Reproducible Research, Open Science
Motivation, Challenges, Approaches, ...

Arnaud Legrand
CNRS, Inria, University of Grenoble

March 7, 2016 – Reproducible Research Webinar

Foreword about the organization (1/2)

There is currently a screencast of this seminar:

https://mi2s.imag.fr/pm/direct

The resulting video will be edited and available from github that gathers all the information, slides, and resources:

https://github.com/alegrand/RR_webinars/blob/master/README.org

There is a 5 second delay between what I say and the screencast. We can have almost live interaction with other sites by using pad to comment and ask questions

http://tinyurl.com/RRW-pad1

Foreword about the organization (2/2)

1. General introduction ($\approx$ an hour) on reproducible research.and briefly present many aspects that should be considered to improve our practice I will then try to answer the questions that have been written on the pad
2: A short break Finish installing software (instructions given on github)
3. Demo Reproducible research is an endless quest, which can be quite discouraging. So I'll focus on one particular point and demo some some tools (Rstudio/knitr, the ipython notebook, emacs/org-mode) that can help

One such webinar per month to cover other topics related to reproducible research

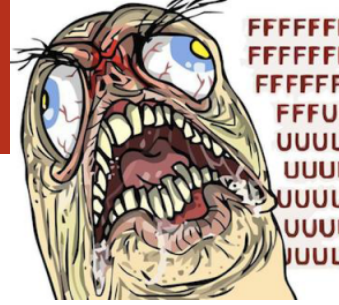
- The goal of these talks is to provide the audience with a general background and possibly tools that are ready to use
- Announcement on recherche-reproductible@listes.univ-orleans.fr, which is a rather low traffic mailing list
- Notifications by using the watch button of our Github repository

Just a few words about myself

- 2000-2003: PhD in from ENS Lyon, France on *Scheduling and Parallel Algorithms for Heterogeneous Platforms*
- 2004-present: CNRS Researcher in Grenoble working at Inria
- Research topics: large scale distributed computing infrastructures
  - Management (scheduling, load balancing, fairness, game theory. . . .)
  - Performance evaluation (simulation, visualization, statistical analysis, . . .)
  - Since 2000, I am one of the main developers of the SimGrid project

Outline

Frustration



As an Author

- Advisor: "Did you take care of setting this?" Me: "Uh?"
- I thought I used the same parameters but I'm getting different results! I swear it worked yesterday!
- A new student wants to compare with the method I proposed last year
- The damned fourth reviewer asked for a major revision and wants me to change figure 3. ☹ Which code and which data set did I use to generate this figure?
- 6 months later: Why did I do that?

As a Reviewer

This may be an interesting contribution but:

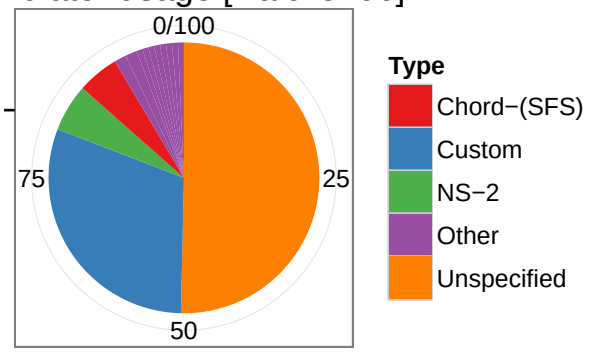
- There is no label/legend/... What is the meaning of this graph? If only I could access the generation script and get rid of the logscale
- This average value must hide something. As usual, no confidence interval... I wonder whether the difference is significant at all
- That can't be true, I'm sure they removed some points or decided to show only a subset of the data. I wonder what the rest looks like
- It this improvement solely the result of this naive idea?

A Few Edifying Examples

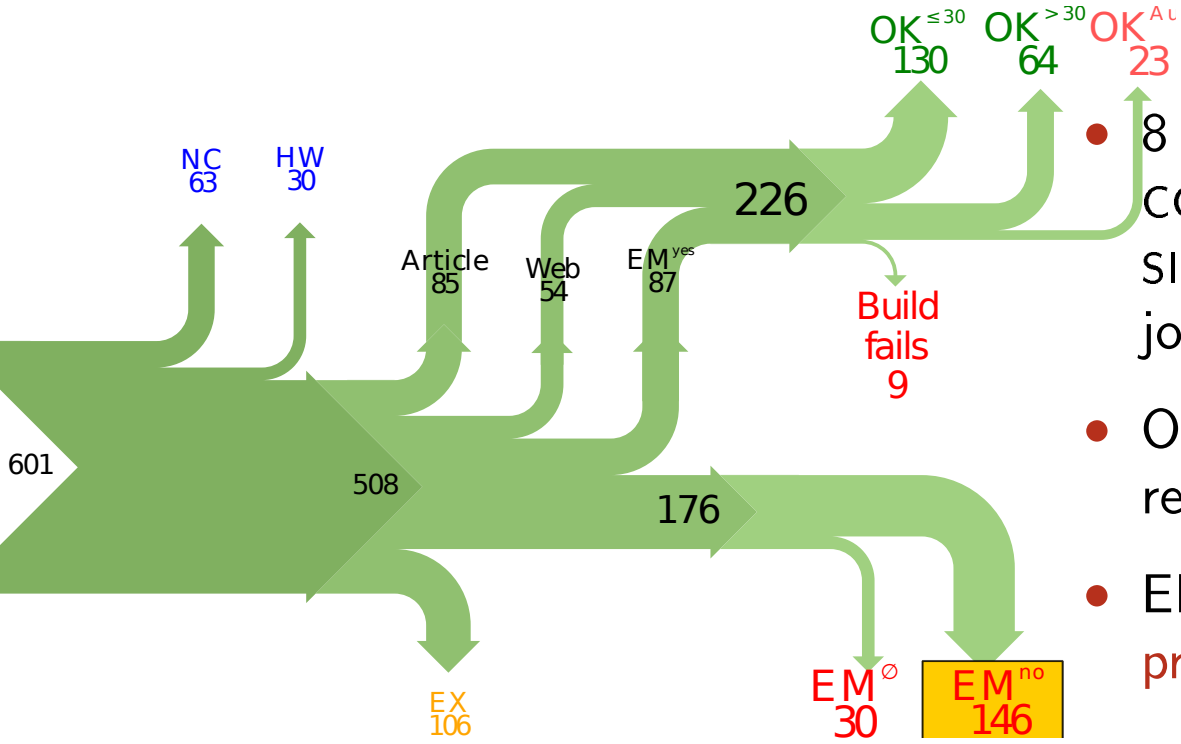
Naicken, Stephen *et Al.*, *Towards Yet Another Peer-to-Peer Simulator*, HET-NETs'06.

From 141 P2P sim.papers, 30% use a custom tool,
50% don't report used tool

Simulator usage [Naicken06]



Collberg, Christian *et Al.*, *Measuring Reproducibility in Computer Systems Research*, http://reproducibility.cs.arizona.edu/ 2014,2015



- 8 ACM conferences (ASPLOS'12, CCS'12, OOPSLA'12, OSDI'12, PLDI'12, SIGMOD'12, SOSP'11, VLDB'12) and 5 journals
- Original study = 80% of non reproducible work
- EM^{no}= the code cannot be provided

The Dog Ate my Homework !!!

- Versioning Problems

Thanks for your interest in the implementation of our paper. The good news is that I was able to find some code. I am just hoping that it is a stable working version of the code, and matches the implementation we finally used for the paper. Unfortunately, I have lost some data when my laptop was stolen last year. The bad news is that the code is not commented and/or clean.

Attached is the ⟨system⟩ source code of our algorithm. I'm not very sure whether it is the final version of the code used in our paper, but it should be at least 99% close. Hope it will help.

The Dog Ate my Homework !!!

- Versioning Problems
- Bad Backup Practices

Unfortunately, the server in which my implementation was stored had a disk crash in April and three disks crashed simultaneously. While the help desk made significant effort to save the data, my entire implementation for this paper was not found.

The Dog Ate my Homework !!!

- Versioning Problems
- Bad Backup Practices
- Code Will be Available Soon

Unfortunately the current system is not mature enough at the moment, so it's not yet publicly available. We are actively working on a number of extensions and things are somewhat volatile. However, once things stabilize we plan to release it to outside users. At that point, we would be happy to send you a copy.

The Dog Ate my Homework !!!

- Versioning Problems
- Bad Backup Practices
- Code Will be Available Soon
- No Intention to Release

> *I am afraid that the source code was never released. The code was never intended to be released so is not in any shape for general use.*

The Dog Ate my Homework !!!

- Versioning Problems
- Bad Backup Practices
- Code Will be Available Soon
- No Intention to Release
- Programmer Left

⟨STUDENT⟩ was a graduate student in our program but he left a while back so I am responding instead. For the paper we used a prototype that included many moving pieces that only ⟨STUDENT⟩ knew how to operate and we did not have the time to integrate them in a ready-to-share implementation before he left. Still, I hope you can build on the ideas/technique of the paper.

Unfortunately, the author who has done most of the coding for this paper has passed away and the code is no longer maintained.

The Dog Ate my Homework !!!

- Versioning Problems
- Bad Backup Practices
- Code Will be Available Soon
- No Intention to Release
- Programmer Left
- Commercial Code

Since this work has been done at ⟨COMPANY⟩ we don't open-source code unless there is a compelling business reason to do so. So unfortunately I don't think we'll be able to share it with you.

The code owned by ⟨COMPANY⟩, and AFAIK the code is not open-source. Your best bet is to reimplement :(Sorry.

The Dog Ate my Homework !!!

- Versioning Problems
- Bad Backup Practices
- Code Will be Available Soon
- No Intention to Release
- Programmer Left
- Commercial Code
- Proprietary Academic Code

Unfortunately, the ⟨SYSTEM⟩ sources are not meant to be opensource (the code is partially property of ⟨UNIVERSITY 1⟩, ⟨UNIVERSITY 2⟩ and ⟨UNIVERSITY 3⟩.)

If this will change I will let you know, albeit I do not think there is an intention to make the ⟨SYSTEM⟩ sources opensource in the near future.

If you're interested in obtaining the code, we only ask for a description of the research project that the code will be used in (which may lead to some joint research), and we also have a software license agreement that the University would need to sign.

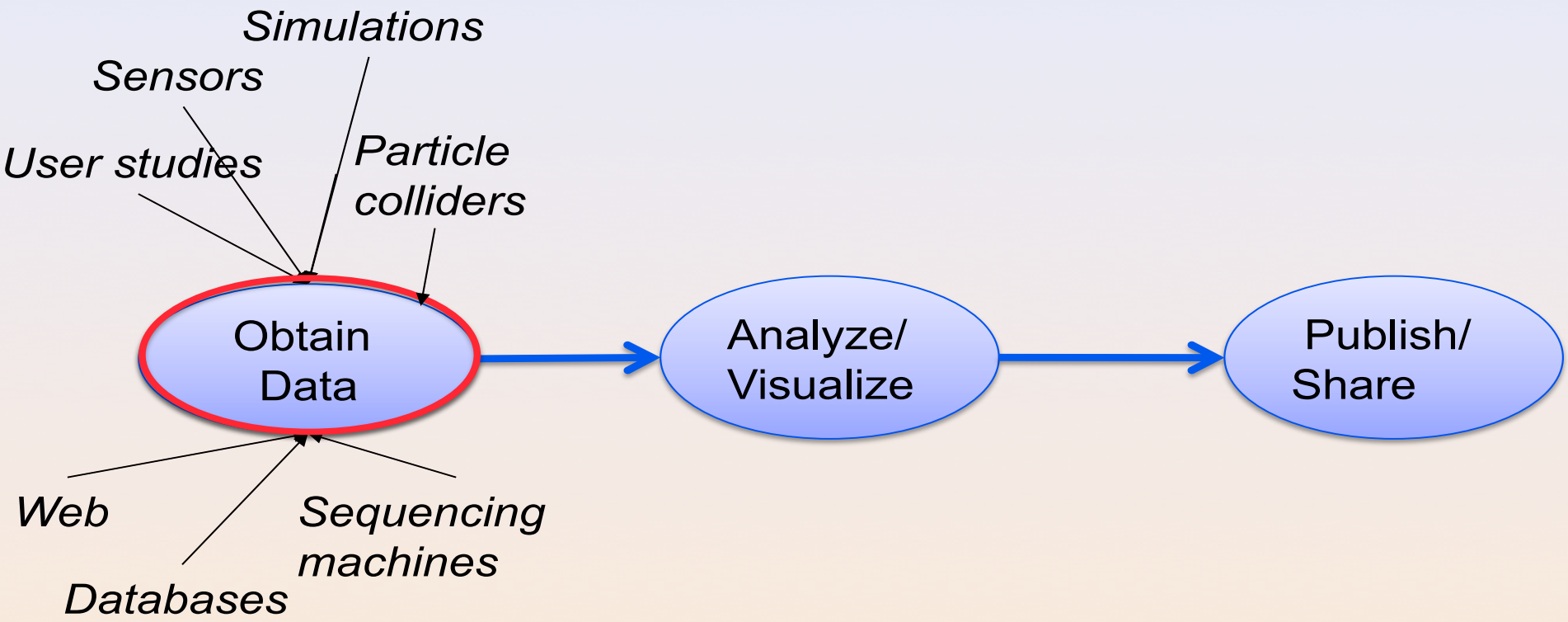
The Dog Ate my Homework !!!

