Reproducible Research for Computer Scientists

Arnaud Legrand

MOSIG/PDES, Performance Evaluation Workshop,
Grenoble, October 13, 2014

Outline

1. Reproducible Research
   - Looks familiar?
   - How does it work in other sciences?
   - Reproducible Research/Open Science
   - Many Different Alternatives for Reproducible Analysis

2. Reporting Results
   - An IMRAD Report
   - Good Practice for Setting up a Laboratory Notebook
   - Emacs Demo of How to Keep Things Tidy
   - To do for the Next Time

Outline

Frustration as a Reviewer

This may be an interesting contribution but:

- This average value must hide something
- As usual, there is no confidence interval, I wonder about the variability and whether the difference is significant or not
- That can't be true, I'm sure they removed some points
- Why is this graph in logscale? How would it look like otherwise?
- The authors decided to show only a subset of the data. I wonder what the rest looks like
- There is no label/legend/... What is the meaning of this graph? If only I could access the generation script

Frustration as an Author

- I thought I used the same parameters but I'm getting different results!
- The new student wants to compare with the method I proposed last year
- My advisor asked me whether I took care of setting this or this but I can't remember
- 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?
- It worked yesterday!
- 6 months later: why did I do that?

My Feeling

Computer scientists have an incredibly poor training in probabilities, statistics, experiment management

Why should we? Computer are deterministic machines after all, right? ;)

Ten years ago, I've started realizing how lame the articles I reviewed (as well as those I wrote) were in term of experimental methodology.

- Yeah, I know, your method/algorithm is better than the others as demonstrated by the figures
- Not enough information to discriminate real effects from noise
- Little information about the workload
- Would the "conclusion" still hold with a slightly different workload?
- I'm tired of awful combination of tools (perl, gnuplot, sql, . . .) and bad methodology

Common practice in CS

Computer scientists tend to either:

- vary one factor at a time, use a very fine sampling of the parameter range,
- run millions of experiments for a week varying a lot of parameters and then try to get something of it. Most of the time, they (1) don't know how to analyze the results (2) realize something went wrong...

Interestingly, most other scientists do the exact opposite.

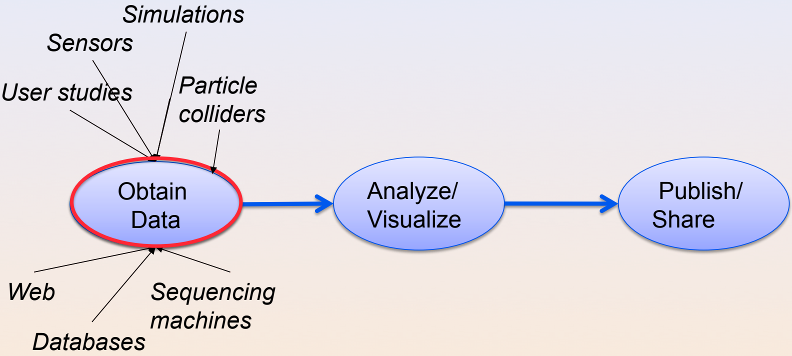
These two flaws come from poor training and from the fact that C.S. experiments are almost free and very fast to conduct

- Most strategies of experimentation (DoE) have been designed to provide sound answers despite all the randomness and uncontrollable factors
- Maximize the amount of information provided by a given set of experiments
- Reduce as much as possible the number of experiments to perform to answer a given question under a given level of confidence

Takes a few lectures on Design of Experiments to improve. But anyone can start by reading Jain's book on The Art of Computer Systems Performance Analysis

