

NASA/CP-2012-217366



Open Source Summit Proceedings

March 29-30, 2011

*Nicholas Skytland
NASA Johnson Space Center
Houston, Texas 77058*

NASA STI Program ... in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA scientific and technical information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NASA Aeronautics and Space Database and its public interface, the NASA Technical Report Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- **TECHNICAL PUBLICATION.** Reports of completed research or a major significant phase of research that present the results of NASA Programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.
- **TECHNICAL MEMORANDUM.** Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- **CONTRACTOR REPORT.** Scientific and technical findings by NASA-sponsored contractors and grantees.
- **CONFERENCE PUBLICATION.** Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.
- **SPECIAL PUBLICATION.** Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- **TECHNICAL TRANSLATION.** English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include organizing and publishing research results, distributing specialized research announcements and feeds, providing help desk and personal search support, and enabling data exchange services.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at <http://www.sti.nasa.gov>
- E-mail your question via the Internet to help@sti.nasa.gov
- Fax your question to the NASA STI Help Desk at 443-757-5803
- Phone the NASA STI Help Desk at 443-757-5802
- Write to:
NASA STI Help Desk
NASA Center for AeroSpace Information
7115 Standard Drive
Hanover, MD 21076-1320

NASA/CP-2012-217366



Open Source Summit Proceedings

March 29-30, 2011

*Nicholas Skytland
NASA Johnson Space Center
Houston, Texas 77058*

Available from:

NASA Center for AeroSpace Information
7115 Standard Drive
Hanover, MD 21076-1320
301-621-0390

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161
703-605-6000

This report is also available in electronic form at <http://ston.jsc.nasa.gov/collections/TRS/>

Contents

Introduction	3
Overview of Event.....	4
Analysis of Input Collected.....	5
Major Issues and Recommendations.....	6
Issue #1: Communication and Publicizing NASA’s Open Source Efforts.....	6
Issue #2: Licensing.....	6
Issue #3: Barriers to Involvement from the Open Source Community.....	8
Issue #4: Barriers to Development Models and Ongoing Support	9
Issue #5: Government Restrictions	9
Issue #6: Limitations on Contributing to External Open Source Projects.....	10
Issue #7: How does Open Source Governance look within NASA?	10
Issue #8: How Should Open Source Efforts be Supported?	11
Issue #9: How does NASA Open Source *Everything?*	12
Issue #10: How to Close the Feedback Loop Between Policy Makers, Developers and End Users?	13
Issue #11: How to Encourage Cultural Change in Hiring Practices?	14
Issue #12: How to Package Open Source Software to be More Accessible.....	14
Issue #13: Combining Open Source Software Development Standards with Office of the Chief Engineer Policies	15
Unanswered Issues:	15
Appendix	16
Registered Participant List	16
Registered In-Person Attendees	16
Registered Remote Attendees.....	22
Press.....	43
Staff	43
Agenda and Schedule.....	45
Day 1 – March 29, 2011	45
Day 2 – March 30, 2011	46
Keynote Presentations.....	47
Excerpts from Ustream Chat.....	48
Tuesday	48

Wednesday	52
UserVoice Submissions	55
Communication/Press Summary	83

Introduction

On March 29 & 30, 2011, NASA hosted its first Open Source Summit (OSS) at Ames Research Center in Mountain View, California. The event brought engineers and policy makers from across NASA together with well-respected members of the open source community to discuss current challenges with NASA's open source policy framework, and propose modifications that would make it easier for NASA to develop, release, and use open source software.

Open source brings numerous benefits to NASA software projects, including increased software quality, reduced development costs, faster development cycles, and reduced barriers to public-private collaboration through new opportunities to commercialize NASA technology. This inherently transparent, participatory, and collaborative approach is revolutionizing the way software is created, improved, and used.

Although open source release has already provided some of the potential benefits to NASA, the full benefits of open source can only be realized if NASA is able to establish the processes, policies, and culture needed to encourage and support open source development. This will require expanding open source activities beyond releasing software only after completion and finding new ways to support two-way collaboration with an open development community throughout the entire software lifecycle.

This document provides a snapshot of the activities during the OSS and summarizes the major issues and recommendations received not only from the in-person attendees, but also through the various online venues utilized during the event.

Overview of Event

The OSS was modeled after the popular OpenGov Community Summits that were co-hosted by the Department of Transportation, U.S. Department of Agriculture, NASA, Treasury, Environmental Protection Agency, and General Services Administration throughout 2010. Called a Focus Forum, the format of the OSS was geared toward innovating new thinking and capturing solutions on a specific topic. Attendees were encouraged to engage in conversation and share ideas as well as participate digitally in various online venues.

Attendees heard from a variety of stakeholders in the open source community, including the Department of Defense (DoD)-affiliated Institute for Defense Analyses' David Wheeler, IBM's Bob Sutor, Google's Chris DiBona, and GitHub's Chris Wanstrath. Additionally, a number of leaders of NASA's open source efforts presented their projects and shared their experiences working within the current software development constraints at the agency. Targeted breakout sessions, which occurred between speaker presentations, addressed a number of critical issues and gathered ideas for potential solutions.

A breakdown of the event's agenda and breakout session topics may be found in the appendix.

Analysis of Input Collected

Ideas and input were collected at the OSS through a variety of methods that included participation from both participants physically at the event and remote participants via online platforms. Five communication vehicles for discussion were used: Google Docs, Ustream, UserVoice, Twitter, and Maestro (a virtual teleconference platform). Each platform served a different purpose and audience, with many participants using multiple services simultaneously.

Google Docs

Participants used Google Docs, primarily, as a tool to take notes and write down proposed solutions to the issues raised at the event. Google Docs serves as the most substantial documentation of the event. Twenty-three different documents were created and shared with the public, each covering a separate topic. From these topics, 66 proposed solutions were offered, a full listing of which is offered in the **Major Issues and Recommendations** section.

Ustream

Remote participants used Ustream to watch live video of keynote speakers during the event. Additionally, an extensive chatroom was formed where participants discussed many aspects of the event (see **Ustream Chat** in the Appendix). There were a combined total of 2,315 views on Ustream throughout the event.

UserVoice

Participants used UserVoice, an ideation tool, to record and vote on specific actions or ideas during the event. Forty-seven unique ideas were posted on UserVoice and attracted a total of 638 votes and 125 comments. A full record of UserVoice activity is located in the **UserVoice Submissions** section in the Appendix.

Twitter

Participants also used Twitter extensively to communicate using the hashtag #NASAOSS. Over 1,250 tweets were recorded originating from the event, reaching more than 3 million people.

Maestro

Remote participants who wished to take part in various breakout sessions could use Maestro as a teleconference tool. Sixty participants used Maestro at the event for a total of 2,578 minutes of call time.

A synthesis of the input collected through all the above tools is located in the **Major Issues and Recommendations** section.

Major Issues and Recommendations

Issue #1: Communication and Publicizing NASA's Open Source Efforts

For an open source policy to be successful, NASA must make an effort to encourage both internal and external parties to participate in open source development. What does the agency need to do to make NASA's open source efforts well known?

Proposed solutions:

1. Portal page for NASA workers to locate agency open source projects

Create an online portal with historical and existing open source projects within the agency. The portal should have wiki-style tools for projects to assist with collaboration and publicity. Perhaps expanding this to be an entire "forge" repository, similar to Forge.mil. This could be a platform to base an entire federal-wide code repository on.

2. More contests

Utilize coding contests to draw people in to work on open source projects, similar to Google's Summer of Code. Rewards could include cash prizes, publicity, or simply the prospect of adoption.

3. Encourage contractor participation

NASA would benefit from increased contractor participation in open source activities, perhaps by requiring software developed under government contract to be released as open source software by default unless restricted by export regulations or other security restriction.

4. Define main user categories

NASA open source developers, general public, worldwide open source developer community, science community.

5. Attend additional open source conferences

OSCON was discussed as a venue for NASA presentations or participation.

6. Create monthly meetup groups

Monthly local meetups (perhaps with bimonthly nationwide phone tagups) would galvanize community interest.

7. Create a community manager position

Someone to ensure the community is healthy and to research new initiatives, and to run top coder competitions, unconferences, and code-a-thons.

Issue #2: Licensing

The NASA Open Source Agreement license (NOSA) was originally developed in 2003 to enable NASA to provide software in source code form to the public; however, software must already be considered

complete prior to public release. This precludes the ability to develop software iteratively with other agencies and the public. To participate in the open source developer community, NASA needs to be involved in the development process from the beginning.

Two issues need to be addressed:

- How does NASA license the code it develops internally? Should it use NOSA or drop it? Why does such code need to be licensed at all? Shouldn't it be public domain?
- What licenses are conducive to government agencies using non-government code? For those that aren't conducive, the government needs a model for using those licenses in a way that makes lawyers happy.

Proposed solutions:

1. Drop NOSA in favor of existing mainstream open source licenses

Although innovative when first developed over 7 years ago, NOSA is not a well-received license in the open source community, and the purpose for its development no longer exists. NASA software should be released under whatever mainstream open source license makes sense within the development environment in which it is being released. Further, re-license software that was previously released under NOSA using one of the mainstream open source licenses.

2. Create a policy and licensing for unfinished or in-progress development

The NOSA was created for releasing software that is complete. NASA needs a policy that addresses licensing options for iterative software development with a community that includes non-NASA workforce.

3. Be aware of licensed software within other open source software covered by a different open source license

An open source software package often will list only the primary license but may include modules from other sources in its distribution that are covered under a different license. Source code files may reveal additional licenses, as well as original copyright holders.

4. Approve a subset of OSI-approved licenses for NASA use

Review and approve a set of mainstream open source licenses (BSD, MIT, GPL, Apache, etc.) that can be used to license publicly released NASA software so that such a review does not have to be performed for each release of NASA software.

5. Provide a one-stop shop for NASA guidance with regard to licensing open source software

Provide a comprehensive source of the current regulations and restrictions. Create a FAQ that can be used to explain how different licenses may impact the release software.

6. Define NASA requirements for Contributor License Agreements (CLAs).

Currently, a CLA is required to accept and use third-party contributions. Yahoo! Is currently

using “Harmony CLAs” which are broadly accepted. Others include Fedora CLA, GNU Contributor agreement, and DARPA F6.

Issue #3: Barriers to Involvement from the Open Source Community

What are the limits of community contribution for NASA, as a government agency bound by significant regulation and bureaucracy? For example, could a NASA-originated codebase ever be handed over to a non-NASA community member for long-term support and maintenance? Is this legal? If it were legal, would it be practical?

Things to consider:

- Open source developers should not be considered free labor.
- You need to have a mutual benefit for people to have motivation to contribute.
- Project/mission needs must be balanced with community needs.
- License compatibility issues are a barrier to open contribution and distribution of code.
- If NASA has the NOSA, but project contributors take patches from the community without requiring copyright assignment paperwork, it short-circuits the intention of the NOSA. People are going to route around the problem of overly restrictive licenses in practice.
- People need to learn about the code base before they can contribute.
- Opening things up early and accepting contributions would be better than trying to open things up later.
- NASA, as an agency, has to come to terms with “letting go” of a project and turning it over to the community for the community to thrive.
- NASA has a slew of exciting projects, but is laden with bureaucratic processes and delays.

Potential Solutions:

1. Have NASA engage openly with the OSS community

Using and finding engagement in OSS communities is a great way of driving expertise inward. NASA teams are already using Drupal, Django, Plone, Zope, PostgreSQL, Apache, memcache, Trac, SVN and many others. However, there is a need to incentivize (or at least explicitly encourage) involvement with open source communities. How does NASA contribute without implying endorsement of a technology or product?

2. Use existing open source development tools

Using proprietary or internal tools in the development of open source software can limit the number of developers who can participate. Software tools need to encourage open source participation. For example, using Subversion to manage source control is a barrier to entry because an open source developer cannot commit to a Subversion repository without having commit access. Using a distributed version control system such as Git, Bazaar, or Mercurial allows developers to freely commit to his or her own copy of an open source project and allows for NASA to more easily select which commits it wants to integrate into a version specialized for mission use.

3. Make mailing lists and internal communication public

A large part of the development of open source software is having access to communication related to its development. Use separate, public mailing lists for each open source project at NASA. Set up Wikis and document repositories to host design documents, requirement analyses. Or better, host NASA open source projects on public repositories such as SourceForge and GitHub.

4. Describe how developers can contribute (and follow through)

Each project should create a public facing webpage that describes the process of contributing changes to open source projects. Actively work with developers to get their contributions committed to the main repository and respond promptly to questions on mailing lists.

Issue #4: Barriers to Development Models and Ongoing Support

How do we ensure that “open-ness” does not conflict with rigor? How narrow should the definition of “development team” be?

Proposed Solutions:

1. Get folks with passion for a project involved, even if they’re not formally assigned to that project

Those motivated by passion tend to do fantastic work. Open forums for suggestions tend to get great ideas. At times, it makes sense to borrow or lend team members to contribute to other projects – the TEAM wins in the end.

2. “Bug Bounty” program

Some organizations pay developers to fix bugs in open source software, particularly bugs related to security. NASA could have a similar program to encourage developers to work on issues that are important to NASA.

Issue #5: Government Restrictions

How do we mesh open source software with decidedly un-open policies such as International Traffic in Arms Regulation (ITAR)?

Proposed Solutions:

1. Clearly describe applicable legalities

To assist the public with understanding applicable government regulations and/or contractual obligations, each project can write a document that outlines the legal issues that affect the project.

2. **Specify who owns what**

Trademarks, intellectual resources, copyrights, patents... Which organization or person owns which?

Issue #6: Limitations on Contributing to External Open Source Projects

What are the differences between contributing minimal, incremental improvements or bug fixes and new features (as per NPR 2210.01)? And, who makes this decision? What lessons learned / best practices can be drawn from current NASA open source development “pathfinder” projects (and can these be applied “generally”)? What should be the policy of NASA personnel contributing in their off-hours? How do you handle / treat situations where people work off-hours on things that derive directly, or are inspired, by what they worked on during duty hours?

Proposed Solutions:

1. **Clarify existing policy and/or restrictions on contributing to open source projects**

Given the issues stated in the description and a general sense of confusion, and a prevailing attitude of “asking forgiveness is easier than asking permission,” NASA needs to clarify what the existing policy means for software developers who want to contribute to open source software projects that may or may not also be used within NASA. Publish this information prominently online and make it widely available. Simple, unambiguous language without a lot of caveats is critical.

