COMPUTATIONAL INFRASTRUCTURE for GEODYNAMICS

I. Advancing Earth Science through Best Practices in Open Source Software: CIG

II. Software Attribution for Geoscience Applications in CIG

Lorraine J. Hwang

Joe Dumit, Alison E. Fish, LOUISE H. KELLOGG, MacKenzie Smith, and Laura Soito

UC DAVIS
UNIVERSITY OF CALIFORNIA

SAGA Software Attribution for Geoscience Applications

INNOVATING COMMUNICATION IN SCHOLARSHIP

UC DAVIS INTERDISCIPLINARY FRONTIERS IN THE HUMANITIES AND ARTS

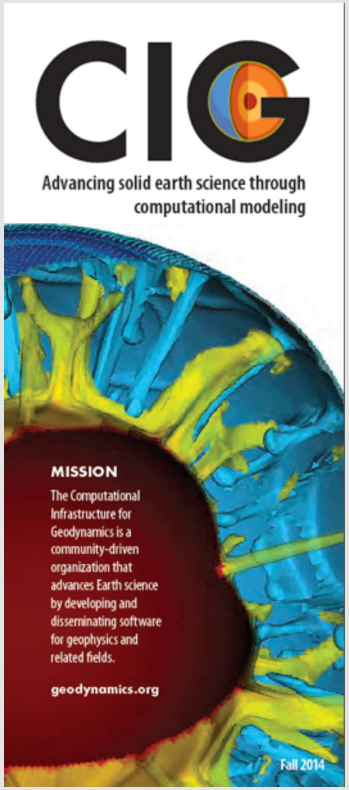
NSF

ABOUT CIG

The Computational Infrastructure for Geodynamics is a community-driven organization that advances Earth science by **developing and disseminating software** for geophysics and related fields.

Primary Scientific domains:

- Geodynamo
- Mantle convection
- Seismology
- Short and long term deformation of crust & lithosphere
- Computational science
- Fluid migration/multiphysics



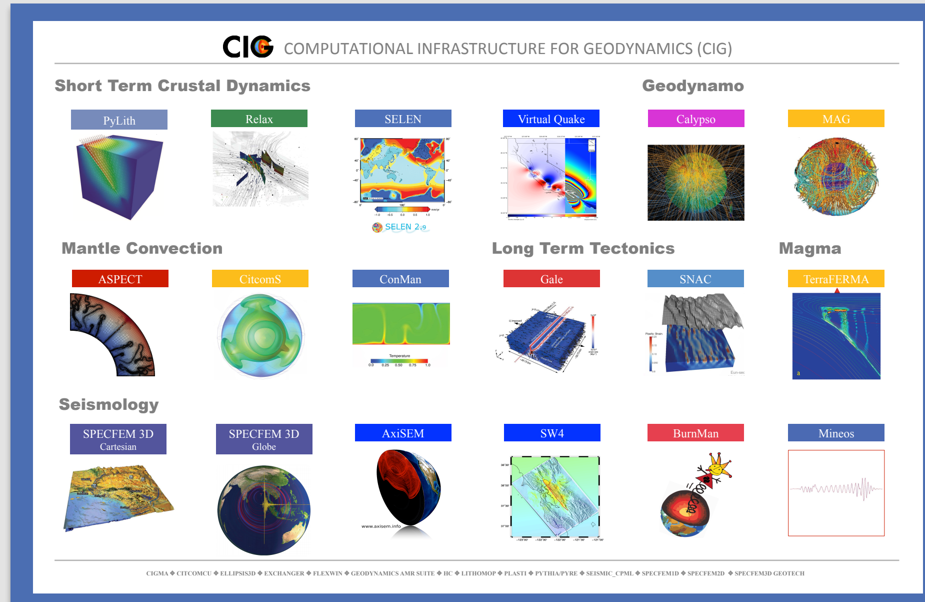
Funded primarily by the US National Science Foundation (NSF) Geoinformatics program