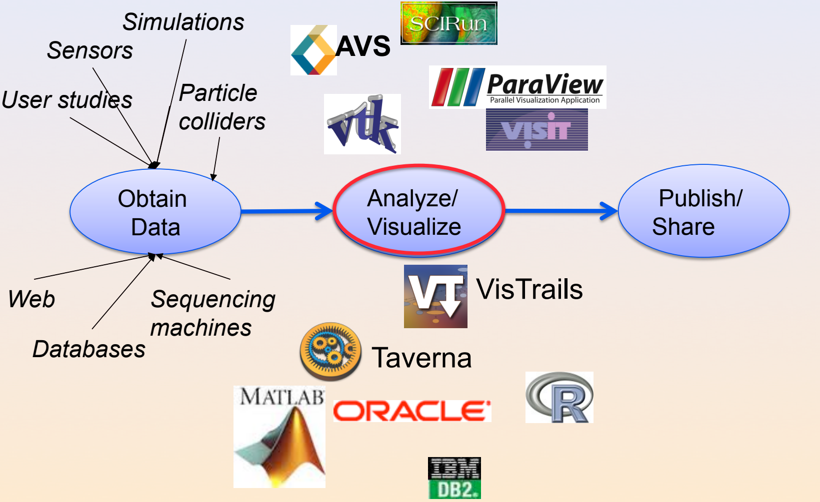
Outline

Science Today: Data Intensive



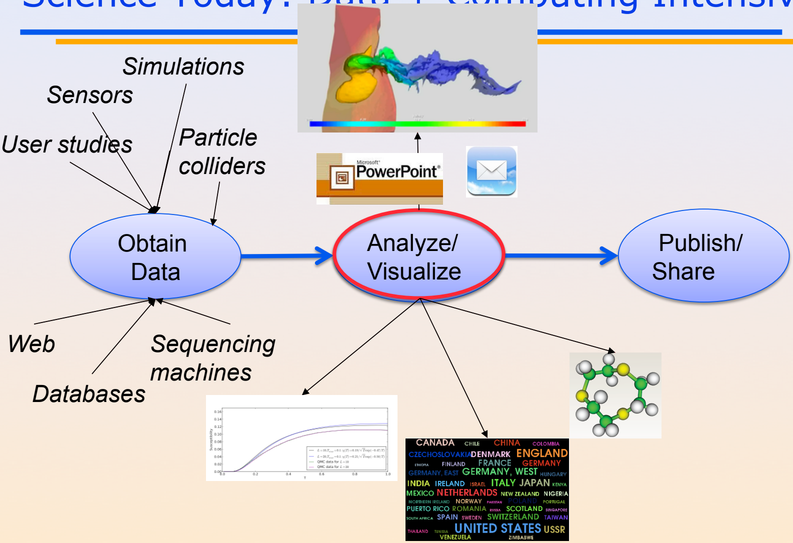
Courtesy of Juliana Freire (AMP Workshop on Reproducible research)

Science Today: Data + Computing Intensive



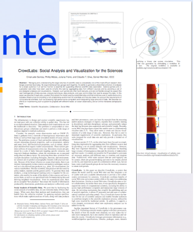
Courtesy of Juliana Freire (AMP Workshop on Reproducible research)

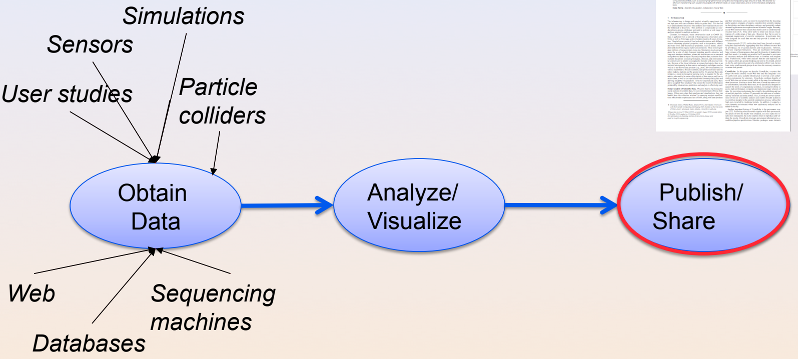
Science Today: Data + Computing Intensive



Courtesy of Juliana Freire (AMP Workshop on Reproducible research)

Science Today: Data + Computing Inte


CrowdLabs: Social Analysis and Visualization for the Sciences



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

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

"*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)

A few Words on Scientific Foundation

- Falsifiability or refutability of a statement, hypothesis, or theory is an inherent possibility to prove it to be false (not "*commit fraud*" but "*prove to be false*").
- Karl Popper makes falsifiability the demarcation criterion to distinguish the scientific from the unscientific

  > *It is not only not right, it is not even wrong!*
  > *– Wolfgang Pauli*

- Theories cannot be proved correct but they can be disproved. Only a few stand the test of batteries of critical experiments.
- It is not all black and white. There are many stories where scientists stick with their theories despite evidences and sometimes, they were even right to do so...

