Report on the Software Infrastructure for Sustained Innovation (SI^2) PI Meeting January 17-18, 2013, Arlington VA

https://sites.google.com/site/si2pimeeting/

Contributors: Workshop Participants
(as listed in Appendix I)

Editor: Ewa Deelman
[1]University of Southern California
deelman@isi.edu

Executive Summary

This workshop brought together Principal Investigators of the Software Infrastructure for Sustained Innovation (SI^2) software cyberinfrastructure projects to discuss issues relevant to the community as it moves into the future. 62 SI^2 projects were represented in total. These included 12 Institute Conceptualization, 20 SSI, and 30 SSE awards. 60 principal investigators represented the projects. Additionally, 9 participants from the community were invited to share their experiences and ideas. The workshop was organized to include guest presentations, panel discussions, and a poster session.

Some of the workshop findings included: 1) outreach is a critical part of software adoption; 2) it is important to be open with the community and contributors; 3) workforce development is important to the success of scientific software but it is hard to do, because it often requires multi-disciplinary training; 4) it is sometimes hard and yet important to have your software credited by others; and 5) software institutes can play an important role in the scientific software ecosystem.

There was a survey conducted before the workshop to gather community best practices.

Objectives

The goal of this workshop was to bring together Principal Investigators of Software Infrastructure for Sustained Innovation (SI^2) software cyberinfrastructure projects to discuss issues relevant to the community as it moves into the future. It is critical that the funds used by NSF for software development result in software being used effectively by scientists and engineers so that researchers in a number of domains can make advances in their respective fields without being overly burdened by interactions with the cyberinfrastructure.

Motivation

Since the publication of the "Dear Colleague Letter: Cyberinfrastructure Framework for 21st Century Science and Engineering (CF21)," much attention has been devoted to the development and support of software that can have a significant impact across scientific domains. As part of these efforts NSF/OCI established six Task Forces to examine cyberinfrastructure challenges from a number of angles: Campus Bridging, Grand Challenges, Software and Tools, Data, High Performance Computing, and Work Force Development. In 2011 the task forces published their reports: http://www.nsf.gov/od/oci/taskforces/. At the same time, OCI established a program: Software Infrastructure for Sustained Innovation (SI^2). In 2011 and 2012 the program funded software efforts in Scientific Software Elements (SSE) and Scientific Software Integration (SSI) categories. SSE focused primarily on small development efforts that can provide software pieces that can be integrated into the larger cyberinfrastructure. SSI targeted larger collaborations that were delivering significant community software

ACI is not alone is driving cyberinfrastructure efforts. Software is not developed in a vacuum, but rather through larger community efforts, where community needs are being translated into concrete roadmaps for the development of the needed software capabilities. An example of such an effort is EarthCube, where the geoscience and cyberinfrastructure communities are coming together to discuss the computational challenges faced by geoscientists, the advances in software, and to set a trajectory of innovation needed to make it easier to do science.

Recently, as data being collected and simulated grows to unprecedented scales, the focus of NSF is also shifting towards data-intensive solutions. This has been seen in the recent letter: "Dear Colleague Letter: New Solutions to Create Integrative Data Management Infrastructure(s) for Research Across the Sciences," which launched a new effort within ACI and broader NSF to examine issues of data access, management, and processing at scale. Similarly to EarthCube, DataWay was planned to convene community meetings and charrettes to define a comprehensive data management vision, architecture, and software to transform science in the 21st century.

The workshop discussed in this report brought together the Principal Investigators of the SI^2 awards made in 2011 and 2012 to discuss potential synergies and collaborations, define challenges ahead, discuss the relationship of the SI^2 efforts to the planned Software Institutes, and explore the relationship of the ACI-funded software in the context of the broad NSF initiatives such as EarthCube, DataWay, and other planned community-focused efforts.

Survey Findings

Ahead of the workshop, a survey was conducted to gauge the community interest in a number of topics and to gather information about the practices employed within the various SI^2 projects.

