TRAINING & BEST PRACTICES TO DEVELOP PORTABLE YET PERFORMANT CODE

Sunita Chandrasekaran
Assistant Professor, University of Delaware
Dept. of Computer & Information Sciences
schandra@udel.edu

SIAM PP MS42: Feb 14, 2020

CREATE BETTER SOFTWARE!

- Programmer Productivity
- Software Sustainability
- Facilitate Reproducibility of Computational Results
- Incremental Code Improvement
- Re-usability of code
- Reduced development effort

image credit: https://www.pnas.org/content/115/20/5042

BEST PRACTICES - 7 OF MANY!

- Best Practice #1 Profiling
- Best Practice #2: Systematic Testing
- Best Practice #3: Report bugs
- Best Practice #4: Automate
- Best Practice #5: Document
- Best Practice #6: Pair Programming
- Best Practice #7: Open Source but…

My group in action :-)

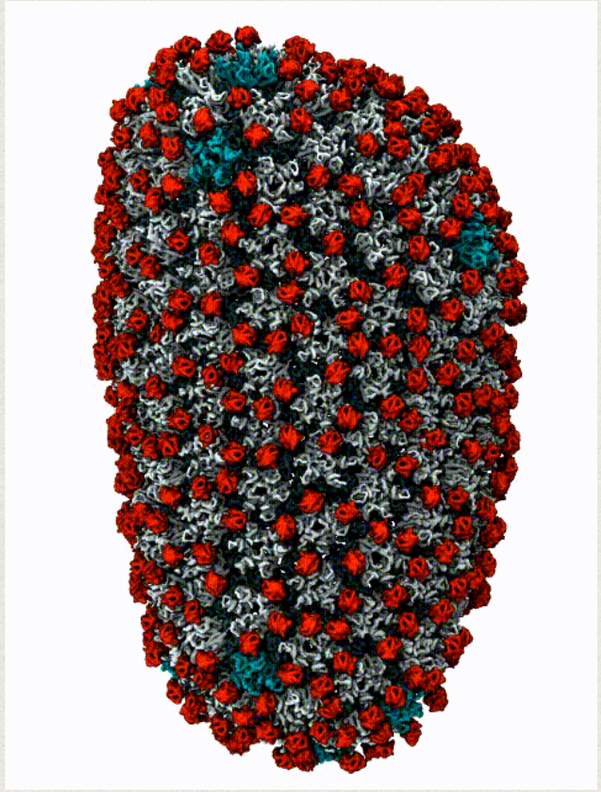
CRPL
Sunita Chandrasekaran

Computational Research and Programming Lab

BEST PRACTICE #1:
PROFILING

ACCELERATING A BIOPHYSICS PROBLEM ON GPUS

- Nuclear Magnetic Resonance (NMR) is a vital tool in structural biology and biochemistry
- NMR spectroscopy measures chemical shifts
- Predicting chemical shift has important uses in scientific areas such as drug discovery

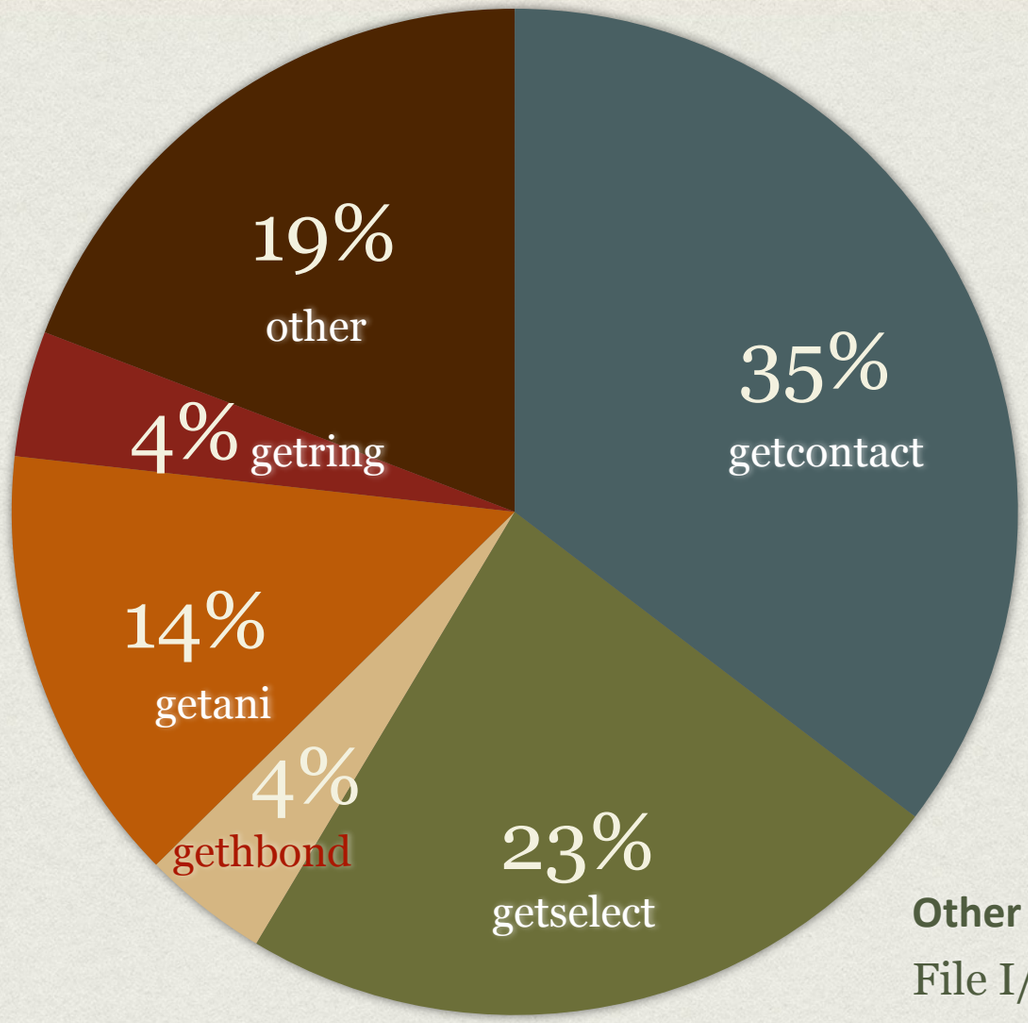


Our goal:

