VirginiaTech
Invent the Future

DISCOVERY ANALYTICS CENTER

Software Engineers as Partners in Astronomy Software Development

John Wenskovitch (@wenskovitch)
Discovery Analytics Center
Department of Computer Science
Virginia Tech

“Software is the most prevalent of all the instruments used in modern science.”

C. Goble, "Better Software, Better Research," in *IEEE Internet Computing*, vol. 18, no. 5, pp. 4-8, Sept.-Oct. 2014.

Scientists and Software

- 92% of academics use software [2]
- 91% say using software is important for their research [1]
- 84% say developing software is important for their research [1]
- 69% say that their research would not be practical without software [2]
- 54% claimed to spend more time developing software than they did 10 years ago [1]
- 38% spend at least 1/5 of their time developing software [1]

[1] J. E. Hannay, C. MacLeod, J. Singer, H. P. Langtangen, D. Pfahl and G. Wilson, "How do scientists develop and use scientific software?," *2009 ICSE Workshop on Software Engineering for Computational Science and Engineering*, Vancouver, BC, 2009, pp. 1-8.

[2] E. Hayden. Rule rewrite aims to clean up scientific software. Nature Biotechnology 520, 7547 (2015).

“Research institutions need individuals with a new professional designation—the *research software engineer*.

These individuals combine a **professional attitude to the exercise of software engineering** with a **deep understanding of research topics**.

They lead the ***design and construction of increasingly complex research software systems***, and play an important part in the ***co-design of research requirements***, understanding and addressing ***software engineering questions that arise in research planning***.”

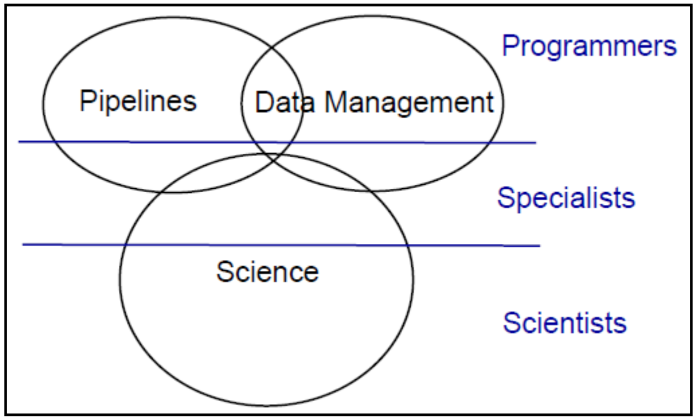
Rob Baxter, Neil Chue Hong, Dirk Gorissen, James Hetherington, and Ilian Todorov, "The Research Software Engineer," in *Digital Research Conference, Oxford*, 2012.