- Versioning Problems
- Bad Backup Practices
- Code Will be Available Soon
- No Intention to Release
- Programmer Left
- Commercial Code
- Proprietary Academic Code
- Research vs. Sharing
- ...
- ...

In the past when we attempted to share it, we found ourselves spending more time getting outsiders up to speed than on our own research. So I finally had to establish the policy that we will not provide the source code outside the group.

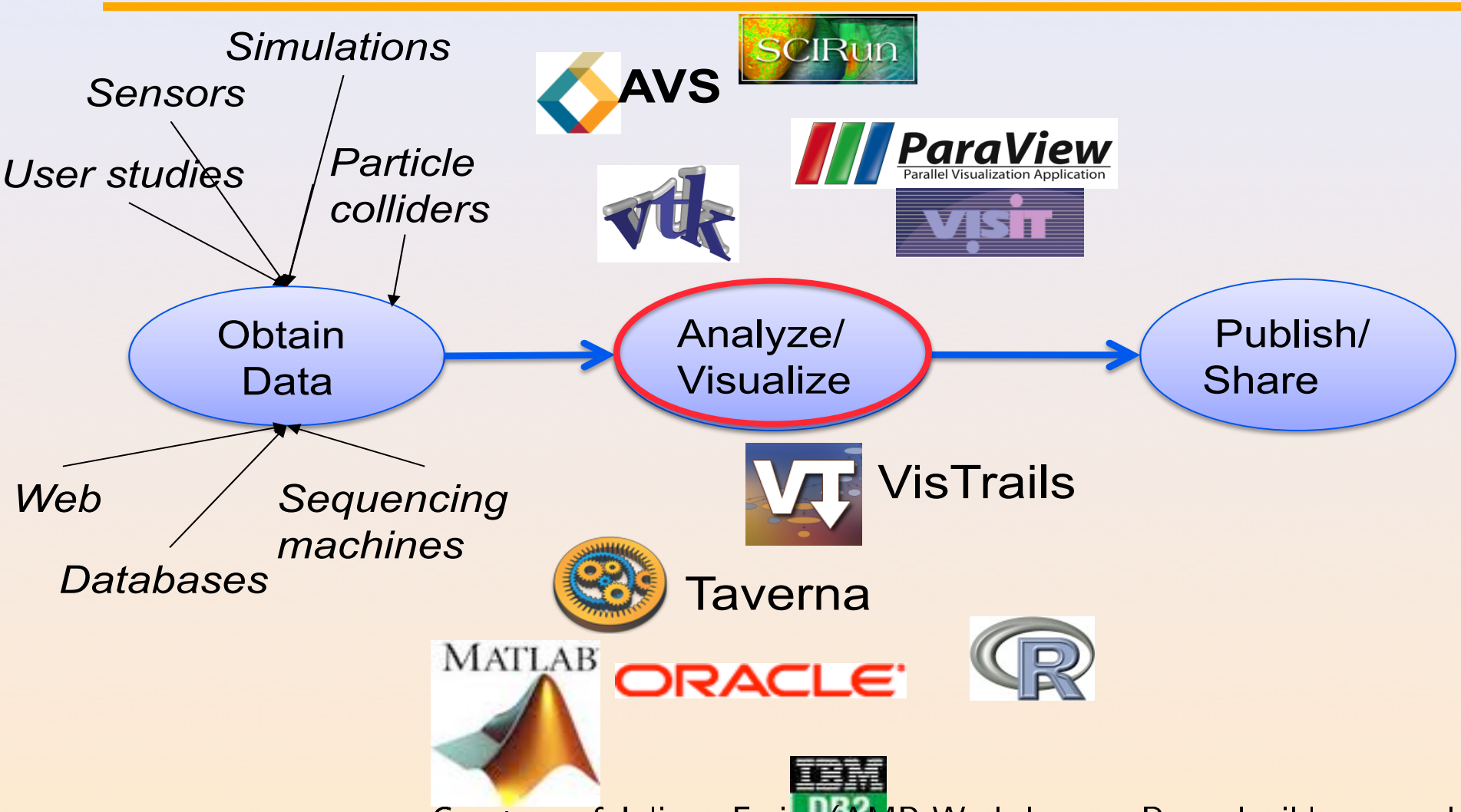
Outline

Science Today: Data Intensive



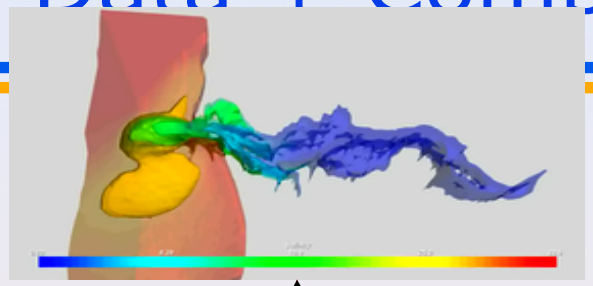
Courtesy of Juliana Freire (AMP Workshop on Reproducible research)

Science Today: Data + Computing Intensive



Courtesy of Juliana Freire (AMP Workshop on Reproducible research)

Science Today: Data + Computing Intensive



Simulations

Sensors

User studies

Particle colliders

Obtain Data

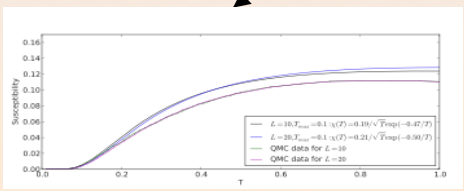
Analyze/ Visualize

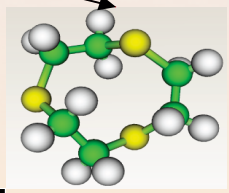
Publish/ Share

Web

Sequencing machines

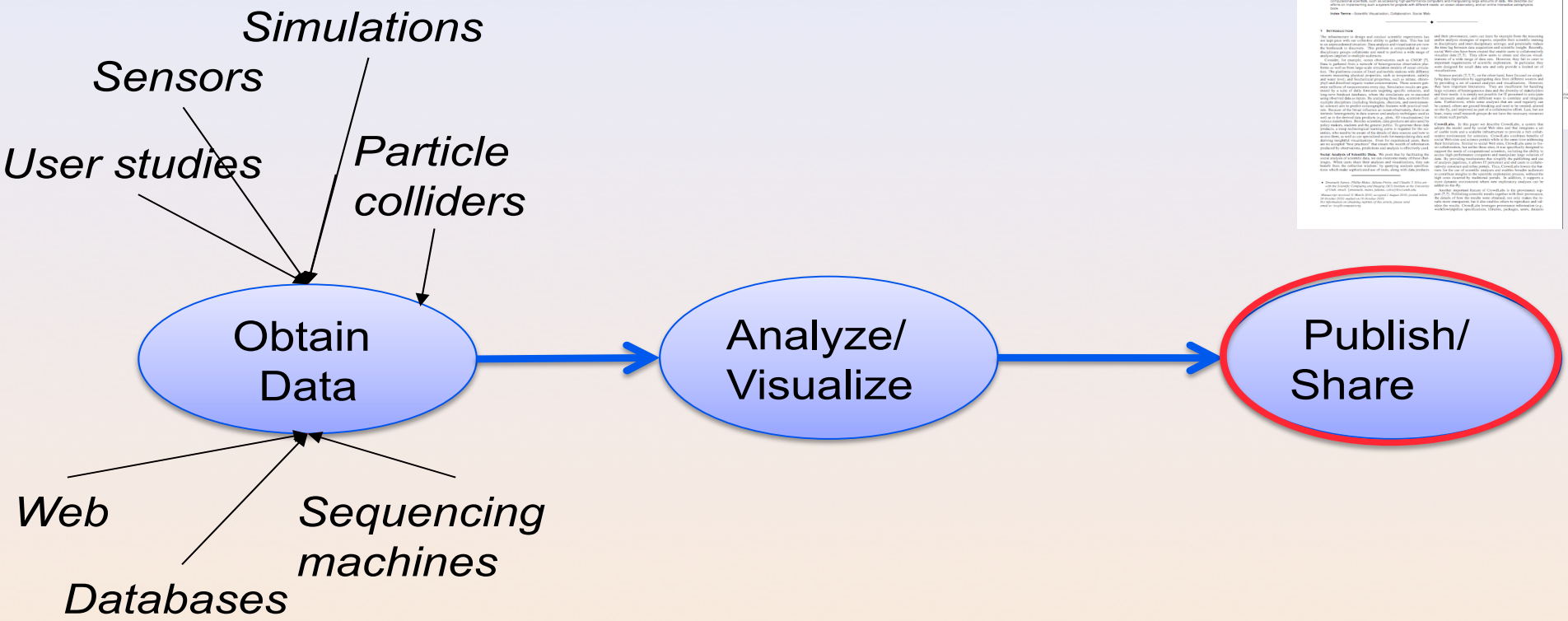
Databases





Courtesy of Juliana Freire (AMP Workshop on Reproducible research)

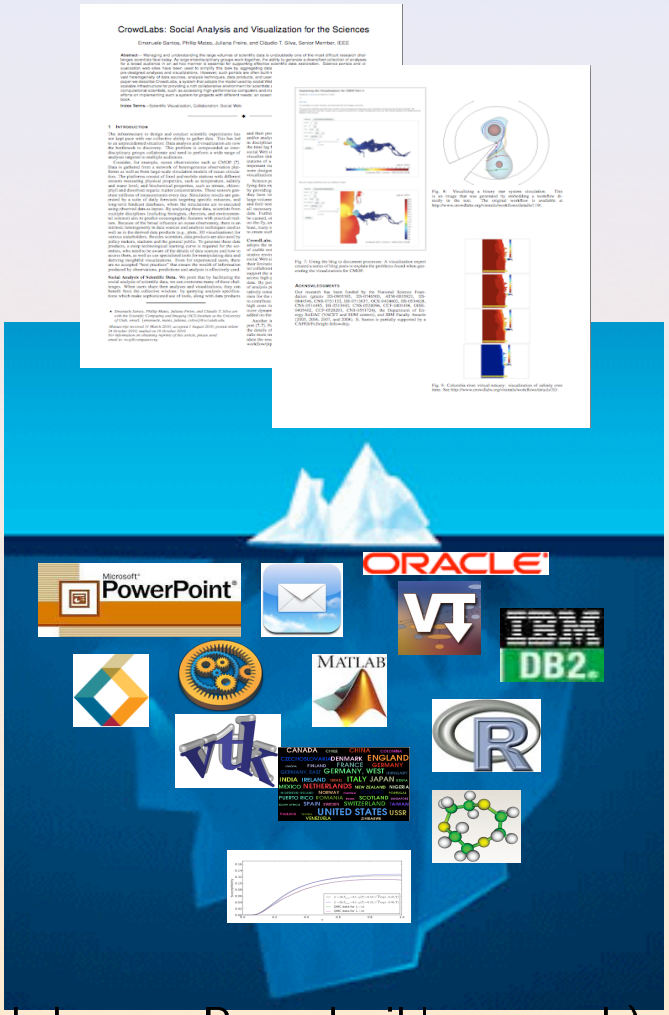
Science Today: Data + Computing Inte



Courtesy of Juliana Freire (AMP Workshop on Reproducible research)

Science Today: Incomplete Publications

- Publications are just the tip of the iceberg
  - Scientific record is incomplete---to large to fit in a paper
  - Large volumes of data
  - Complex processes
- Can't (easily) reproduce results

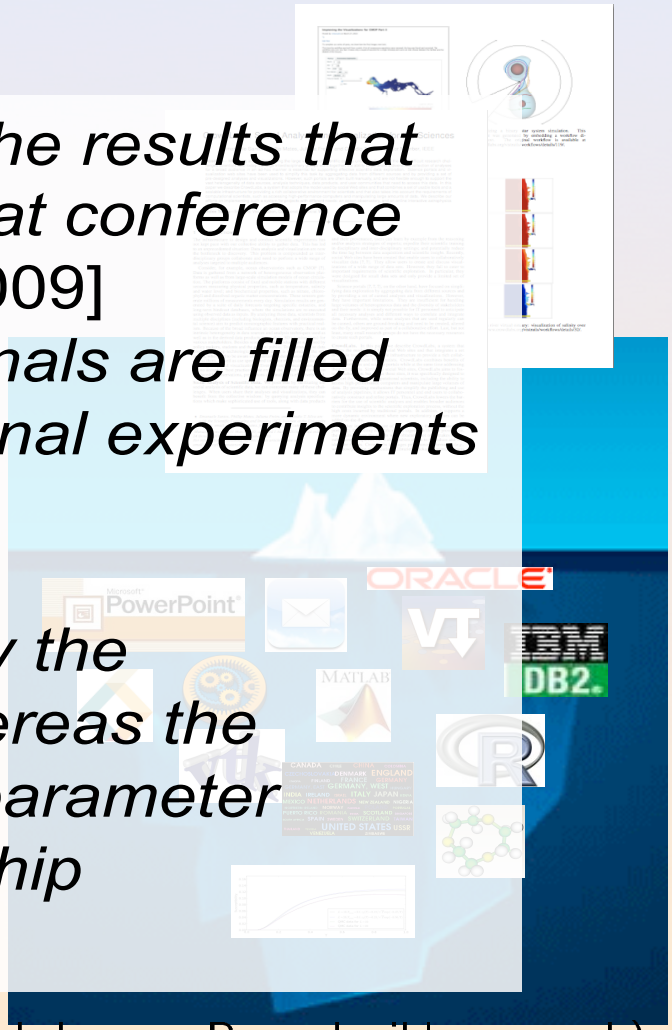


Courtesy of Juliana Freire (AMP Workshop on Reproducible research)

Science Today: Incomplete Publications

- Publications are just the tip of the iceberg
  - Scientific record is incomplete—too large to fit in a paper
  - Large volumes of data
  - Complex processes
- Can't (easily) reproduce results

> "*It's impossible to verify most of the results that computational scientists present at conference and in papers.*" [Donoho et al., 2009]
>
> "*Scientific and mathematical journals are filled with pretty pictures of computational experiments that the reader has no hope of repeating.*" [LeVeque, 2009]
>
> "*Published documents are merely the advertisement of scholarship whereas the computer programs, input data, parameter values, etc. embody the scholarship itself.*" [Schwab et al., 2007]



Courtesy of Juliana Freire (AMP Workshop on Reproducible research)

Why Are Scientific Studies so Difficult to Reproduce?