- Accelerate the prediction of chemical shift
  - To enable execution of multiple chemical shift predictions repeatedly
  - To allow chemical shift predictions for larger scale structures

SERIAL PROFILE VISUAL

- Serial code profiling NVProf
- Obtained large overview without needing to read thousands of lines of code
- Identified hotspots within the code
- 2 undergrads - 1 year project



Other Contains:

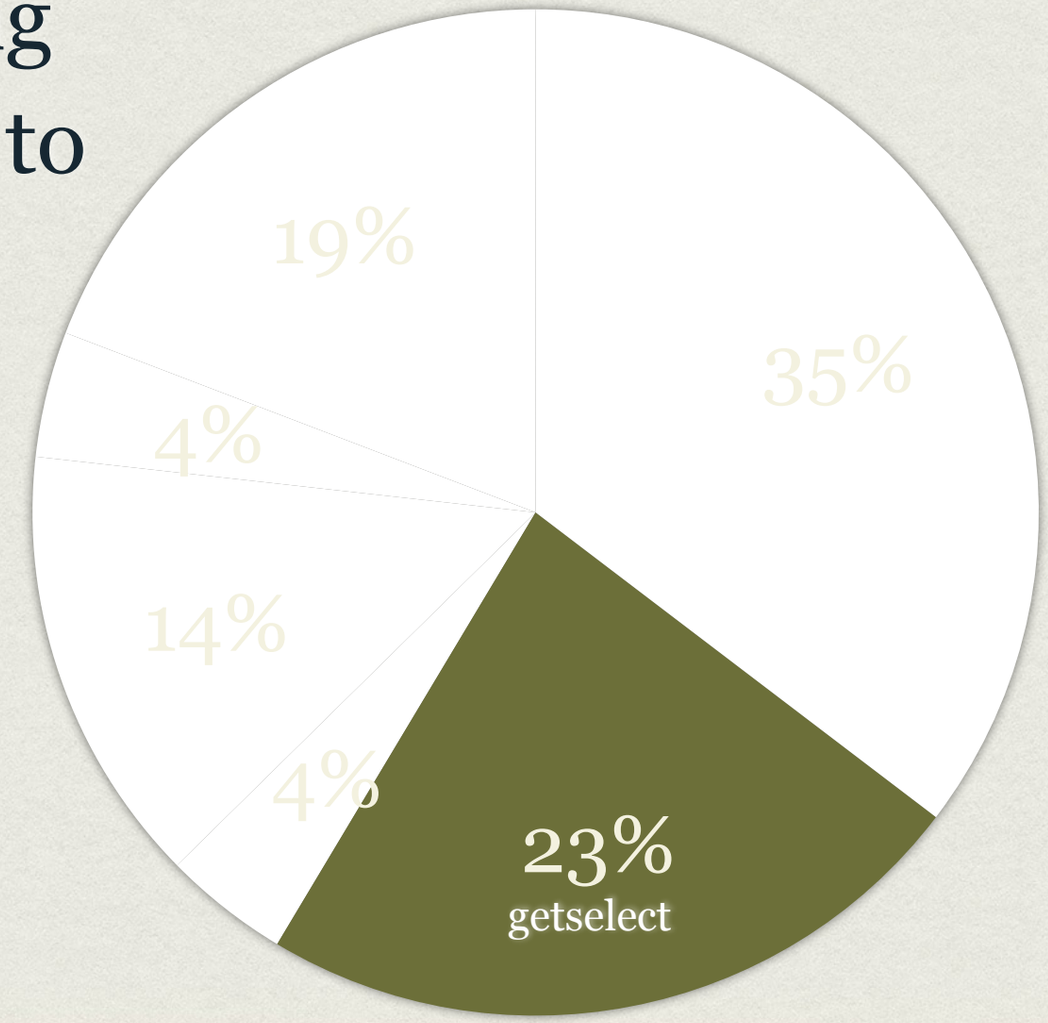
File I/O

PDB Structure

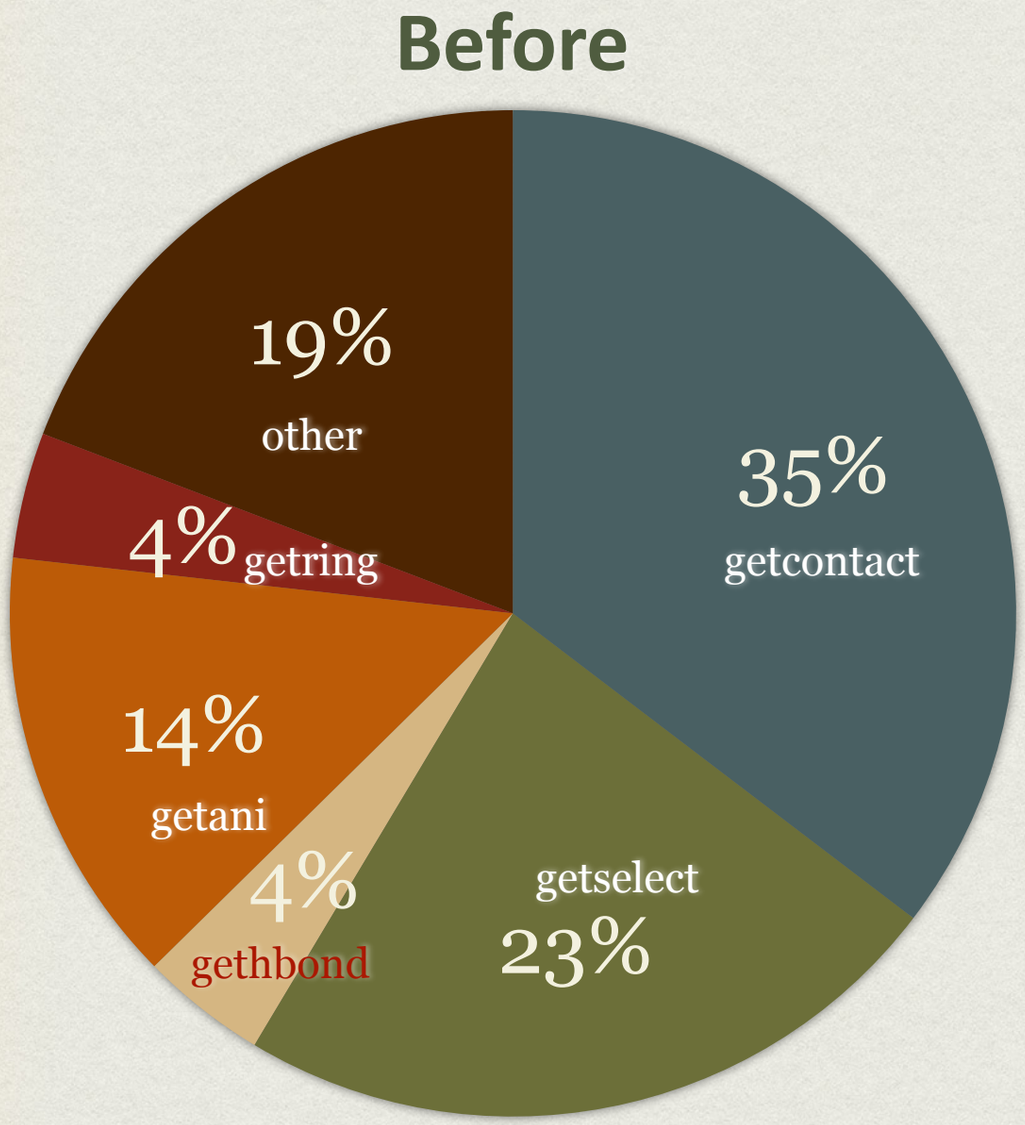
Initialization

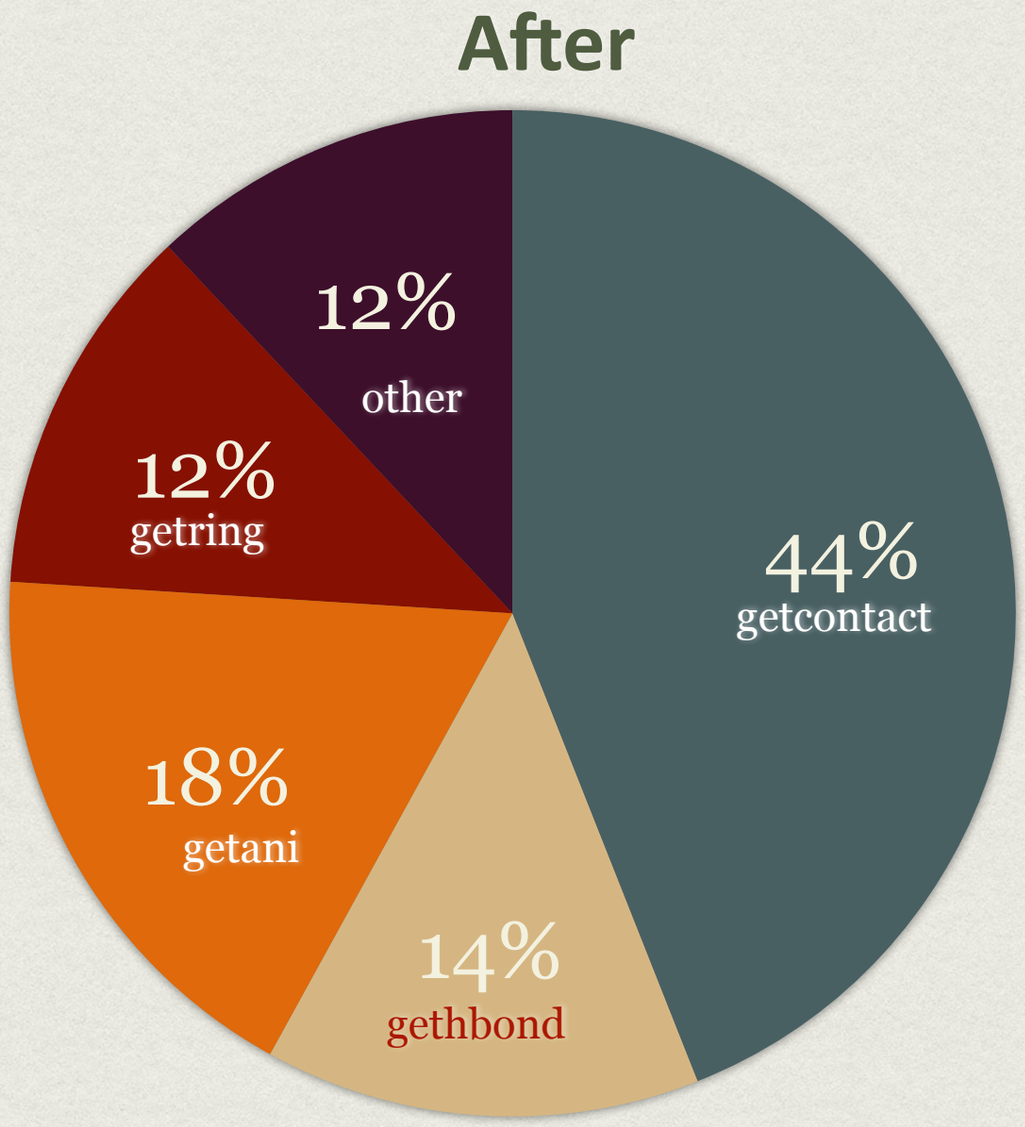
SERIAL CODE CLEAN UP!!

- getselect()
- Looking into cleaning the serial code prior to parallelizing it



PROFILER MAGIC!