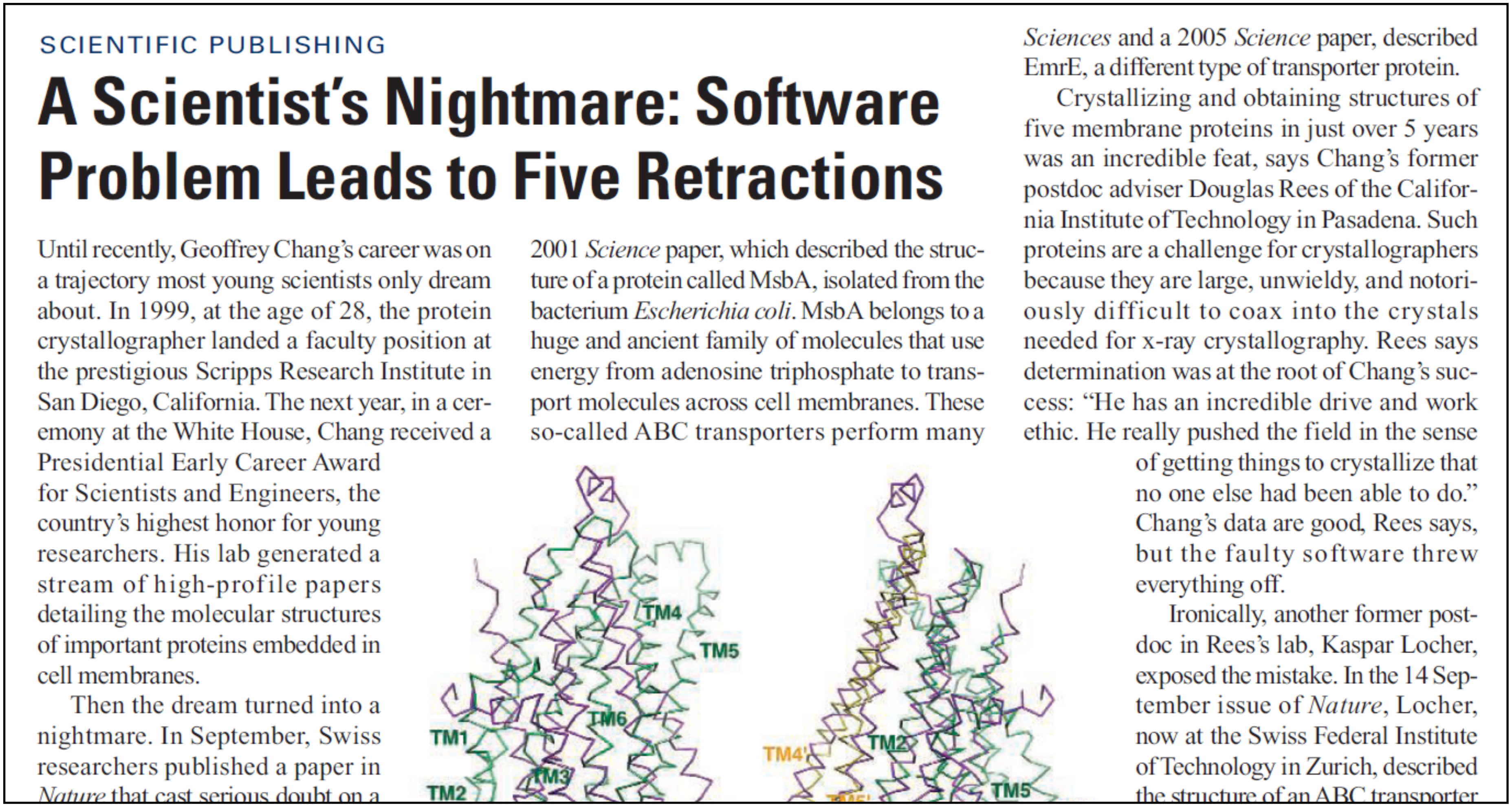
Benefits of a Specialist

- Scientists don’t have to:
  - Focus on data mgmt
  - Manage large databases
  - (most) Debugging

Many Ideas and Image from:
Nilsson, Kim K., and Ole Möller-Nilsson. "Future management needs of a "software-driven” science community." *Software and Cyberinfrastructure for Astronomy*. Vol. 7740. International Society for Optics and Photonics, 2010.

- Specialist should not be
  - IT/helpdesk
  - Downtrodden developer





SCIENTIFIC PUBLISHING

A Scientist's Nightmare: Software Problem Leads to Five Retractions

Until recently, Geoffrey Chang's career was on a trajectory most young scientists only dream about. In 1999, at the age of 28, the protein crystallographer landed a faculty position at the prestigious Scripps Research Institute in San Diego, California. The next year, in a ceremony at the White House, Chang received a Presidential Early Career Award for Scientists and Engineers, the country's highest honor for young researchers. His lab generated a stream of high-profile papers detailing the molecular structures of important proteins embedded in cell membranes.