2. **Agency-wide blanket authorization covering contributions to external open source software**

The agency could create a blanket agreement permitting contributions to open source projects. This would be greatly preferable to having one-off contributor agreements for individual projects/contributions. This may require an incremental implementation, given current contract language, contractor agreements, etc.

Issue #7: How does Open Source Governance look within NASA?

Currently, there is a lack of information and awareness toward licensing, legal issues, and activity in the open source software community at NASA. How does one receive guidance on open source contributions? What does the process look like?

Proposed Solutions:

1. **Create a single point of contact for open source information/governance**

There should exist one central point of contact and process to initiate an open source project at NASA. Should be similar to 508 or ITAR reps. The process should be short and non-legalese, and

should make sense.

2. Create an open source “review board” (community of practice)

Create a group of subject matter experts who are not responsible for reviewing every line of code, but who can act as resources for answering questions, recommending policy guidelines, etc. Should have technical, legal, and policy knowledge (IBM uses 1/3 attorneys, 1/3 technical, 1/3 management).

3. Limit approval time to 1 week (on average)

Develop a new structure to run with five pilot projects to walk through the new process and try it out. It doesn't have to be perfect.

Issue #8: How Should Open Source Efforts be Supported?

NASA needs to develop cooperative support into project structure

- Project Level - permitting code deemed outside the purview of ITAR/Export Administration Regulations (EAR) to be open source
- Budget Level - allowing for hiring of floating talent as temporary staff augmentation
- Organization Level - designing organization to support habits and practices of open source development

Proposed Solutions:

1. Assist with ways to get financial support

Provide directory of possible resources, including:

- a. Consortium/Forum membership
- b. Standards groups
- c. Identifying sponsors

2. Offer non-financial resources

Provide various support mechanisms, including:

- a. Unit testing
- b. Documentation
- c. Bug fixes
- d. Security validation

3. Convene conversations with organization experts on the appropriate management and org structures for supporting open source approaches

To sustain open source policy and practices over the long run, we need to think through the organizational and institutional structures that support this type of cooperative interactions. Material exists on how to run a successful open source project and we are generally discussing

open source policy. However, projects and policy exist within an organizational framework and open source projects within government must happen within the agency's legacy organizational structures.

Issue #9: How does NASA Open Source *Everything?*

Proposed Solutions:

1. Implement a "Default Open Source" duration on code

There is a growing incentive to open source everything. Is this the right path to take? How does this fit into ITAR issues? Moving to data-driven approaches allows the software to be generic and exportable, and the quality of the data loaded into it determines whether it is ITAR. Open source projects are only useful where code can be generalized to solve many problems for many people.

Small-but-useful tools not developed in accordance to NPR 7150.2A are probably not available for any kind of public release. Should these be considered as well?

2. NASA open source software repositories

NASA could make use of two agency-wide source-code repositories (GitHub or something similar). Allow the "NASA-wide" open sourcing of projects, including potential ITAR/EAR projects, to be shared on this repository. The external (publicly accessible) repository would be a home for those projects that contain no ITAR/EAR issues. In conjunction with Proposed Solution #1, non ITAR/EAR NASA software could graduate to this location after its "Open Source Duration" is reached.

3. Establish requirements prior to start of project

Proposed software projects should describe the proposed functionality and requirements so that a preliminary ITAR review can occur. ITAR issues can be foreseen based on proposed functionality.

4. New projects should declare their license and get contributor agreements signed

When a new project starts, it should be declared something that can (or cannot) be shared openly; if not, proper justification for closing should be provided and documented, ideally with ways of how to make the project open-able (e.g., componentize into open and closed portions).

5. Acceptance of non-licensed community code

Much software is openly developed and freely shared by the scientific community. Unfortunately this software is often not released under any license. This requires NASA to obtain copyright releases from everyone involved in its development in order to redistribute it as part of a NASA project release. Determining who has touched this community code is often very difficult. If the community is freely distributing code without regard for copyright and licensing issues, then NASA should be allowed to follow suit.

Issue #10: How to Close the Feedback Loop Between Policy Makers, Developers and End Users?

How do we ensure that draft policies have enough eyes on them, particularly from the people who may be most affected or who have the most detailed knowledge of those areas, and who can thus best understand the implications? Policies should not put anyone in the position of performing the mission by violating the policy, or adhering to the policy and thereby reducing the effectiveness or inducing the failure of the mission. Many times, our policies derive (or are simply copied) from Federal law, regulation, guidance or, more commonly, from the policies of other agencies. Effective feedback provides opportunities to modify or adjust policy based on practical, realistic feedback. Are we taking advantage of these flexibilities?

Potential Solutions:

1. Experiment with open policy reviews online

Instead of performing relatively closed reviews of draft policy, place draft policies online where each individual paragraph of the draft policy can be commented on by anyone within NASA, and perhaps those outside of NASA. Those comments can then be responded to online and explanations given for why a particular comment was accepted, rejected or rewritten. This would allow those for whom a policy will affect to provide accurate, more relevant and useful feedback on implications and issues with each portion of a draft policy before it becomes cast in stone.

2. Reverse the policy development process

Policy written at the agency level may often be written by those with little or no direct experience in the areas where the new policy will apply. Where possible, have communities that will be affected develop the policy, and have the Agency organizations that would normally write the policy serve as reviewers of it. This may not work for all policies, but for some would it be possible to provide the purpose of a new policy, the laws and regulations that constrain the scope or reach of the new policy, and allow the community to develop the new policy to meet those requirements? Doing so would likely reduce bad policies that create unnecessary obstacles and waste time and resources in working around them. Things we would need to determine first:

- Who writes current policies within NASA for Information Technology (IT) and other areas?
- What is the process for reviewing draft policies before they are implemented?
- Who reviews draft policies, and are they versed in the area to which the policy applies?
- How does someone get access to the comments that are fed back during the review process?
- How do we know that draft policies are receiving thorough reviews given that everyone seems oversubscribed already?
- Is there a mechanism for anyone within NASA to provide feedback to draft policies?
- How does someone who provides feedback know that the feedback is being looked at?
- Are there cultural barriers to providing feedback?
- What are the organizational barriers to providing feedback?
- How to track that each section has been reviewed by appropriate stakeholders?

- How often are policies updated? What constitutes the need to update a policy (what is the policy for updating a policy)?

Issue #11: How to Encourage Cultural Change in Hiring Practices?

How can NASA attract more open source-savvy people in a world where companies like Red Hat and Google offer careers that encourage such participation? NASA is competing against these companies for the same skills.

Potential Solutions:

1. Highlight NASA's ongoing open source participation

Do a better job of marketing NASA's current OSS participation

- Make 'Heroes and Rockstars' out of NASA personnel working in OSS
- Get these people out on OSS speaking circuits (evangelists)

2. Streamline the process to participate in external OSS communities

Allow a 20% time for individuals to contribute to/participate in OSS communities on NASA's behalf

3. Proposed Solution #3: Allow for remote/distributed work

Don't limit your search for OSS talent to just locations/centers that NASA has a physical presence in. Tap into university programs and other academic institutions, etc.

Issue #12: How to Package Open Source Software to be More Accessible

Collaboration on open source software is dependent on others who find the software useful. The barrier for adoption of OSS must be kept low. This also prevents the project from dying on the vine. Is there a way we can package OSS to make it easy for others to try and adapt to their needs?

Proposed Solutions:

1. Develop generic Application Programming Interfaces (APIs)

2. Market the software such that it addresses a "general" need. For example, "commodity services" – a wiki, an email system, a content management system

Marketing of the OSS should not be strictly through the perspective of the specific application. Present scenarios for wider application. May want to look at dividing existing software packages into useful tools; e.g., break down something like World Wind into its modules and then combine modules into logically separate projects. Encourage reuse and visibility by cataloging existing projects such as <http://ti.arc.nasa.gov/opensource/projects/>.

3. Leverage cloud computing and virtual memory images

Create "turnkey" systems that demo OSS in a way that will speak to larger community needs. Include extra support material targeted to needs of the Federal Government such as security documentation for Federal Information Security Management Act (FISMA) conformance. Add security features that would make these work both inside and outside the NASA environment.

- 4. Package so software can be distributed by a third party that has a reputation**
Standardization improves trust in the product. Providing packaging specification (such as RPM spec files) would also help.
- 5. Distribute software binary executables compatible with a specific software stack**
Packaging binaries compatible with a specific software stack could increase adoption; however, still having a reliable path to build from source code in a repeatable process is needed.
- 6. Provide simple, accessible documentation on the software and its use**
Good open source software systems have a common set of basic documentation that includes some features:
 - system requirements and constraints
 - installation instructions
 - tutorial introduction
 - a way to download the source code

Issue #13: Combining Open Source Software Development Standards with Office of the Chief Engineer Policies

NASA Procedural Requirement (NPR) 7150.2A applies to all software development by and for NASA. In many cases, the mapping of NPR 7150.2A requirement to the open source development processes is apparent. However, the mapping is not always obvious.

Proposed Solutions:

- 1. Ensure there is a dialog between open source community and Office of the Chief Engineer (OCE) regarding mapping NPR 7150.2A to open source development process.**

Unanswered Issues:

The following questions were topics considered at the OSS that did not receive meaningful discussion or responses:

1. How do we offer project managers guidance regarding the best open source models for their projects?
2. What are the obstacles to two-way sharing of information?
3. How to untangle licensing regulations?
4. Different applications of open source licensing (i.e., research vs. ops)?
5. Open data: how to release more data and make it part of how we do business

Appendix

Registered Participant List

Registered In-Person Attendees

Name		Title	Organization
Arash	Aghevli		
Avinash	Agrawal	Director, Open Innovation	Seti Institute
Andrew	Aitken	Gm & Svp	Olliance Group, A Black Duck Company
Jesse	Andrews		NASA Ames - Dell
Javier	Barreiro	Software Engineer	NASA Ames - Sgt Inc.
Greg	Barrett	Geospatial Specialist	Independent Consultant
David	Bell	Director, Usra Research Institute For Advanced Com	NASA Ames - Sgt Inc.
Genesis	Berlanga	Nlsi Intern	NASA Ames Research Center
Ross	Beyeer	Research Scientist	Sagan Center (SETI) And NASA Ames Research Center
Zaheda	Bhorat		
Gary	Borda	Agency Counsel For Intellectual Property, Ogc	NASA HQ
Chris	Boshuizen	Space Mission Architect	NASA Ames
Fred	Bourgeois		Team Frednet
Deborah	Bryant	Public Sector Communities Manager	Oregon State University Open Source Lab
Richard	Bullington-McGuire	Director, Technology	Three Pillar Global, Inc.
Bob	Burbach	Founder, Open Source Developer	Critical Juncture, LLC
Heather	Burke	Program Manager	Space And Naval Warfare Systems Center Atlantic (Ssc Lant)
Matthew	Burton	Technical Architect	Consumer Financial Protection Bureau
Steven	Carmine	Federal Account Rep.	Red Hat
Edward	Chan	Task Area Manager, NASA Soc	NASA Ames Dell Services, Federal Government
Soo	Choi		NASA Ames - Dell/Anso Labs, LLC

Op	Choudhary	President	Opal Soft
Cyrus	Chow		
Kim	Chrestenson	Ames Software Release Coordinator	Deltha/Critique
Bob	Ciotti	Supercomputing Systems Lead	NASA Ames Tne
Yvonne	Clearwater	New Media Innovation Leader	NASA/Ames/Nmit
Keith	Cowing		Spaceref Interactive
Diana	Cox	Patent Attorney	NASA Ames
Jason	Crusan		NASA HQ
Linda	Cureton	Chief Information Officer	NASA
Joshua	Davis	Manager, Open Technology & Outreach	Georgia Tech Research Institute
Martha	Del Alto	Software Release	NASA
Bosco	Dias	Software Engineer	SAIC
Christopher	Dibona	Open Source Programs Manager	Google
Estelle	Dodson		NASA Ames - Lm
Aaron	Duley		NASA
Casey	Dunn		Czwx LLC
Greg	Elin		Federal Communications Commission
Stuart	Engelhardt	Software Developer	NASA JSC - Tietronix
James	Farmer	Director	Sigma Systems Inc.
Pascal	Finette	Director Of Mozilla Labs	Mozilla Corporation
Jane	Finette	Director Of Global User Engagement	Mozilla Corporation
Elizabeth	Foughty	It Project Coordinator	MCT Inc, NASA Arc
Jennifer	Fung	Principal Se	Sgi
Nicolas	Garcia Belmonte	Senior Software Architect	Sencha Labs
Bryan	Geurts	Chief Patent Counsel	NASA GSFC
Richard	Golding		NASA Ames - Ktsi
Lester	Gong		

Lon	Gowen	Associate Director For Innovation	NASA GSFC
Justin	Gray	Aerospace Engineer	NASA Glenn Research Center
Saugata	Guha	Senior Software Engineer	Uarc/NASA
Guru	Guruswamy	Sr. Scientist	NASA Advanced Supercomputing Division
Matthew	Hancher	Senior Software Engineer, Special Projects	Google
Gunnar	Hellekson	Chief Technology Strategist For Us Public Sector G	Red Hat
Sarah	Hobart	Dep. Mission Systems Manager	NASA
Patrick	Hogan	World Wind Project Manager	NASA Ames
Julie	Holland	Partnership Development Manager	Dryden Ipp Office
Jamie	Hoover		Yummly / Ninja Ui
Paula	Hunter	Executive Director	Outercurve Foundation
Phillip	Hyett	COO	GitHub
Jeetendra	Jagasia	Architect	Microsoft Corporation
Jean-Marie	Jean-Pierre	Code 750 End-User Services Manager	NASA GSFC
John	Kelly	Oce, Program Executive For Software Engineering	NASA HQ
Jim	Kerevala	Coo	Ubm
Taemin	Kim	NASA Postdoctoral Fellow	NASA Ames Research Center
Hyeongwoo	Kim	Visiting Researcher	NASA Ames Research Center
Christopher	Koenig	Software Engineer	Microsoft Corp.
Amanda	Koenig	Attorney-Advisor, Commercial And IP Law	NASA HQ
Paul	Kolano	Software Engineer, Advanced Supercomputing Divisio	NASA Ames - Computer Sciences Corporation
Eric	Kutner	Consultant	Erdg
Rick	Kwan	Computer Scientist	American Academy Of Aeronautics
Tsengdar	Lee	(Acting) CTO for IT	NASA
David	Lees	Senior Project Scientist	NASA Ames/Cmu Silicon Valley