COMMUNITY SOFTWARE

CIG COMPUTATIONAL INFRASTRUCTURE for GEODYNAMICS

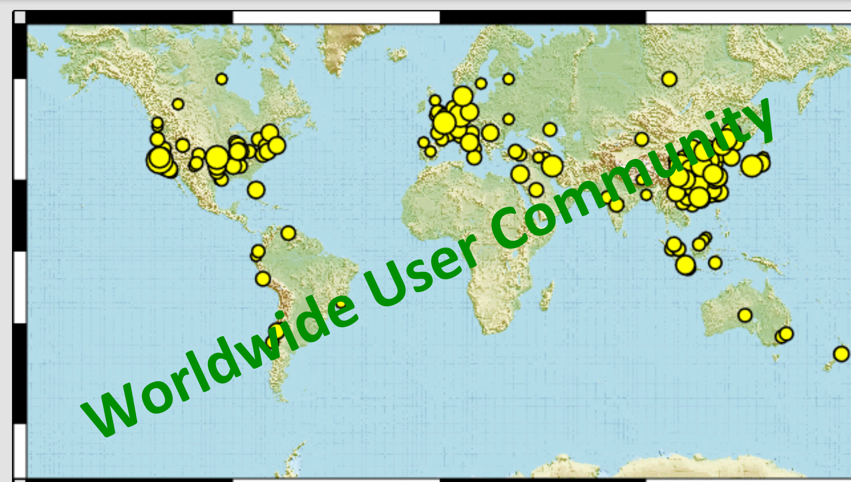
Domain relevant open source software contributed and developed by CIG and independent researchers.

- Domain scientists in collaboration with computational scientists



Maintained and developed by:

- Community - at-large
- Developer(s) – “hero” or small team
- Developers – large team



CIG REPOSITORY HOSTS 32 CODES IN A GITHUB REPOSITORY GITHUB.COM/GEODYNAMICS

Software Best Practices: Goals

- Usability

  Software is effective and is designed to promote ease-of-use.
- Sustainability

  Code can be improved and adapted to a changed environment; resilient.
- Reproducibility

  Scientific method built on reproducible and reliable results.

SBP: Supporting Community

Communications

- Mailing lists (8)
- Wikis

Training & Community building

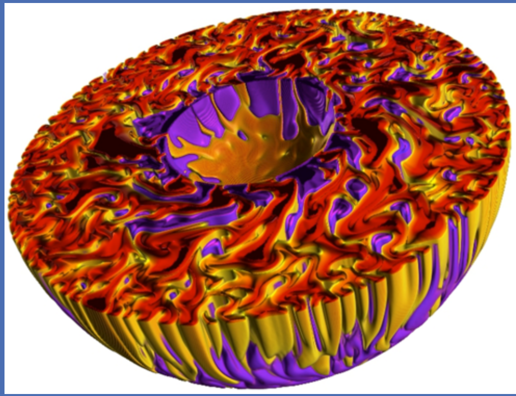
We coordinate and collaborate with other organizations to provide:

- Workshops
- Hackathons
- Tutorials
- Webinars

We target early career researchers.

Facilitating access to high performance computing

- Allocation on XSEDE for testing
- Coordinate code performance & accuracy benchmarks



CIG has invested in the development of *Rayleigh* a massively parallel spherical harmonic code. Running on ALCF Mira *5th fastest.*

SBP: CIG Standards

COMPUTATIONAL INFRASTRUCTURE for GEODYNAMICS

ALL → IDEAL

	Minimum	Standard	Target
Licensing	Open source	Same as Minimum.	Same as Minimum.
Version Control git	All source in version control.	Differentiation between maintenance and new development.	(a) New features added in separate branches. (b) Stable development branches for rapid release of new features.
Coding Write good code!		(a) User-friendly specification of parameters at run time. (b) Development plan, updated annually. (c) Comments in code with purpose of each function. (d) Users can add features or alternative implementations without modifying main branch. (e) User errors generate message that helps user correct the problem.	Standard + (a) Functionality implemented as a library rather than an application. (b) Output of provenance information. (c) Parallel access to inputs/outputs. (d) Checkpointing.
Portability, configuration and building Can I build it?	(a) Codes builds on Unix-like machines with free tools. (b) Portable build system.	Minimum + (a) Dependency checking. (b) Automation and portability of configuration and building. (c) Each simulation outputs all configuration and build options for reproducibility.	Standard + (a) Selection of compilers, optimization, build flags during configuration without modifying files under version control. (b) Multiple builds using same source. (c) Allows installation to a central location.
Testing Does it run?	(a) Code includes tests that verify it runs properly. (b) Results of accuracy and/or performance benchmarks (if established by the community).	Code includes pass/fail tests that verify it runs properly.	(a) Pass/fail unit testing for verification at a fine grain level. (b) Method of Manufactured Solutions for verification at a coarse grain level.
Documentation Can I use it?	(a) Instructions for installation. (b) Description of all parameters. (c) Explanation of physics the code simulates. (d) Cookbook examples with input files. (e) Citable publication.	(a) Description of workflow for research use. (b) Description of how to extend code in anticipated ways.	Standard + (a) Guidelines on parameter scales/combinations for which code is designed/tested. (b) FAQs or knowledge base.
User workflow But is it easy?		(a) Changing simulation parameters does not require rebuilding. (b) User-specified directories and filenames for input/output. (c) Use of standard binary formats. (d) Citation for code version.	Standard + Reproducibility via archiving of workflow.