Human error:

- Experimenter bias (crowdsourced research?)
- Programming errors or data manipulation mistakes
- Poorly selected statistical test

There is just no real incentive in doing so:

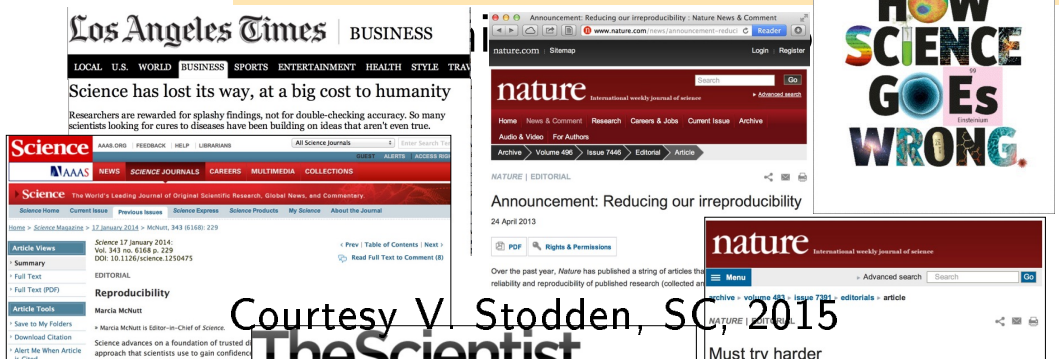
- Legal barriers, copyright (*many ongoing thoughts on this in the US*)
- Competition issue (*researchware*, bibliometry, . . .)
- Publication bias (only the idea matters, not the gory details)
- Rewards for positive results, not for consolidating results

Technical difficulty:

- ~~Hardware and software evolve too quickly. It's not worth it~~
- ~~No resources for storing somuch data/information~~
- ~~Lack of easy-to-use tools~~

Evidence for a Lack of Reproducibility

- Studies showing that scientific papers commonly leave out experimental details essential for reproduction and showing difficulties with replicating published experimental results:
  - J.P. Ioannidis. *Why Most Published Research Findings Are False* PLoS Med. 2005 August; 2(8)
  - *Reproducibility: A tragedy of errors* Nature, Feb 2016.
- High number of failing clinical trials.
  - *Do We Really Know What Makes Us Healthy?*, New-York Times — September 16, 2007
  - *Lies, Damned Lies, and Medical Science*, The Atlantic. Nov, 2010
- Increase in retracted papers:
  - Steen RG, Retractions in the scientific literature: is the incidence of research fraud increasing?



Courtesy V. Stodden, SC, 2015

A Reproducibility Crisis?

The Duke University scandal with scientific misconduct on lung cancer

- *Nature Medicine* - 12, 1294 - 1300 (2006) Genomic signatures to guide the use of chemotherapeutics, by Anil Potti and 16 other researchers from Duke University and University of South Florida
- Major commercial labs licensed it and were about to start using it before two statisticians discovered and publicized its faults

Dr. Baggerly and Dr. Coombes found errors almost immediately. Some seemed careless — moving a row or a column over by one in a giant spreadsheet — while others seemed inexplicable. The Duke team shrugged them off as "clerical errors."

The Duke researchers continued to publish papers on their genomic signatures in prestigious journals. Meanwhile, they started three trials using the work to decide which drugs to give patients.

- Retractions: January 2011. Ten papers that Potti coauthored in prestigious journals were retracted for varying reasons
- Some people die and may be getting worthless information that is based on bad science

Courtesy of Adam J. Richards

Definitely

A recent scandal In 2013, Dong-Pyou Han, a former assistant professor of biomedical sciences at Iowa State University was disgraced:

- Falsified blood results to make it appear as though a vaccine he was working on had exhibited anti-HIV activity
- Han and his team received $\approx$ \$19 million from NIH
- Retraction and resignation of university
- Han was sentenced in 2015 to 57 months imprisonment for fabricating and falsifying data in HIV vaccine trials. He was also fined US \$7.2 million!

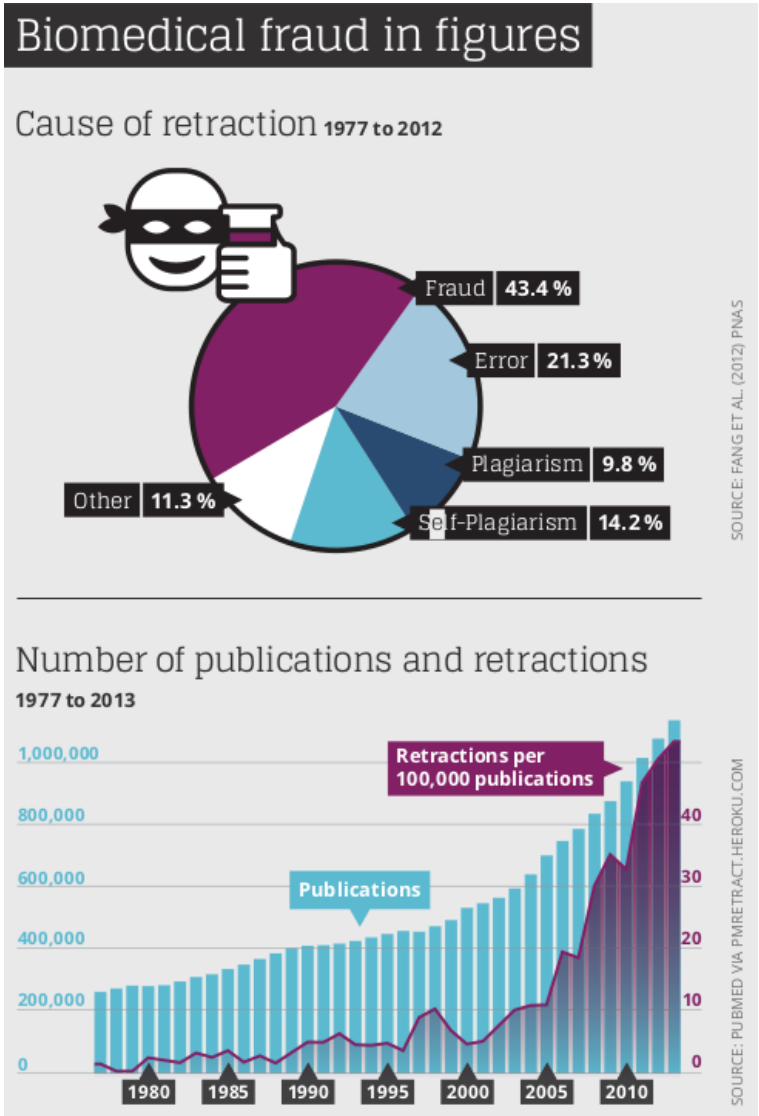
We should avoid witch-hunt

- August 5, 2014, Yoshiki Sasai (stem cell, considered for Nobel Prize) hanged in his laboratory at the RIKEN (Japan). Fraud suspicion...
- In 1986, a young postdoctoral fellow at MIT accused her director, Thereza Imanishi-Kari, of falsifying the results of a study published in Cell and co-signed by the Nobel laureate David Baltimore. [..] Declared guilty, Univ. presidency resignation, and finally cleared. This put their careers of two outstanding researchers on hold for ten years based on unfounded accusations.

Scientific fraud is bad. Let's try to improve Have a look at the wikipedia *list of academic scandals*. *Plagiarism* is also a problem and partly the result of the worldwide *publish or perish*/*bibliometry* craziness

The Battle against Scientific Fraud in the CNRS International Magazine

Is Fraud a new phenomenon?



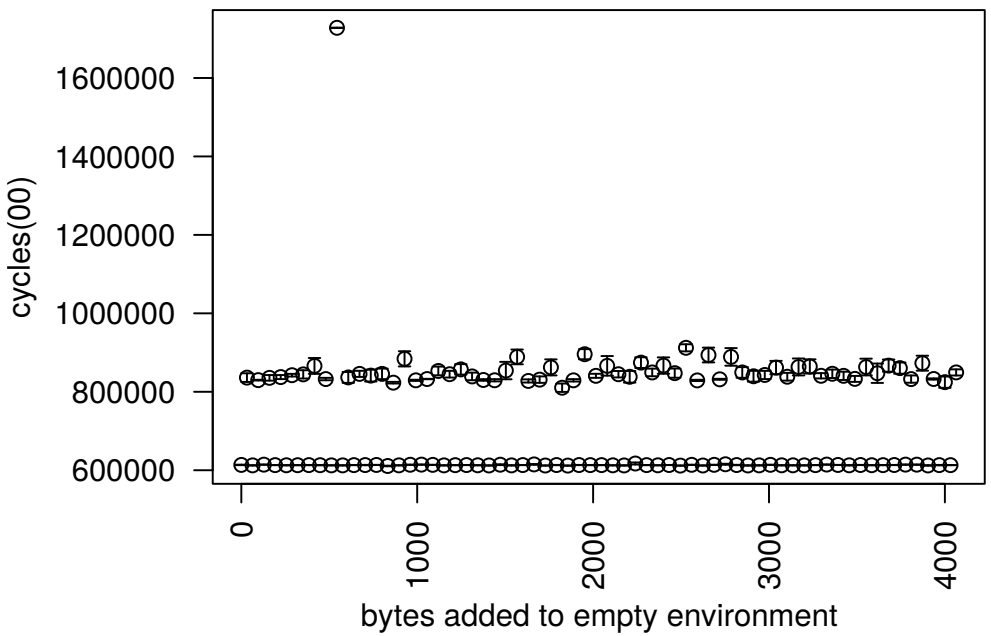
- Galileo (data fabrication), Ptolemy (plagiarism), Mendel (data enhancement), Pasteur (rigorous but hid failures), ...

Outline

But do we **really** have to care in CS?

Yes. **Model** $\neq$ **Reality**. Although designed and built by human beings, computers are so complex that mistakes are easy to do...

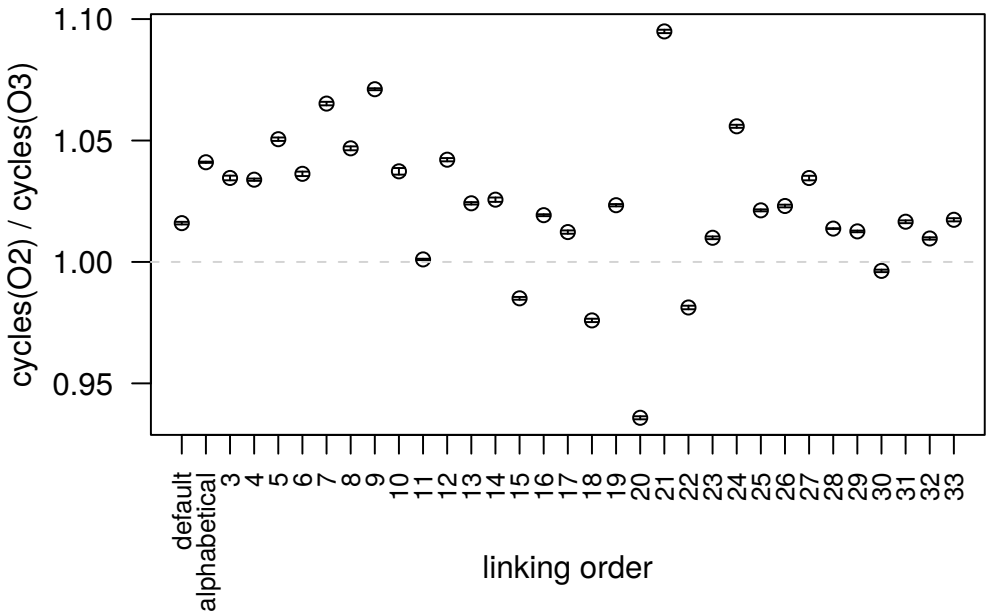
- Experiments: Mytkowicz, Diwan, Hauswirth, Sweeney. Producing wrong data without doing anything obviously wrong!. SIGPLAN Not. 44(3), March 2009



But do we **really** have to care in CS?