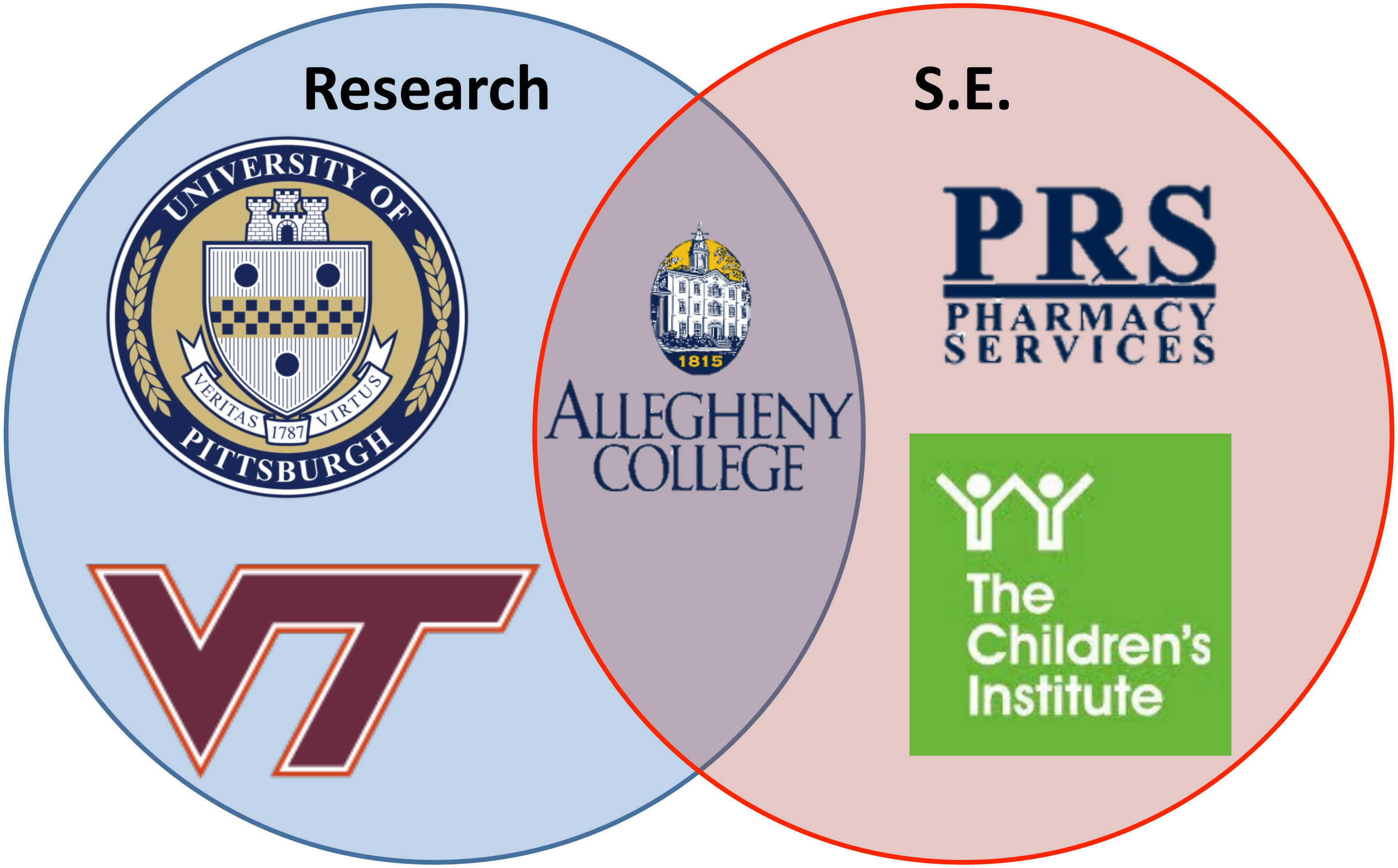
Then the dream turned into a nightmare. In September, Swiss researchers published a paper in *Nature* that cast serious doubt on a 2001 *Science* paper, which described the structure of a protein called MsbA, isolated from the bacterium *Escherichia coli*. MsbA belongs to a huge and ancient family of molecules that use energy from adenosine triphosphate to transport molecules across cell membranes. These so-called ABC transporters perform many

Sciences and a 2005 *Science* paper, described EmrE, a different type of transporter protein.

Crystallizing and obtaining structures of five membrane proteins in just over 5 years was an incredible feat, says Chang's former postdoc adviser Douglas Rees of the California Institute of Technology in Pasadena. Such proteins are a challenge for crystallographers because they are large, unwieldy, and notoriously difficult to coax into the crystals needed for x-ray crystallography. Rees says determination was at the root of Chang's success: "He has an incredible drive and work ethic. He really pushed the field in the sense of getting things to crystallize that no one else had been able to do." Chang's data are good, Rees says, but the faulty software threw everything off.

Ironically, another former postdoc in Rees's lab, Kaspar Locher, exposed the mistake. In the 14 September issue of *Nature*, Locher, now at the Swiss Federal Institute of Technology in Zurich, described the structure of an ABC transporter

Research + Software Engineer



Research

S.E.

Disclaimers

- I am not a professional software developer
- My experience with astronomy research software development is $n = 1$
- My interview with a “senior research scientist” is also $n = 1$
- We don’t speak for all computer scientists

Keep the Engineer Involved

- Nobody (probably) gets into CS to only write code
  - Keep the programmer involved in scientific decisions; they may have some surprising insights
    - Give plenty of feedback