Testing and checking is thus one of the basis of science

Further readings: A Summary of Scientific Method, Peter Kosso, Springer

Why Are Scientific Studies so Difficult to Reproduce?

- Copyright/competition issue
- Publication bias (only the idea matters, not the gory details)
- Rewards for positive results
- Experimenter bias
- Programming errors or data manipulation mistakes
- Poorly selected statistical tests
- Multiple testing, multiple looks at the data, multiple statistical analyses
- ~~Lack of easy-to-use tools~~

Courtesy of Adam J. Richards

Outline

1. Reproducible Research
   - Looks familiar?
   - How does it work in other sciences?
   - Reproducible Research/Open Science
   - Many Different Alternatives for Reproducible Analysis

2. Reporting Results
   - An IMRAD Report
   - Good Practice for Setting up a Laboratory Notebook
   - Emacs Demo of How to Keep Things Tidy
   - To do for the Next Time

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) (2014 edition)
- Reproducible Research: Tools and Strategies for Scientific Computing (2011)
- Working towards Sustainable Software for Science: Practice and Experiences (2013)
- REPPAR'14: 1st International Workshop on Reproducibility in Parallel Computing
- Reproducibility@XSEDE: An XSEDE14 Workshop
- Reproduce/HPCA 2014
- TRUST 2014

Should be seen as opportunities to share experience.

Reproducibility: What Are We Talking About?

Replicability ⟷ **Reproducibility**

Reproduction of the original results using the same tools

- by the original author on the same machine
- by someone in the same lab/using a different machine
- by someone in a different lab

Reproduction using different software, but with access to the original code

Completely independent reproduction based only on text description, without access to the original code

Courtesy of Andrew Davison (AMP Workshop on Reproducible research)

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)
- 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. 2010, Nov.
- Increase in retracted papers:
  - Steen RG, *Retractions in the scientific literature: is the incidence of research fraud increasing?* J Med Ethics 37: 249–253.

Courtesy of Adam J. Richards

Why Bother Making Our Work Reproducible?

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.

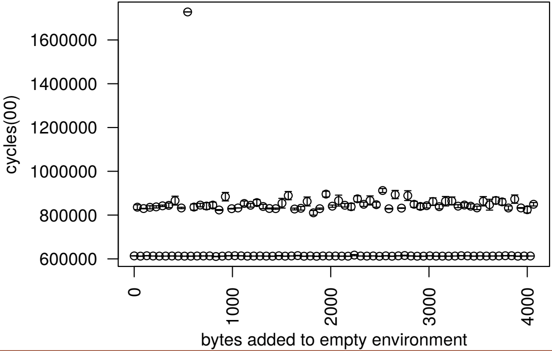
- Retractions: January 2011. Ten papers that Potti coauthored in prestigious journals were retracted for varying reasons
- Some people die and may be getting worthless information that is based on bad science

Courtesy of Adam J. Richards

Is CS Concerned Really With This?

Yes, although designed and built by human beings, computers are so complex that mistakes are easy to do. . .

