);
  - easy to try;
  - WIP, https://github.com/everware (open-source, care to join?);
    - feature requests are welcome
    - pull requests are most welcome
  - See talk on LHCb open data masterclass for an extensive example.

Thank you!

Andrey Ustyuzhanin, anaderiru @ twitter

Slideshow created using remark

Backup slides

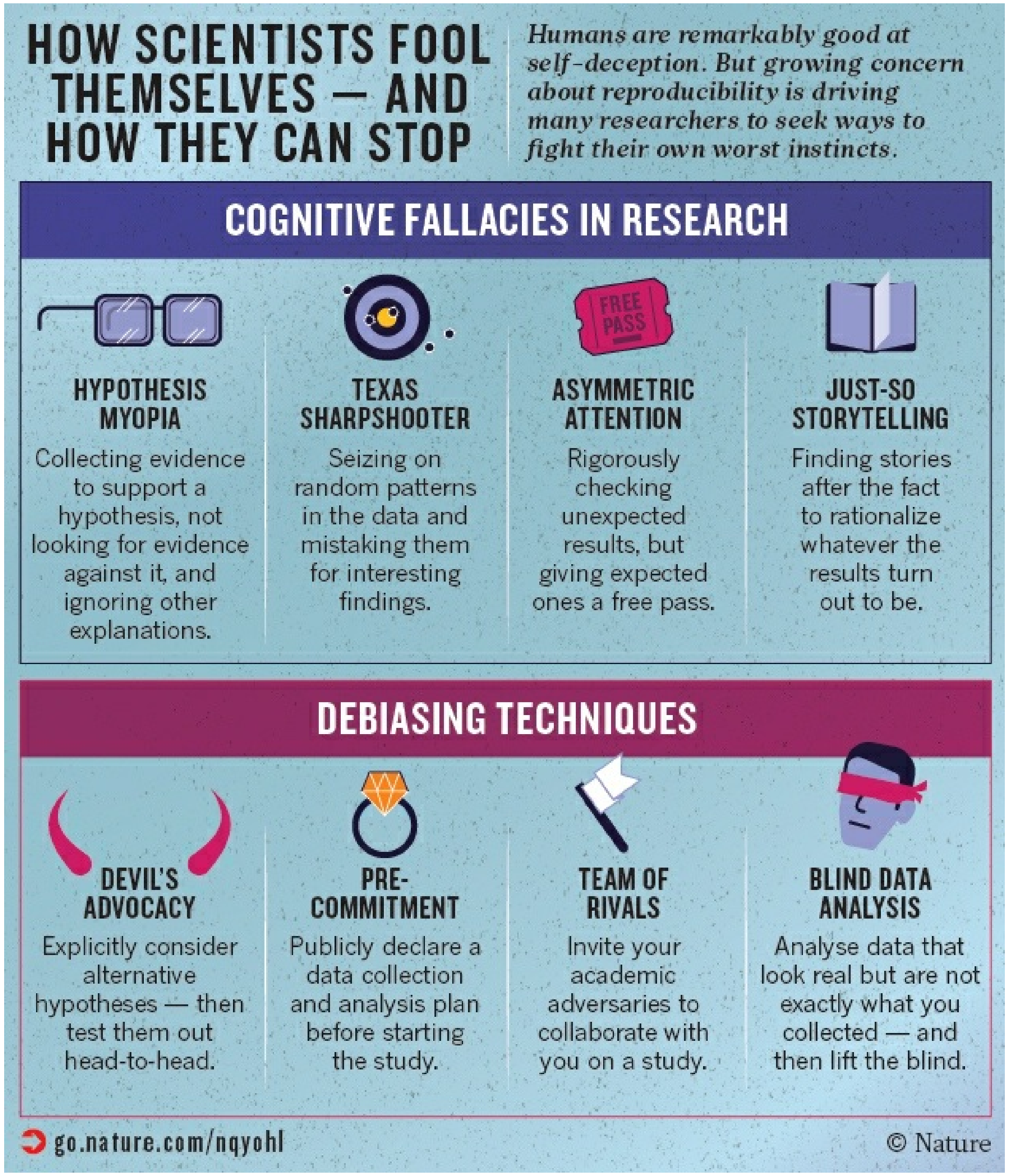
Yandex School of Data Analysis is

- non commercial private university https://yandexdataschool.com (separate from Yandex)
- 450+ students graduated since 2007
- Graduate students receive strong education in Data & Computer Science (main supply of Yandex employees)
- Interest in interdisciplinary research – Data Science methods to Information Retrieval and Fundamental Sciences
- organizes bi-yearly international Machine Learning Conference, YAC https://yandexdataschool.com/conference/
- 25% of our students have background in Physics
- full member of LHCb since 2015, associate member during 2014-2015

References

- http://www.nature.com/news/1-500-scientists-lift-the-lid-on-reproducibility-1.19970
- https://rescience.github.io/read/
- http://push.cwcon.org/
- https://openml.org
- https://figshare.com/
- https://gitlab.cern.ch/lhcb-bandq-exotics/Lb2LcD0K
- https://osf.io/ezcuj/wiki/home/
- https://osf.io/e81xl/wiki/home/
- Center for open science, https://cos.io/
- IPFS, https://github.com/ipfs/
- Nature, keyword: reproducibility, http://www.nature.com/news/reproducibility-1.17552

Dealing with cognitive bias



http://go.nature.com/nqyohl

Research workflow with everware

- User creates a git repository for his project
- User creates some code, notebooks, figures out what libraries he needs
- User creates `Dockerfile` where he writes all the dependencies for his code (use `everware-cli`)
- User creates `Makefile` that simplifies start one of the targets in `Makefile` passes through all the essential steps of analysis
- (optional) User tests that his analysis is runnable by one of the CI systems (e.g. on travis, adding, `.travis.yml`)
- User tests that analysis is also runnable by everware
- User completes his research and checks that he/she can reproduce all the figures/tables supporting his hypothesis by running corresponding notebooks (or automates cascade of notebooks execution by single `Makefile` target)
- User publishes paper, filling-in special form link to his git repository and to everware that any member of the researcher community can pick-up from to improve his research

https://github.com/everware/everware/wiki/How-to-embed-everware-into-research-use-cases