Creon	Levit		
Martin	Levy	Director, Ipv6 Strategy	Hurricane Electric
Brian	Lewis		NASA Ames
Aaron	Lippold		
Ali	Llewellyn	NASA Open Government Initiative	NASA JSC - Valador
Remy	Loubradou	Student Computer Science	Supinfo
Mike	Lundy		NASA Ames - Sgt Inc.
Mike	Lundy		
Rachel	Lunsford	Special Assistant To The Chief Technology Officer	Department Of Veterans Affairs
Jeff	Luszcz	Founder	Palamida, Inc.
Will	Marshall	NASA Ames - Affiliate	
Guy	Martin	Community Manager At Forge.Mil	Collabnet, Inc.
Chris	Mattmann		NASA JPL - Affiliate
Paul	Mcmillan	Cto	Zero Coordinate
Alfred	Mecum	Technology Manager	GSFC Ippo
Cristina	Milesi	Research Scientist	Csumb/NASA Ames
Eugene	Miya		NASA Ames
Patrick	Moran	Computer Scientist, Advanced Supercomputing Divisi	NASA Ames
Scott	Murman	Code Tnf	NASA Ames
Gwyn	Murray	Founder And Principal	Matau Legal Group
James	Neushul	Amphibious Communication Officer	Us Third Fleet
Ray	O'Brien	CTO for IT	NASA Ames
Ryan	Ozimek	President	Joomla (Open Source Matters)
Muni	P	Technical Architect	Opal Soft
Rob Padilla	Padilla	Chief Patent Counsel	NASA Ames
Steve	Parker	V.P. Technology Research And Projects	Energy Sector Security Consortium, Inc. (Energysec)

Tiago	Pereira	Post-Doctoral Fellow	NASA ARC
Ab	Periasamy	CTO	Gluster
Tom	Preston-Werner	CTO	GitHub
Margarita	Quihuis	Director, Peace Innovation Lab	Stanford
Jon	Richards	Software Engineer	The Seti Institute
Peter	Robinson	Computer Scientist	NASA
Enidia	Santiago	Technology Manager	GSFC Ippo
Edward	Scharff	Research Engineer	Sgt
Robbie	Schlinger		NASA
Andrea	Schneider	Director	Opengov In Action
John	Scott	Mil-Oss Founder / Open Technology Lead / Sr. Syste	Radiantblue Technologies, Inc.
Sharad	Sharma	Director, Operations & Business Development	Opalsoft, Inc.
Sandeep	Shetye	Information Architect	NASA
Michael	Sims		NASA Ames
Amalie	Sinclair	Administrator	Space For Progress
Ryan	Singer	Researcher	Stanford Peace Innovation Lab
James	Snow	Software Engineer	NASA
Harlan	Steen	Project Director	Network Time Protocol
Brian	Stevens	Cto And Vice President, Worldwide Engineering	Red Hat
Martin	Still		NASA Ames - Baeri
Ian	Struken	Deputy Manager/Service Office Integration Lead, Oc	NASA Ames
Sunny	Sundstrom	Product Manager	Sgi
Bob	Sutor	Vice President, Open Systems Strategy	IBM
Andy	Tainter	Senior Consultant	U.S. Department Of State
Joshua	To	Director	Soup

J.J.	Toothman		NASA Ames - Dell
Tom	Trainer		Gluster
Philip	Tsao		
Christopher	Tucker	Information Technology Director	Laane
Bill	Van Dalsem	Systems & Software Lead Discipline Engineer, Ames	NASA Ames
Alexander	Van Dijk	Web Lead Flight Opportunities Program	MCT @ NASA Ames Research Center
Alonso	Vera	Chief (Acting), Human Systems Integration Division	NASA Ames - Sgt Inc.
Eric	Wahl		
Jeffrey	Walpole	Ceo	Phase2 Technology, LLC
Chris	Wanstrath	CEO	GitHub
Tony	Wasserman	Professor Of Software Management Practice And Exec	Carnegie Mellon Silicon Valley
Phil	Webster	Chief Cisto	NASA GSFC
Heather	Wellington		Gluster
David	Wheeler		Institute For Defense Analysis
James	Williams	CIO	NASA Ames
Michael	Williams		
Patti	Yamakido	Account Exec	Sgi
Sara	Yap	Development Consultant	Soup, Inc. / Mozilla
Sara	Yap		
Gil	Yehuda	Director Of Open Source	Yahoo! Inc.
Alana	Zimmer	New Media Intern	NASA/Ames/Nmit
Robert	Zimmerman		Symbiotek Systems

Registered Remote Attendees

Name		Title	Organization
Avinash	Agrawal	Director, Open Innovation	Seti Institute
Ahmed	Akour	Student	Jordan Open Source Association
Nouf	Aljalaud	Computer Engineering Student	Prince Mohammad Bin Fahd University
Stuart	Anderson	President	Netsweng, Llc
Emma	Antunes	Web Manager	NASA
John	Archdeacon	Project Lead Engineer	Th
Michael	Arida	Sr. Programmer/Analyst	Adnet
Chad	Armond	Software Engineer	Db Consulting
Jesse	Ash	Software Engineer	Synapse Information Resources
Sohail	Aslam	Software Developer	Consultant, NASA HQ
Ditjon	Baboci	Architect	Studio Aa
Ben	Bac	System Administrator	Benbac
Darrell	Bailey	Technical Assistant	NASA/Msfc/Es50
Uri	Barkan		
William Chiquito	Barreto	Barreto	Ica2 Innovación Y Tecnología
Merlin	Barschke		
Jean-Francois	Barthe	Systems Engineer	San Mateo County
Marco	Battistoni	Architect	Unisys Belgium
Mary	Bauer	Customer Service Rep	Contact Centers Of America
Guy	Bedette	Kennedy Space Center	NASA
James	Bee		
Reid	Beels		Open Source Bridge
Steve	Belvin		Honeywell
Alfredo	Bencomo	TI	Sgt

Andrew	Benson	Senior Research Fellow	Caltech
Bachir	Benyami	System Administrator	Benbac
Sarah	Berman	Sociologist	Unt
César	Bernal		Esa
Steve	Berrick	Earth Science Data Systems	NASA - Hq
Stephen	Berrick		
Thomas	Berry	Software Engineer	NASA JPL
Nancy	Bertolino	Counsel & President	Sml Professional Services Llc
Erna Giles	Beverly	Enterprise Applications Service Executive	NASA HQ
Colin	Bidwell	Aerospace Engineer	NASA Glenn/ Icing Branch
Conrad	Bielski	Researcher	Joint Research Centre Of The European Commission
Ralph	Bischof	Ipam Engineer	SAIC/Unites
Nicole	Bishop	Analyst	NASA
Brian	Black	Robotics Alliance Project	NASA
Steven	Bliss	Human Resources Information Systems	NASA Shared Services Center
Scott	Blomquist	Systems Administrator	N/A
Andrew	Bodker	It Specialist	Chicago Public Media
Randy	Bolanos	Nisn Csr	Unites
Artus	Bolzanni		
Carolyne	Borel		
David	Bovill	Director	Opn Technologies Llp
Meagan	Brant		
William	Brantley	Hr Specialist (It)	Opm
Martin	Braun		
T Hall	Brehme		
Marty	Brenner	Aerospace Engineer	NASA Dryden Frc

Linda	Brewster	Branch Chief	NASA/Msfc Es53
Bill	Brod	Experimental Fac Dev Engr	NASA HQ
Oscar	Brooks	Chief, System Software Engineering	NASA, Ksc
Lee	Brotzman	Ims Architect	NASA Soc
Pam	Broviak	City Engineer/Asst Dpw	City Of Geneva
Christopher	Brown	Structures Analysis Lead	NASA
Gregory	Brown	It Specialist	NASA
Darline	Brown	Technical It Security Manager	NASA/Hq Itcd
Robert	Bruchs	Consulting Software Engineer	A.I. Solutions (NASA Ksc Esc)
Stephen	Buckley	CIO	U.S.Transparency
Lori	Budin		Gluster
Phillip	Bui	Senior Software Engineer	Usa
Lori	Burke	Software Developer	A.I. Solutions, Inc.
Alexander	Burke		
Matt	Burkhardt	Owner	Impari Systems, Inc.
Paul	Burr	Senior Technology Systems Developer	Washoe County Technical Services
Chris	Burroughs		NASA/SOMD
Roland	Burton		
Johnny	Busby		
Michelle	Cadieux	Space Educator	Creative Communications /
David	Calderón		
Ricky	Campbell	Aerospace Engineer	NASA Marshall Space Flight Center
Louis	Cancino	Information Architect	Johnson Space Center
Guillermo Rodríguez	Cano		
Chuxiao	Cao		
William	Carlin	Computer Specialist	Washoe County Library System

Christopher	Carmichael	Ssc Deputy Cto	NASA
Bruce	Caron	Executive Director	The New Media Studio
Cory	Casazza	Chief Information Management Officer	Washoe County Technical Services
Sarah	Caufield		
Peter	Cauwels	Esb Lob Manager	NASA / Apexio
Ravi	Chandra	Sv	Honeywell Technology Solutions Ltd.
Jessica	Charlesworth	Designer	Self-Employed
Beau	Charvet	It Specialist	NASA
Op	Choudhary	President	Opal Soft
Cyrus	Chow		
Trisha	Christensen	Web & Marketing Consultant	Dharma Creative
Emmanuel	Christophe		
Bob	Ciotti	Supercomputing Systems Lead	NASA Ames Tne
Matthew	Clark		Honeywell Aerospace
Brian	Claywell	Graduate Research Assistant	Center For Geospatial Intelligence, University Of Missouri - Columbia
Tom	Clune	Senior Computational Scientist	NASA - GSFC
James	Cochran	Forms Designer	Db Consulting Group
Tom	Cochrane		
Michael	Cohen	Software Lead	NASA Ames (Jacobs Asg Contractor)
Doug	Coleman		Dougcoleman
James	Comstock	Aerospace Technologist	NASA Langley Res Ctr
Casey	Conaghan	Consultant	Ops International
Max	Connelly	Computer Technician	Washoe County Library System
Alberto	Conti	Archive Scientist	Space Telescope Science Institute
Christopher	Cook	It Systems And Security Director	Inuteq Llc
Shawnee	Cook	Principal	Zentu, Llc

David	Cox	Senior Research Engineer	NASA Langley/ D316
Wayne	Crane		
John	Crockett	Library Assistant li	Washoe County Library System
Alan	Cudmore	Computer Engineer	NASA / Goddard Space Flight Center
Angela	Czupta	Secretary	Innovative Health Applications
Mike	Dalal	Computer Scientist	Sgt, Inc. At NASA Ames
Mat	Dagleish		University Of Wolverhampton
Saurabh	Das		
Arun	Datta	Director, Technology R & D (Nucri)	National University
David	Davis	Risk Manager	Dynetics Technical Services/MSFC Mits
Jerry	Davis	Deputy Assistant Secretary, Info Sec	Us Department Of Veterans Affairs
Galen	Debord	Coordinator	Missouri Ubuntu Loco Team
Martha	Del	Software Release	NASA
Tishampati	Dhar	Software Engineer	Csro Cmar
Kimberly	Diorio	Senior Aerospace Engineer	SAIC
Portia	Dischinger	It Project Manager	NASA Msfc
Ridley	Disiena	Emerging Technology Integration Engineer	NASA / Etads / Icam Engineering / Db Consulting
Kevin	Divico	Entrepreneur	
John	Dockendorf		Harris Corp
Visal	Doek	Core Planning Engineer	Camgsm Company Limited
Jessie	Dotson	Astrophysicist	NASA Ames Research Center / Ssa
Faith	Dow		
Bob	Downing	Engineering Manager	Arcata Associates
Prateek	Dubey	Technical Specialist	Zensar Technologies Inc
Brian	Duggan	Scientific Programmer	Adnet Systems
Leslie	Dugger	I.T. Associate	Laane
Casey	Dunn		Czwx Llc

Robin	Edwards		
Priscilla Chang	Eines	Web Administrator	Magnet Media Norway
William	Eisenhauer	National Director Of Veterans Engineering Resource	Department Of Veterans Affairs - Vha
Sheri	Elgin	Itsd Technician	Jr Simplot Co
Ehab	Elhanafy	Freelance Translator	Elhanfy Limited
Drew	Elliott		
Michael	Enescu	Cto Open Source	Cisco Systems
David	Engelbrecht	Systems Engineer	Dell Federal Services/NASA Ames Code I
Betsy	Ensley	Marketing Manager	Phase2 Technology
Josh	Ermentrout		
Hassan	Eslami	Deputy Dir. Tech Area Mgr	Uarc
Jeff	Estefan	Division Technologist	Jet Propulsion Laboratory
John	Evans		Global Science & Technology, Inc.
Dave	Eveland	Managing Librarian	Washoe County Library System
Matt	Everingham	Aerospace Engineer	Nlsi
Mark	Faine	Software Engineer	Dts
Rob	Falck	Aerospace Engineer	NASA Glenn Research Center
James	Farmer	Researcher	Instructional Media + Magic Inc.
Brandon	Farmer	Programmer / Analyst	Mssu
John	Ferrell	Principal	Clearly Media
Marcus	Fisher		GSFC/Iv&V
James	Flanagan		
Cecelia	Fletcher	Management And Program Analyst	NASA Langley Research Center
Karl	Fogel	Open Civics Development Specialist	O'Reilly Media, Inc.
Richard	Fong		Govdelivery
James	Francis	Software Engineer	Aurora Flight Sciences
Robin	Friedrich	Project Engineer	United Space Alliance