- T. Mytkowicz, A. Diwan, M. Hauswirth, and P. F. Sweeney. Producing wrong data without doing anything obviously wrong!. SIGPLAN Not. 44(3), March 2009



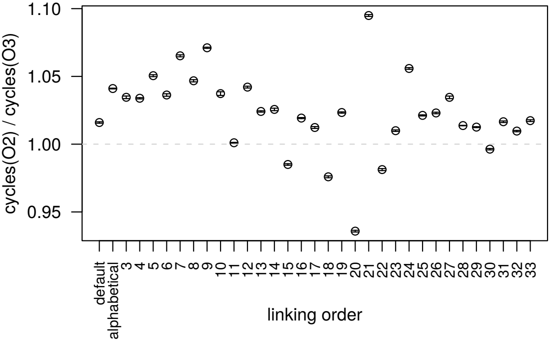
Key principles of experiment design

- Randomize to reduce bias
- Replicate to increase reliability

Is CS Concerned Really With This?

Yes, although designed and built by human beings, computers are so complex that mistakes are easy to do...

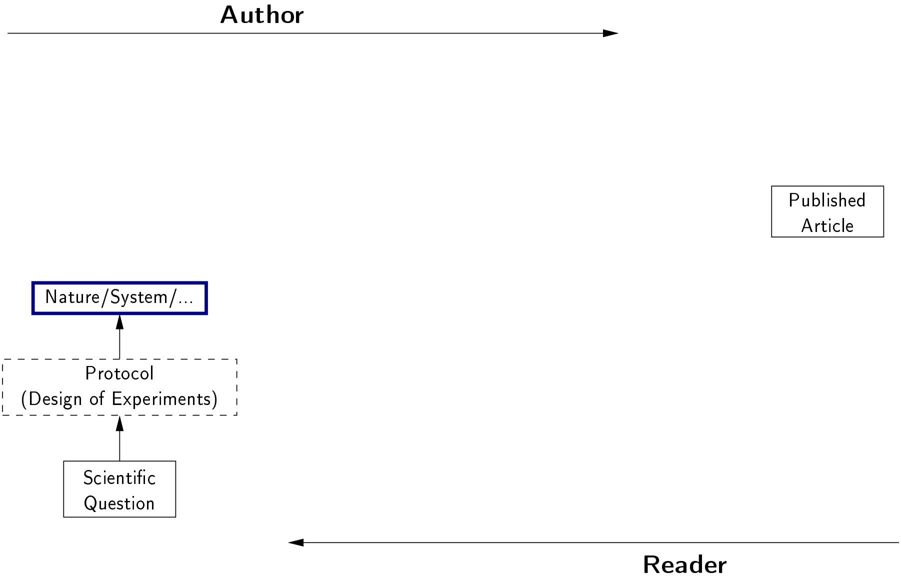
- T. Mytkowicz, A. Diwan, M. Hauswirth, and P. F. Sweeney. Producing wrong data without doing anything obviously wrong!. SIGPLAN Not. 44(3), March 2009



Key principles of experiment design

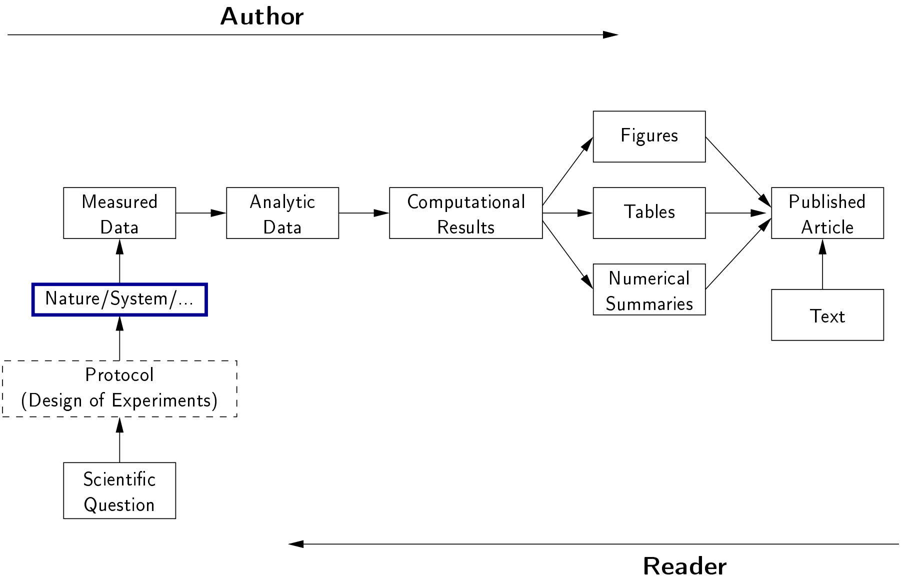
- Randomize to reduce bias
- Replicate to increase reliability

