

# Software Communities Productivity in Extreme Science

Andy R. Terrel  
Chief Computational Scientist  
Continuum Analytics, Inc

SWP4XS  
Jan 2014



# People Involved



Andy Terrel  
Continuum Analytics  
Computational Scientist



Aron Ahmadia  
US Army ERDC  
Computational Scientist



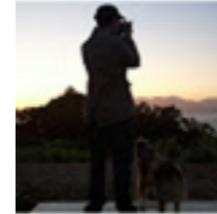
Chris Kees  
US Army ERDC  
Computational Geoscientist



Ondřej Čertík  
LANL  
Chemical Physicist



Dag Sverre Seljebotn  
mCASH Norge AS  
Computational Astrophysicist

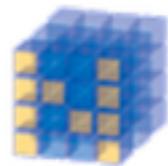


Matthew Turk  
Columbia  
Astrophysicist

# My Perspective



FENICS  
PROJECT



NumPy  
Base N-dimensional array  
package



SciPy library  
Fundamental library for  
scientific computing



Matplotlib  
Comprehensive 2D  
Plotting

IP[y]:  
IPython

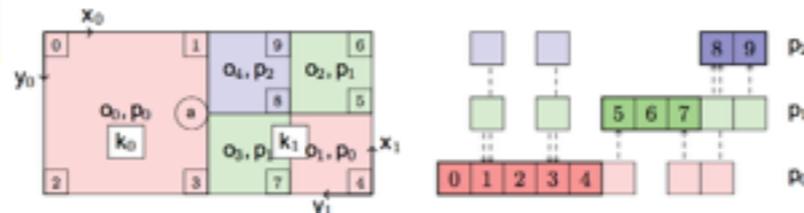
IPython  
Enhanced Interactive  
Console



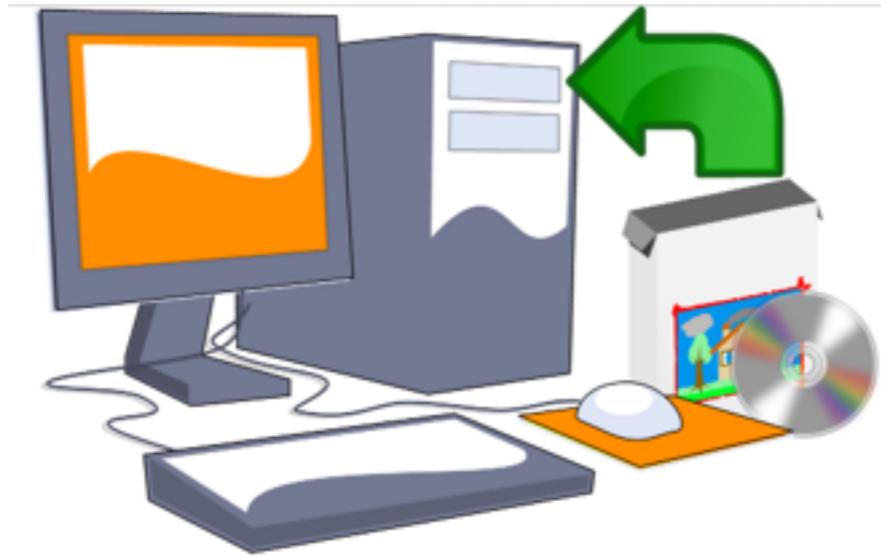
Sympy  
Symbolic mathematics



pandas  
Data structures & analysis



# Two persistent problems



Building scientific software stacks

Developing communities of practice to share tools and experience



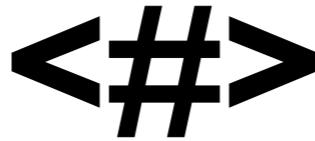
# Building Stacks

- Installing code has become a very expensive task
- Many different stacks have different needs and require customization.
- Exotic hardware is making it worse
- How much does it cost?
  - Typical HPC Center\* employs ~ dozen of people to install and manage the stack
  - If half their time is installing looking at  $\$.5 \text{ MM} * \#\text{Centers}$
  - Spawned a new field: DevOp Engineer
- Problem even in industry, especially the data science world

# Many Solutions

- Environment modules are very fragile
- Virtualization has a high amount of performance cost and is not composable
- Grids and portals limit the available resources requiring many man-months of work to add new software
- Typical package managers require root access which is not happening on production systems

# Better Solutions



HashDist

Anaconda



Better user space  
build managers

Better containers and  
portals that emulate the  
desktop



SageMathCloud™



VAGRANT

# Practice

- Current practices divide producers and consumers of scientific codes
  - Application codes
  - Libraries
- Creates a separation from science and coders that use in HPC environment.
- Productivity comes from a community helping on another.