Todd	Fries		Free Daemon Consulting, Llc
Thomas	Gaeng	It Manager	Adnet Systems
David	Galiel	Executive Director	Beyond War
Manuel	Gamelas		
Victor	Gandarillas		
Gary	Gapinski	Senior System Engineer	NASA Glenn Research Center — Db Consulting Group, Inc.
Sumedha	Garud	Marketing/Outreach	Simlabs At NASA Ames
Richard	Gaskin		Fourth World
Mark	Gayler	Open Software Evangelist	Microsoft
David	Gaylor		
James	Geiger		
Stephan	Gerard		
Jeffrey	Gilbert	Software Lead	Db Consulting / NASA
Sean	Gill		
Sebastian	Gillen	Equipment Technician	Jacob Burns Film Center
Kim	Gillies	Software Architect	Thirty Meter Telescope Corporation
Mark	Gisi	Sr. IP Manager	Wind River Systems
Jeremy	Glassenberg		
Todd	Goatley	Computer Technician	Washoe County Library System
Viktor	Godard	Sr Linux Engineer	
Stephane	Goldstein	Student	
Daniel	Golombek	Chief Of Strategic Communications	Space Telescope Science Institute
Lester	Gong	Lead Associate	Booz Allen Hamilton
Lester	Gong		
Jose Manuel	Gonzalez	Systems Administrator	
Kathleen	Gordon	Systems Engineer	SAIC / Is40
Aman	Goyal	Software Developer	Reflexis Systems

Tracy	Graham	Systems Administrator	Tbe/Msfc Poif
Patrick	Grayson		
Scott	Gries		
Tony	Grimshaw	Information Security Architect	
Manuel	Grizonnet		French Space Agency (Cnes)
Rebs	Guarina	Systems Developer	N/A
Sreedhar	Gudur	Software Engineer	Zensar Technologies Ltd
Arthur	Guest		
Saugata	Guha	Senior Software Engineer	Uarc/NASA
Piyush	Gupta	Programmer	Accenture
Aman	Gupta		
Guru	Guruswamy	Sr. Scientist	NASA Advanced Supercomputing Division
David	Guterrez	Solution Architect	Centauri
Sandy	Gutheinz	Project Manager	JPL
Valerie	Gwaltney	Senior System Administrator	Ssai
Therese	Haar	Sgt	NASA/Goes-R
Jon	Hamkins	Supervisor	JPL
Marc	Handelman	Information Security Officer/Research	Infosecurity.Us
Andy	Harb		
Brian	Hargrave	Systems Analyst	Mits
Kazuo	Hayashida	Electronics Engineer	NASA/Msfc
April	Hayman		
Phillip	Hebert	Lead, Software Engineering	NASA / Ssc
Richard	Hecker		
Ryan	Heggem		Hopper Cosmic & Company
Timothy	Hemphill	Glide Developer	Asrc
Joel	Henry	Engineer	NASA

Sam	Henry	It Security Analyst	NASA/Jsc
Jim	Henze		
Tom	Higgins		
Evelyn	Hill	Software Release Authority	NASA/Msfc
Trudy	Hill		
Nancy	Hine	Projects/Payload Coordinator	Ukseds/MyPocketqub
Randall	Hirsch	Attorney/Physician	
Akira	Hiyama		
Alex	Hocking	Student	Open University
Terry	Hodgson	Sr Software Engineer	Meit / NASA
Erich	Hohman	Technology Network Engineer Ii	Washoe County Technical Services
Vera	Holman	Managing Principal	V_Lan Technologies
Corbin	Holtz	Systems Engineer	Lockheed Martin/Jsc/Mod
David	Hoppe	Solution Architect	Oh4
Mack	Houston		
James	Howard	Senior Research Systems Analyst	Board Of Governors Of The Federal Reserve System
Carrie	Howard	It Manager	Washoe County Technical Services
Chunyah	Hsu		
Sara	Hunter	Office Manager	Metis Technology Solutions, Inc
Paul	Hunter	Cto For It	NASA - GSFC
Paul	Hunter	Cto	NASA Goddard Sfc
Alan	Hylton	Software Engineer	NASA Glenn/Communication Technology
Dennis	Iannicca	Software Engineer	NASA Glenn/Communication Technology
Luis	Ibanez	Technical Leader	Kitware Inc.
Chaudhry	Ibrahim		Ics Nett
Daniel	Ihonzbere	Ceo	Tech Prognosis
Joseph	Isach	Software Engineer	NASA Glenn/Communication Technology

Mashru	Islam	Msc Student (Dept. Of Eee)	Khulna University Of Engineering Technology
Marin	Ivan	Researcher	Is Geo
Will	Ivancic	Senior Research Engineer	NASA Glenn/Communication Technology
Carlos	Ize	Sr. Engineer	American Airlines
Sheila	Jackson	Human Resources Development Specialist	NASA
Bruce	Jackson	Senior Research Engineer	NASA Langley Research Center
Andrew	Jalics	Flight Software Engineer	NASA Grc Qinetiq North America Contractor
Mark	James	Software Release Authority	NASA/JPL
Jinyoung	Jang	Ceo	Uenginesolutions
Erik	Johnson	Research Associate	Moves Institute, Naval Postgraduate School
Kirk	Johnson	Information Assurance Delivery Manager	SAIC / Neacc Msvc
Michael	Johnson		
Jeff	Johnstone	Retired	Open Source Space Ca
Ellen	Jones	Software Engineer	Dtsi
Patricia	Jones	Deputy Director, Expl Technology	NASA Ames Research Center
Brian	Jones		
Jim	Justen	Assoc Editor	
Elia	Kabanov	Founder	Metkere.Com
Sharmin	Kamal	Technology Systems Developer I	Washoe County Technical Services
Amy	Kaminski		
Nancy	Keener	Systems & Services Access Librarian	Washoe County Library System
Paul	Keller	Research Engineer	NASA Ames
Keith	Keller	Associate Chief, Strategy & Planning It	NASA Goddard Space Flight Center
Megan	Kelly		Gscnc
Sean	Kelly	Technologist	JPL

Brian	Kennedy		
Jim	Keravala		
Ellen	Keulemans		Escg - Mei Technologies / JSC - Ev
Alvin	Khaled		
Robert	Kilker	Director Of Federal Operations	Talend
Stephen	King	Esmd Ice Pdm/Plm Service Manager	Dki Consulting/ Esmd Ice
Roy	King	Technology Systems Administrator Iior I	Washoe County Technical Services
Morgan	King	Cip Auditor	Wecc
Jon	Kocen	It Professional	Ares/Oh4
Joe	Kochocki	Principle Member Tech Staff	Draper Laboratory
Christopher	Koenig	Software Engineer	Microsoft Corp.
Markus	Korn		
Stefanos	Koutsoutos		
Aleksandar	Kracun		
Kevin	Kreitman	Senior Engineering Specialist	Aerospace Corporation/Computers And Software Division
Bhagwat	Kshirsagar	Professor	Mit
Iwanka	Kultschyckyj		
Kieran	Kunhya	Ceo	Open Broadcast Systems
Eric	Kutner		
Jose	Lagares	Principal Software Engineer	A.I. Solutions (NASA Ksc Esc)
Amit	Lakhanpal		
Chatwin	Lansdowne	Electronics Engineer	NASA - JSC - Ev611
Joe	Latone		Ibm
Carsten	Lawrenz	Application Engineer	Ptc
Anh	Le		
Mike	Ledbetter	Principal Investigator / Manager	MSFC Es61 / Davidson Technologies, Inc.

Clarissa	Lee	Graduate Student/Graduate Instructor	Duke University
Henry	Lee		Government Of British Columbia
Rebecca	Lee		Grant Thornton Llp
Lucia	Lee	Senior Computer Scientist	Ssai/NASA Larc
Brett	Lewinski	Technical Area Expert	Indyne / NASA
Edward	Lewinson		
Tristan	Li		
Roger	Liang	Software Release Authority	NASA Kennedy Space Center
Teresa	Liao		
Evan	Lim	Consulting Engineer	SAIC
Greg	Lind		Govhub
Van	Lindberg		
Dennis	Linders	Phd Student	University Of Maryland
Francis	Lindsay		Esdis
Jeff	Lohrmann	Director, Sales And Marketing	Advanced Computer Consulting
Chris	Long	Senior Technology Network Engineer	Washoe County Technical Services
Remy	Loubradou	Student	Supinfo
Filipe	Lourenço		
Greta	Lowe	It Project Manager	NASA
Roberto	Lozano	Astronomer And Scientific Difusser	Astronomical Observatory Giefa
Mark	Luboski	Software Assurance Engineer	Ares Corporation
Chris	Lucas		
Franklin	Lue	Group Leader	Drdc Toronto
James	Lux	Task Manager	Jet Propulsion Laboratory
Mark	Lyon	Adv Sw Group Leader	Draper Laboratory
Cosimo	Macafore	Software Developer	Saitech / Nssc
Jeffrey	Macdonald	It	Ksc-Lx

Gail	Macdonald	Sr System Support Engineer	Qinetiq Na
Ana Ligia	Machado	Advertising	Tam
Bob	Mader	Vp Linux	Citigroup/Platform Engineering
Dhruv	Mahajan	Student	
Issa	Mahasneh	President	Jordan Open Source Association
Kevin	Mahoney	Senior Developer	Toolhouse
Richard	Mains	Senior Scientist	Mains Associates
Saurabh	Mallik	Marketing Manager	Pushtotest
Joseph	Mangan	Ceo	Coanda Aerospace Llc U.S.A. / Coanda Aerospace Software Gmbh Austria Europe
James	Mantovani	Physicist	NASA Kennedy Space Center
Jane	Maples	It Specialist	NASA/Neacc
Lawrence	Markosian	Computer Scientist	Sgt, Inc.
Jason	Maron	Associate Health Fellow	New England Veterans Affairs Resource Center
Jane	Marquart	Computer Specialist	NASA/GSFC
James	Marshall	Senior Staff Scientist	Innovim / NASA Goddard Space Flight Center
Sam	Martin		I360Technologies, Inc.
Victor Hugo	Martín	Planning	Davi
Wayne	Matthews		Escg/Estl
Chris	Matthews	E-Government Information Officer	Washoe County
Mary	Matthews	Director/Producer	
Arnie	Maurins	Director Washoe County Library System	Washoe County Library System
Tom	McBride	Patent Attorney	NASA Larc
Neal	McBurnett	Founder/Developer	Electionaudits Open Source Project
David	McComas	Gpm Flight Software Product Lead	NASA Goddard Space Flight Center
Perry	McDowell	Executive Director	Delta3D

James	McGroary	Chief Patent Counsel	MSFC NASA
Dawn	McIntosh		Darpa/NASA Ames
Edward	McLarney	Chief Technology Officer	NASA Larc Ocio
Fred	McMullen	Web 2.0 Platforms Tead Lead	NASA Mits
Erin	Means	Software Systems Engineer	JPL Ocio
Luis	Mederos		
Simon	Mehalek		Gold Street Labs
Cade	Metz	Us Editor	The Register
Kenneth	Meyer	Software Engineer	JPL
Joel	Miller		NASA Ames - Dell
Sheena	Miller	Sr. Software Systems Engineer	NASA JSC (Barrios)/Er6
Chase	Million	Applications Developer	California Institute Of Technology
Robbo	Mills		Hunky Dorey Entertainment
Bill	Monroe	Global Account Manager	AT&T
Bruce	Montgomery	Chief Technology Officer	Technology Access Television
Tristan	Moody	Graduate Research Assistant	University Of Kansas
Kenneth	Moore	Senior Analyst 2	Db Consulting / NASA GRC
Neal	Most	Project Manager	Innovim
Catalina	Movileanu		
Troy	Moyers	Engineer	Mits/Is70
Erik	Mullinix	Managing Partner	Rainworks
Stuart	Murray		
Syed	Nadeeb	Software Engineer	
Theodore	Nathanson	Aerospace Engineer	Consultant
Robert	Navarre	System Administrator	Lockheed Martin Es&S
Diego	Naya	Ceo	Plugtree
Michael	Needler	Sr. Software Engineer	Tietronix/Mod/JSC/NASA

Ara	Nefian	Senior Scientist	NASA
Brent	Newhall	Software Release Assistant, Goddard Ipp Office	NASA - GSFC
David	Newton	System Admin	Dts/MSFC
William	Neyer		Hhs/Fda
Beth	Nguyen		Arts
Ben	Nickell		Idaho State University
Rolando	Nieves	Software Architect	NASA
Jeffrey	Norris	Section Manager, Planning And Execution Systems	NASA Jet Propulsion Laboratory
Stefan	Novak		
Daniel	Oldham	Computer Engineer	NASA Grc
Bonita	Oliver	Computer Scientist	NASA
Robin	Orans	Technology Partnership Manager	NASA Ames Research Center
Roger	Orellana		
Jeffrey	Osier-Mixon	Community Manager, Yocto Project	Intel Corporation / Open Source Technology Center
Edward	Ost	Technical Director	Talend
Billy	Otrs		
Pete	Overstreet	It Manager	Barrios Technology, Space/Life Sciences, Jsc
Steve	Padgett	It Manager	Arts
Rabi	Palikonda	Sr. Research Scientist	Science Systems & Applications Inc.
Aravind	Parchuri		
Joel	Parker	Flight Dynamics Engineer	NASA Goddard Space Flight Center
Pilar	Passmore	Librarian	
Tripp	Pearson	Engineer	NASA
Tiago	Pereira	Post-Doctoral Fellow	NASA Arc
Jennifer	Perez	Software Engineering Lead	E301
Ab	Perisamy		Gluster

Nick	Peters	Pm	Iv
Kenneth	Peters	Advanced Technology R&D; Software	JPL
Kristina	Petrosyan	Security Analyst	Province Of BC
Long	Pham	Computer Engineer	NASA - Goddard Space Flight Center
Shawn	Pierce	Software Engineer	Lockheed Martin
Nick	Pink	Security Intern	NASA Ames
Alex	Pline	It Project Manager	NASA Headquarters
Chris	Preimesberger		Eweek
Dave	Proffer		
Dan	Proper	Technology Systems Administrator I	Washoe County Technical Services
Jessica	Prue	Masters Student	Maxwell School, Syracuse University
David	Pruett		Geocontrol Systems Inc.
Greg	Pryzby	Solutions Architect	Red Hat
Martha	Przysucha		Gsa Office Of Governmentwide Policy
Siegfried-Angel	Pujals	Free Software Developer	
Cuong	Quach	Research Engineer	
Garry	Qualls	Systems Engineer	NASA/Oct
Mark	Radcliffe	Partner	Dla Piper
Frankie	Ramos		IHA
Thomas	Rando	Open Source Team Lead	Indyne, Inc / NASA HQ
Lee	Razo	Director	Datasphere Systems
Michael	Redmon	Program Manager	Sgt, Inc.
David	Reese	Mobile Computing/Emcs Technician	Jacobs Technology Maf-G
Chris	Regan	Aerospace Engineer	NASA Dryden
Scott	Reinhart	Software Engineer, Staff	Lockheed Martin Space Systems Company
David	Retherford	Retherford	Erc Inc.
Jon	Richards	Senior Software Engineer	The Seti Institute