The survey was sent to the SI^2 PIs and invited guests. Altogether 59 people responded to the survey (42 finished it). Below, we provide an overview of the questions and responses.

Question 1: Rank the goals of the project (1) most important, (5) least:
1) Novel Science (20 responses ranking #1)
2) Quality Software (13 responses ranking #1)
3) Community Growth (7 responses ranking #1)
4) Sustainability (3 responses ranking #1)
5) Others: (written in by participant)
   - "reproducible science" -- 2 responses
   - "impact to field", "integrative collaborative science that is novel"
   - "support improved science code development processes" --- 2 responses
   - "change and improve the software and research culture of the science community"
   - "transfer of skills and best practice", "community education"

Question 2: ""What metrics will your project collect?"

The participants were free to provide their own metrics, which we organized based on the metrics developed by Shaowen Wang, as part of the CyberGIS Software Integration for Sustained Geospatial Innovation project [1].

The following tables summarize the participant responses.

Category	Metrics	Number of responses (41 total)
Communities and Users	Number and diversity of contributors/users	12
	Number of different application domains	2
	User feedback and experiences/surveys	10
	Number of end user issues/tickets, avg resolve time	3
	Number of software elements integrated	1
	Performance statistics against benchmarks	5
Usage	Software usage (CPU hours)	10
	Amount of quality datasets accessible	2
	Number of software elements used in integrated ways	2
	Number of visitors/visits	2
	Number of members on mailing list	2
Software	Interactions with the open source community/current open source mechanisms employed/what is contributed back to open source community	2
	New standards defined	1
	Number of downloads	17
Science	Number of publications and their impact measures	8
	Number of citations	11

Question 3: What project infrastructure are you using?

We also surveyed the infrastructure that the various projects are using. These are summarized in the following table.

Infrastructure/Capability	Software used	# of responses
Version Control	SVN	24
	Git	19
	Mercurial	3
Software hosting	Github	5 (4 more moving)
	Bitbucket	3
	Google code	2
	Sourceforge	1
Mailing lists	Mailman and others	19
Websites	Custom	13 (probably an undercount)
	Wordpress	4
Bundled Infrastructure	HUBzero	3
	Apache	1
	Eclipse	1
Project Management	Jira	3
Build and Test	Jenkins	3

We also asked the PIs if they allowed contributions from outside. 28 said yes, 5 said no.

Question 4: What is your project's approach to software sustainability?

The answers to that question can be categorized as follows:

- Integration with other projects
- Being part of larger projects/gateways
- Open access/ Open to contributions
- Technology transfer
- Diversification of users/communities / Outreach
- Establishing foundations/be part of consortiums
- Building ties with industry
- Be written into others' grants
- Standardization
- Software reuse
- Automated software generation

Since Software Institutes were starting to be conceptualized at the time of the workshop, we asked the following:
Question 5: What services could a software institute offer that you would be most likely to use?

We organized the answers into the following categories:

- **Workforce development**
  - Training and cross training of CS and domain scientists

  - "Access to qualified and motivated human resources"
- **Mentoring**
  - Fostering collaborations between SI^2s and outside
  - Expert advice/Project mentoring
  - Financial support
- **Community Building**
  - Meetings to discuss shared challenges and solutions
- **Sustainability**
  - Sustaining codes
  - Software repository
  - Ranking of institute content
- **Best Practices/Standardization**
  - Establishing protocols for interoperability
  - Tool recommendation
  - Technology forecasts
- **Services**
  - Provide auditing services (security, privacy, automated testing)
  - Code reviews
  - Integration of toolkits
  - GUI/website development
  - Organizing data

This clearly shows the need for software institutes in building a scientific software ecosystem. They can provide the "glue" and expertise that is not available within individual software projects.