CRPL
Sunita Chandrasekaran

UNIVERSITY OF DELAWARE

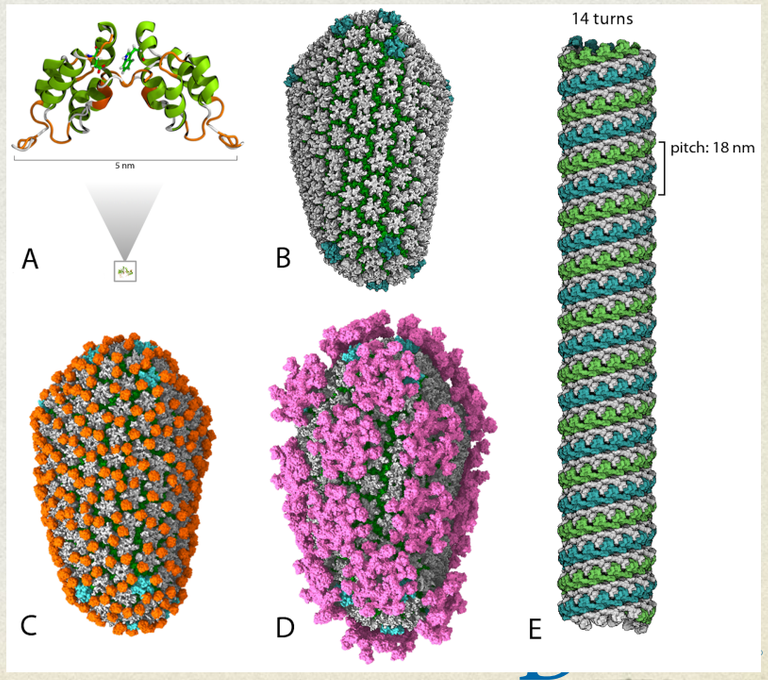
OPENACC-GPU PERFORMANCE RESULTS

	100K	1.5M	5M atoms	6.8M	11.3M
Serial (Unoptimized)	167.11s	572.01s	3547.07s	7 hours	14 hours (estimate)
Serial (Optimized)	53.57s	196.12s	2003.6s	1510.71s	2614.4s
Multicore	4.67s	32.82s	116.66s	153.8s	146.06s
NVIDIA P40	3.47s	17.15s	56.2s	78.57s	72.55s
NVIDIA V100	3.11s	13.62s	39.79s	49.63s	47.71s

https://github.com/UD-CRPL/ppm_one

Biorxiv: https://www.biorxiv.org/content/10.1101/2020.01.12.903468v1.full.pdf

[Under Review]



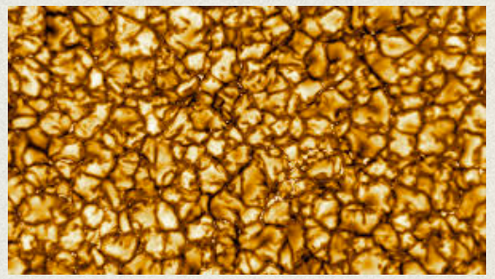
CRPL
Sunita Chandrasekaran

BEST PRACTICE #2:
SYSTEMATIC TESTING

PREPARE MURAM (MAX PLANCK UNIVERSITY OF CHICAGO RADIATIVE MHD) FOR NEXT-GENERATION SYSTEMS

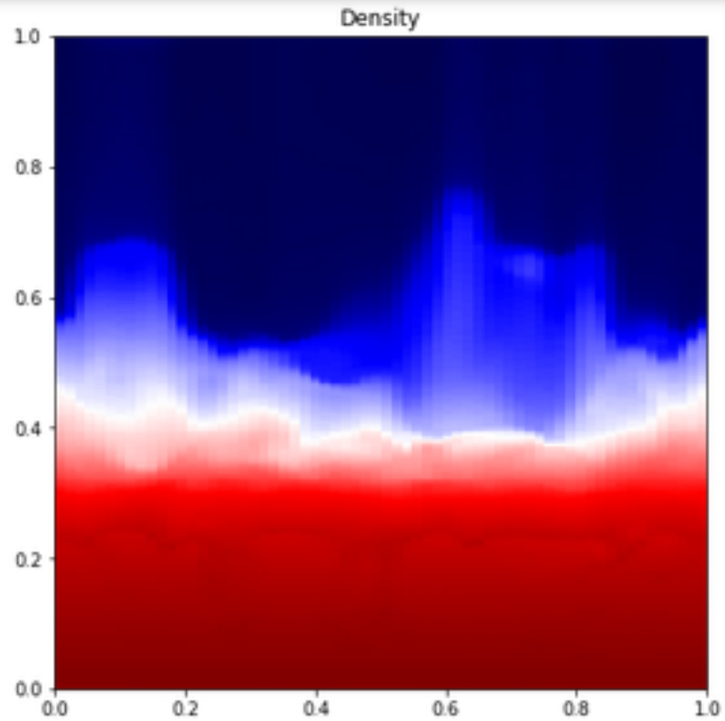
- Primary solar model for simulations of upper convection zone, photosphere, and corona
- Typical runs across ~10,000 CPU cores, runs ~100x slower than real-time
- New physics will also require more computation power
- Goals: Port and Accelerate MURaM to GPU, achieve real-time simulation, prepare for new physics

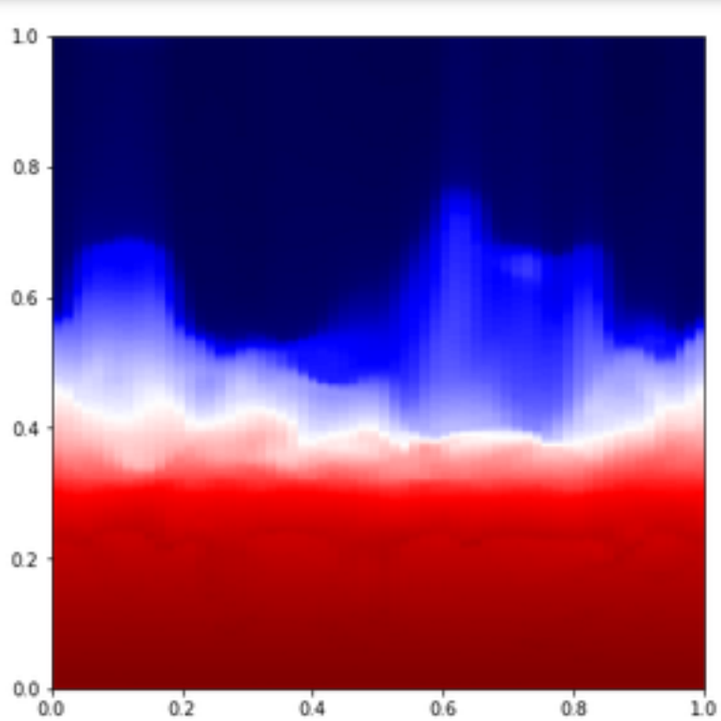
Comprehensive model of entire life cycle of a solar flare (Cheung et al 2018)