Gary	Richmond	Web Service Developer	Jet Propulsion Laboratory
Norm	Ritchie	New Programmes	Internet Systems Consortium
Luis	Rivas	It Manager	Ls
Marco	Rivers	Student	Miami Dade College
Michael	Roberts	Process Engineering Consultant	Qna/Ksc Epg
Erin	Robinson		Foundation For Earth Science
Peter	Robinson	Computer Scientist	NASA
Conrad	Roche		
Paul	Rodgers		Londonbikecompany
Jacob	Rodriguez	Tier Ii Lead Software Support Engineer	Irdeto
Ginny	Rogers	Web Designer/Development	Simply Bridges Group, Inc.
Bryon	Rollins	Program Manager	Ultra Electronics, Prologic, Inc.
Lyford	Rome	NASA/JPL Solar System Ambassador	
Alfredo Jimenez	Ronssin		
Ellis	Rubio		Ellis Engineering Solutions Ltd.
Lamont	Ruley	Supervisory Computer Engineer	NASA/GSFC
Kyle	Ryan	Electrical Engineer	Orbital Sciences/ Nsroc Ii
Stephen	Ryan		
Padma	S		
Alvaro	Salamanca	Sales Rep	Nexsys Latinoamerica
John	Samson		Honeywell Aerospace, Defense & Space
Anthony	Santopietro	Software Integration Team Lead	Chugach Federal Solutions, Inc / NASA CASI
Virat	Sarin		
Kayla	Sather	Environmental Intern	Jq
Godfrey	Sauti	Research Scientist	National Institute Of Aerospace
Craig	Sayler	Sr. Unix-Linux, High Performance Cluster Engineer	Dfrc NASA Dryden Flighth Research Center

Edward	Scharff	Research Engineer	Sgt
John	Schipper	Principle Systems Engineer	L-3 -NASA Ivv Facility
Weston	Schmidt	Software / Hardware Architect	Open Roadster
Otto	Schnarr	Embedded Systems Developer	NASA Dryden
John	Schnase	Senior Computer Scientist	NASA Goddard Space Flight Center
Bruce	Schubert		Emxsys
Kathryn	Scoggin		NASA
Peter	Scott	Analyst	JPL
Dave	Scott	Computer Engineer	MSFC / Eo50 - Hosc Ground Systems Development & Integration
Anthony	Sellers		
Chad	Sells	I.T. System Analyst	Laane
Timothy	Sesow	Chief Technical Officer	Open Source Storage, Inc.
Aarshey	Shah	Student	Cept University
Ehsan	Shahrokhi	IT	
Warren	Shearer	Senior Network Administrator	Dfrc
Salman	Sheikh	Ee	NASA/GSFC
Jack	Sheldon	Sr. Systems Engineer	Arcata Associates
Chris	Shenton	Consultant, Agent Provocateur	NASA HQ
Sandeep	Shetye	Information Architect	NASA
Fu-Tai	Shih	Software Engineer	Sgt/Ti
David	Siedband	CEO	Zentraal, Inc.
Jacob	Silvia	Software Configuration Management	J&P Technologies
Luca	Simoncelli		
M.	Simos		
Amalie	Sinclair	Organiser	Space For Progress
Jerome	Slangen		
Laura	Sleasman	Project Manager	Dell Inc.

Ifan	Slv		
Michael	Smith	Consultant	Booz Allen Hamilton
Joe	Smith	Enterprise Architect	Information Dynamics
Tristan	Smith	Program Lead	Mission Critical Technologies, Inc.
Jeffrey	Smith	Optical Engineer	NASA Goddard Space Flight Ctr.
Dan	Smith	Software Division - New Business Lead	NASA/GSFC
Kelly	Smith	Aerospace Engineer	NASA/Jsc
Ernest	Smith	Project Integration Manager	Usra/Ames/Ti
Kraig	Smith	It Manager	Washoe County Technical Services
Chaise	Smith		
Paul	Snow	Rules Management Chief Architect	Sourcepulse
Sunil	Soprey	Ceo/Developer/Geek	Derigo Technology Inc
Jared	Sperli	Co-Founder	Itsoftware
John	Sprague	End User Services Exec	NASA\OCIO
Amy	Stapleton	Neacc Management - It Supv	NASA / Msfc
Gerald	Steeman	It Specialist For Cm	NASA Sti Program Office
Sheila	Steffenson		Esri
Haley	Stephenson	Writer	Valador/ NASA Appel
Joseph	Steurer	Student	Missouri University Of Science And Technology
Martin	Still	Kepler Guest Observer Office Director	NASA Ames
Ted	Stodgell	Engineer	NASA
Kim	Stroeger	Scrum Master	Nokia
Lynnette	Stroeger	Developer	Trm
Bala	Sundarraaj	Chief Consultant	Podhigai Open Source Services
Patrick	Svenburg	Director, Platform Strategy	Microsoft
Thomas	Swanson	Server Administrator	Abacus Technology

Keith	Swanson	Computer Scientist	NASA Ames Research Center
Eric	Sweigard	Program Manager	Innovative Defense Technologies (Idt)
Alasdair	Swenson	Digital Creative	The Prototype Committee
Brian	Takemoto	Senior Technology Systems Developer	Washoe County Technical Services
Eric	Tatara	Software Engineer	Argonne National Lab
Graeme	Taylor	Undergraduate Student	University Of Southampton
Harbles	Theskepticalcat		
Peter	Thompson	Senior Systems Engineer	Sgt Inc/NASA NPP Msi&T
Grace B	Thomson	Owner/Ceo	Organic Software & Computers
Ed	Thomson	Software Engineer	Pratt & Whitney Rocketdyne
Marc	Tiar	Librarian li	Washoe County Library System
Beth	Todd	Technology Systems Developer li	Washoe County Technical Services
Igor	Toledo		
Mark	Tomizawa		Humankindx (An Mtv For Democracy Where The Audience Is The Star; In Beta)
Carlos	Torrez	Bussiness Manager	NASA
Joel	Tosi		
Tom	Trainer		Gluster
Daniel	Trebbien		
Shui-Ay	Tseng	Database Engineer Lead	Computer Science Corp.
Tina	Tsui	Computer Engineer	NASA/GSFC
Christopher	Tucker	I.T. Director	Laane
Nathan	Uitenbroek	Orion Sw Architect	L3/NASA
Luigi	Usai	Hacker	
Andreea	Vaculisteanu	Event Coordinator	Unawe Romania
Clementine	Valayer	Senior Consultant	Trasys Sa (On A Contract For Esa)
Michele	Vallisneri	Research Scientist	JPL

Anoop	Valluthadam	Student	Vmkv
Michael	Van Chau	Director Of Cyber Strategy & Development	Mei Technologies, Inc.
Alexander	Van Dijk	Web Lead Flight Opportunities Program	Mct @ NASA Ames Research Center
Jaap	Van Doorn	Consultant	Capgemini Nederland
Jay	Vasa	Cto	Applios, Inc.
Rishi	Verma	Software Engineer	NASA Jet Propulsion Lab
Jon	Verville	Engineering Collaboration Lead	NASA GSFC
Huy	Vo	Engineer	LMSSC @ Maf
Stephen	Voels	Sr. Engineer	SAIC
Mary	Volz-Peacock	Program Manager Open Government	Opm
Jerry	Wagner	ISS Trajectory Operations Officer	United Space Alliance
Eric	Wahl		
Jeff	Walter	Deputy Project Manager - Technical	NASA GSFC
David	Walters	SSC CTO	
Andrea	Warmbier	Patent Attorney	NASA Langley
Keith	Watson	Research Engineer	Cerias, Purdue University
Phil	Webster	Chief	NASA GSFC Cisto
Lee	Wedgeworth	Engineer	Wyle/Sd24
Jon	Welch	It Consultant	Ae Wood And Associates
Heather	Wellington		Gluster
R	Wertenberg		
Jason	Wickard	Manager, Software Development	
Michael	Williams		
Samuel	Wong		
Michael	Worsham	Systems Support Coordinator	Src/Spawar
Michael	Wright	Deputy Branch Chief	NASA/Msfc/Es12

Aaron	Wright	G.I.S. Student	San Antonio College
Lawrence	Wu		
Patti	Yamakido	Account Exec	Sgi
Masaki	Yamaya	Consultant	Just Skill, Inc.
Gil	Yehuda	Director Of Open Source	Yahoo! Inc.
Ku Chung	Ying	Computer Maintenance Engineer	
Hans	Zaunere	Managing Member	New York Php
William	Ziebell	Senior Software Engineer	Mct
Angela	Ziegenhorn		

Press

Name		Title	Organization
Luke	Fretwell	Founder, Media Correspondent	Govfresh
Debbie	Gage	Reporter	Dow Jones Venturewire
Sarah	Granger	Founder	Center For Technology, Media & Society
Alex	Howard	Government 2.0 Washington Correspondent	O'Reilly Media, Inc.
Cade	Metz		
Chris	Preimesberger	Senior Writer	Eweek
Mike	Wall	Reporter	Space.Com

Staff

Name		Title	Organization
Wayne	Burke	Executive Director	Open Forum Foundation
Lucas	Cioffi		Onlinetownhalls Inc.
Jessica	Culler	Public Affairs Officer	NASA Ames Deltha-Critique
Gretchen	Curtis	CTO-IT Communications Manager	NASA Ames Dell Federal Services
Pat	Elson		NASA Ames Dell Federal Services

William	Eshagh		
Terry	Fong	Director, Intelligent Robotics Group	NASA Ames
Rita	Galan		NASA Ames Dell Federal Services
Chris	Gerty	Open Government Analyst	NASA HQ
Scott	Goodwin	CIO For Space Operations	NASA HQ
Gabriel	Hurley-Ramstad	Senior Web Developer	NASA Ames Dell Federal Services
Cristina	LeClerc	CTO-IT Project Coordinator	NASA Ames Dell Federal Services
Kim	Lembo	Technology Partnership Manager	NASA Ames Lockheed Martin
Nick	Skytland	Director, Open Government Initiative	NASA HQ

Agenda and Schedule

Day 1 - March 29, 2011

8:00am	Registration Opens	
	<i>Main Room</i>	<i>Breakout Rooms</i>
9:00am	Dr. Tsengdar Lee, NASA Introduction and Framing	
9:15am	Wayne Burke, Open Forum Foundation Overview, Goals, and Format of the Summit	
9:30am	David Wheeler, Institute for Defense Analysis/DoD What is possible? Key differences between open source development inside and outside federal government	Breakout Session #1
10:00am	Terry Fong, NASA ARC Open Source at NASA	Breakout Session #2
10:30am	Patrick Hogan, NASA World Wind, Implementing an Open Source Project at NASA	Breakout Session #3
11:00am		Breakout Session #4
12:00pm	Lunch	
1:15pm	Wayne Burke, Open Forum Foundation Welcome back, afternoon goals.	
1:30pm	Richard Bullington-McGuire & Guy Martin DoD's forge.mil: Collaborating on Software Development within DoD	Breakout Session #1 Develop solutions to Communication and publicizing NASA's open source efforts
2:00pm		Breakout Session #2 Licensing
2:30pm		Breakout Session #3 How deeply can the community be involved?
3:00pm	Bob Sutor, IBM The Value of Open Source	Breakout Session #1 Development models & ongoing support
3:30pm	Chris Mattmann, NASA JPL Open Source at NASA	Breakout Session #2 Government restrictions, e.g. ITAR
4:00pm		Breakout Session #3 Limitations on contributing to external open source projects

4:35pm Wayne Burke, Open Forum Foundation
Summary of Day 1, Overview of Day 2

4:40pm Ray O'Brien, NASA
Moving Forward

Day 2 - March 30, 2011

	<i>Main Room</i>	<i>Breakout Rooms</i>
9:00am	James Williams, NASA Welcome to Day Two	
9:15am	Wayne Burke, Open Forum Foundation Overview, Goals, and Format of the Summit	
9:30am	Group Discussion Review the Issue List, what's missing?	
10:00am	Chris Wanstrath, GitHub GitHub	
10:30am	Chris DiBona, Google Google, NASA, and Open Source	
11:00am		Breakout Session #1
12:00pm	Lunch	
1:15pm	Wayne Burke, Open Forum Foundation Welcome back, afternoon goals.	
1:30pm	Brian Stevens, RedHat	
2:00pm		Breakout Session #2 Open discussions that may revisit any topic from the rest of the event or cover entirely new topics that have been missed thus far.
3:00pm	Pascal Finette The Mozilla example: What's happening with open source in industry?	
3:30pm	Linda Cureton, NASA Thank you, Wrap up	
3:40pm		Breakout Session #3
4:30pm	Wayne Burke, Open Forum Foundation Conclusion	

Keynote Presentations

Listed in Order of Appearance

David Wheeler, Institute for Defense Analysis/DoD

What is possible? Key differences between open source development inside and outside federal government

<http://open.NASA.gov/wp-content/uploads/2011/05/OSS-DavidWheeler-DOD.pdf>

Terry Fong, NASA ARC

Open Source at NASA

<http://open.NASA.gov/wp-content/uploads/2011/05/OSS-TerryFong-Ames.pdf>

Patrick Hogan, NASA World Wind

Implementing an Open Source Project at NASA

<http://open.NASA.gov/wp-content/uploads/2011/05/OSS-PatrickHogan-WorldWind.pdf>

Richard Bullington-McGuire & Guy Martin

DoD's forge.mil: Collaborating on Software Development within DoD

<http://open.NASA.gov/wp-content/uploads/2011/05/OSS-RichardBullingtonMcGuire-GuyMartin-ForgeMil.pdf>

Bob Sutor, IBM

The Value of Open Source

<http://open.NASA.gov/wp-content/uploads/2011/05/sutor-NASA-opensource-2011.pdf>

Chris Mattmann, NASA JPL

Open Source at NASA

<http://open.NASA.gov/wp-content/uploads/2011/05/OSS-ChrisMattmann-OpenSourceAtNASA.pdf>

Chris Wanstrath, GitHub

GitHub

<http://open.NASA.gov/wp-content/uploads/2011/05/OSS-ChrisWanstrath-GitHub.pdf>

Chris DiBona, Google

Google, NASA, and Open Source

<http://open.NASA.gov/wp-content/uploads/2011/05/OSS-ChrisDiBona-Google.pdf>

Brian Stevens, RedHat

<http://open.NASA.gov/wp-content/uploads/2011/05/OSS-BrianStevens-RedHat.pdf>

Pascal Finette

The Mozilla example: What's happening with open source in industry?