Finally, the participants were asked to name projects that they look up to with the following:
Question 6: Are there any projects or communities, in or outside of science, that your project looks to as exemplars of what you are hoping to accomplish?

The following table lists the projects put forward by the participants.

Astronomy/HEP projects	iRODS	PETSc
HTCondor	Protein Data Bank	HUBzero
Dropbox	Amazon	Google docs

The participants highlighted the following positive aspects of the exemplar projects:

User-centered aspects	**Community aspects**	**Technology aspects**
"Easy to use"	"Community trusts it"	"Impressive technology"
"Devotion and responsiveness to users"	"Active vibrant community"	"Integrative services"
"Great user services"	"Active development group"	"Simplicity"

"Good documentation"	"International consortium model"	"Open API"
"Serves both large users and long tail users"	"Great Community"	"Successful models for supporting sustainability"
	"Broad appeal among govt, commercial & academia"	"Sound underlying infrastructure"
	"Diverse community"	

Workshop Findings

The workshop consisted of a number of panels, invited presentations, and discussions. The results of these activities focused on five main areas: 1) outreach 2) community building, 3) workforce development 4) software credit, and 5) software institutes.

In many cases, the workshop participants emphasized the importance of developing software tools that are directly applicable to their research. This was the case, for example, with the Galaxy software [2], which the PIs initially developed for their own purposes. They then grew the software, its capabilities, and community over time. The talk given by James Taylor captured many of the approaches that software providers adopt in their work. Some key points from James Taylor's talk included:

- Be useful
- Be useful as fast as possible
- Stay close to the science (present on the science not the tools)
- Don't implement things that are not useful
- Clearly articulate the value add
- Provide detailed news briefs for new features, make sure to also provide tutorials, and related materials
- Organize workshops (and provide free food)
- Don't pay too much attention to user demands.

Other software providers have also recognized that the new enhancements are often driven by the research interest and vision of the providers.

Outreach

Outreach was seen as a very important part of a software project. Some examples of the outreach performed by Galaxy included:

- Evangelize and be passionate
- Attend "tons" of meetings
- Give power to the users
- Python is a low barrier to entry because it is easy to learn, so a good choice for user interfaces
- Always do live demos, and do them perfectly—this has driven many usability and stability improvements
- Hire a community director and invest in community resources and outreach (nearly half of Galaxy team's is involved in it)

Community Building

Related to the outreach effort is community building. Most of the projects developed open source software, some of them also encouraged community contributions to the code base. It was recognized however, that most of the code was developed within individual projects.

Part of the community building effort involves consulting the community when determining new features to add to the software. Some of the users also serve as beta testers of the software, for the example in the BOINC project. However, there is always a tension between investing in existing components when incentives demand new functionalities.

Much of the community communications are done on mailing lists, so that support of the software can be distributed across all the users.

Workforce Development

Although students are often involved in the projects and learn from the research and software development processes, it is often not beneficial to the project to rely on students for software development. Professional programmers provide the necessary continuity, focus, and skills necessary to develop robust and usable software.

Students, however, are often targeted by the software institutes. For example the Water Institute (as being conceptualized) intends to teach undergraduate and graduate students better software engineering practices. It has also hosted Software Carpentry workshops for undergraduate students and scientists.

It is also difficult for software projects to fund the right people for software development, especially with many opportunities that are present in the commercial sector.

Credit for Software Contributions

There are two issues related to credit for software contributions: 1) how does one value the software? And 2) how does one value the contributions of a developer to the software and to the broader community?

In general, there is a lack of career path for students, developers, and researchers contributing to software development. Part of the issue is that there is an absence of credit for software contributions. It is difficult to track citations for a piece of software, especially when in today's systems the software is very layered and some layers are not visible to the end user. Ideally, authors would cite all the software they used for their work. However, it is also not clear how far down the software stack one needs to go—should we credit the compiler, the job scheduler?

So the question still remained open: how does a software contributor measure the impact of their work?