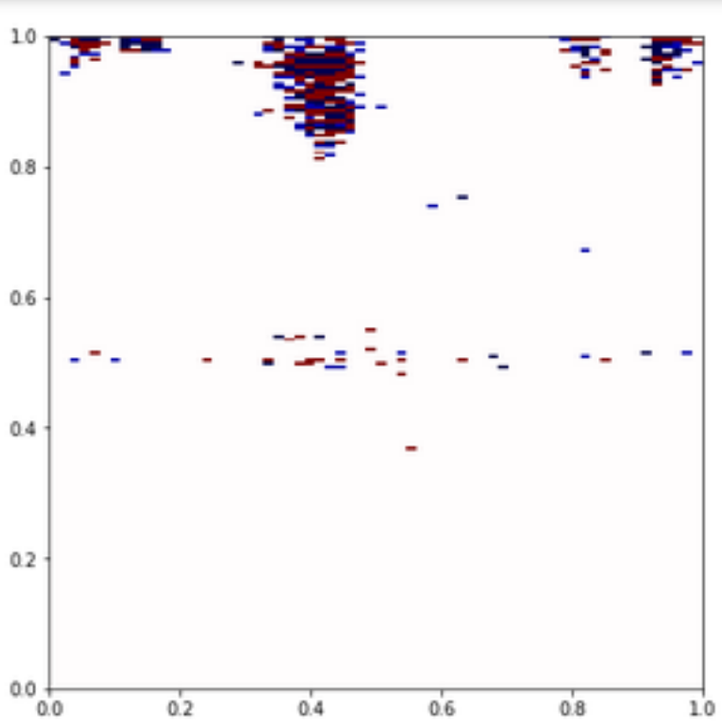


The Daniel K. Inouye Solar Telescope (DKIST), a ~$300M NSF investment, is expected to advance the resolution of ground based observational solar physics by an order of magnitude.

ERROR IN DENSITY CALCULATION







Difference in density between reference and the test runs

CRPL
Sunita Chandrasekaran

UNIVERSITY OF DELAWARE

UNRESOLVABLE BUG

- **LOC:** 20-30K Lines of Code
- **Team:** 4 Computer Scientists and 2 Solar physicists
- **Tools used:** Valgrind, GDB, Python notebooks, PCAST (PGI tool)
- **Test bed:** 3 different compilers and 2 different platforms
- **Version Control:** GitHub
- **Time spent:** 3 months and the clock is still ticking!!!!

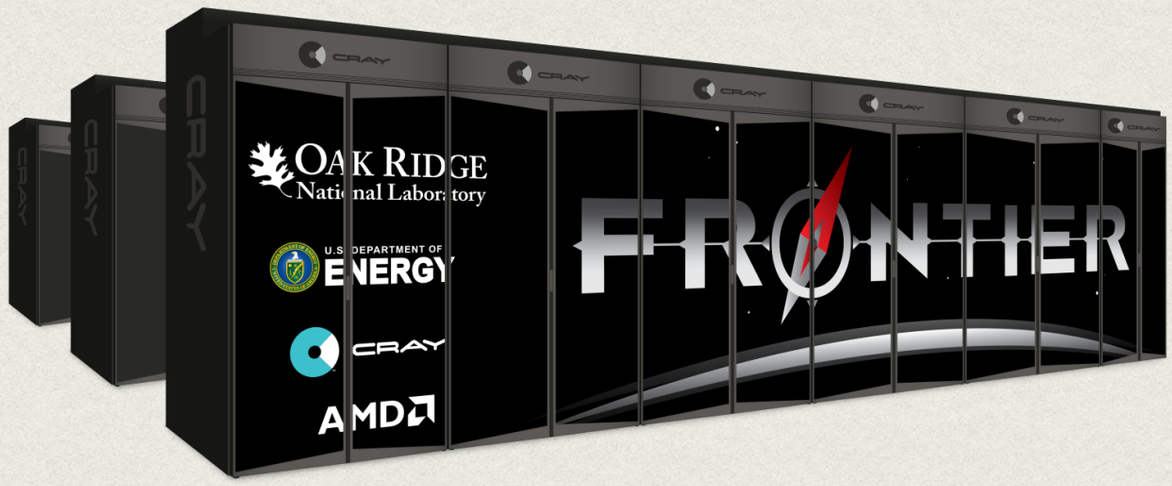
LESSONS LEARNT

- **Work in small steps:** Make incremental changes
- **Testing:** Weekly/bi-weekly testing
- DOCUMENT!!!
- **Test bed:** 3 diff. compilers, more than 2 hardware setup, more than 2 versions of a compiler (perhaps?)
- **Optimization:** ONLY after a thorough verification of the serial version
- **Verification:** Use tools to verify divergence of results between CPU and GPU