<http://open.NASA.gov/wp-content/uploads/2011/05/OSS-PascalFinette-Mozilla.pdf>

Excerpts from Ustream Chat

The follow is an excerpt of the Ustream chat from the Summit. Relevant sections of conversation have been included, while unrelated or logistical conversation has been removed.

Tuesday

Time	Username	Comment
11:51	user16	Does the phrase "works for hire" not apply to contractors making software for the government?
11:51	User10	Correct, does NOT apply.
11:52	user17	So the contractor firm 'owns' the software the Gov is paying for?
11:52	user30	If only gnu had taken that advice prior to unleashing gpl3
11:52	user24	depends on the contracting agreement
11:52	user24	They can go either way
11:53	user16	Not only is license proliferation a problem, but satisfying all of the requirements of subcomponents.
11:54	user24	Yeah. the license is pretty horrible. The lawyers who wrote it clearly didn't spend any time studying other similar types of licenses
11:54	user30	wow, that's really bad
11:54	User10	The government's "unlimited rights" specifically do NOT allow distribution to the public. Yes, the contractor generally can control the software.
11:54	user23	Nice licence table here: http://fedoraproject.org/wiki/Licensing:Main#GPL_Compatibility_Matrix
11:55	User10	For the contractor to not control the software, we generally need a deviation to the FAR.
11:55	user30	@User10: so a contractor could be the "leader" of an OSS project, thereby absolving NASA of its crazy NOSA and liability requirements
11:55	User10	No.
11:55	user30	*absolving
11:55	user16	Should that change? If I work as an independent contractor, then I automatically assign my rights to software that I produce to the company.
11:56	user16	Actually, when I work as a bonified employee as well.
11:56	user24	user16 it depends on your state, and your employment agreement. Usually.
11:56	User12	bona fide?
11:57	user16	*bona fide. Yes sorry.
11:57	user30	If you want to retain ownership of your own code, or write OSS on the clock, make sure it is in your contract.
11:57	user24	NASA doesn't effectively monetize it's patents
11:57	user24	so that argument is bunk
11:57	User12	bonified employees might be people who retired (and died) in place. :-}
11:58	user22	hehe
11:58	user16	Does NASA hold patents?
11:58	User10	The default is US General Release so that taxpayers do have access to NASA software, but foreign persons do NIT get benefit paid for by US taxpayers except in certain situations.

11:58	<i>user24</i>	he's absolutely right about all this. I hope it isn't news to msot of the people in this room
11:58	<i>user16</i>	I know that other branches do.
11:58	<i>user17</i>	NASA can make money off patents? I though we were not allowed by Congress to take in funds not allocated by Congress??
11:58	<i>user24</i>	well, if a contractor produces something patentable, that patent probably goes to NASA
11:59	<i>user24</i>	(makes not sense to do anything else)
11:59	<i>User10</i>	BASA does get patent royalties and the developers can also get a split of the funds. The patent goes to the contractor.
11:59	<i>User4</i>	@user17, yes, our group has numerous patents. The individual doesn't make money, or even the group though.
11:59	<i>user17</i>	That is probably true but we cannot take money for it, I think?
11:59	<i>user24</i>	but then that patent is used as an excuse not to release patent encumbered software
11:59	<i>user26</i>	I believe patent-able products share the patent between NASA and the contractor (if appl)
12:00	<i>User10</i>	Shared roylaties is correct.
12:00	<i>User10</i>	There are very few NASA codes that have ever been patented so it is NOT used a tool to prevent release to others.
12:01	<i>user24</i>	But if the contractor has the patent, it's hard for NASA to release the code
12:01	<i>User10</i>	Correct, if others hold the patent NASA can only release for government purposes.
12:02	<i>user24</i>	which is why patents related to software developed for NASA should revert to NASA ownership
12:02	<i>user26</i>	if NASA and a contractor "share" a patent, can NASA "buy out" the contractors interest?
12:03	<i>user24</i>	you'd never be able to justify the funding
12:03	<i>user17</i>	Could he clarify how NASA can make money or take in funds for NASA patents?
12:03	<i>user22</i>	User1: What's the best way for Open Source projects to communicate with Government agencies?
12:03	<i>user16</i>	https://secure.wikimedia.org/wikipedia/en/wiki/Copyright_status_of_work_by_the_U.S._government
12:04	<i>User10</i>	Public Domain is a legal term and government works, while not copyrightable, are patentable so they are NOT automatically public domain.
12:05	<i>User11</i>	everything he mentioned is interesting- but wading through it requires a lawyer. NASA needs a a no-nonsense and easy-to-comprehend policy that invites developers to make the decision easy
12:05	<i>user17</i>	good idea!
12:05	<i>user21</i>	great question
12:05	<i>user24</i>	User11: hopefully that's what will come out of this
12:05	<i>user17</i>	here here User11!
12:05	<i>User11</i>	excellent
12:06	<i>user16</i>	Yes. It's useful when OSS licenses have FAQ pages.

12:06	<i>user16</i>	A similar FAQ can be created for the legal issues involved with OSS at NASA.
12:07	<i>User8</i>	Development groups should not have to become legal experts to get the benefits of collaboration and more eyes on the code.
12:08	<i>user24</i>	I'd love to see software contracted for NASA released by default under BSD or GPL unless otherwise necessary
12:08	<i>user24</i>	make it part of the agreement
12:08	<i>User8</i>	@user24: +1
12:08	<i>User10</i>	They don't need to become lawyers. Every Center has a Software Release Authority that answer any questions about this subject.
12:09	<i>user30-1</i>	man that would be wonderful. it's so frustrating to manage a proprietary SW project with multiple contractors... everybody has to sign NDA's etc
12:09	<i>user24</i>	yeah. Hell, have a requirement that it be released as it's developed. Solve that "never done" problem he was talking about
12:10	<i>user24</i>	also actually get an eye on what's really being done, as compared to what the contractor wants you to think is done
12:10	<i>user30-1</i>	then you have the problem of a contractor claiming copyright over some trivial piece of software, and then they turn around and sell their "secret" modification to the government at ridiculous prices
12:10	<i>User6</i>	And avoid the possibility that a software vendor can use it to milk the contract.
12:10	<i>user24</i>	woot! continious release process!
12:10	<i>user24</i>	that's what NASA needs as default
12:11	<i>user24</i>	ted: I'm sure NASAs clever lawyers could figure out how to discourage that.
12:11	<i>user21</i>	thats very true!
12:13	<i>user24</i>	This is neat software. I'll need to check it out
12:13	<i>user24</i>	but did it start as an OSS project?
12:13	<i>user24</i>	or a NASA project?
12:14	<i>user20</i>	This is the Software Release site for NASA Goddard: http://ipp.gsfc.nasa.gov/SRA/Release-process.htm
12:14	<i>User11</i>	put it on apps.gov
12:15	<i>user30-1</i>	speaking of software release sites... does anyone know what happened to CosmosCode?
12:15	<i>User10</i>	Do you mean COSMIC?
12:15	<i>user24</i>	saw the comment about that in the uservice stuff.
12:15	<i>user24</i>	I'm curious as well
12:15	<i>user30-1</i>	it was an ARC project. part of their Co-Lab
12:15	<i>user30-1</i>	sort of NASA's version of sourceforge
12:16	<i>User10</i>	The software was transferred to the National Technology Transfer Center in West Virginia. The codes are available.
12:16	<i>user30-1</i>	umm, they were using svn and drupal, etc. nothing real earth shattering
12:16	<i>User3</i>	@User10: can we use LGPL instead of NOSA?
12:16	<i>user24</i>	The codes aren't really relevant. It's the people working to make it happen
12:16	<i>user30-1</i>	the interesting part was that it was a NASA hosted service

12:17	User10	It depends on lawyer approval from your Center but we are currently looking at allowing other licenses.
12:17	user20	@Stuart - is this IV&V facility?
12:17	user26	isn't the use of and collaboration on open source initiatives covered under the Space Act Agreement?
12:18	user24	<sigh> it's really unfortunate that robotics/flight code is generally not going to be released
12:18	User3	@User10: I'm at LaRC. Can I use LGPL? How do I ask for approval?
12:18	user30-1	basically, if your center won't let you host a project on BerliOS or GitHub, then CosmosCode at NASA/ARC would have been a very good "Safe" alternative
12:22	user24	"software is never finished, so don't write policies that depend on that" seems to be a solid takeaway so far
12:23	user30-1	at every bullet point i'm thinking "YES! YES!"
12:23	user24	Clone this guy and have him as an internal open source missionary
12:24	User11	ouch- a week for filling out forms?
12:24	User10	No, not a week. Other people experienced less than hour.
12:24	user24	so once it's approved for release, does this whole process have to happen every time they push a bugfix?
12:25	User10	No, bugfixes are exempt from review.
12:25	user30-1	releasing OSS requires lawyers? i didn't know it took a lawyer to type "svn commit"
12:25	user24	what about minor feature improvements?
12:25	User7	I like Terry's perspective on this - very helpful information to see 2-5 months or more for releases
12:25	user24	I'm wondering about the issues related to having a public source repo for a NASA project (as opposed to NASA working with some other project)
12:25	user19	user30-1: releasing/distributing s/w is when the licenses *kick in*
12:25	User10	His week experience depends on whether one has complied with NPD7150 (OCE rules) during development. We release many codes in under a week total.
12:26	user30-1	mathematically, there's no difference between a bugfix vs. any other change to software
12:26	user30-1	therefore, ALL releases could be interpreted as "bugfixes"
12:26	User11	@user24 i agree, clone him and have him evangelize. he is on point
12:27	User10	Minor upgrades CAN be exempted depending on what it represents in the big picture.
12:27	user20	CIO/OCE should give Terry the latitude to evangelize on these issues
12:27	user19	not true.... licenses are all based on how the s/w is distributed. If using internally, you are not releasing the code as a external distribution
12:27	User10	No, bugfixes are determined by the Software Release Authority to be exempt or not. Not the devlopers call because we know the game.
12:27	User6	@ted: We're a long way from an information-theoretic basis for copyright. Mathematics are unlikely to be considered.
12:27	user30-1	alas :p
12:28	user30-1	if all initial releases were the empty set, then you'd only have to go

		through one trivial approval process, once, ever.
12:28	User6	And everything would be a derived work of that set.
12:29	User1	Questions for Fong?
12:30	user29	They should do a Lean Six Sigma event to look at the open source process to get the software approved more quickly.
12:31	user19	People need to recall that crowd sourcing != open source... the differences are subtle, but important
12:33	user26	@User1: Can web developers use open APIs (e.g., GoogleMaps API, etc.) simply under the Space Act Agreement, or is additional approval required?
12:34	user30-1	@user26: why would they be restricted in the first place? are developers restricted from using other open standards?
12:34	user26	@User1: not sure, hence the question.

Wednesday

Time	Username	Comment
11:41	user16	Open source by default. That's an interesting idea.
11:41	User2	Is there any NASA lawyer attending or participating in this event?
11:44	user15	We tried to have every affected group represented, and yes, the law component will be critical in any solutions we implement - so we need them involved in the discussion from the beginning.
11:44	User2	Encouraging NASA scientists and engineers, and LAWYERS to participate in OSS development
11:49	user18	and webmasters
11:50	user22	User5: are you interested in using open source for your work?
11:51	User5	@user22...unfortunately, that's a discussion that has to happen at a much higher level than where I sit.
11:52	User5	We are constrained with the tools, gear, manpower and tasking given to us.
11:52	User2	cute answer
11:52	User5	At this time, the tools we are given work and are similar across the Agency.
11:53	User5	Open Source would be a valuable tool, but one question is how to distribute across the Agency, spanning 11 Centers...and a country!
11:58	user22	Basically I am working to get open source into broadcasting with 2 major networks in the US and lots in Europe
12:00	user22	what's quite interesting is that they are interested in replacing mission critical parts of their broadcast infrastructure with open source, not just the backoffice stuff
12:19	user16	Maybe Chris can talk about whether open source projects should use central VCS (like Subversion) or a distributed VCS (like git, Bazaar, Mercurial, ...)
12:21	User11	need somebody to translate what was said on day 1 (legal stuff) w/ how NASA can use GitHub. exciting stuff
12:21	user28	@User11 +1. I'd love to hear NASA Legal make a pronouncement on our

		ability to use GitHub et al
12:24	User11	NASA needs to ask: how can we afford NOT to use this
12:26	user19	Wonder how many of those contributions were *useful*
12:27	User11	Gov't version?
12:27	user16	Please ask Chris about centralized version control systems (VCSs) versus distributed VCSs.
12:41	user16	*project
12:41	user19	cool!
12:42	user19	Looking forward to svn 1.7 !
12:42	user16	Yes! Me too.
12:42	user19	It IS quite slow. git is fast
12:42	user16	svnsync --source-prop-encoding
12:42	user16	That's my little niche.
12:42	user19	In fact, I use git-svn on most the the apache projects I hack
12:42	user16	True. git-svn works pretty well.
12:43	user16	I *love* that ASF has official git mirrors, btw!
12:43	user16	https://github.com/apache
12:43	user19	yep... we're trying to also see the feasibility of "real" git as well...
12:43	user16	Oh, okay.
12:45	user16	900 projects. I wonder if that includes projects by Googlers.
12:45	user16	not just official Google projects
12:45	user27	yeah, don't they get like one day a week to work on private stuff or something?
12:45	user16	in their 20% time
12:52	User9	That's an interesting discussion, regarding OSS from government agencies
12:53	User9	considering NASA is not empowered to copyright anything, can they actually enforce any sort of FOSS license?
12:53	User9	Is there an appeal process regarding ITAR'ed information?
12:54	user19	I wonder why NASA employee's can't copyright stuff done as part of their job, but they *can* patent it...
12:55	user16	There was some dicussion about that on day 1.
12:55	user16	David Wheeler discussed it.
12:56	User11	c'mon
12:57	User1	and there's a liveblog here: http://gov20.govfresh.com/2011-NASA-open-source-summit-convenes-innovators-and-technologists/
12:57	user16	Why does Google commonly use the Apache License?
12:58	user14	@user16: he mentioned this earlier, along with others. BTW, was that cdibena@google.com?
12:59	user16	Did he discuss dual-licensing?
12:59	User1	discussing licensing now
12:59	user14	@user16: nope. @AH: nope, thanks.
12:59	User5	He also provided a link - http://code.google.com/opensource
1:00	User5	It was on his final graphic