Yes. **Model** $\neq$ **Reality**. Although designed and built by human beings, computers are so complex that mistakes are easy to do...

- Experiments: Mytkowicz, Diwan, Hauswirth, Sweeney. Producing wrong data without doing anything obviously wrong!. SIGPLAN Not. 44(3), March 2009



But do we **really** have to care in CS?

Yes. **Model** $\neq$ **Reality**. Although designed and built by human beings, computers are so complex that mistakes are easy to do...

- Experiments: Mytkowicz, Diwan, Hauswirth, Sweeney. Producing wrong data without doing anything obviously wrong!. SIGPLAN Not. 44(3), March 2009

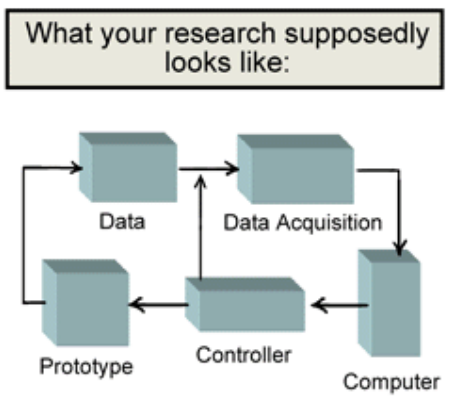


Figure 1. Experimental Diagram

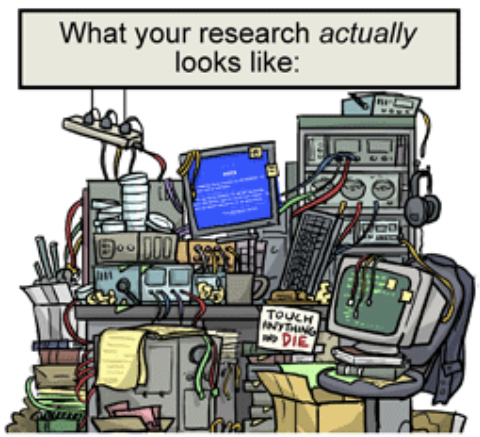


Figure 2. Experimental Mess

C.S. suffers from the same difficulties as natural sciences.

- Rely on large, distributed, evolving, prototype hard/software
- Validation on a few datasets/scenarios? ☹

But do we **really** have to care in CS?

Yes. **Model** $\neq$ **Reality**. Although designed and built by human beings, computers are so complex that mistakes are easy to do...

- Experiments: Mytkowicz, Diwan, Hauswirth, Sweeney. Producing wrong data without doing anything obviously wrong!. SIGPLAN Not. 44(3), March 2009

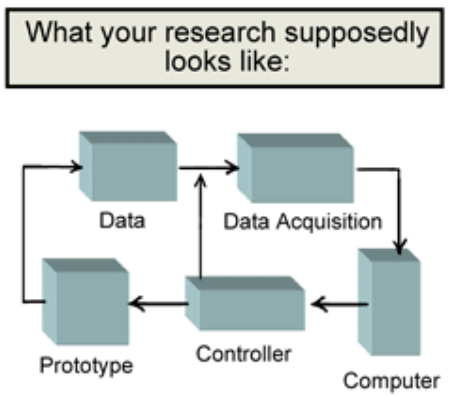


Figure 1. Experimental Diagram

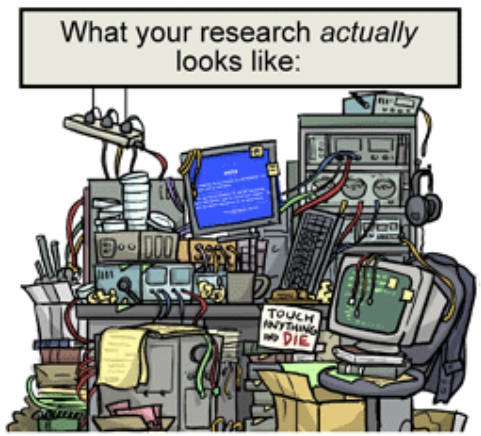


Figure 2. Experimental Mess

C.S. suffers from the same difficulties as natural sciences

- Rely on large, distributed, evolving, prototype hard/software
- Validation on a few datasets/scenarios? ☹

- Statistics: Trouble at the lab, The Economist 2013

  According to some estimates, three-quarters of published scientific papers in the field of machine learning are bunk because of this "overfitting".
  Sandy Pentland, MIT

- Numerical reproducibility: simulated results are often non reproducible when moving from a platform to another or exploiting parallel architectures

Reproducible Research: the New Buzzword?

H2020-EINFRA-2014-2015

A key element will be capacity building to link literature and data in order to enable a more transparent evaluation of research and reproducibility of results.

More and more workshops

- Workshop on Duplicating, Deconstructing and Debunking (WDDD) (2002-2016 edition)
- AMP Workshop. Reproducible Research: Tools and Strategies for Scientific Computing (2011)
- Working towards Sustainable Software for Science: Practice and Experiences (2013)
- REPPAR'16: 3rd International Workshop on Reproducibility in Parallel Computing
- Reproducibility@XSEDE: An XSEDE14 Workshop
- Reproduce/HPCA 2014
- TRUST 2014, 2015
- Talk at SC by V. Stodden a few months ago

Should be seen as opportunities to share experience

Reproducibility: What Are We Talking About?

1934: Karl Popper introduces the notion of falsifiability and crucial experiment and puts reproducing the work of others at the core of science
Short introduction: A Summary of Scientific Method, Peter Kosso, Springer

Reproducibility of experimental results is the hallmark of science
[Drummond, 2009]

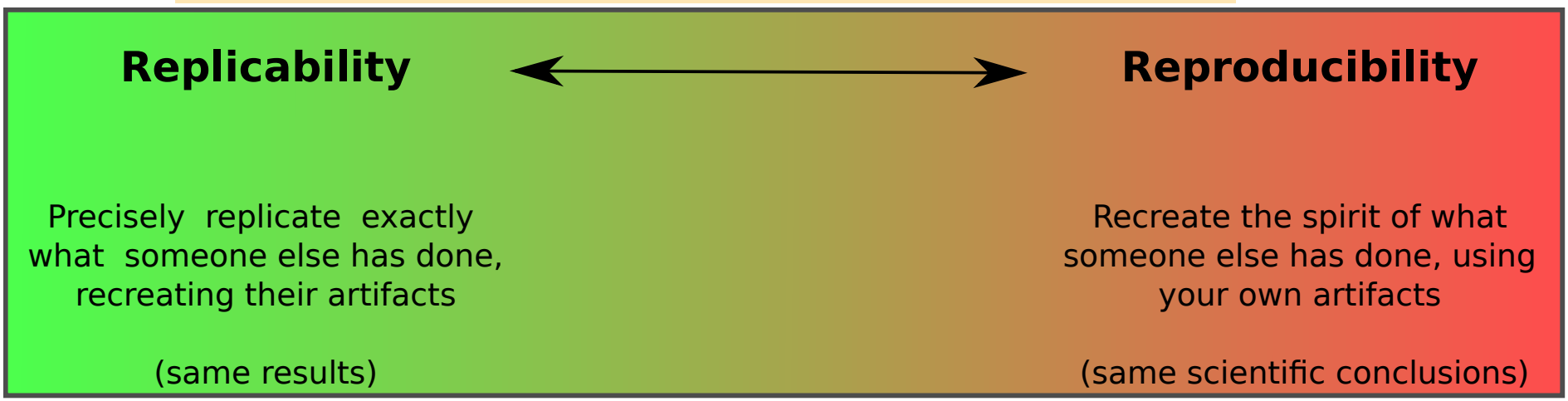
Reproducibility: What Are We Talking About?

1934: Karl Popper introduces the notion of falsifiability and crucial experiment and puts reproducing the work of others at the core of science
Short introduction: A Summary of Scientific Method, Peter Kosso, Springer

Reproducibility of experimental results is the hallmark of science
[Drummond, 2009]

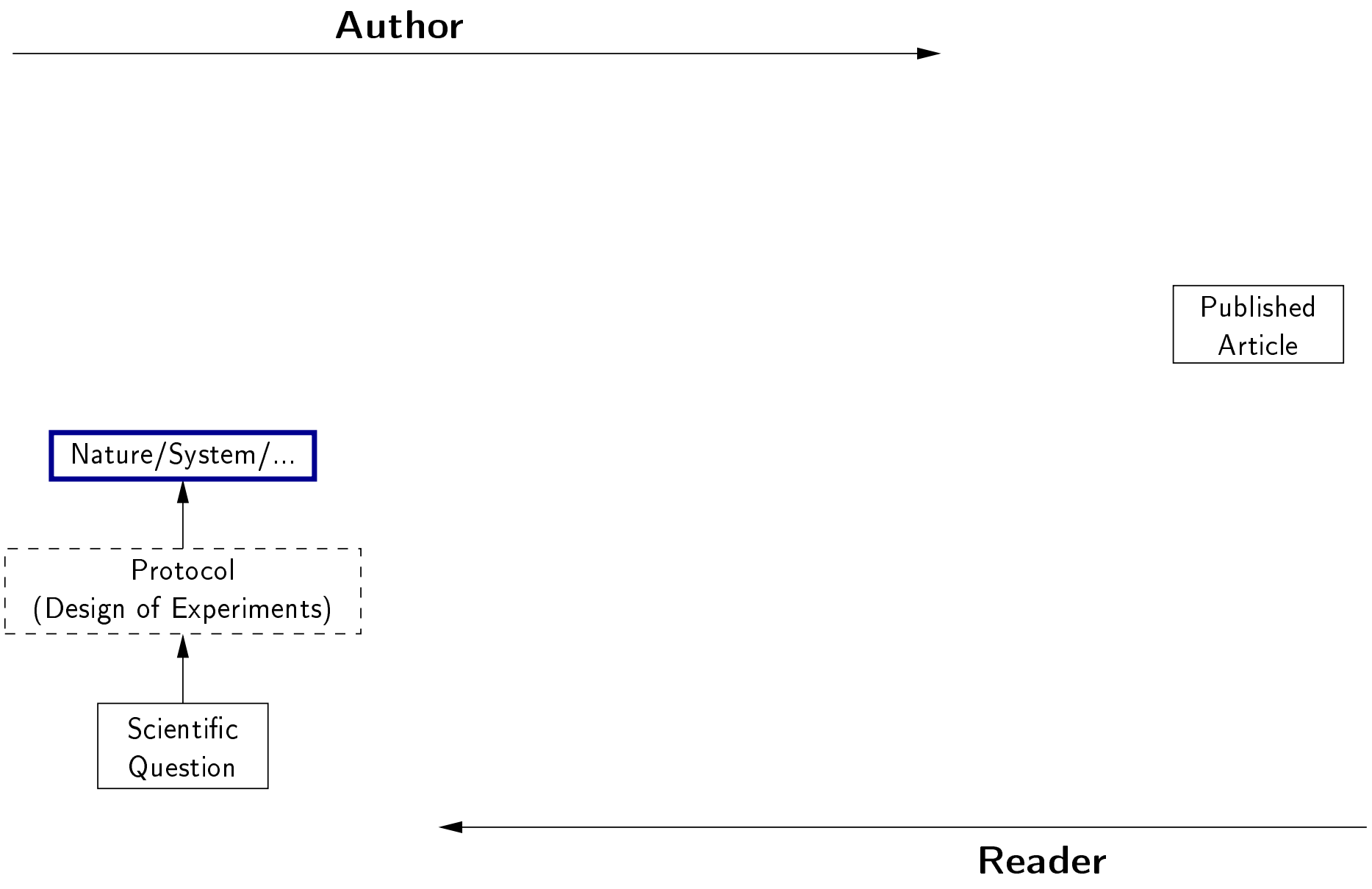
⚠ **Terminology varies** ⚠

Repetition, replication, variation, reproduction, corroboration (Feitelson, 2015)
But also International Vocabulary of Metrology, JCGM 200:2012 is inspiring ACM