SEE: https://geodynamics.org/cig/dev/best-practices/

SBP: Citation

COMPUTATIONAL INFRASTRUCTURE for GEODYNAMICS

Why?

- Provide credit and recognition to developers
- Aid in discoverability, reuse, and reproducibility

Are we following our own Best Practices?

"citable paper"

Science Paper (15)
Paper on the Code (4)
User Manual (3)
Website (1)
Additional Attribution (9)

14 with no citation information:

Archived/legacy (7), variants (3), other (3), missing (1)

DEVELOPMENT OF SCIENTIFIC SOFTWARE REPRESENTS A SIGNIFICANT INTELLECTUAL CONTRIBUTION TO THE FIELD.

NSF

CIG Citation Statistics

Sample: 5 years of self-reported or "searchable" publications (journals, conference proceedings, thesis): 308

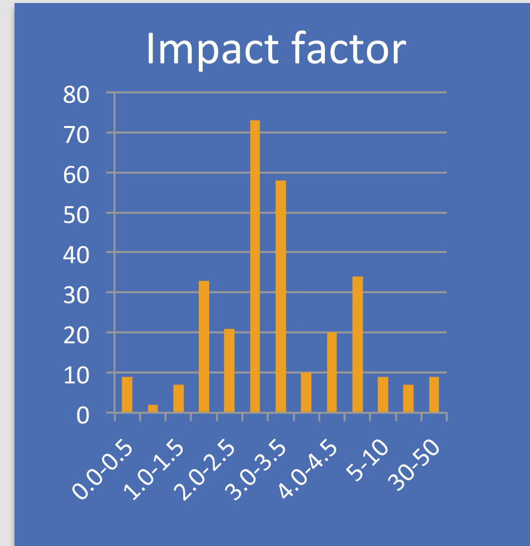
Total Code Mentions: 500

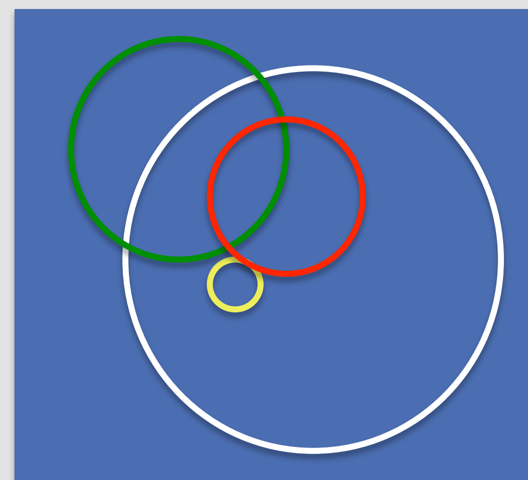
Version:	13% (65)
URL:	21% (104)
Licensing:	<1% (4)

Includes non-CIG codes both commercial and open source.

<u>**CIG ONLY codes**</u>

Mentions Code Name:	83% (257)
Citation:	75% (206)
Acknowledge CIG:	19% (58)





CITATION PRACTICES SHOW SOME CONSISTENCY WITHIN RESEARCH GROUPS.

SBP: Ideal Citation ?

TEXT

We use ***PyLith 2.1.0 for linux*** (***PyLith, 2015; Aagard and Williams, 2013; Aagaard et al., 2015***) published under the open source ***MIT*** license freely available through ***the Computational Infrastructure for Geodynamics (geodynamics.org)***.

ACKNOWLEDGEMENTS

We thank ***the Computational Infrastructure for Geodynamics (geodynamics.org)*** which is funded by the ***National Science Foundation under award NSF-094946.***

IN REFERENCE

PyLith. Computer software. geodynamics.org. Vers. 2.1.0. Computational Infrastructure for Geodynamics, 15 Feb. 19. Web. 01 July 2015. <geodynamics.org>. DOI

www.nature.com

Questions:

What is citable?