Reproducible Research: Trying to Bridge the Gap



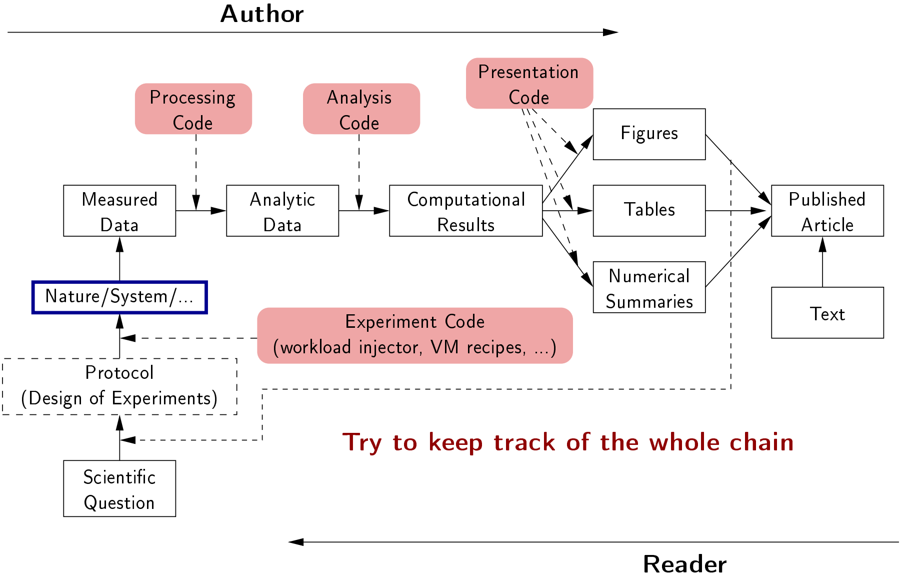
Inspired by Roger D. Peng's lecture on reproducible research, May 2014

Reproducible Research: Trying to Bridge the Gap



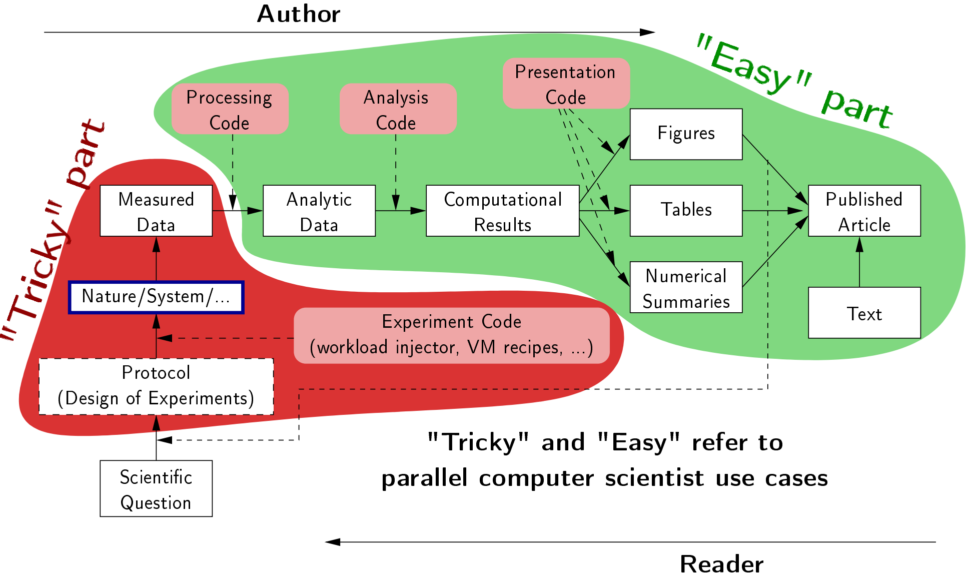
Inspired by Roger D. Peng's lecture on reproducible research, May 2014