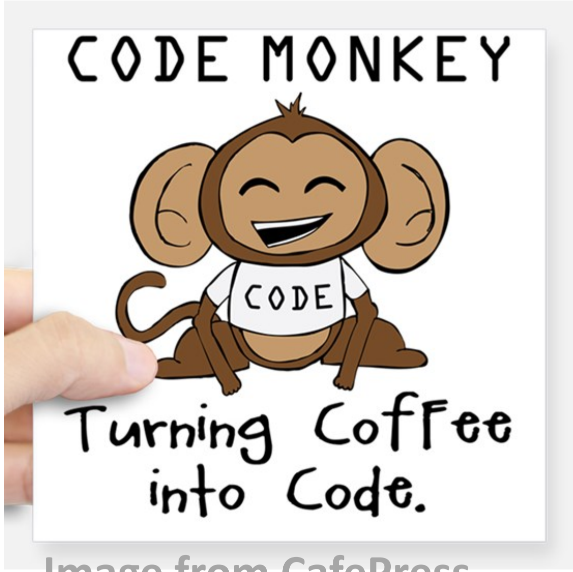
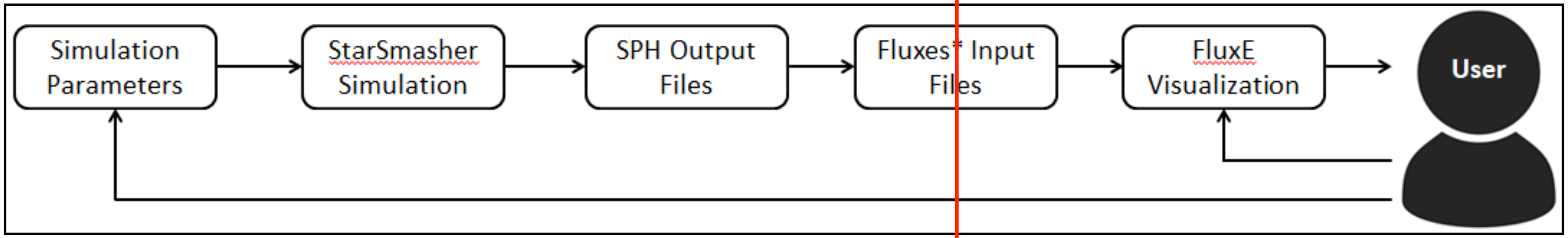


Image from CafePress
https://www.cafepress.com/+code-monkey+bumper-stickers

> He/she must not be seen as a "second grade" scientist, and should be encouraged to take active part in the scientific life of the institute.
>
> Nilsson, Kim K., and Ole Möller-Nilsson. "Future management needs of a "software-driven" science community." *Software and Cyberinfrastructure for Astronomy*. Vol. 7740. International Society for Optics and Photonics, 2010.