BEST PRACTICE #3:
REPORT BUGS

CAAR-ORNL-PICONGPU PROJECT

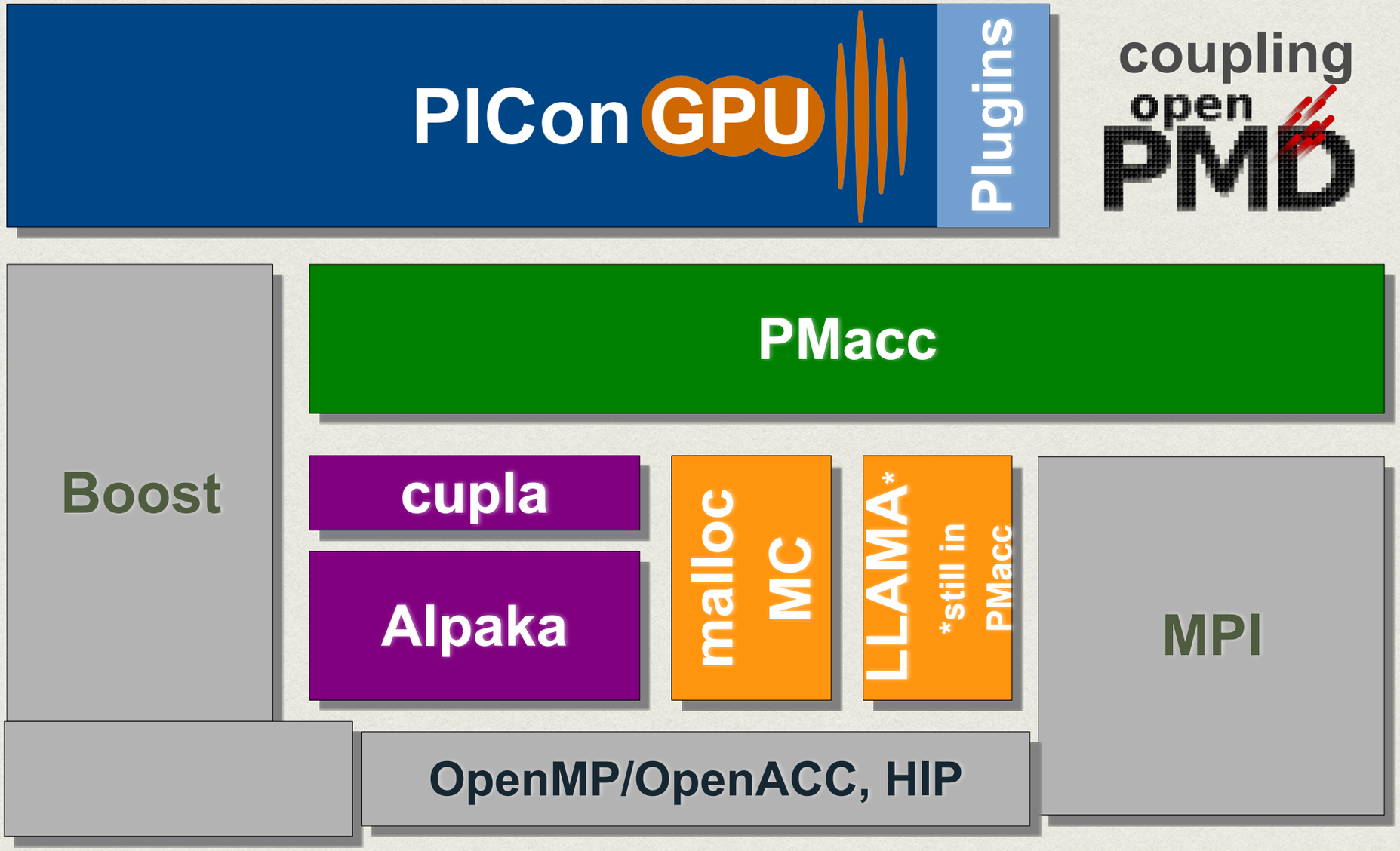
- Preparing PIConGPU, a plasma Physics application for the upcoming exascale system - Frontier



OAK RIDGE National Laboratory | UNIVERSITY OF DELAWARE | CASUS CENTER FOR ADVANCED SYSTEMS UNDERSTANDING | Georgia Tech | Partner von DRESDEN concept exzellenz-dresden.de | HZDR HELMHOLTZ ZENTRUM DRESDEN ROSSENDORF

CAAR Project in Collaboration with COE (AMD + Cray) developers

OUR OPEN HPC SOFTWARE STACK



CRPL
Sunita Chandrasekaran

University of Delaware

REPORT BUGS!

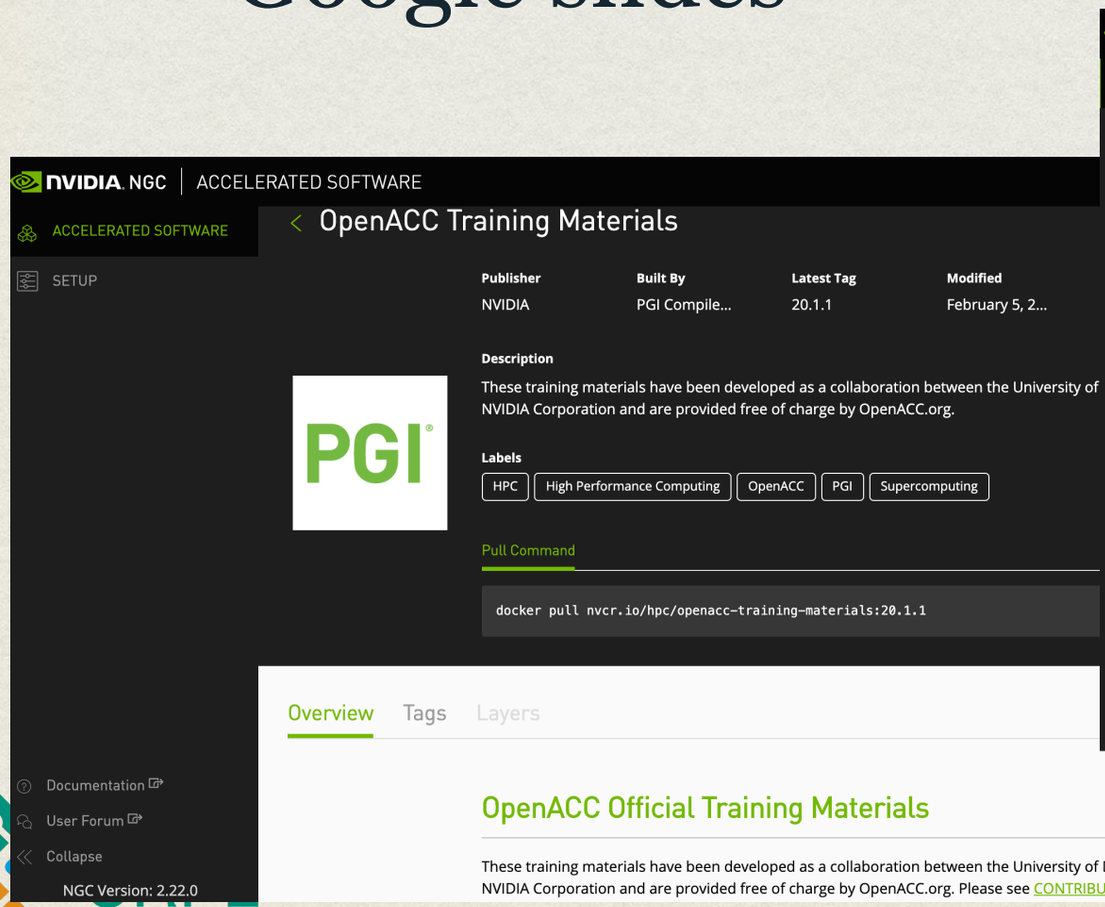
- **Code Review:** author of a PR cannot merge his/her own PR)
- **Report bugs:** help improve compilers
- **Reproducible code:** Useful to debug