## A Survey of the Practice of Computational Science

Prakash Prabhu   Thomas B. Jablin   Arun Raman   Yun Zhang   Jialu Huang  
Hanjun Kim   Nick P. Johnson   Feng Liu   Soumyadeep Ghosh   Stephen Beard  
Taewook Oh   Matthew Zoufaly   David Walker   David I. August

Princeton University  
{pprabhu,tjablin,arun,yunzhang,jialuh}@princeton.edu  
{hanjunk,npjohnso,fengliu,soumyade,abeard}@princeton.edu  
{twoh,mzoufaly,dpw,august}@princeton.edu

### ABSTRACT

Computing plays an indispensable role in scientific research. Presently, researchers in science have different problems,

even for computer scientists [3]. Given this background, this paper seeks to answer the question: How are scientists coping with the growing computing demands?

# Others talking about Practice

in 2013

## How to Scale a Code in the Human Dimension

Matthew J. Turk ([matthewturk@gmail.com](mailto:matthewturk@gmail.com))  
Columbia University

---

*This is a re-telling of a talk given at Scientific Software Days in December, 2012, at the Texas Advanced Computing Center. The slides and video of this talk can be found at <http://actsoftdays.org/meetings/2012/>.*



**Steve Yegge**  
Shared publicly · Oct 21, 2011

Last week I accidentally posted an internal rant about service platforms to my public Google+ account (i.e. this one). It somehow went viral, which is nothing short of stupefying given that it was a massive Wall of Text. The whole thing still feels surreal.

Amazingly, nothing bad happened to me at Google. Everyone just laughed at me a lot, all the way up to the top, for having committed what must be the great-granddaddy of all Reply-All screwups in tech history.

But they also listened, which is super cool. I probably shouldn't talk much about it, but there's already floating out how to deal with some of the issues I raised. I want

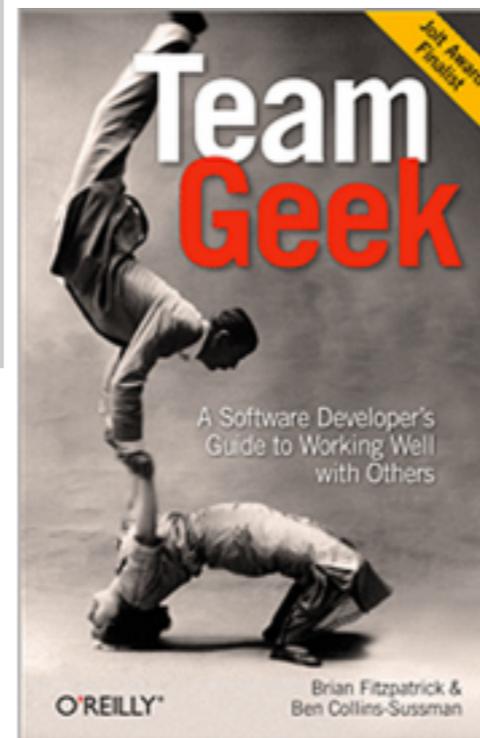
Steve Yegge  
37,047 people have them in circles

## COMPUTATIONAL SCIENCE & DISCOVERY

### What makes computational open source software libraries successful?

Wolfgang Bangerth<sup>1</sup> and Timo Heister<sup>2</sup>

<sup>1</sup> Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, USA  
<sup>2</sup> Mathematical Sciences, O-110 Martin Hall, Clemson University, Clemson, SC 29634-0975, USA  
E-mail: [bangerth@math.tamu.edu](mailto:bangerth@math.tamu.edu) and [heister@clemson.edu](mailto:heister@clemson.edu)



Why Coverity Products Services Customers Partners Company

## Coverity Finds Python Sets New Level of Quality for Open Source Software

Coverity, Inc., the leader in **development testing**, today announced the results of its second Coverity Scan™ Project Spotlight, which analyzed the Python open source software project, including defect density as compared to the industry average defect density for good quality software and types of defects identified.

# Conclusions

- Build Communities of Practice
  - Open communities provide continuous feedback
  - Usability requires dog-fooding
  - Increase the number of practitioners through broader communication / shared experience
- Build Better Software Stacks
  - Invest in Smarter Build Managers
  - Emulate the install process to the user desktop
    - Even custom builds for machine
  - Specialized Hardware needs to be easy to integrate
  - Test installs