Reproducible Research: Trying to Bridge the Gap



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

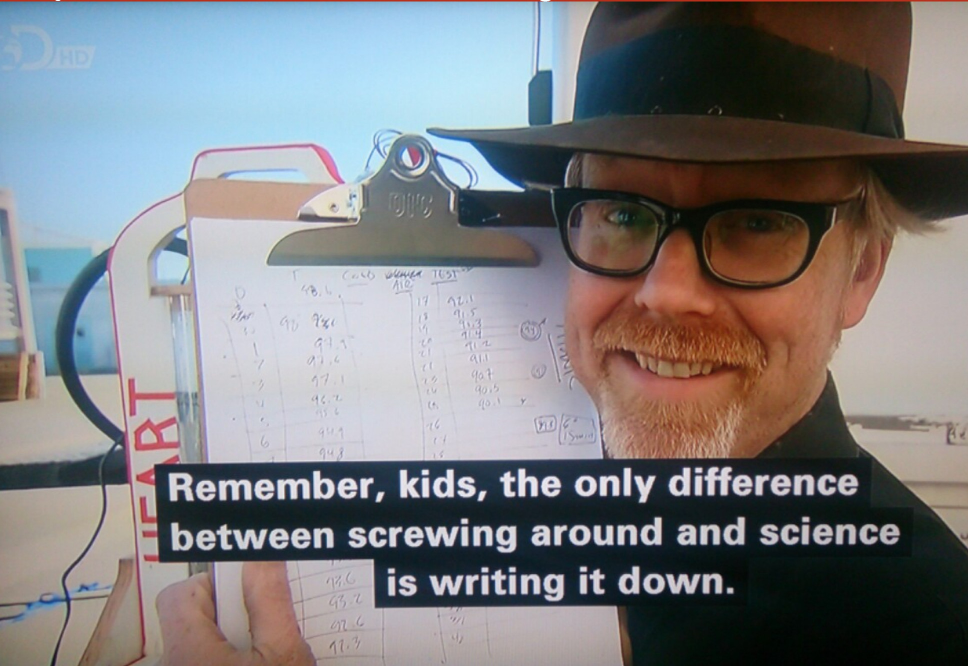
Reproducible Research: Trying to Bridge the Gap



"Tricky" and "Easy" refer to parallel computer scientist use cases

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

Mythbusters: Science vs. Screwing Around



A Difficult Trade-off

Many different tools/approaches developped in various communities

Automatically keeping track of everything

- the code that was run (source code, libraries, compilation procedure)
- processor architecture, OS, machine, date, ...

VM-based solutions and experiment engines

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?

Laboratory notebook and recipes

Outline

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

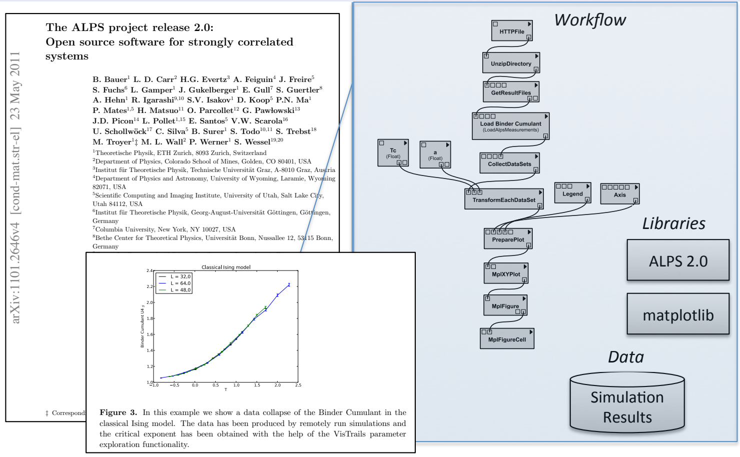
- Tools for *authors* to create reproducible papers
  - Specifications that encode the computational processes
  - Package the results
  - Link from publications