Acknowledgment of Strengths

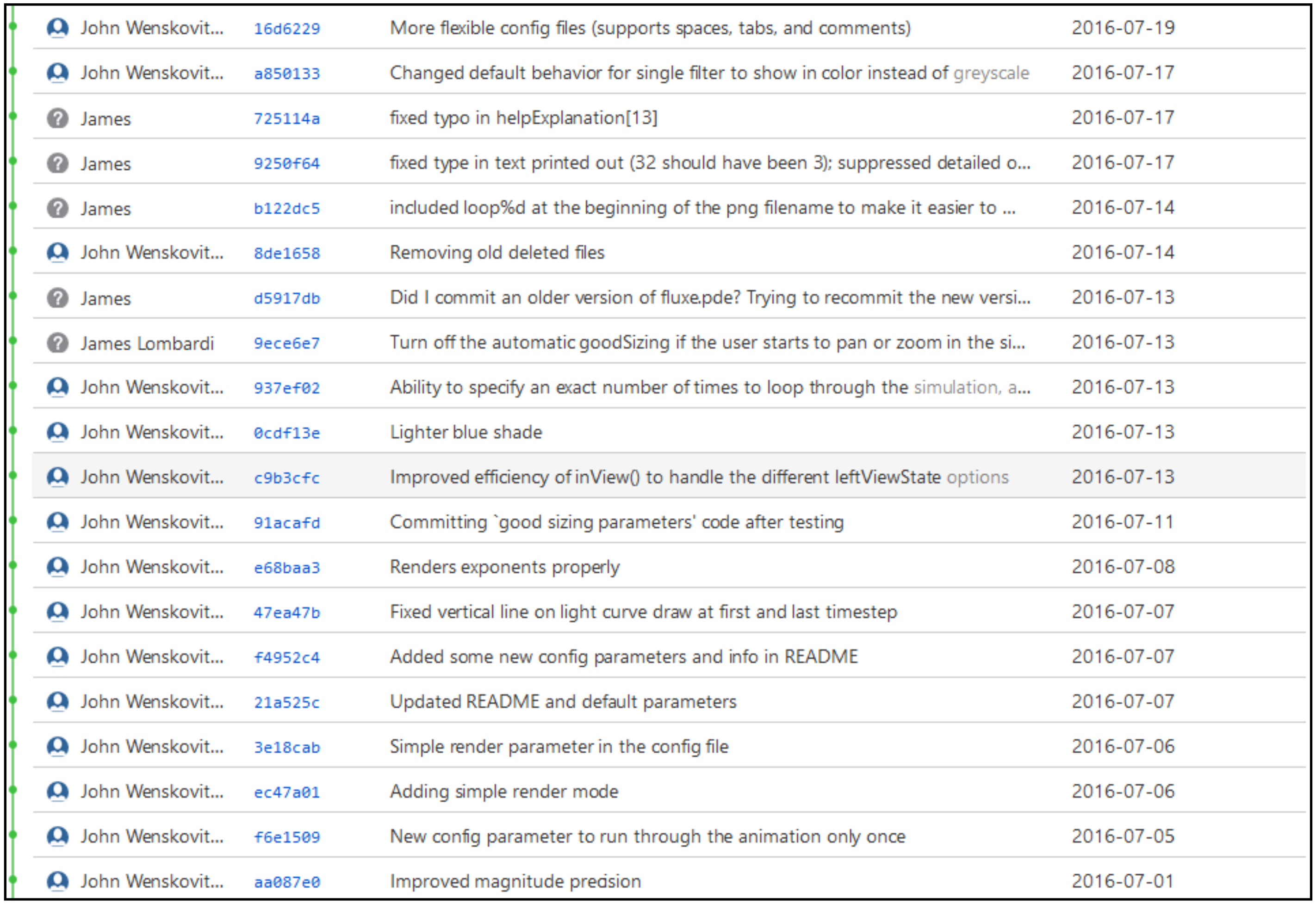


Version Control – Use It

THIS IS GIT. IT TRACKS COLLABORATIVE WORK ON PROJECTS THROUGH A BEAUTIFUL DISTRIBUTED GRAPH THEORY TREE MODEL.
COOL. HOW DO WE USE IT?
NO IDEA. JUST MEMORIZE THESE SHELL COMMANDS AND TYPE THEM TO SYNC UP. IF YOU GET ERRORS, SAVE YOUR WORK ELSEWHERE, DELETE THE PROJECT, AND DOWNLOAD A FRESH COPY.