1:01	User1	asked Apache & ITAR question. thank you
1:03	user13	have you looked at how long unmodified code can survive - since licenses may impacts contributions/updates
1:04	user13	rate of code rot, how long before that 2B lines of code becomes useless or stops working
1:05	user16	That was a great question from the audience.
1:06	user16	Which open source license does Google prefer?
1:06	user14	Apache2
1:07	user14	I had this question that I didn't get out in time: Can "useless/rotting/unattended" code be turned into open-source, by default? Something like the FOIA after a certain time?
1:13	user13	Thats a good question for the lawyers. Code is really a living thing, if its sits around too long, it becomes useless or is replaced with something else.
1:15	user13	Any value that the code had is often lost - and that is a bad thing, because the value returned to the taxpayer is diminished.
1:16	User10	How does that diminish value to taxpayers?
1:17	User10	The default release level for all NASA code without export/programmatic restriction is currently US General release which allows taxpayers to request the codes.
1:21	user13	The assertion is that license type impacts community participation, and community participation determines how long and how much value certain code may have (not all).
1:22	user13	there may also be code thats never officially released because of the level of effort to do so.
1:23	User10	Bob, I do understand those points.
1:24	User10	However, widely disseminated codes when adapted by Airbus or Embraer also serve to diminish taxpayer value because they aid foreign competition.
1:27	user13	ITAR is an important but orthogonal issue.
1:32	user13	Clearly - we don't want to run afoul of ITAR or sensitive engineering data. But there is a large class of code that can and should be pushed to anyone that can use it.
1:33	User10	There is truth in that Bob. Agency policy limits us to releasing code that is for health, human safety, quality of life at this point.
3:43	user13	FYI: The initial work for large SSI was done at NASA Ames. We currently have a 2048 CPU LINUX system.
3:44	User9	<3 XFS
3:51	user16	libguestfs?
3:51	user13	question: clearly RH has benefited from OSS. How does RH view debranded derivatives like centos or oracle? positive/negative/neutral.
3:52	user19	user16: http://libguestfs.org/
3:52	user19	v. cool tech
3:52	user16	Nice
3:54	User1	Any questions from the Internet for the RedHat CTO?

3:54	user13	question: clearly RH has benefited from OSS. How does RH view debranded derivatives like centos or oracle? positive/negative/neutral.
3:54	User1	thanks, Bob.
3:55	user25	Great answer.
3:56	user16	What open source licenses does RedHat recommend for new projects?
5:22	User1	Interesting to see the new parts, comparing to Lilly's preso from 3 years ago http://www.slideshare.net/johnolilly/7-lessons-from-mozilla-presentation
5:23	user16	Some questions for Pascal:
5:23	user16	In what ways does Mozilla need to improve, to increase the effectiveness of its open source development efforts?
5:23	user16	Do you believe that Mozilla's focus on web technology helps Mozilla's efforts to maintain active open source projects?
5:23	user16	Do you think that a single organization can maintain active communities of developers of software that cover a wider range of purposes?
5:24	user16	https://github.com/mozilla
5:26	User1	thx. asked
5:27	user16	Being able to *build* a piece of software is crucial. That's a good answer.

UserVoice Submissions

85 Votes	Licensing
	This may include many issues, including the fact that NOSA (NASA Open Source Agreement) deters outsiders from using NASA produced software.
Comments:	
March 28, 2011 1:45 PM	Lot of votes here, and this is the key topic. But I'm curious what details of the NASA Open Source Agreement, http://www.opensource.org/licenses/NASA1.3 do outsiders find limiting?
March 28, 2011 1:59 PM	Agreed. the NOSA license is unnecessarily complicated. NASA doesn't even need its own vanity license when there are so many appropriate choices already. Per the National Aeronautics and Space Act of 1958, anything other than public domain is questionable. Where "fancy" licenses come in handy are twofold:1) For respecting the IP created through joint projects between NASA and its private contractors.2) For protecting the US Government's own authorship and copyright where appropriate, e.g. through a 2 clause BSD style license instead of plain old public domain.
March 28, 2011	Also can have cases where NASA/center legal staff decide not to release software based on external open source, if they have trouble deciding

8:07 PM	whether the external licenses are satisfied.
March 29, 2011 1:03 AM	I would agree with him here. If you have a new product which is a derivative work with components from a variety of different open source licenses, the legal review burden is substantial, and unfunded. So the cost-effective option is "don't release the work" since external release is rarely a requirement for the funding project.
March 29, 2011 1:30 PM	I am looking closely at NOSA 1.3 for the first time, and from the perspective of Recipient. The most worrisome part to me is clause 3.J. "...Recipient is hereby put on notice that export of any goods or technical data from the United States may require some form of export license from the U.S. Government. Failure to obtain necessary export licenses may result in criminal liability under U.S. laws.." This is not compatible with the Internet as it exists today, and puts a legal burden that many grass-roots open source projects cannot handle. These and perhaps many academic institutions would be wary of basing their open source efforts on top of it. Having this clause in the NOSA would very nearly doom a code base to being the same as closed source, i.e., the people who use open source and respect US law will stay away for fear of criminal prosecution.
March 29, 2011 1:57 PM	He has a valid concern, but I don't know that there is any way to get around the fact that NASA does stuff in space and technology, and it's in just such areas that export controls often apply. However, lots of academic and grass roots efforts manage to work within the export control rules. Sure, it's not the total carefree world of publish what you will.
March 29, 2011 3:04 PM	Unless the appropriate part of the source code is specifically marked as export controlled, then how is anyone supposed to meet this license? If something is not clearly marked as export controlled, then it should be fair game.
March 29, 2011 3:22 PM	Our project, OpenMDAO is currently released under NOSA. We have lots of outside interest from Academia and Industry. No one has yet expressed concern over the license. That being said, I see that there are some gotchas in the license. It would be nice to have other options, but in our case it was definitely better than nothing.
March 29, 2011 3:30 PM	IMHO (and IANAL)*, if every file is marked with NOSA v1.3 (as is often practice with GPLv2), then there is no way to know. Assuming that any release from NASA needs to go through NASA legal review, then it should be possible to catch export issues there. If there is no export issue, then that wording should not be applied. Granted, there still may be many cases of gray area, but I'm guessing that a lot of progress can be made if this option of a non-export/non-ITAR-encumbered license is made available to the NASA code originators by NASA legal. This, of course, may mean an alternative license, or the option to release under something like 3 clause BSD. (*IMHO = in my humble opinion

	*IANAL = I am not a lawyer)
March 29, 2011 4:07 PM	Other projects have deal with ITAR restrictions. OpenBSD, for instance, has cryptography that falls under ITAR. (see http://www.openbsd.org/cvsync.html and scroll down)."IMPORTANT NOTE: There are a few issues relating to cryptographic software that everyone should be aware of: * The OpenBSD sources are from Canada. As researched by a Canadian individual and as described in the Export Control list of Canada, it is legal to export crypto software from Canada to the world. * However, if you are outside the USA or Canada, you should not fetch the cryptographic sections of the OpenBSD sources from a CVSync server located in the USA. The files in question are [.....]Because of the USA ITAR munitions list, crypto software may only be exported to Canada from the USA."Another thing to consider might be dual licensing: release stuff under a dual NOSA / [something else] license. NASA and other government entities can employ the software under NOSA to keep their people happy. outside users could be free to hack and distribute the code under LGPL or something similarly aligned with the spirit of the NOSA license.
March 29, 2011 4:22 PM	I will never understand lawyer speak, but I always read that clause as a catchall. It sort of says, we give you this open source software, but that does not mean you can put ITAR/export controlled stuff in it and use the license as cover. If you put ITAR stuff into what we gave you and then distribute it, you're still liable. Again, it's not ideal. However, for our project it has not been a show stopper, yet!
March 29, 2011 4:23 PM	When a project really has ITAR issues, such as CVsync, then I think it is appropriate to have that additional statement. But that statement is outside the standard BSD license.If there *really* is an ITAR issue with a piece of source code, I personally would not apply LGPL to it to circumvent the issue.
March 29, 2011 4:28 PM	NASA needs to be able to make use of other OS licenses (e.g., Apache 2, GPL 3) in its work. NOSA is not sufficient.Perhaps the biggest problem with NOSA is that very few people really understand it, or are willing to take the effort to understand it. Although there are many OS licenses, the most successful (widely used, visible, etc.) projects make use of a small set of licenses (Apache 2, GPL 3, etc). Each of these "most popular" licenses has been extensively reviewed and "tested" by the community (developers, lawyers, etc.) and so their implications, constraints, etc are well understood... and thus are NOT a barrier to open source innovation.

March 29, 2011 4:30 PM	it's not CVsync, it's the OpenBSD project as a whole. CVSync is just their version control system. Furthermore, they don't release anything under GNU licenses (they're BSD, for goodness sake!).The only "circumvention" going on is the fact th
March 29, 2011 4:32 PM	Federal agencies needs a fungible model for sharing GPL modifications. If one agency can find a way to do it that makes its lawyers happy, it would have a huge impact on OSS government-wide.
March 29, 2011 4:37 PM	The "disclaimer of warranty" and "limitation of liability" clauses in GPL3 (and similar language in other licenses) ought to CYA for any contributions from civil servants. It's pretty clear that contributors, the project leaders, and cont
March 29, 2011 4:43 PM	furthermore, contributions come from *individuals*, not federal agencies. All we have to do is empower civil servants to make contributions to OSS projects as individuals.If NASA wants to bless an official patch to an OSS project, then
March 29, 2011 4:50 PM	Amusing. NOSA Ver. 1.3 incompatible with all other OSS licenses! (according to speaker). FSF considers it non-free?! (Clause G)
March 29, 2011 4:53 PM	My roadblock has not been liability, but convincing lawyers that the GPL doesn't conflict with the public domain. Public domain, they say, must not be subjected to the GPL. In effect, their argument is that we *protect* the public domain mo
March 29, 2011 4:55 PM	Oh, great point! Not to start a flamewar here, or reveal my personal bias, but perhaps this is yet another reason to prefer BSD over GNU. ;)
March 29, 2011 5:03 PM	After listening to Mr W, see how problematic NOSA can be. As I have said previously, I am happy to have NOSA as apposed to nothing else. But perhaps future releases of OpenMDAO can be released under a different license, if we can figure out a way for that to happen at NASA.
March 29, 2011 5:33 PM	Public domain code is always public domain (in the US), even if it is embedded into a source code file that is otherwise covered under the GPL. In other words, I could extract the public domain portion from that source code file and use it however I please, without putting what I use it for under the GPL.

<p>March 29, 2011 8:17 PM</p>	<p>The exception is where a public domain section of code has so many mods to it that you can't reasonable extract the untouched portions. Anyway, the problem is knowing that a portion of code is public domain if it's not marked as such in a GPL source code file.</p> <p>This is perhaps a refinement of the main issue, but something I've run into... I think it is worth talking to Silicon Valley companies that have made contributions to open source. Some are okay with GPL v2, but not GPL v3; others have a problem with MozillaPL. The issue is: they see GPL v3 or MPL as potentially compromising whatever patents they have. I want to see commercial enterprises pick up NASA code, build on it, sell products based on it, and contribute back. But uninformed selection of open source licenses would still hinder acceptance of the software.</p>
<p>March 29, 2011 8:48 PM</p>	<p>The central issue seems to be that users dislike NOSA for the following reasons: 1) Not one of the standard set of OSS licenses, so it should not be used2) Incompatible with other licenses3) Unnecessary, as some of the other standard OSS licenses should be sufficient for our needs. Other government agencies have not needed their own custom license, and neither should NASA4) Having a special NASA license is a barrier to collaboration</p>
<p>March 29, 2011 8:55 PM</p>	<p>Potential users are not aware of NOSA, due to it's lack of use. So NOSA ends up being a barrier to adoption.</p>
<p>March 29, 2011 8:56 PM</p>	<p>Education to the developers regarding the need for disclosure of code usage and their accompanying licenses. Developers should be taught that disclosures are just like citations in a research paper, they are required and necessary!</p>
<p>March 29, 2011 9:03 PM</p>	<p>If NASA still wants to use NOSA, it at the very least needs to be modified to be compatible with other licenses... but NOSA should not continue to be used.</p>
<p>March 29, 2011 9:06 PM</p>	<p>Educate developers on the basics of open source licensing and its implications</p>
<p>March 29, 2011 9:09 PM</p>	<p>I agree with [the] statement that open source release should be the default option as much as possible for software done at NASA. Having that default option streamlined, with the use of one of the most popular standard OSS licenses instead of NOSA would have great impact NASA and the outside world benefitting from OSS.</p>
<p>March 29, 2011 9:16 PM</p>	<p>The gaol should be to no longer user NOSA... because it is not a standard license!</p>

March 29, 2011 9:19 PM	Google uses the apache 2 license for almost everything. Also more modern licenses like MPL, LGPL (for libraries). BSD and MIT are a little antiquated.
March 29, 2011 9:20 PM	NASA Nebula is licensed under Apache 2, so there is a precedent there.
March 30, 2011 12:24 AM	Perhaps NASA should consider a dual-licensing model, such as Apache License, Version 2, and GNU GPL?
March 30, 2011 2:54 AM	"Standard" licenses become so because of popularity. It doesn't necessarily mean they're the best choices. As an example, GPL v2 versus GPL v3: version 3 adds a ton of complexity with not much added value (GNU fans may disagree here). If you look at, say, the MPL and Apache licenses, they're also pretty complex, and are far less popular than GPL and BSD. Simple is good. Not just for the lawyers, but for the developers, managers, customers, and end users who need to figure out what the heck the license means. Dual licensing is an interesting anomaly. The examples I've seen are worded so that any end user can choose which license they wish to use. I've also seen this backfire where the two licenses have incompatible clauses and the copyright owners / authors get rubbed the wrong way by people running with the project under an "undesirable" license. Balancing freedom with liability protection: that's how I see software licenses. "Freedom" includes liberty to distribute and modify the code, as well as liberty to sell it for a profit and the liberty to make proprietary mods if that's what floats your boat (or pays your bills).
March 30, 2011 1:33 PM	What do NASA folks do when they contribute to a project they do NOT own? If I contribute (say) an RSA authentication module to Trac, I have to abide by the Trac project's license. If I contribute similar to Django, I have to abide by theirs. And these may be different. I may need to agree to BSD, GPL, Apache, or others. We need to ensure that this is permitted by NASA policy.
March 30, 2011 2:10 PM	AFAIK it's not forbidden by any policy. If we needed explicit orders to do anything, nothing would ever get done.