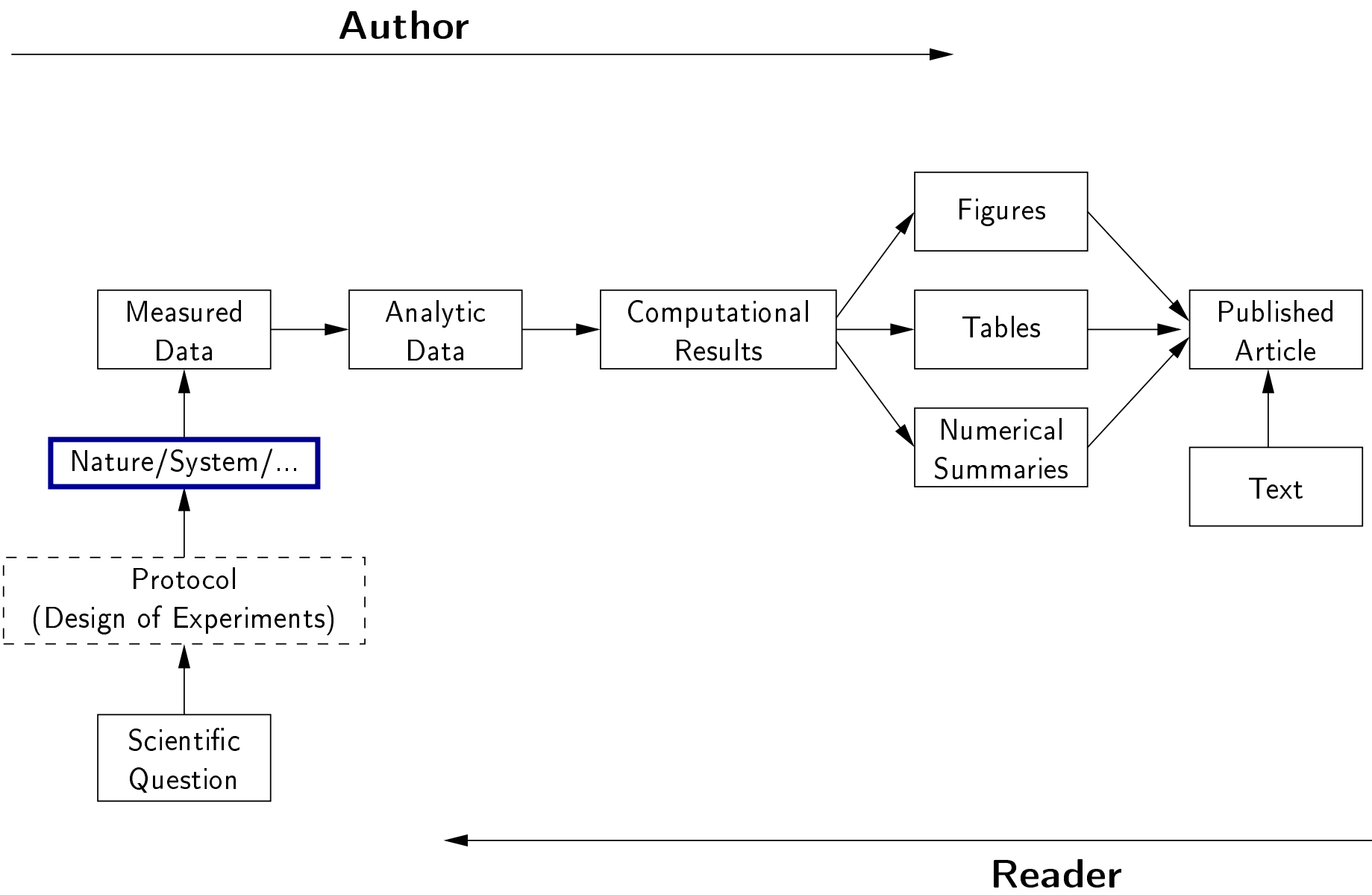
Inspired by Andrew Davison (AMP Workshop on Reproducible research)

Reproducible Research: Trying to Bridge the Gap



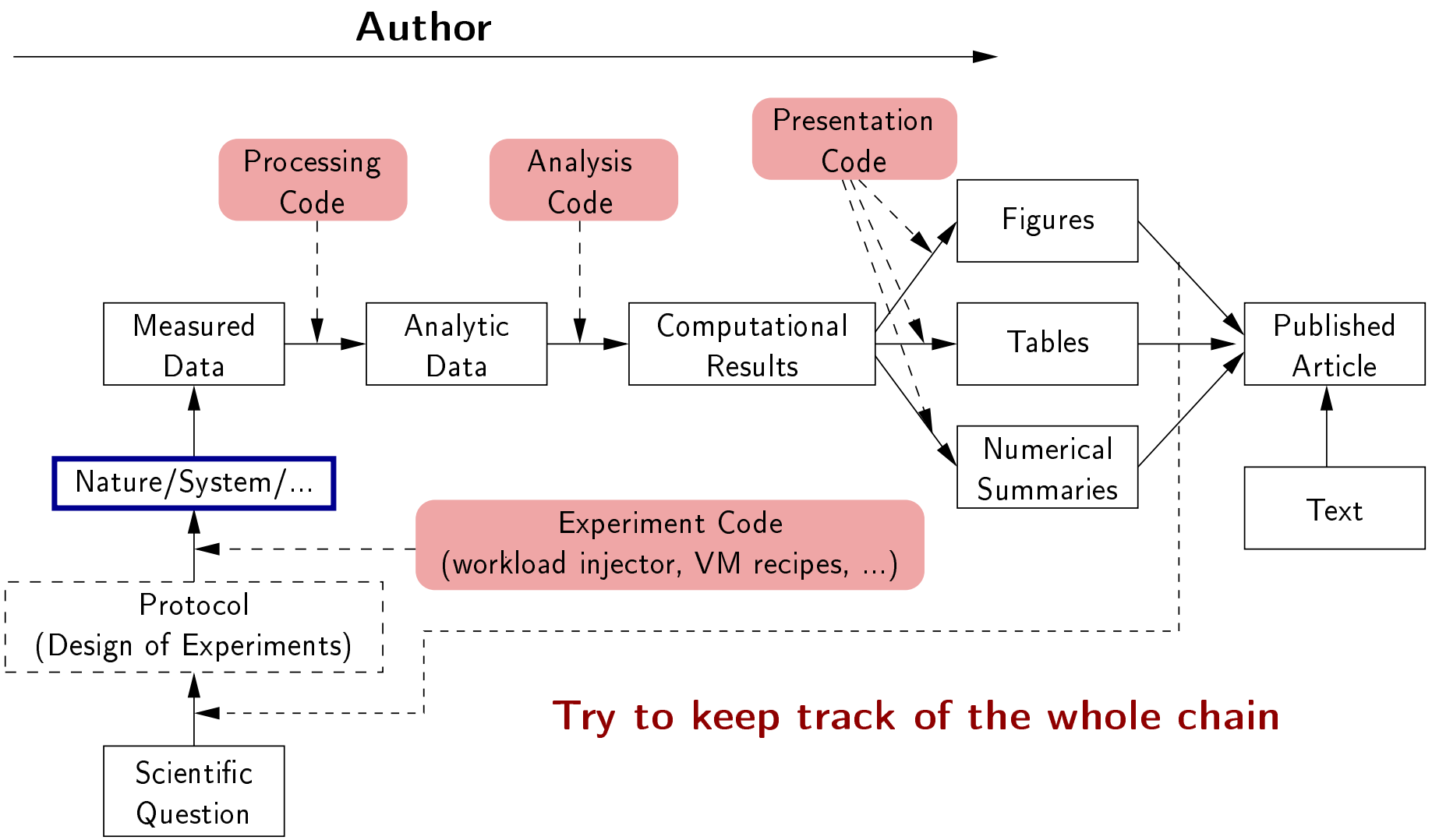
Inspired by Roger D. Peng's lecture on reproducible research, May 2014

Reproducible Research: Trying to Bridge the Gap



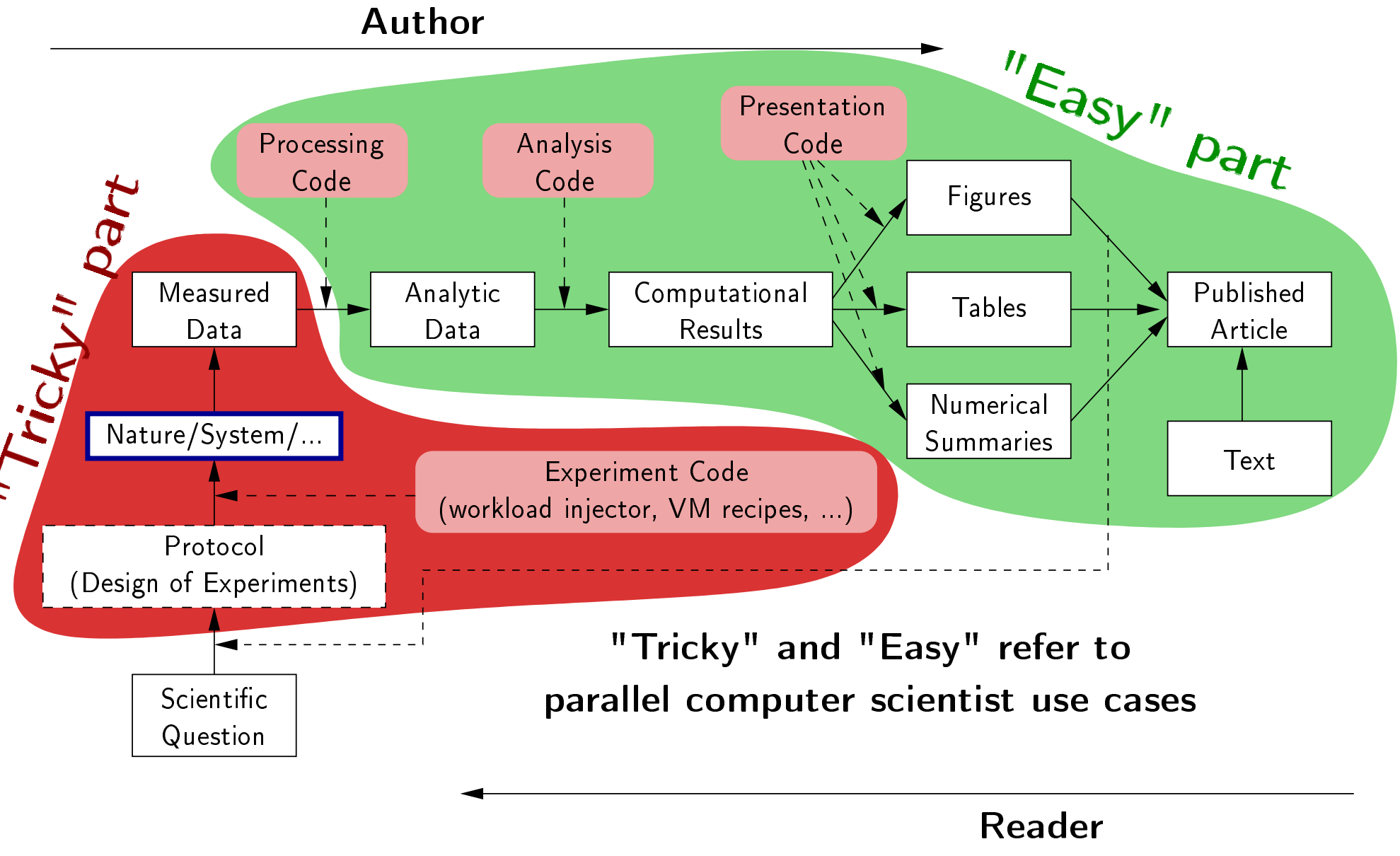
Inspired by Roger D. Peng's lecture on reproducible research, May 2014

Reproducible Research: Trying to Bridge the Gap



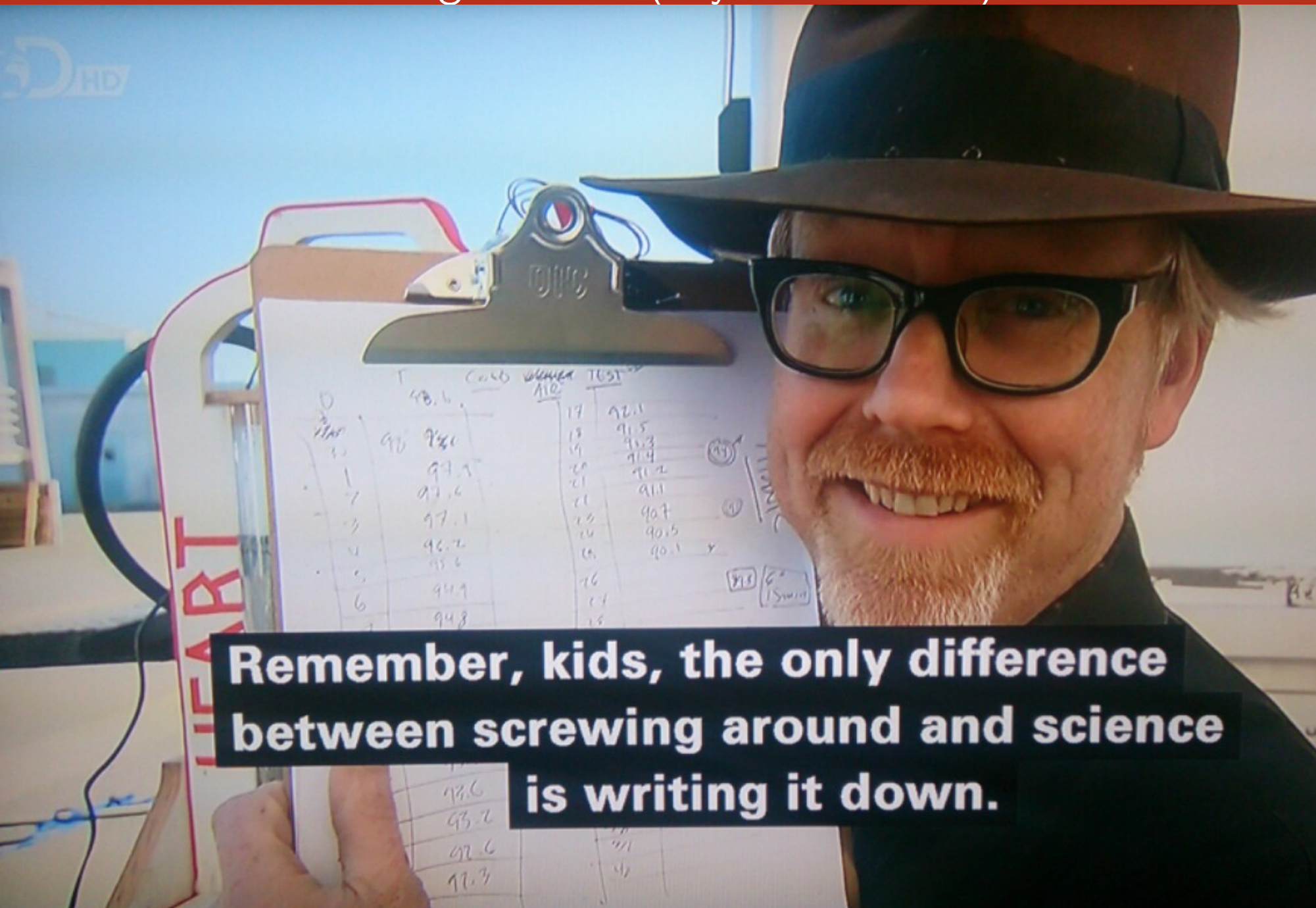
Inspired by Roger D. Peng's lecture on reproducible research, May 2014