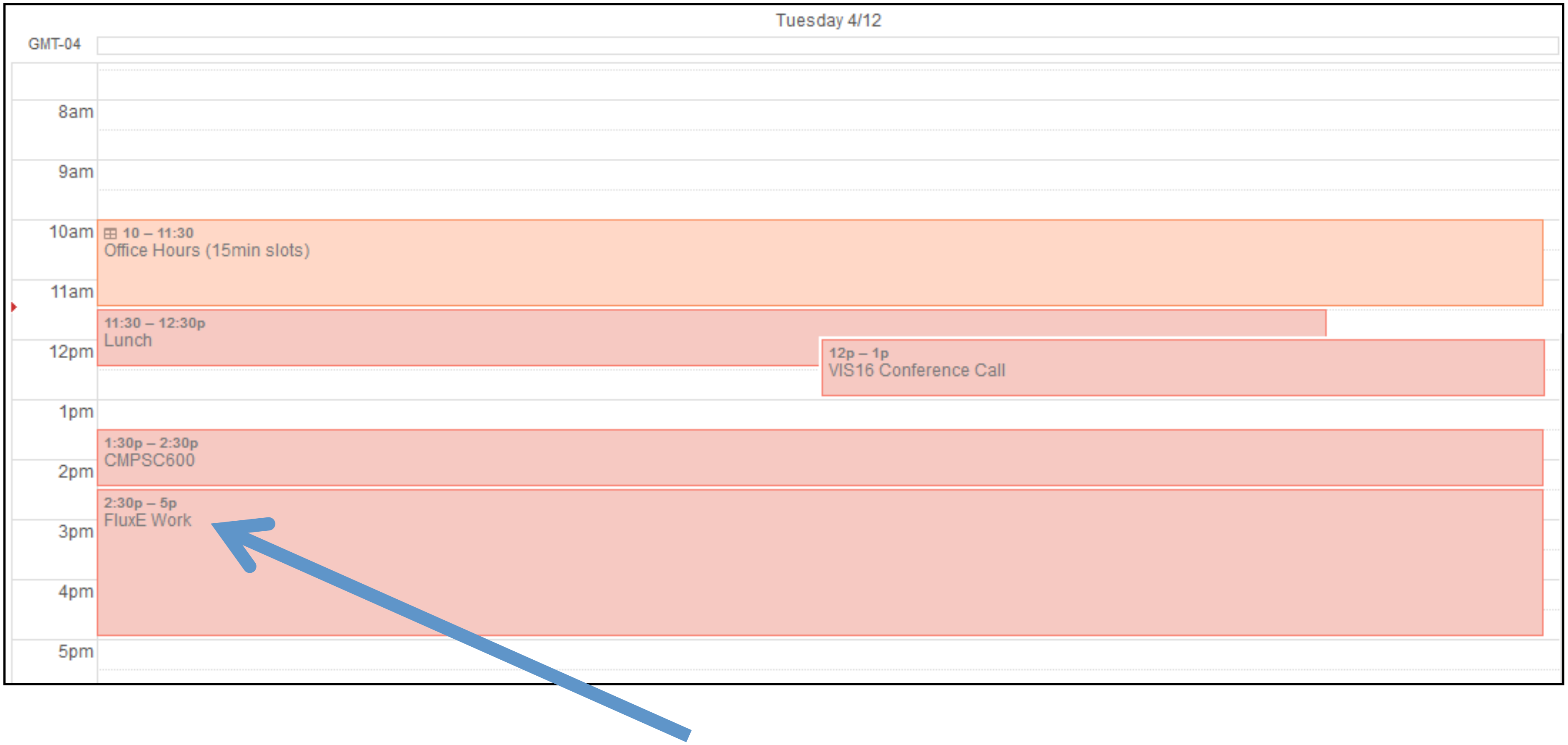
https://xkcd.com/1597/

Version Control – Commit Often



John Wenskovit...	16d6229	More flexible config files (supports spaces, tabs, and comments)	2016-07-19
John Wenskovit...	a850133	Changed default behavior for single filter to show in color instead of greyscale	2016-07-17
James	725114a	fixed typo in helpExplanation[13]	2016-07-17
James	9250f64	fixed type in text printed out (32 should have been 3); suppressed detailed o...	2016-07-17
James	b122dc5	included loop%d at the beginning of the png filename to make it easier to ...	2016-07-14
John Wenskovit...	8de1658	Removing old deleted files	2016-07-14
James	d5917db	Did I commit an older version of fluxe.pde? Trying to recommit the new versi...	2016-07-13
James Lombardi	9ece6e7	Turn off the automatic goodSizing if the user starts to pan or zoom in the si...	2016-07-13
John Wenskovit...	937ef02	Ability to specify an exact number of times to loop through the simulation, a...	2016-07-13
John Wenskovit...	0cdf13e	Lighter blue shade	2016-07-13
John Wenskovit...	c9b3cfc	Improved efficiency of inView() to handle the different leftViewState options	2016-07-13
John Wenskovit...	91acafd	Committing `good sizing parameters' code after testing	2016-07-11
John Wenskovit...	e68baa3	Renders exponents properly	2016-07-08
John Wenskovit...	47ea47b	Fixed vertical line on light curve draw at first and last timestep	2016-07-07
John Wenskovit...	f4952c4	Added some new config parameters and info in README	2016-07-07
John Wenskovit...	21a525c	Updated README and default parameters	2016-07-07
John Wenskovit...	3e18cab	Simple render parameter in the config file	2016-07-06
John Wenskovit...	ec47a01	Adding simple render mode	2016-07-06
John Wenskovit...	f6e1509	New config parameter to run through the animation only once	2016-07-05
John Wenskovit...	aa087e0	Improved magnitude precision	2016-07-01

Frequent Communication



10 – 11:30
Office Hours (15min slots)