Who should be cited?

What is a persistent archive?

CREATING BOILERPLATE LANGUAGE INCREASES USE.

Implementation

1. **Website** Easy

   Standardized language accessible from landing page.

2. **Software** Not too difficult

   Plug in to generate citations at runtime

   Future Questions:

   - Transitive credit
   - Promote reproducibility
   - *Others?*

COMPUTATIONAL INFRASTRUCTURE for GEODYNAMICS

Home | About | Working Groups | Software | Developers | Projects | Events | News & Publications

Home > Software

Software Package List

PYLITH
Current Release
User Resources
Developer Resources
User Map

PyLith

PyLith is a finite-element code for dynamic and quasistatic simulations of crustal deformation, primarily earthquakes and volcanoes.

PyLith v2.1.0 provides a few new minor features and bugfixes.

A few small changes are needed in .cfg files to switch from v2.0.x to v2.1.0. See the README file for details.

Binaries are available from the links below. Detailed installation instructions for the binary packages are in the User Manual. Use the PyLith Installer to install from source. Detailed building instructions for a few platforms in the INSTALL file bundled with the PyLith Installer.

Release Notes

- Station names are required for output at arbitrary points (OutputSolnPoints) and are included in a /stations dataset in HDF5 files.
- A progress monitor will update a text file with the progress of a simulation (time in the time stepping loop or the number of impulses completed) and given an estimate of when the simulation will be completed.
- Bugfixes
  - A few bugs related to creating cohesive cells for fault intersections have been fixed. Faults can now meet at T intersections provided the buried edges of the faults are clamped. In other words, the fault ending at the T intersection has a clamped edge along the intersection. The fault ending at the intersection must also come AFTER the through-going fault in the list of fault interfaces.
  - There have been two major bug fixes for Drucker-Prager plasticity, for both DruckerPrager3D and DruckerPragerPlaneStrain. The first fix was a missing initial pressure term for the plastic multiplier in the Drucker-Prager formulation. This affects plasticity calculations when initial stresses are used. The error has been corrected in the code, the manual, and the unit tests. The second bug was an incorrect test for tensile yield that could cause PyLith to exit with an error when plastic yield had not actually occurred. The error would only occur when the allow_tensile_yield flag was set to False. This bug has been fixed in the code, and the new test is also described in the manual. This should prevent problems that previously existed when allow_tensile_yield was set to False (as it should be for most quasi-static problems).
  - Fixed bug in DataWriterHDF5Ext associated with multiple processes writing information to the HDF5 file. With external datasets the HDF5 file is limited to metadata and is maintained by process 0.
  - A two-dimensional gravity example has been added, based on the tutorial from the June, 2014 workshop at Stanford University. The tutorial itself is in examples/2d/gravity, and a new section has also been added to the manual describing the example.
  - Fixed inconsistent fault orientation when running in parallel for 2-D domains.

PyLith User Manual Version 2.1.0
Brad Aagaard, Matthew Knepley, Charles Williams, Leif Strand, Sue Kientz
www.geodynamics.org

Status:
Actively adding features to support improved science or performance by CIG.

Code changes:
4 commits this past month, 130 commits this past year.

Contact:
cig-short@geodynamics.org

Bug reports:
Github Issue Tracker

License:
MIT License

Current Release

BINARIES

These binaries are suitable for use on single core/processor computers as well as multi-core or multi-processor computers. They can only be used on a single compute node of cluster. Users wanting to run PyLith on multiple compute nodes of a cluster must build from source.

pylith-2.1.0-darwin-10.6.8.tgz[2015-02-19]
Version 2.1.0 binary for Darwin Intel, Mac OS X 10.6 and later.

pylith-2.1.0-linux-x86_64.tgz[2015-02-19]
Version 2.1.0 binary for Linux ~ x86_64 (64 bit), GLIBC 2.11 or later

3. **Other** Needs work

   Archiving, discoverability, workflows, etc.

KEEP IT SIMPLE AND STRAIGHTFORWARD.

Contact Us

Director
Prof. Louise H. Kellogg

Associate Director
Dr. Lorraine J. Hwang

contact@geodynamics.org

CIG
2119 Earth and Physical Sciences Building
One Shields Avenue
University of California, Davis, CA 95616

lhkellogg@ucdavis.edu
ljhwang@ucdavis.edu