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

‡ Correspond

arXiv:1101.2646v4 [cond-mat.str-el] 23 May 2011

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]

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)

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)

Word-processor plugin App

LaTeX source

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

LaTeX source with VCR package

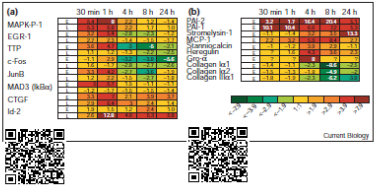
```
\includeresult{vcr.thelancet.com/ffaaffb148d7}
```

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 tran-scripts were hyper-induced in serum-stimulated senescent

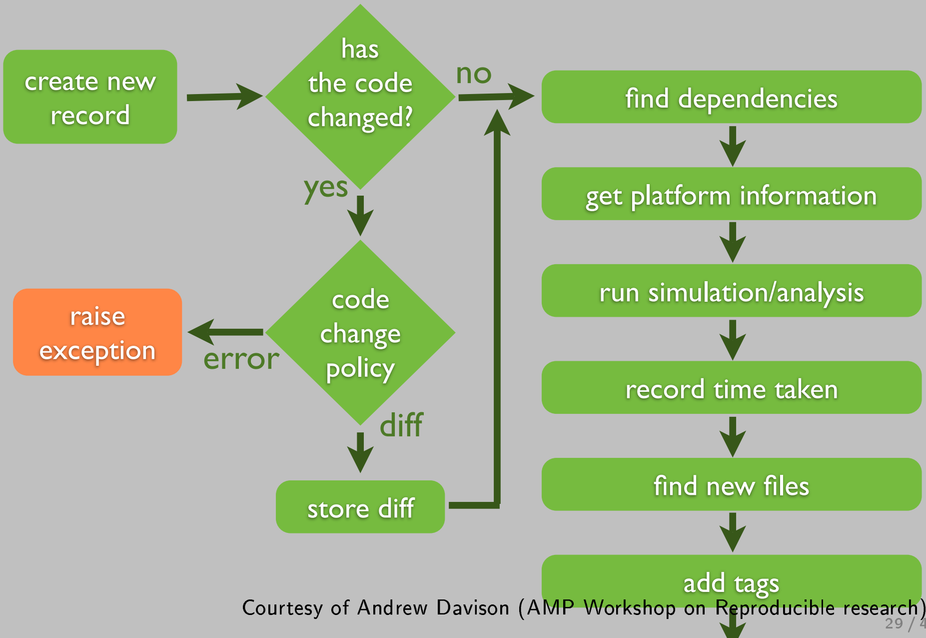
states overlap substantially with those in telomere-induced senescence (W.F., D.N.S., R. Allsopp, S. Lowe, and G. Ferbeyre, unpublished observations) and thus are likely to use many of the same activation processes.

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)

So many new tools

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: **Figshare**, **ActivePapers**, **Elsevier executable paper**, ...

Outline

Structure

Research articles are often structured in this basic order:

Introduction Why was the study undertaken? What was the research question, the tested hypothesis or the purpose of the research?

Methods When, where, and how was the study done? What materials/hardware were used? How was it configured?

Results What answer was found to the research question; what did the study find? Was the tested hypothesis true? Present useful results in a synthetic way with a logical order.

Discussion What might the answer imply and why does it matter? How does it fit in with what other researchers have found? What are the possible bias and points to improve? What are the perspectives for future research?

Such structure facilitates literature review and is a very effective way to convey information.

If the report is a few pages long then an abstract is required.