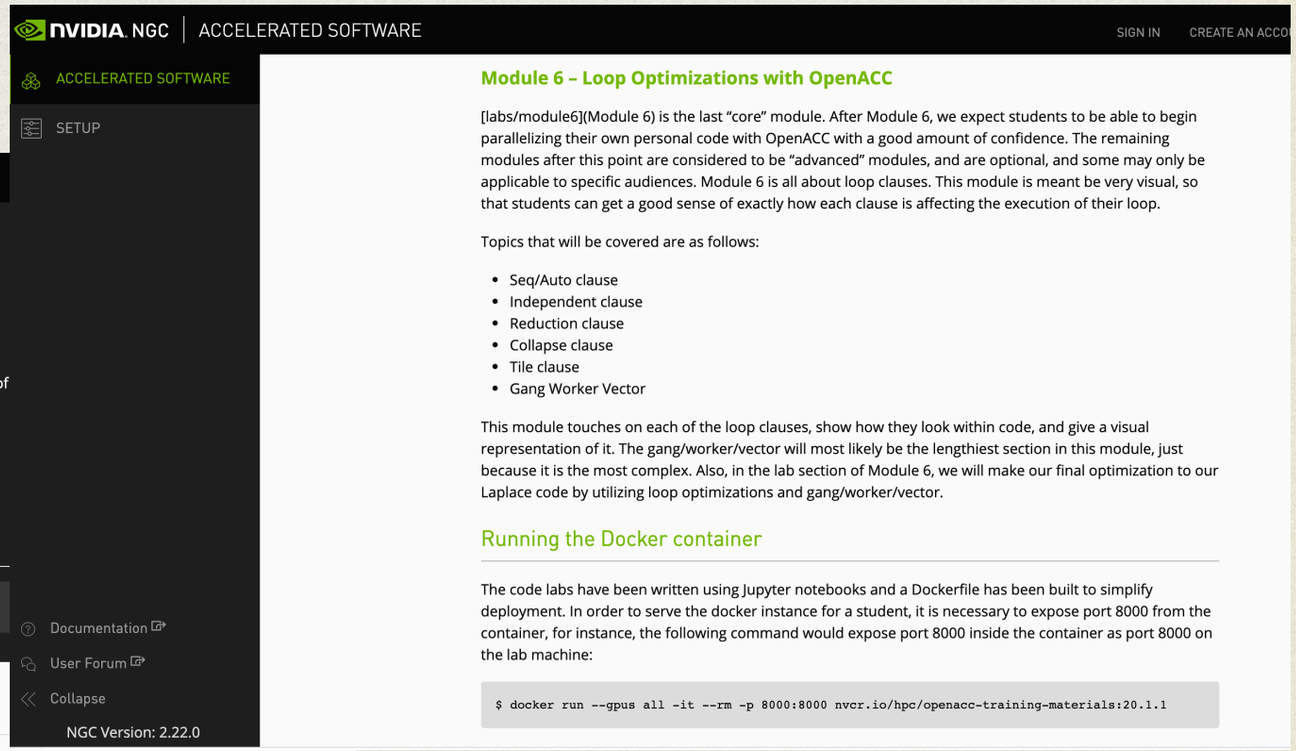
- **REPORT:** Workarounds OK but ”REPORT” bugs
- Report bugs via a ticket system (say Trac – wiki + issue tracker) and not via email – PLEASE! ☺
  - The bug and its fix got to be recorded
  - Documented
  - Code changes to be tracked
- **Time critical bugs** - communicate with the developers directly

BEST PRACTICE #4:
DOCKER

COLLABORATIVE SOFTWARE DEVELOPMENT

- Tools/Platforms such as Docker, Container, GitHub
  - Dramatically reduces barrier to collaboration
  - Google slides





BEST PRACTICE #4:
AUTOMATE

OPENMP VALIDATION & VERIFICATION TESTSUITE

OMP Specifications

Uses specs as guide for implementation

Compiler implementer

Uses vendor's compiler to support programming model

Running system

Uses specs as guide for programming

OpenMP User

Uses system as programming platform

ECP SOLLVE OpenMP V&V

https://github.com/SOLLVE/sollve_vv

https://crpl.cis.udel.edu/ompvvsollve/

OpenMP

U.S. DEPARTMENT OF ENERGY | Office of Science

OAK RIDGE National Laboratory | OAK RIDGE LEADERSHIP COMPUTING FACILITY

ECP EXASCALE COMPUTING PROJECT

CRPL Sunita Chandrasekaran

UNIVERSITY OF DELAWARE

ECP SOLLVE OPENMP V&V

GitHub	→	Collaborative Environment
Integrated with issue tracker	→	Keep track of bugs and fixes
Pull request review system	→	Maintain code history
Website	→	One stop shop for info
Publication, Presentations	→	Documentation
Coding standards	→	Templates - external collaboration