Reproducible Research: Trying to Bridge the Gap



Inspired by Roger D. Peng's lecture on reproducible research, May 2014

Science vs. Screwing Around (Mythbusters 😉)



Outline

Our Approach: An Infrastructure to Support Provenance-Rich Papers [Koop et al., ICCS 2011]

- Tools for *authors* to create reproducible papers
  - Specifications that encode the computational processes
  - Package the results
  - Link from publications

Support different approaches

- Tools for testers to repeat and validate results
  - Explore different parameters, data sets, algorithms
- Interfaces for searching, comparing and analyzing experiments and results
  - Can we discover better approaches to a given problem?
  - Or discover relationships among workflows and the problems?
  - How to describe experiments?

Courtesy of Juliana Freire (AMP Workshop on Reproducible research)

An *Provenance-Rich* Paper: ALPS2.0

The ALPS project release 2.0: Open source software for strongly correlated systems

B. Bauer[1] L. D. Carr[2] H.G. Evertz[3] A. Feiguin[4] J. Freire[5] S. Fuchs[6] L. Gamper[1] J. Gukelberger[1] E. Gull[7] S. Guertler[8] A. Hehn[1] R. Igarashi[9,10] S.V. Isakov[1] D. Koop[5] P.N. Ma[1] P. Mates[1,5] H. Matsuo[11] O. Parcollet[12] G. Pawłowski[13] J.D. Picon[14] L. Pollet[1,15] E. Santos[5] V.W. Scarola[16] U. Schollwöck[17] C. Silva[5] B. Surer[1] S. Todo[10,11] S. Trebst[18] M. Troyer[1]‡ M. L. Wall[2] P. Werner[1] S. Wessel[19,20]

[1]Theoretische Physik, ETH Zurich, 8093 Zurich, Switzerland
[2]Department of Physics, Colorado School of Mines, Golden, CO 80401, USA
[3]Institut für Theoretische Physik, Technische Universität Graz, A-8010 Graz, Austria
[4]Department of Physics and Astronomy, University of Wyoming, Laramie, Wyoming 82071, USA
[5]Scientific Computing and Imaging Institute, University of Utah, Salt Lake City, Utah 84112, USA
[6]Institut für Theoretische Physik, Georg-August-Universität Göttingen, Göttingen, Germany
[7]Columbia University, New York, NY 10027, USA
[8]Bethe Center for Theoretical Physics, Universität Bonn, Nussallee 12, 53115 Bonn, Germany

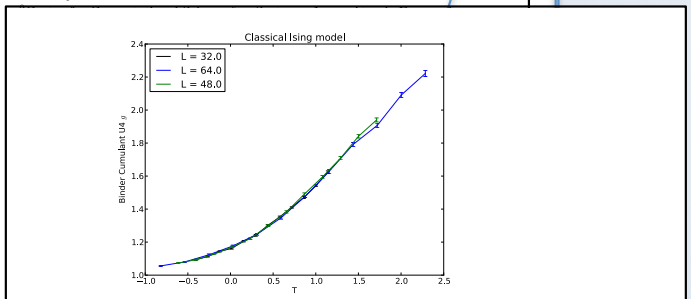
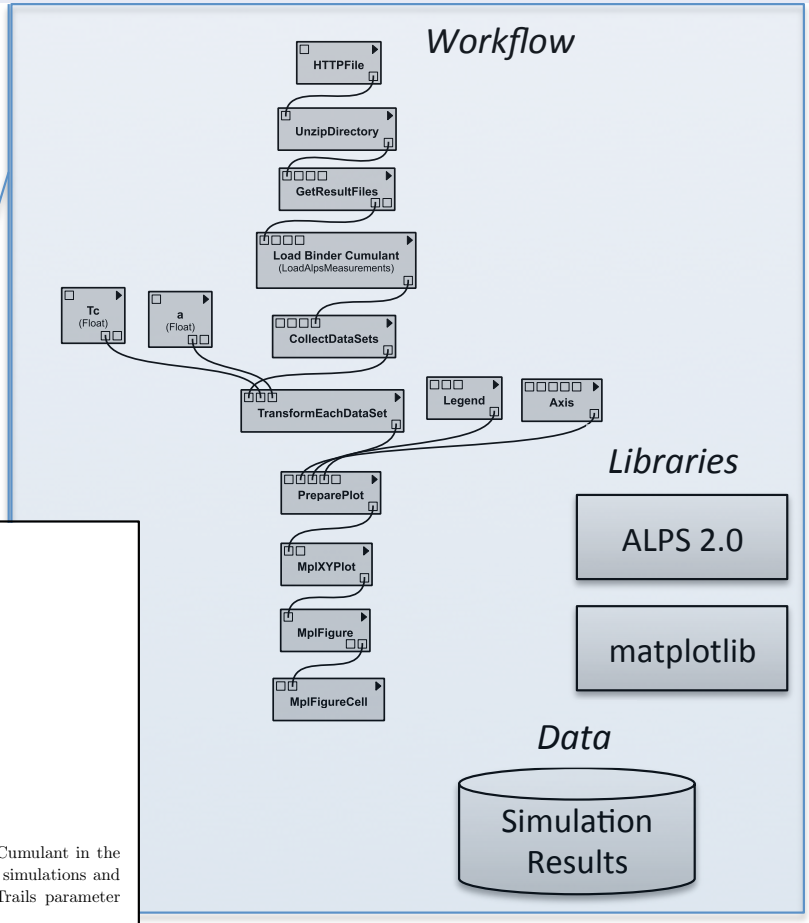


Figure 3. In this example we show a data collapse of the Binder Cumulant in the classical Ising model. The data has been produced by remotely run simulations and the critical exponent has been obtained with the help of the VisTrails parameter exploration functionality.



[Bauer et al., JSTAT 2011]

Courtesy of Juliana Freire (AMP Workshop on Reproducible research)

VCR: A Universal Identifier for Computational Results

Chronicing computations in real-time

VCR computation platform Plugin = Computation recorder

Regular program code

```
figure1 = plot(x)
save(figure1,'figure1.eps')
```

```
> file /home/figure1.eps saved
>
```

Courtesy of Matan Gavish and David Donoho (AMP Workshop on Reproducible research)

VCR: A Universal Identifier for Computational Results

Chronicing computations in real-time

VCR computation platform Plugin = Computation recorder

Program code with VCR plugin

```
repository vcr.nature.com
verifiable figure1 = plot(x)
```

```
> vcr.nature.com approved:
> access figure1 at https://vcr.nature.com/ffaaffb148d7
```

Courtesy of Matan Gavish and David Donoho (AMP Workshop on Reproducible research)

VCR: A Universal Identifier for Computational Results

Word-processor plugin App

LaTeX source

```
\includegraphics{figure1.eps}
```

LaTeX source with VCR package

```
\includeresult{vcr.thelancet.com/ffaaffb148d7}
```

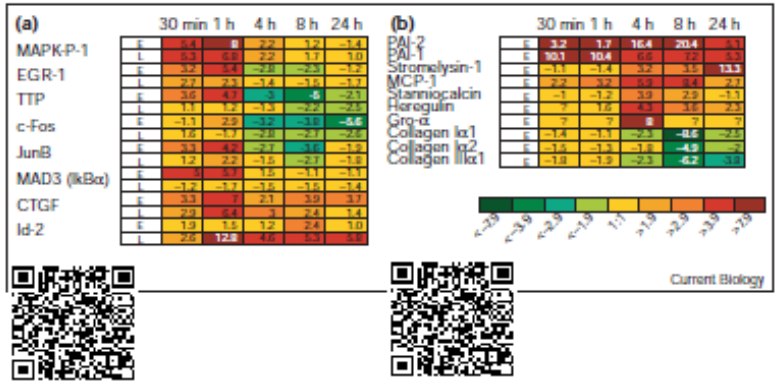
Permanently bind printed graphics to underlying result content

Courtesy of Matan Gavish and David Donoho (AMP Workshop on Reproducible research)

VCR: A Universal Identifier for Computational Results

Figure 3

Time course of serum stimulation. **(a)** Early passage (E: PD30) or late passage (L: PD89) BJ cultures were held in 0.5% serum for 2 days, then stimulated with 10% FBS. RNA levels from cultures at the indicated time points (Cy5 channel) were compared with the uninduced starting culture (Cy3 channel). Positive values indicate higher expression in induced cells; negative values indicate lower expression in induced cells. Question marks indicate that there was insufficient signal for detection. A complete listing of serum-responsive genes from this analysis is provided in Supplementary material. **(b)** The serum-responsiveness of select senescence-regulated genes in early passage (PD30) BJ fibroblasts.



senescence response appears to overlap substantially with gene expression patterns observed in activated fibroblasts during wound healing [24–26]. MCP-1, Gro-α, IL-1β and IL-15 are strong effectors of macrophage and neutrophil recruitment and activation [27,28]. The upregulation of Toll (Tlr-4) in senescent fibroblasts confirms the overall immune response behavior at senescence. Tlr-4 is an IL-1 receptor homolog and is implicated in the activation of the gene regulatory protein NF-κB, a function proposed to be part of the innate immune response [29]. The induction of IL-15 at senescence is also consistent with an innate immune response, as IL-15 can be induced by NF-κB-dependent transcription [30] and also participates in inflammatory disease processes [28].