In general it is hard to measure the worth of a piece of software. One cannot just consider popularity (downloads for example), one also needs to consider impact—a piece of software can potentially be a one-off that enables someone to win the Nobel prize.

Role of Software Institutes

The workshop participants saw a number of different roles for the planed software institutes. Clearly, broader reach for the institutes was desired. An important outcome for the institutes would be to accelerate and sustain tools for science. However, there needs to be a symbiotic relationship between users and institutes, where each side benefits from each other's work.

The institutes face sustainability problems and need to think about a business model from day one. The following were the models discussed during the meeting:

- keep applying for funding from NSF
- seek funds from the open access community
- provide a service that is useful enough for people to pay for it
- provide open content and data for free but charge for add-on services.

In the UK, some projects started with a single grant, and then eventually were able to sustain themselves by getting multiple, diverse grants. Other projects were funded by university overheads.

At the time of the workshop the software institutes were just being conceptualized, so much work still lay ahead.

Acknowledgments: We would like to thank Program Director Daniel S. Katz (National Science Foundation) for his helpful comments and suggestions.

References:

[1] CyberGIS http://cgwiki.cigi.uiuc.edu:8080/mediawiki/index.php/CyberGISMetricsDiscussion
[2] Galaxy http://galaxyproject.org/

Appendix I: Workshop Participants

Invited Participants
Neil Chue Hong, Director, Software Sustainability Institute, EPCC
Gideon Juve, University of Southern California
Jim Herbsleb, Carnegie Mellon University
James Howison, University of Texas at Austin
Michael McLennan, Purdue University
Jarek Nabrzyski, University of Notre Dame
Jason Priem, University of North Carolina at Chapel Hill
James Taylor, Emory University
Von Welch, Indiana University

PI	Affiliation	Project Name
Stanley Ahalt	RENCI	Conceptualization of a Water Science Software Institute
Jay Alameda	NCSA	SSI: A Productive and Accessible Development Workbench for HPC Applications Using the Eclipse Parallel Tools Platform
David Anderson	UC Berkeley	SSI: Next-Generation Volunteer Computing
Ian Anderson	Utah State University	SSE: Interdisciplinary Software Infrastructure for Differential Geometry, Lie Theory and Applications
David August	Princeton University	SSI: Accelerating the Pace of Research through Implicitly Parallel Programming
Jogesh Babu	Pennsylvania State Univ. University Park	SSE: Statistical software for astronomical surveys
Paul Barbone	Boston University	SSE: Collaborative Research: Advanced Software Infrastructure for Biomechanical Inverse Problems
George Biros	Georgia Tech Research Corporation	SSE: Software for integral equation solvers on manycore and heterogeneous architectures
Philip Bourne	University of California-San Diego	Conceptualization and Analysis of a 3D Virtual Cell
Richard Brower	Boston University	SSI: Scalable Hierarchical Algorithms for Extreme Computing (SHARE)
Daniel Bump	Stanford University	SSE: Sage-combinat: Developing and Sharing Open Source Software for Algebraic Combinatorics
Markus Bürg	Texas A&M University	SSI: Open Source Support for Massively Parallel, Generic Finite Element Methods
Garnet Chan	Cornell University	SSE: General Tensor Software Elements for Quantum Chemistry, Tensor Network Theories, and Beyond