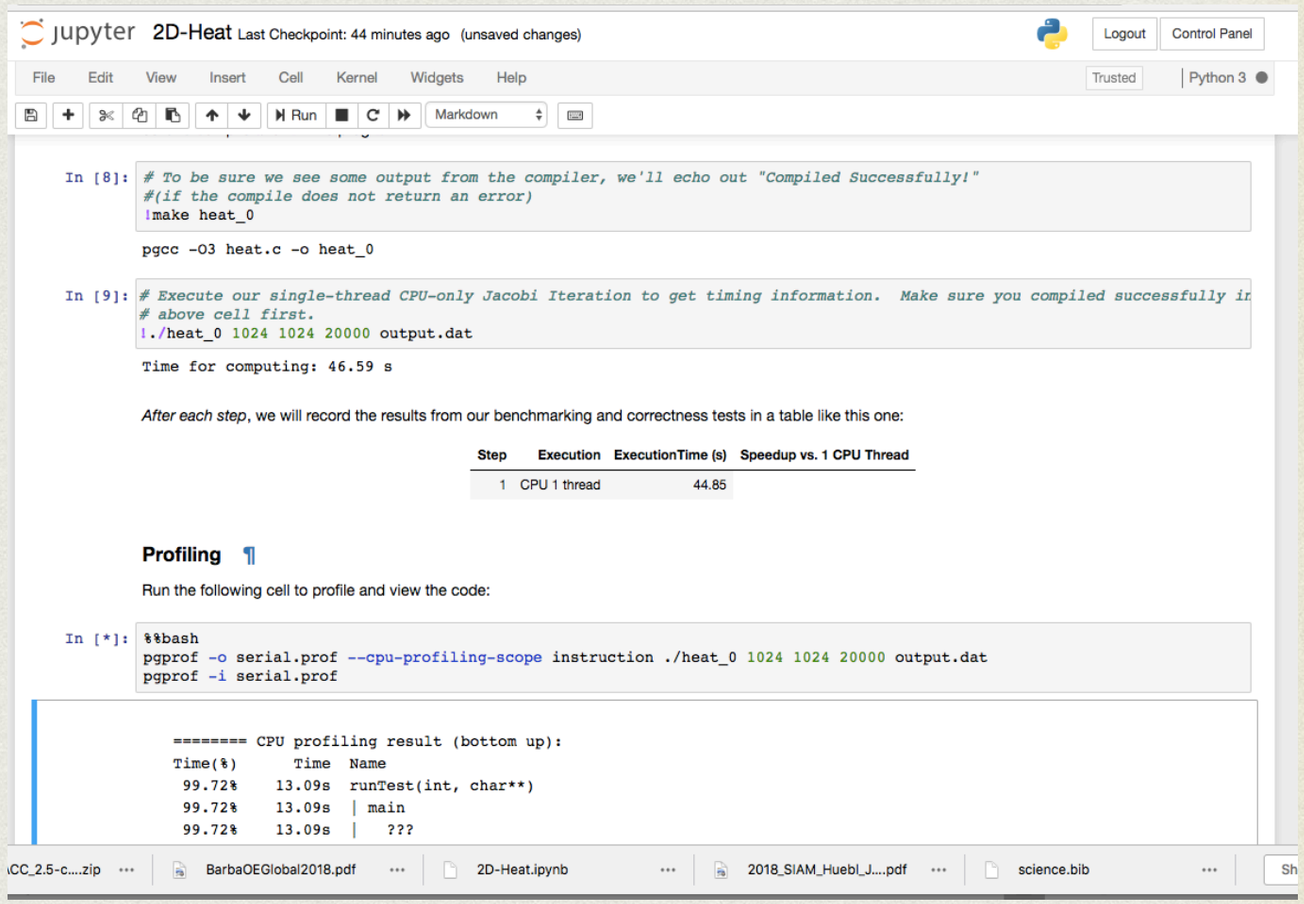
FURTHER IMPROVEMENTS

- Currently implementing a CI/CD ecosystem
  - Jenkins Automation System
  - Build, Deploy and Automate
  - Scale software quality
  - Save time
  - Prone to lesser errors
- More specifically to the V&V
  - Decoupling offloading tests from the rest of the code
  - Clear pass/fail messages

BEST PRACTICE #5: DOCUMENT (pretty pls! :-))

DOCUMENTATION

- IPython notebooks
- Embed code as part of documentation
- Probability of a programmer updating the documentation when the code changes is high!
- Useful for training and education



BEST PRACTICE #6:
PAIR PROGRAMMING

HACKATHONS

- Profile the code, understand performance bottlenecks
- Start with a smaller kernel
- Use a simple system to compile/execute the code for starters
- Identify a good starting point
- Debugging - “when in doubt, comment it out, re-test”
- https://www.gpuhackathons.org/

Chandrasekaran, Sunita, Guido Juckeland, Meifeng Lin, Matthew Otten, Dirk Pleiter, John E. Stone, Juan Lucio-Vega, Michael Zingale, and Fernanda Foertter. "Best Practices in Running Collaborative GPU Hackathons: Advancing Scientific Applications with a Sustained Impact." *Computing in Science & Engineering* 20, no. 4 (2018): 95-106.

BEST PRACTICE #7:
OPEN SOURCE BUT...

OPEN SOURCE IS GREAT BUT….

- Not sustainable unless there is a community that will help with the sustainability
- Need help
  - From sponsors, funding organizations, interested vendors
  - To ensure continuity of software maintenance beyond the funded project period

SUMMARY

- Best Practice #1 Profiling
- Best Practice #2: Systematic Testing
- Best Practice #3: Report bugs
- Best Practice #4: Automate
- Best Practice #5: Document
- Best Practice #6: Pair Programming
- Best Practice #7: Open Source but…

My group in action :-)

CRPL
Sunita Chandrasekaran

Computational Research and Programming Lab

ACKNOWLEDGEMENTS

- **CRPL: Sanhu Li, Mauricio Ferrato, Eric Wright, Mayara Gimenes, Fabian Mora, Matt Leinhauser, Thomas Huber, Matt Stack, Brad Atmiller, Hayden Carter, Kyle Friedline, Josh Davis, Jose Diaz**

- **PPMONE: Juan Perilla, Eric Wright, Mauricio Ferrato, Robert Searles, Alexander Bryer**

- **MURAM :Rich Loft, Matthias Rempel, Shiquan Su, Supreeth Suresh, Cena Miller, Raghu Kumar**

- **CAAR: Alex Debus, Thomas Kluge, Rene Widera, Sergei Bastrakov, Klaus Steiniger, Marco Garten, Matthias Werner, Jeff Kelling, Matt Leinhauser, Jeff Young, Josh Davis, Jose Diaz, Ronnie Chatterjee, Axel Huebl, Ronnie Chatterjee, Guido Juckeland, Michael Bussman**

- **ECP SOLLVE: Jose Diaz, Josh Davis, Thomas Huber, Swaroop Pophale, Oscar Hernandez, David Bernholdt**