Deficiencies in the response of senescent cells to serum stimulation have been reported, and include an inability to induce the expression of c-*fos* mRNA [31] and markers of late G1 and S phase [32]. In response to serum, expression of inflammatory chemokines, matrix-degrading proteases and their modulators is induced in early-passage dermal fibroblasts, and expression of matrix collagens is reduced. This transient burst of activity may represent the natural response of these cells in wound repair [24]. Id-2 transcripts were hyper-induced in serum-stimulated senescent

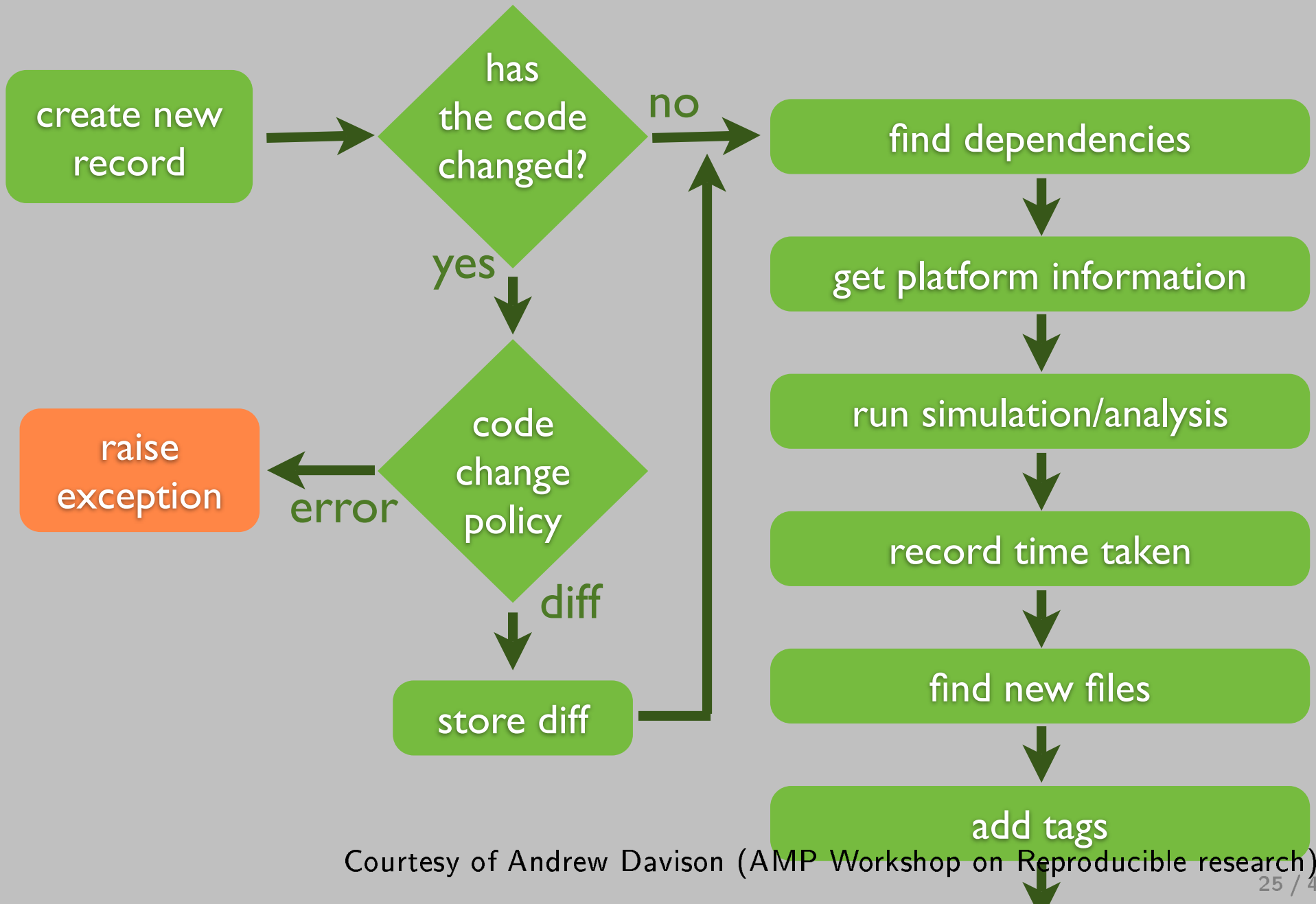
states overlap substantially with those in telomere-induced senescence (W.F., D.N.S., R. Allsopp, S. Lowe, and G. Ferbeyre, unpublished observations) and thus are likely to use many of the same activation processes.

The pattern of gene expression at senescence varies substantially in different cell types. Although the expression of matrix and structural proteins, such as the collagens, keratins and auxiliary factors, is repressed in RPE cells, inflammatory regulators are not induced, in contrast to dermal fibroblasts. Physiologically, this would make sense, as an acute inflammatory response in a tissue critical for normal vision would be likely to have deleterious consequences. However, as the RPE layer has a central role in the deposition and maintenance of extracellular matrix in the retina, decrements in the ability of senescent RPE cells to maintain appropriate expression patterns, as evidenced by decreased expression of collagens, keratins, aggrecan, transglutaminase and so on, would be predicted to have adverse affects on retinal architecture. Dysfunction of the RPE cell layer is considered to be a substantial factor in the development of age-related macular degeneration [36].

Surprisingly, early-passage HUVECs overexpress many of the markers associated with senescence in dermal fibro-

Courtesy of Matan Gavish and David Donoho (AMP Workshop on Reproducible research)

Sumatra: an "experiment engine" that helps taking notes



Courtesy of Andrew Davison (AMP Workshop on Reproducible research)

Sumatra: an "experiment engine" that helps taking notes

```
$ smt comment 20110713-174949 "Eureka! Nobel prize
here we come."
```

Courtesy of Andrew Davison (AMP Workshop on Reproducible research)

Sumatra: an "experiment engine" that helps taking notes

```
$ smt tag “Figure 6”
```

Courtesy of Andrew Davison (AMP Workshop on Reproducible research)

Sumatra: an "experiment engine" that helps taking notes

Sumatra: TestProject: List of records

http://127.0.0.1:8002/

Google

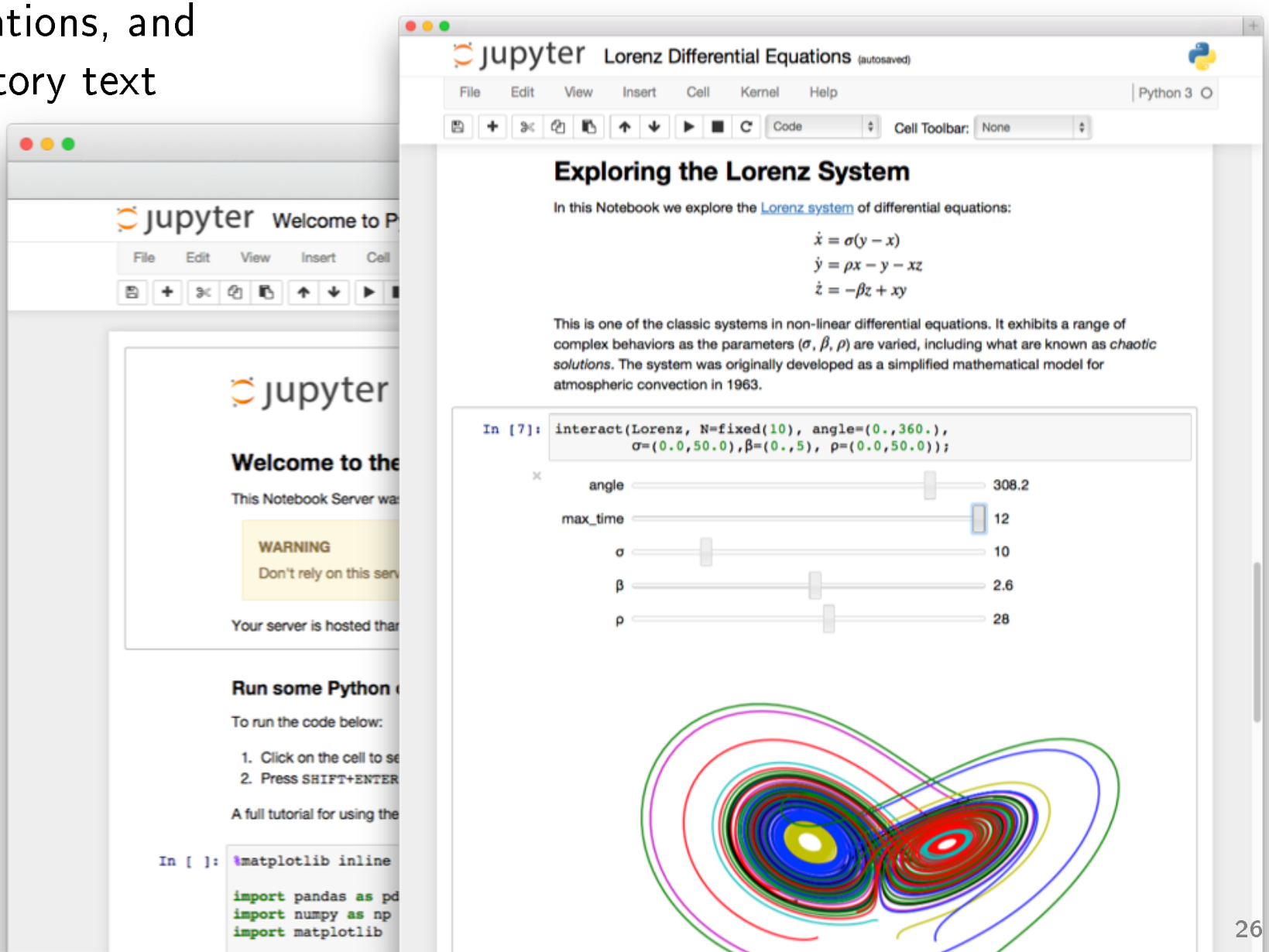
TestProject: List of records

Delete / include data	Label	Reason	Outcome	Duration	Processes	Simulator		Script			Date	Time	Tags
						Name	Version	Repository	Main file	Version			
☐	20100709-154255		'Eureka! Nobel prize here we come.'	0.59 s		Python	2.5.2	/Users/andrew/tmp/SumatraTest	main.py	396c2020ca50	09/07/2010	15:42:55	
☐	20100709-154309			0.59 s		Python	2.5.2	/Users/andrew/tmp/SumatraTest	main.py	396c2020ca50	09/07/2010	15:43:09	
☐	haggling	'determine whether the gourd is worth 3 or 4 shekels'	'apparently, it is worth NaN shekels.'	0.59 s		Python	2.5.2	/Users/andrew/tmp/SumatraTest	main.py	396c2020ca50	09/07/2010	15:43:20	foobar
☐	20100709-154338	'test effect of a smaller time constant'		0.59 s		Python	2.5.2	/Users/andrew/tmp/SumatraTest	main.py	396c2020ca50	09/07/2010	15:43:38	
☐	haggling_repeat	Repeat experiment haggling	The new record exactly matches the original.	0.58 s		Python	2.5.2	/Users/andrew/tmp/SumatraTest	main.py	396c2020ca50	09/07/2010	15:43:47	

Courtesy of Andrew Davison (AMP Workshop on Reproducible research)

Ipython/Jupyter Notebook

Web app: create and share documents that contain live code, equations, visualizations, and explanatory text



Reprozip

Automagically pack your experiment to fight dependency hell

ON THE ORIGINAL MACHINE

```
$ pip install reprozip
$ reprozip trace ./myexperiment -my --options inputs/somefile.csv other_file_here.bin
experiment: 0%... 25%... 50%... 75%... 100%
result: 42.137
Configuration file written in .reprozip/config.yml
Edit that file then run the packer -- use 'reprozip pack -h' for help
$ reprozip pack my_experiment.rpz
[REPROZIP] 17:26:42.588 INFO: Creating pack my_experiment.rpz...
[REPROZIP] 17:26:42.589 INFO: Adding files from package coreutils...
[REPROZIP] 17:26:42.601 INFO: Adding files from package libc6...
[REPROZIP] 17:26:42.906 INFO: Adding other files...
[REPROZIP] 17:26:43.450 INFO: Adding metadata...
```

ON ANOTHER MACHINE

```
$ pip install reprounzip[all]
$ reprounzip vagrant setup my_experiment.rpz mydirectory
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Importing base box 'remram/debian-7-amd64'...
==> default: Booting VM...
==> default: Machine booted and ready!
==> default: Running provisioner: shell...
$ reprounzip vagrant run mydirectory
experiment: 0%... 25%... 50%... 75%... 100%
result: 42.137
$ reprounzip vagrant upload /tmp/new_config:global-config
$ reprounzip vagrant run mydirectory --cmdline ./myexperiment --other --options
inputs/somefile.csv
experiment: 0%... 25%... 50%... 75%... 100%
result: -17.814
```

New Tools for Computational Reproducibility

- Dissemination Platforms:
  - ResearchCompendia.org
  - IPOL
  - Madagascar
  - MLOSS.org
  - thedatahub.org
  - nanoHUB.org
  - Open Science Framework
  - The DataVerse Network
  - RunMyCode.org
- Workflow Tracking and Research Environments:
  - VisTrails
  - Kepler
  - CDE
  - Galaxy
  - GenePattern
  - Synapse
  - Sumatra
  - Taverna
  - Pegasus
- Embedded Publishing:
  - Verifiable Computational Research
  - Sweave
  - knitR
  - Collage Authoring Environment
  - SHARE

Courtesy of Victoria Stodden (UC Davis, Feb 13, 2014)

And also: **Org-Mode** ☺, **Figshare**, **Zenodo**, **ActivePapers** ☺, **Elsevier executable paper** ☹, ...

Outline

A Difficult Trade-off

Many different tools/approaches developed in various communities

But mainly two approaches: Automatic vs. Explicit

- **Automatically keeping track of everything**
  - the code that was run (source code, libraries, compilation procedure)
  - processor architecture, OS, machine, date, ...
- **Ensuring others can understand/adapt what was done**
  - Why did I run this? Does it still work when I change this piece of code for this one?