Outline

Step 0: Taking Notes

Document your:

- Hypotheses: keep track of your ideas/line of thoughts
- Experiments: details on how and why an experiment was run, including failed or ambiguous attempts.
- Initial analysis or interpretation of these experiments: was the outcome conform to the expectation or not? does it (in)validate the hypothesis?
- Organization: keep track of things to do/fix/test/improve

Structure:

1. General information about the document and organization conventions (e.g., directory structure, notebook structure, experimental result storing mechanism, . . .)
2. Documentation of commonly used commands and of how to set up experiments (e.g., git cloning, environment deployment, connection to machines, compiling scripts)
3. Experiment results can be either structured by dates ($\rightsquigarrow$ add tags) or by experiment campaigns ($\rightsquigarrow$ add date/time)

Which format should I use ?

- Wikis are encouraged to favor collaboration but I do not find them really effective
- Blogging systems are also a way of managing such notebook but they should rather be considered as an effective way to share information with others
- I recommend to use basic plain-text format and to structure it hierarchically

Here is a link to one of my PhD student's journal managed with git/org-mode.

Last but not least:

Provide links to Raw Data!!!

When/How Often Should I Use it?

I have a very intense usage (demo to general journal and specific BOINC journal) and I tend to capture a lot of information but you do not have to be as extreme as I am. Here are a few advices:

- 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, write what the next steps are
  - Summarize in a 2-4 lines (for your advisor) what you did, what was difficult, what you learnt.

Step 1: Sharing Code and Data

What kinds of systems are available?

- "Good" - The cloud (Dropbox, Google Drive, Figshare)
- Better - Version control systems (SVN, Git and Mercurial)
- "Best" - Version control systems on the cloud (GitHub, Bitbucket)

Depends on the level of privacy you expect but you probably already know these tools. **Few handle GB files...**

Is this enough?

1. Use a workflow that documents both data and process
2. Use the machine readable CSV format
3. Provide raw data and meta data, not just statistical outputs
4. Never do data manipulation and statistical tests by hand
5. Use R, Python or another free software to read and process raw data (ideally to produce complete reports with code, results and prose)

Courtesy of Adam J. Richards

Step 2: 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 `3l33t` to program this way but that's
exactly what we need for writing a reproducible article/analysis!

Org-mode (requires emacs)

My favorite tool.

- plain text, very smooth, works both for html, pdf, . . .
- allows to combine all my favorite languages even with sessions

Ipython notebook

If you are a python user, go for it! Web app, easy to use/setup. . .

KnitR (a.k.a. Sweave)

For non-emacs users and as a first step toward *reproducible papers*:

- Click and play with a modern IDE (e.g., Rstudio)

Outline

Literate Programming on a Daily Basis

Mastering Emacs

- C-g: get me out of here!
- C-_: undo
- Activate CUA keys in the Options menu

Mastering Org-mode

- Tab will fold/unfold stuff
- C-c C-c: do something (context-sensitive) where you are
- <s + Tab, <b, <l, <r, <h, ... for creating code blocks
- C-c C-e: export
- C-c c: capture content
- C-c C-o / C-c l / C-c C-l: open/store/insert links
- C-c C-a: attach a file
- C-c C-d: set deadline, C-c C-t: TODO/DONE

A Few Links to Learn More

Emacs/Org-mode

- *Org for beginners* (worg)
- *My emacs configuration*
- *For Mac OS X users*

Outline

This was way too much information. . .

. . . but keep these slides in mind and re-read them later. You will follow many links when you will realize what they can bring to you.

- We need to put all this in practice.
- During this semester, you will learn how to improve your methodology
- You will apply analysis and reporting techniques to a simple use case:

  One of your colleague just implemented a multi-threaded version of the quicksort algorithm for multi-core machines. He's convinced his code can save significant time saving but unfortunately, he did not follow the performance evaluation lecture and he would like your help to promote his code.

- After you have tried, we will debrief on what you did and discuss how it could be improved

To do for the Next Time

1. Fork on Github

   https://github.com/alegrand/M2R-ParallelQuicksort
2. Experiment this code on various environments (laptop, G5K, . . .)
3. Take notes on what you did and push back your journal on github
4. Create a synthetic one page IMRAD report