PI	Affiliation	Project Name
Ann Chervenak	University of Southern California	Collaborative Research: Conceptualizing an Institute for Empowering Long Tail Research
T. Daniel Crawford	Virginia Polytechnic Institute and State University	A Scientific Software Innovation Institute for Computational Chemistry and Materials Modeling
T. Daniel Crawford	Virginia Polytechnic Institute and State University	SSI: Sustainable Development of Next-Generation Software in Quantum Chemistry
Ewa Deelman	University of Southern California	SSI: Distributed Workflow Management Research and Software in Support of Science
Gabriel Dos Reis	Texas Engineering Experiment Station	SSE: Supporting Generic Programming in C++ for Modular and Reliable Large-Scale Software
Anshu Dubey	University of Chicago	Building Community Codes for Effective Scientific Research on HPC Platforms
Kevin Eliceiri	University of Wisconsin-Madison	SSE: SCIFIO: An Extensible Framework for Scientific Image Interoperability
Ian Foster	University of Chicago & ANL	SSI: SciDaaS - Data Management as a Service
Brent Fultz	California Institute of Technology	Collaborative Research: Scientific Software Innovation Institute for Advanced Analysis of X-Ray and Neutron Scattering Data (SIXNS)
Cynthia Gibas	University of North Carolina at Charlotte	SI2-SSE: Reducing the Complexity of Comparative Genomics with Online Analytical Processing
Ganesh Gopalakrishnan	University of Utah	SSE: Correctness Verification Tools for Extreme Scale Hybrid Concurrency
Boyce Griffith	New York University Medical Center	SSE: Parallel and Adaptive Simulation Infrastructure for Biological Fluid-Structure Interaction
So Hirata	University of Illinois at Urbana-Champaign	SSE: Adaptive Software in Quantum Chemistry
Matt Jones	University of California-Santa Barbara	Conceptualizing an Institute for Sustainable Earth and Environmental Software (ISEES)
George Karypis	University of Minnesota-Twin Cities	SSE: Software Infrastructure For Partitioning Sparse Graphs on Existing and Emerging Computer Architectures
Mike Kirby	University of Utah	SSE: A GPU-Enabled Toolbox for Solving Hamilton-Jacobi and Level Set Equations on Unstructured Meshes
Akos Ledezci	Vanderbilt University	SSI: Development of an Integrated Molecular Design Environment for Lubrication Systems (iMoDELS)

PI	Affiliation	Project Name
Dane Morgan	University of Wisconsin-Madison	SSI: A Computational Materials Data and Design Environment
Dan Negrut	University of Wisconsin-Madison	SSE Collaborative Research: SPIKE-An Implementation of a Recursive Divide-and-Conquer Parallel Strategy for Solving Large Systems of Linear Equations
Vijay Pai	Purdue University	Conceptualizing an Institute for Using Inter-Domain Abstractions to Support Inter-Disciplinary Applications
Dhabaleswar Panda	Ohio State	SSI: A Comprehensive Performance Tuning Framework for the MPI Stack
Philip Papadopoulos	UC San Diego	SSE: Fingerprinting Scientific Codes to Verify and Create Compatible System Software Environments
Beth Plale	Indiana University	SSE: Pipeline Framework for Ensemble Runs on Clouds
Jeffrey Potoff	Wayne State University	SEE: Development of a GPU Accelerated Gibbs Ensemble Monte Carlo Simulation Engine
Viktor Prasanna	University of Southern California	Software Infrastructure for Accelerating Grand Challenge Science with Future Computing Platforms
James Pustejovsky	Brandeis University	SSI: The Language Application Grid: A Framework for Rapid Adaptation and Reuse
John J. Rehr	University of Washington	SSE: Cloud-Computing-Clusters for Scientific Research
Christopher Roland	North Carolina State University	SSE: Software Tools for Biomolecular Free Energy Calculations
Karsten Schwan	Georgia Tech Research Corporation	SSI: A Glass Box Approach to Enabling Open, Deep Interactions in the HPC Toolchain
Shawn Shadden	Illinois Institute of Technology	SSE: Lagrangian Coherent Structures for Accurate Flow Structure Analysis
John Shumway	Arizona State University	SSE: Developing and Deploying Path-Integral Quantum Simulation Tools for a Broad Research Community.
Edgar Spalding	University of Wisconsin-Madison	Determining the Cyberinfrastructure Needs for Efficient Phenomics Research
Michael Stonebraker	Massachusetts Institute of Technology	SSE: SciDB - A Scientific DataManagement System
Ricardo Taborda	CMU	SSI: A Sustainable Community Software Framework for Petascale Earthquake Modeling
David Tarboton	Utah State University	SSI: An interactive software infrastructure for sustaining collaborative community innovation in the hydrologic sciences