A Difficult Trade-off

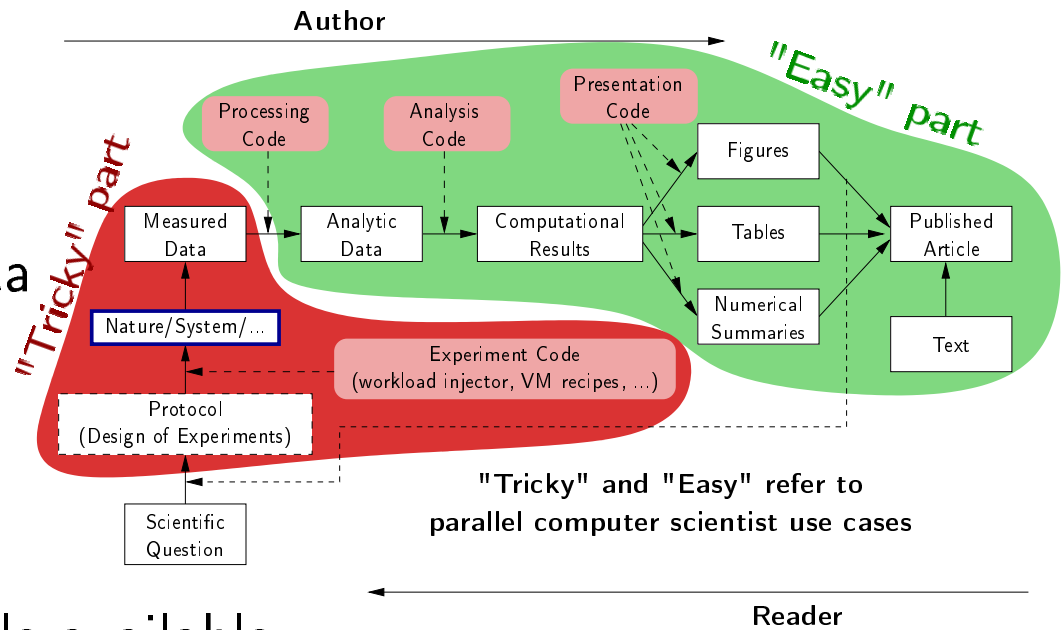
Many different tools/approaches developed in various communities

But mainly two approaches: Automatic vs. Explicit

- **Automatically keeping track of everything**
  - the code that was run (source code, libraries, compilation procedure)
  - processor architecture, OS, machine, date, ...
- **Ensuring others can understand/adapt what was done**
  - Why did I run this? Does it still work when I change this piece of code for this one?

And the following key points:

1. Replicable article
2. Logging your activity
3. Logging and backup your data
4. Organizing your data
5. Mastering your environment
6. Controlling your experiments
7. Making your data/code/article available



3. Logging and backup your data

What are the options?

- Nothing ☹ (remember the funny examples from the beginning. . . ☺)
- Incremental backup mechanisms (e.g., time machine)
- The cloud! (e.g., Dropbox and Google Drive ☹ . . .)
- Flexible version control systems (e.g. git ☺) where you're in control of what's happening
  - Use a crontab if you really do not want to think about it
  - We have come up with a specific git branching workflow for managing experimental results

4. Organizing and managing your data

- Use the machine readable CSV format
- Provide raw data and meta data, not just statistical outputs
- Organization
  - Explain your conventions (e.g., `src/`, `data/`, `script/`, `journal.org`)
  - Git submodules
- Never do data manipulation and statistical tests by hand or with a spreadsheet 
- Use R, Python or another free software to read and process raw data.
  - Use a workflow that documents both data and process
  - The org-mode tangling mechanism may help

5. Mastering your environment

What are the options?

- Nothing 😉 No, it's not, you have to do something...
- Restrict your tools/dependencies to the bare minimum (e.g., python)
  - List them all manually in a README
  - Use custom shell scripts or `sosreport` that log all the dependencies you are aware. Ask your friends to check whether this is sufficient...
  - Combine everything in *activepapers*, i.e., an HDFS5 file combining datasets and programs working on these datasets in a single package, along with meta data, history, ...
- Create and distribute your own virtual image (VM, docker, Singularity)
- Have tools that automatically keep track of dependencies/files and packages up the Code, Data, and Environment
  - CDE (Guo et al., 2011) ReproZip (Freire et al., 2013), CARE (Janin et al., 2014),
  - See Preserve the Mess or Encourage Cleanliness? (Thain et al., 2015)
- Use a specific tool to generate customized appliances (kvm, LXC, Virtualbox, iso, ...): recipes with steps and aliases, execution in contexts, checkpoints, ... (*Kameleon*)

6. Controlling your experiments

- Naive way: sh + ssh + ...
- Better way: use a workflow management system (taverna, galaxy, kepler, vistrails, ...)
- Parallel/distributed experiments require specific experiment engines
  - Expo (2007-, G5K)
  - XPflow (2012-, G5K)
  - Execo (2013-, G5K)

  } although nothing specific to G5K

  - Plush (2006-, Planetlab)
  - OMF (2009-, Wireless)
  - Splay (2008), ...

  They differ in the underlying paradigms and the platforms for which they have been designed

  > A survey of general-purpose experiment management tools for distributed systems, T. Buchert, C. Ruiz, L. Nussbaum, O. Richard, FGCS, 2014

- Control your **numerical results** (random generators, libraries, rounding and non-determinism, ...)

7. Making your data/code/article available

- Your webpage ☹
- Figshare, Zenodo ☺, ...
- Companion websites (elsevier executable paper ☹, runmycode, exec&share ☺, ...)
- Github (damn, they're good! ☺), ...

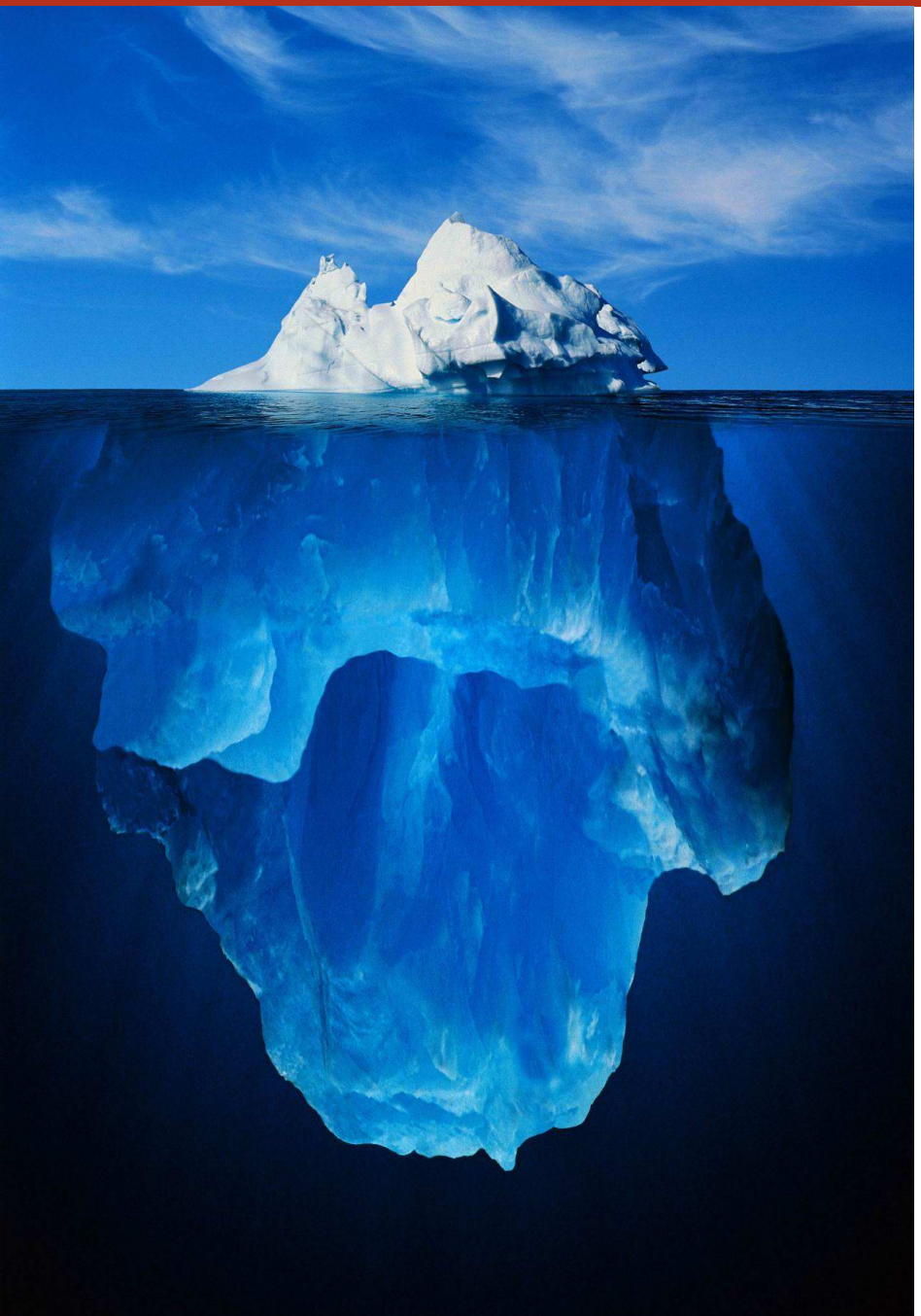
This may seem easy but is more tricky than it looks like:

- Arbitrary limits can make your life painful
- Perennity (Roberto Di Cosmo's talk at R^4)
  - CodeSpaces murdered on Amazon, Google Code termination, Gitorious shutdown, ...
  - Disruption of the web of reference: URLs decay (half-life of 4 years), DOIs have little guarantee, ...

Many legal aspects about data/code/idea sharing

- I am a civil servant and I strongly believe research is a team sport
- Intellectual property is an important topic we do not want to leave to bureaucrats and lawyers...

Remember the general picture



The article is only the top of the iceberg, we need a way to dive and unveil what's behind every graphics and number...

1. Replicable article (Literate programming)

Donald Knuth: explanation of the program logic in a natural language interspersed with snippets of macros and traditional source code.

I'm way too 3133t to program this way 🙂 but that's exactly what we need for writing a reproducible article/analysis!

KnitR (a.k.a. Sweave)

For R and ~~emacs~~ users. Easy replicable articles with a modern IDE (e.g., Rstudio)

Ipython/Jupyter notebook

Python user $\rightsquigarrow$ go for Jupyter. Web app, easy to use/setup... Writing replicable article may be tricky though

Org-mode (my favorite! requires emacs though)

- Org-mode is plain text, very smooth, works both for html, pdf, ...
- Allows to combine all my favorite languages

Note that this generation depends on a computational environment whose preservation is not addressed here (see for example *activepapers*).

A replicable article with Org-Mode

See for example our recent article on the simulation of Multithreaded Sparse Linear Algebra Solvers at ICPADS 2015.
Here are the following important features to exploit:

Structure highly hierarchical

- Sectioning, itemize, enumerate, fonts
- Tags to control what will be exported

Export in several output formats

- Fine control with `#+BEGIN_LaTeX`
- Unfortunate need for verbose headers (because of LaTeX☹) and black magic in the end of the file (for emacs portability ☹)

Babel (the literate programming part of org-mode). Many possible usage:

- Run babel on export
- Or not... and make sure intermediate results are stored (this is how I proceed)
- Dependencies can be expressed
- Caching mechanism
- Side effects are the enemy of reproducibility

2. Logging your activity (Laboratory Notebook)

Do not tie your hands with non-free software like Evernote or OneNote

- Org-mode again!
  - Capture mechanism (notes, todo, . . .)
  - Babel favors code reuse, ssh connections in sessions, meta-programming
  - Tagging mechanism to structure the journal
  - Link mechanism, Todo, Calendar views, Tables, . . .

I have a very intense usage and so do all my master/PhD students (e.g., here)

- Spending more than an hour without at least writing what you're working on is not right. . . Take a 5 minutes break and ask yourself what you're doing, what is keeping you busy and where all this is leading you
- While working on something, you will often notice/think about something you should fix/improve but you just don't want to do it now. Take 20 seconds to write a TODO entry
- There are moments where you have to wait for something (compiling, deployment, . . .). It is generally the perfect time for improving your notes (e.g., detail the steps to accomplish a TODO entry)
- By the end of the day: daily (and weekly) review!
  - Update your lists, decide the next steps, summarize what you did/learnt,. . .

Pros and Cons of these three tools

- Ipython notebook:
  - ☺ Easy to set up, user-friendly, machine readable format (JSON), easy sharing on the cloud
  - ☹ Writing an article, JSON, not fully polyglot
- knitR/Rstudio:
  - ☺ Easy to set up, user-friendly, writing articles, easy publishing on rpubs
  - ☹ not fully polyglot
- Emacs/Org-mode:
  - ☹ Emacs, steep learning curve
  - ☺ Powerful and versatile, yields control to power users, works both for writing articles and a notebook, good integration on github

The ultimate tool would combine an engine in an editor that allows collaborative interactive edition

Outline

Where are we standing now?

- Changes in funding agency requirements
  - Starting? I hardly see how they could really enforce things
- Changes in journal/conferences publication requirements
  - Several attempts (artifact review and branding)
- Cultural changes in our relation to publication

Where are we standing now?

- Changes in funding agency requirements
  - Starting? I hardly see how they could really enforce things
- Changes in journal/conferences publication requirements
  - Several attempts (artifact review and branding)
- Cultural changes in our relation to publication

I think the change has to be profound and cannot be top-down

- We should care. What are the incentives?
  - Reproducible papers are more cited? 😉
  - Definitely more efficient (not only in the long run and for the community)
  - It's simply more satisfying... 🙂
- Train our researchers and students to use better tools, better research methodology, statistics/design of experiments, performance evaluation, ...

Next webinars:

- April 5, 2016: Mastering your environment
- May 3, 2016: Numerical reproducibility

https://github.com/alegrand/RR_webinars