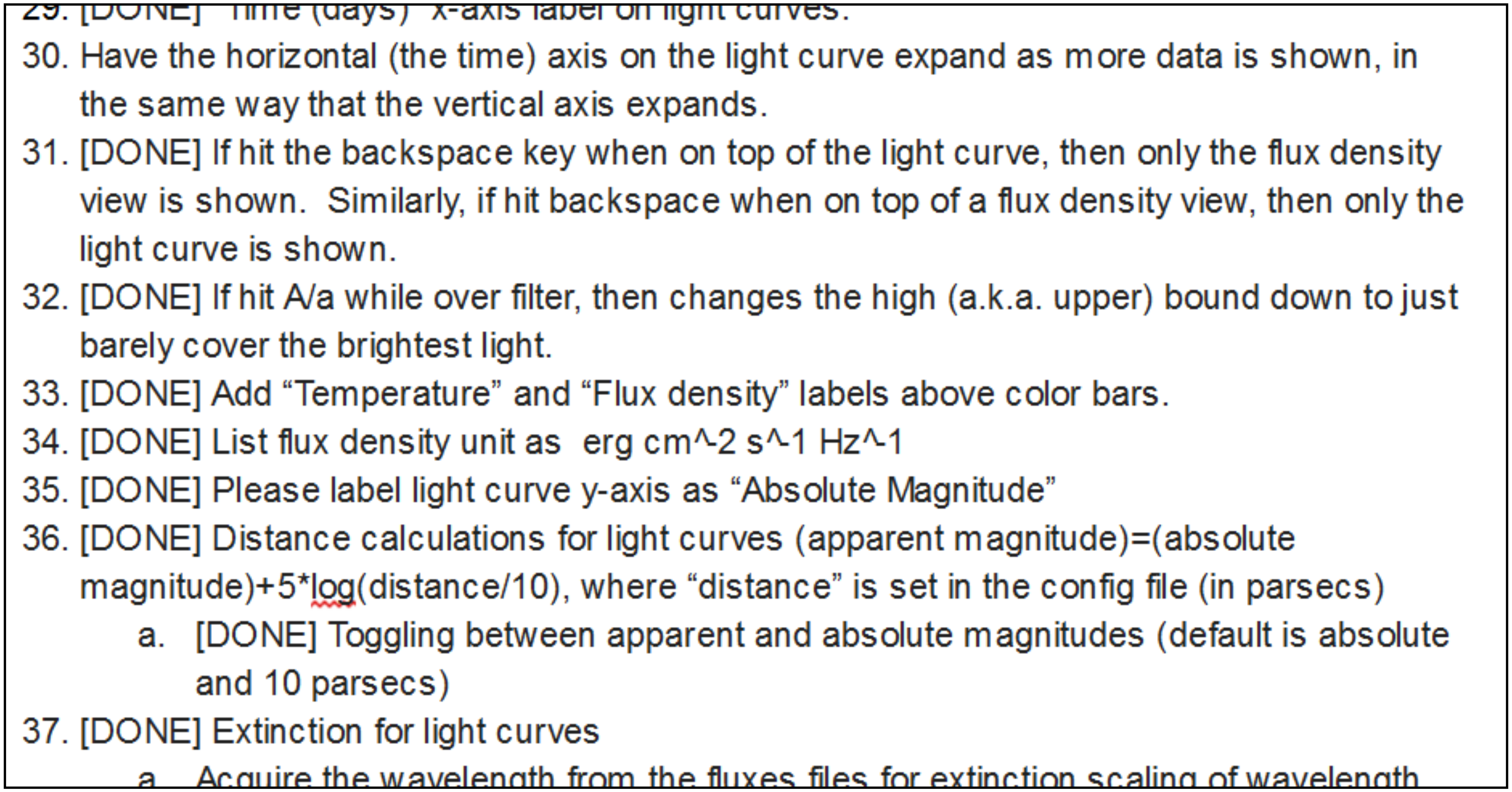
11:30 – 12:30p
Lunch

12p – 1p
VIS16 Conference Call

1:30p – 2:30p
CMPSC600

2:30p – 5p
FluxE Work

Prioritized Feature List



29. [DONE] “Time (days)” x-axis label on light curves.
30. Have the horizontal (the time) axis on the light curve expand as more data is shown, in the same way that the vertical axis expands.
31. [DONE] If hit the backspace key when on top of the light curve, then only the flux density view is shown. Similarly, if hit backspace when on top of a flux density view, then only the light curve is shown.
32. [DONE] If hit A/a while over filter, then changes the high (a.k.a. upper) bound down to just barely cover the brightest light.
33. [DONE] Add “Temperature” and “Flux density” labels above color bars.
34. [DONE] List flux density unit as erg cm^-2 s^-1 Hz^-1
35. [DONE] Please label light curve y-axis as “Absolute Magnitude”
36. [DONE] Distance calculations for light curves (apparent magnitude)=(absolute magnitude)+5*log(distance/10), where “distance” is set in the config file (in parsecs)
    a. [DONE] Toggling between apparent and absolute magnitudes (default is absolute and 10 parsecs)
37. [DONE] Extinction for light curves
    a. Acquire the wavelength from the fluxes files for extinction scaling of wavelength

Test Your Code Thoroughly

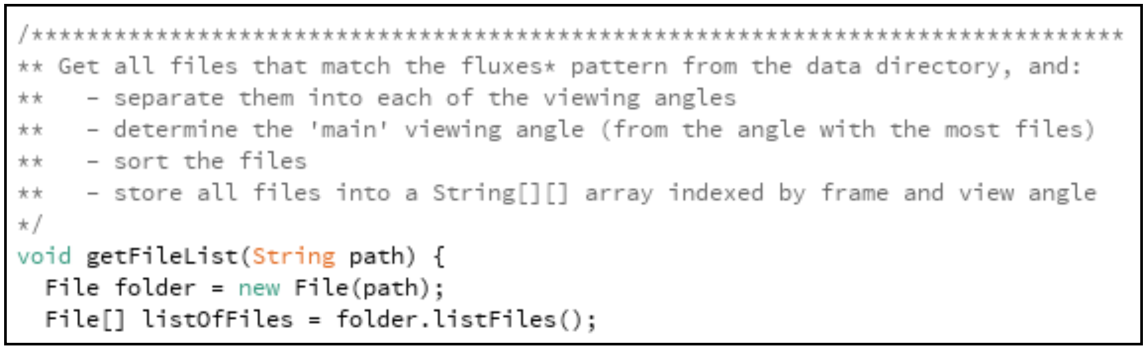
- **Unit test** – Does the functionality work as expected?
- **Regression test** – Did this new code break anything old?
- **Usability test** – Is the interface intuitive and easy to use?