PI	Affiliation	Project Name
Sameer Tilak	UCSD	SSI: Empowering the Scientific Community with Streaming Data Middleware: Software Integration into Complex Science Environments
Douglas Thain	Notre Dame	SSE: Connecting Cyberinfrastructure with the Cooperative Computing Tools
Gregory E Tucker	University of Colorado at Boulder	SSE: Component-Based Software Architecture for Computational Landscape Modeling
Robert van de Geijn	University of Texas at Austin	SSI: A Linear Algebra Software Infrastructure for Sustained Innovation in Computational Chemistry and other Sciences
Eric Van Wyk	University of Minnesota-Twin Cities	SSE: Collaborative: Extensible Languages for Sustainable Development of High Performance Software in Materials Science
Jan Vitek	Purdue	SSE: A Tracing Virtual Machine for Statistical Computing,
Jan Vitek	Purdue	SI2: Conceptualization: Dynamic Languages for Scalable Data Analytics
Ross Walker	University of California-San Diego	SSE:Sustained Innovation in Acceleration of Molecular Dynamics on future computational environments: Power to the People in the Cloud and on Accelerators
Shaowen Wang	University of Illinois at Urbana-Champaign	SSI: CyberGIS Software Integration for Sustained Geospatial Innovation
Michael Wilde	University of Chicago Computation Institute	SSE: Enhancement and Support of Swift Parallel Scripting
Nancy Wilkins-Diehr	University of California-San Diego	The Science Gateway Institute (SGW-I) for the Democratization and Acceleration of Science
Theresa Windus	Iowa State University	SSI: Developments in High Performance Electronic Structure Theory
Xiaodong Zhang	Ohio State University	SSE: A Unified Software Environment to Best Utilize Cache and Memory Systems on Multicores
Vineet Yadav	Standford University	SSI: Real-Time Large-Scale Parallel Intelligent CO2 Data Assimilation System

Appendix II: Workshop Agenda

Thursday 1/17/2013

9:00-9:15am	Welcome and Workshop Goals	Ewa Deelman
9:15-9:45am	NSF OCI Perspective	Alan Blatecky
9:45-10:30am	Keynote 1: Accessible, transparent, reproducible analysis with Galaxy	James Taylor
10:30-11:00am	Break	
11:00am-12:30pm	Panel 1: How to measure the impact of software? *Panelists: Jim Jagielski, Michael McLennan, Jason Priem, Doug Thain, Robert van de Geijn*	James Howison
12:30-1:30pm	Lunch	
1:30pm-2:00pm	CISE and Big Data	Suzi Iacono
2:00-3:30pm	Panel 2: What does it mean to Conceptualize? *Panelists: Stan Ahalt, Phil Bourne, Phil Colella, Ian Foster, Jan Vitek*	Nancy Wilkins-Diehr
3:30-4:00pm	Break	
4:00-4:45pm	Talk and discussion on Software Sustainability	Neil Chue Hong
4:45-5:15pm	SI2 Program Current and Future	Dan Katz
6:00pm-8:00pm	Poster session	

Friday 1/18/2013

8:30-9:15am	Keynote 2: Software Ecosystems and Science	Jim Herbsleb
9:15-10:30am	Panel 3: Managing a software project—the dos and don'ts *Panelists: David Anderson, George Karypis, Dhabaleswar Panda, Von Welch*	Phil Papadopoulos
10:30-11:00am	Break	
11:00-12:15pm	Group Discussion: Getting the scientists on board! How do you make your software useful?	Miron Livny
12:15-12:30pm	Concluding remarks	Miron Livny