Tests should be short:

Verify ONE thing

Image from ToolSQA
http://toolsqa.com/software-testing/software-testing-tutorial/

Document All the Things



```
/*******************************************************************************
** Get all files that match the fluxes* pattern from the data directory, and:
**   - separate them into each of the viewing angles
**   - determine the 'main' viewing angle (from the angle with the most files)
**   - sort the files
**   - store all files into a String[][] array indexed by frame and view angle
*/
void getFileList(String path) {
  File folder = new File(path);
  File[] listOfFiles = folder.listFiles();
```

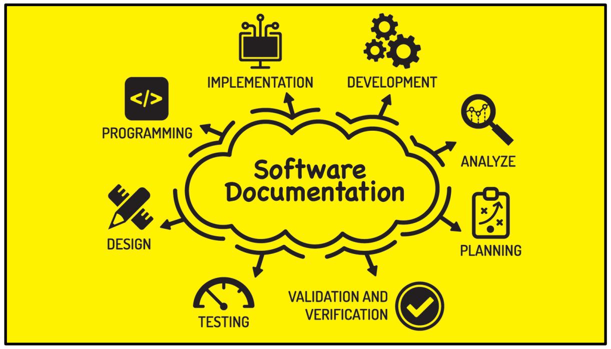
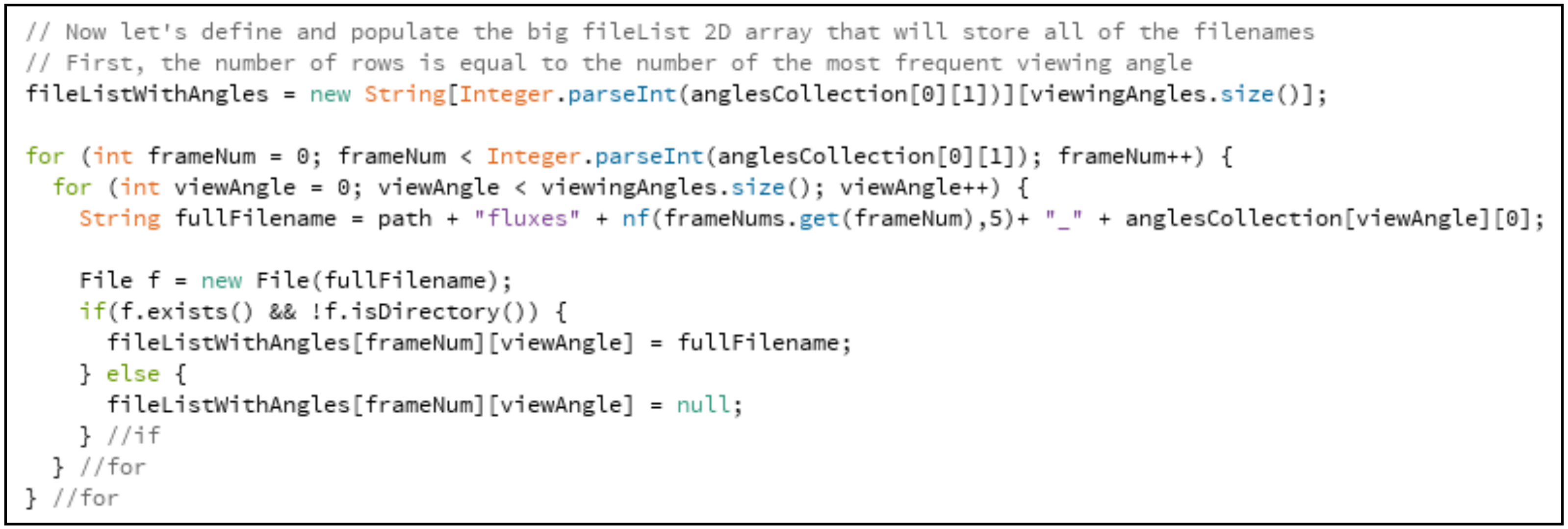


Image from: Haider, Zeeshan. "Role of Documentation and Software Architecture in Cubix." https://www.cubix.co/blog/documentation-software-architecture/



```
// Now let's define and populate the big fileList 2D array that will store all of the filenames
// First, the number of rows is equal to the number of the most frequent viewing angle
fileListWithAngles = new String[Integer.parseInt(anglesCollection[0][1])][viewingAngles.size()];

for (int frameNum = 0; frameNum < Integer.parseInt(anglesCollection[0][1]); frameNum++) {
  for (int viewAngle = 0; viewAngle < viewingAngles.size(); viewAngle++) {
    String fullFilename = path + "fluxes" + nf(frameNums.get(frameNum),5)+ "_" + anglesCollection[viewAngle][0];

    File f = new File(fullFilename);
    if(f.exists() && !f.isDirectory()) {
      fileListWithAngles[frameNum][viewAngle] = fullFilename;
    } else {
      fileListWithAngles[frameNum][viewAngle] = null;
    } //if
  } //for
} //for
```

In summary:

- Scientists use software **a lot**
- Research Software Engineers can make life easier
- If you're working collaboratively, keep in mind:

1. Keep the Engineer Involved
2. Acknowledgment of Strengths
3. Version Control
4. Frequent Communication
5. Prioritized Feature List
6. Test Your Code Thoroughly
7. Document All the Things