81 Votes	Communication and publicizing NASA's open source efforts
	Where do NASA employees go to find out about the process of releasing code under an open source license? Where does the public go to find out what NASA has released? Where do you go to find out how you get involved, contribute, etc.? Is this

centralized or distributed? Let's figure out how to make NASA's open source efforts visible!

Comments:

Marc h 27, 2011 7:10 PM
Do we have a Wiki? More importantly, is there a shepherd who can manage it?

Marc h 29, 2011 4:26 PM
One central website that acts as a clearing house for all the open source projects and resources available (documents, code, etc.) would be essential. The list will probably be large, so everything will have to be sufficiently indexed and searchable.

Marc h 29, 2011 4:36 PM
There are two sides -- releasing NASA-developed code as open source, and the use of open source (developed outside of NASA) within NASA. NASA workers and the public need open communication about use, creation, re-use, and integration of open source software.

Marc h 29, 2011 5:44 PM
Most large organizations that use software engage a developers community. They treat it very seriously and put resources behind the development of the community. Check out:
<http://msdn.microsoft.com/http://code.google.com/http://developer.yahoo.com/http://developer.apple.com/>

Marc h 29, 2011 6:01 PM
I'm more of a consumer than a developer, at JSC. Under Constellation I finally had the opportunity to move around the agency and often asked about the tools people were using. It's really a labor-intensive to try to map out the open source applications and the communities who are using them. If I use MS Windows and MS Office I can call the help desk when I have a question. If I use open source, I'm out there by myself and no one will join me unless I'm looking successful... How can we get some institutional support, some power users who can field questions and investigate the source of problems?

Marc h 29, 2011 7:23 PM
Re: the publicizing of open source efforts to the NASA community - perhaps a service catalog of open source software that is being used within NASA would allow NASA users to be able to find software useful for their projects. Software support could even be available for a fee from the OCIO or some other entity - so some of the more popular open source packages not developed within NASA would be supported. NASA users could also grade the software so a NASA user could easily see what is most popular....

Marc h 29
The state of California has a very similar problem. Lots of local, county, state, and federal projects that overlap in functionality and yet there is little cross communication between the various "agencies". A wiki was deemed an important element of the solution, but developing a

2011 8:24 PM	program with encouraged leadership and "rewarding" those that made significant contributions to CA's complex open source environment.
Marc h 29, 2011 9:39 PM	Enable sites to make use of facebook login, google account login to ease use by new users
Marc h 30, 2011 4:59 PM	Something like the (now-retired) http://www.bbc.co.uk/opensource/ would be interesting

52 Votes	How deeply can the community be involved?
	As a government agency bound by large amounts of regulation and bureaucracy, what are the boundaries of community contribution? For example, could a NASA-originated codebase ever be handed over to a non-NASA community member for long-term support and maintenance? Is this legal? If it is legal, would it be practical?
Comments:	
March 28, 2011 2:06 PM	Unless explicitly forbidden by the license, forking is legal. I see no problem with non-NASA entities doing their own thing or taking over as needed.
March 29, 2011 2:42 PM	I believe there would have to be an organization that can screen any potential code base for proprietary/protected information, but beyond that I would hope that there would be no restrictions. I can foresee problems with software written by contractors having licensing/ownership issues, but NASA should work towards having more open contracting models when it comes to software development & ownership.
March 29, 2011 4:29 PM	NASA will have to commit resources and staff to handle any contributions. Sifting through contributions will be a big task. I wonder is there is a way to quantify the resources needed to actually accept contributions.
March 29, 2011 4:38 PM	There is actually already precedent for this. NASA developed the Numerical Propulsion Systems Simulation software and, thought it's not open source, the code base has been handed over to the NPSS consortium to continue maintenance. The consortium is made up of a number of commercial entities such as GE, Pratt and Whitney, and Boeing.

March 29, 2011 6:07 PM	I think [he] is right here. If bureaucracy is a major problem, why not have NASA contribute to forks instead of trying to lead the community? It's not a handover, it's following the locus of the community.
March 29, 2011 8:03 PM	Already has happened with OODT: http://oodt.apache.org/

40 Votes	Government restrictions, eg ITAR
Comments:	
March 24, 2011 6:50 PM	I think there needs to be a whole session just on ITAR
March 28, 2011 1:28 PM	Wish I'd seen this earlier. I agree with what [he] has said: ITAR and Open Source are not intrinsically incompatible, but anyone working on FOSS who is privy to information covered by an ITAR restriction needs to be aware and careful about never violating the security of that information.
March 28, 2011 2:20 PM	Don't forget SBU! This particular "classification" is a real hindrance to open communication.
March 29, 2011 1:10 AM	The whole thing of dual-use, in particular, needs clarification and some guidelines. One might develop software of general applicability, but, because it will be used on a spacecraft, it might be considered a "defense article", and subject to export controls, even if there's nothing "peculiar" to the space application. I'm thinking here of things like network stacks, utility programs, and the like. Clearly, software that is aimed at Guidance, Navigation, and Control is going to have export control issues. However, OSS and Export control aren't mutually exclusive. The export control restrictions are independent of the Open-Source-ness, so it's more a matter of the mechanics of distribution.
March 29, 2011 5:36 PM	If we use OSS in an ITAR-covered application, and that fact becomes known, that then provides some visibility into that ITAR-covered application. I don't know if that in itself might become an ITAR issue, where we might be able to use OSS but be unable to reveal that fact (possibly even by participating "too heavily" in the particular OSS community).
March 30, 2011	Speaker Patrick Hogan commented that releasing useful libraries as OSS pieces, rather than releasing an entire application, may avoid ITAR issues.

5:42 PM	That's an interesting idea.
March 29, 2011 6:16 PM	This sounds like it's part of a more general subject of improving the release process and speeding that up when/if possible, so it should be a good topic.
March 29, 2011 7:11 PM	NASA doesn't normally deal with "arms", much less their trafficking. Shouldn't this discussion primarily focus on EAR?
March 29, 2011 8:12 PM	Spacecraft, instruments on them, and support equipment for them fall under ITAR as "defense articles" from US Munitions List Category XV. Non-spacecraft "dual use" items would fall under EAR. So different centers may fall mostly in one category or the other. (I just finished my center's annual export compliance training, and I can show off my shiny refreshed knowledge :-)
March 30, 2011 12:07 AM	And given the "if you guess wrong, you go to jail" sort of penalties, there's an incentive to default to "export controlled". That is, you have to work hard to prove it's not export controlled, and if you're developing software to support some specific mission, it's unlikely the project wants to spend much money working that issue: it's not necessarily contributing to mission success. There would need to be overarching institutional support (as in charge numbers and people specifically assigned to the task) to run the gantlet.

37 Votes	legal office approval by default in, say, one week
	<p>The default mode for bureaucracies is to move slowly and avoid taking actions, especially risky ones. Potential solution: after a project fills out a web form declaring its intention to make its information (code, designs, data, etc) open, NASA legal (and export control) have one week to "disapprove", else the release is automatically approved. Here "disapproval" means providing brief, clear, exact, signed feedback describing the problem and suggested remedies. This procedure (inaction==approval) is generalizable to other bureaucratic processes.</p> <p>It can be argued, that in the case of the government, all software must be released open source as it is developed unless an explicit waiver is obtained in advance.</p>
Comments:	
March 29, 2011 6:17 PM	<p>This sounds like it's part of a more general subject of improving the release process and speeding that up when/if possible, so it should be a good topic. I kind of doubt that the policy will be changed to default to open source, especially in such a short time, but it seems worth discussing.</p> <p>YES. And it the onus should be on them to make the case that it should not be</p>

March 29, 2011 6:23 PM	made open. They should have to give clear and legitimate reasons for it not to be approved! The onus should not be on the person seeking the release to make a case for its release. By default it should be released.
March 29, 2011 7:59 PM	Great idea. While I agree with the general idea of this suggestion, I agree with James that the shortness of the duration allowed to Legal will essentially shoot down this suggestion before it see's the light of day. To make this otherwise excellent suggestion acceptable, I propose we make the term "2-3 Weeks" instead of "1".
March 29, 2011 8:58 PM	There seems to be two main observations here:1) let's streamline the approval/review process for all open source activities ((a) either release, (b) converting an ongoing project to open development, and (c) starting a new project that will be openly developed).2) at the beginning of a project where the performer knows they want to develop it openly (case (c) above), there should be a very streamlined approval process.For this to be successful, the performer is going to have to do some work prior to it being approved. The current process right now isn't streamlined and there is considerable back-and-forth between the performer and the review process. This occurs partly because the performer doesn't do the upfront work necessary to make it a streamlined process.In summary, this is not *just* legal, and we can come up with a process which is streamlined. Nobody would admit that it is streamlined right now.

32 Votes	Development models & ongoing support
	Should there be a single standard for how software is developed, how releases are managed, etc.? What happens to a project that is released after the original authors leave? Who maintains that code? Who does code review? Who gets commit access going forward? What happens if no one steps up?
Comments:	
March 28, 2011 1:25 PM	NASA has standards for software development, code-review, and configuration control. They are not uniformly applied, but I think it's a different topic than software licensing. Releasing as open-source is primary a licensing issue. With respect to commit access and maintenance, once it's open the software can always be forked and a branch maintained by people outside of NASA. A key issue in that case would be trademarks, e.g. a forked branch should not be called "NASA Dynamic Flow Estimator" even if the original was. But that is an issue that can be made clear in the license terms.
March 30, 2011 1:29 PM	I think creating, documenting, and blessing some standardized approach*es* would help acceptance from management. But trying to force everyone into a one-size-fits-all won't work, prevents evolution, stifles innovation. The OSS community is evolving rapidly in the tools it uses, the techniques (test driven

development, documentation driven development, continuous integration, continuous deployment). We don't want to be stuck with ill-conceived ideas like "waterfall" wen there are demonstrably better options that have emerged.

31 Votes	Collaboration
	Internal, external, and connecting the two.
Comments:	

26 Votes	Appropriateness of open source release for different types of software produced by NASA.
	Comments:
March 28, 2011 3:25 AM	NASA researches military and non-military applications - and this represents a concern for many developers - as it is possible for NASA to release code to the community, have that code developed further by programmers and then have that code reacquired by NASA for use.It raises real ethical questions on any government (NASA, DoD, etc) use of civilian open source code to develop, for example, weapons used by the armed forces. Most definitely there should be a clear boundary for community contribution that prohibits open source coders from contributing to NASA programs geared toward any kind of explicit military application, as well as open source code used in any kind of military context.
March 28, 2011 1:42 PM	Open source software, and it's derivative products, can be used for any purpose. If a software author wants to control the types of things that get done in derivative works, then a proprietary license would be more appropriate.
March 28, 2011 1:52 PM	on the contrary, publishing software as open source can protect its author from military restrictions. For instance, research code can get branded as "dual use" and end up becoming classified or branded with ITAR restrictions. Once classified,
March 29, 2011 2:58 PM	On the question of a "a boundary for community contribution that prohibits open source coders from contributing to NASA programs geared toward any kind of explicit military application". Anyone who releases software as open source has the flexibility to create their own open source license to restrict the software's use in military applications. I believe there are already certain

<p>March 30, 2011 2:44 AM</p>	<p>open source licenses that prohibit use by the government. I don't think there needs to be any additional "boundaries" put into place. If someone doesn't want their code to be used for military applications, they can put that right into the license, or choose not to contribute to certain projects.</p> <p>While I appreciate pacifist / anti-military sentiments, the fundamental spirit of open source is freedom. That necessarily includes military applications. My personal experiences haven't been with *avoiding* military applications per se, rather avoiding having a project declared ITAR or classified and as a result shutting down my academic work. CYA: I do not condone military secrets or circumventing ITAR, of course.</p>
-----------------------------------	---

<p>25 Votes</p>	<p>Public access to NASA source code repositories</p>
	<p>How will NASA make available its open-source code repositories? Will there be a unified portal (like Sourceforge, Savannah, GitHub, etc.) that will host all NASA code, issue trackers, and forums related to NASA OSS projects? What SCM systems is NASA considering?</p>
<p>Comments:</p>	
<p>March 29, 2011 4:50 PM</p>	<p>Would love to talk about this in the context of a federal-wide SCM platform.</p>
<p>March 29, 2011 5:04 PM</p>	<p>LaRC CIO told me I should use external repositories (GitHub, e.g.) for SCM joint projects</p>
<p>March 29, 2011 5:34 PM</p>	<p>Many of the examples presented by Terry. Fong were hosted in GitHub. One in Sourceforge. I think the more difficult part is <u>finding</u> the projects and discovering the background requirements/goals.</p>
<p>March 29, 2011 5:36 PM</p>	<p>OpenMDAO, from NASA Glenn, uses Launchpad.net</p>
<p>March 29, 2011 5:48 PM</p>	<p>Both SourceForge.net and GitHub are on the APPS.gov storefront. IMHO, GitHub is miles above sourcefourge with continuously iterating their platform and incorporating a micro-community around each piece of software on GitHub.https://www.apps.gov/cloud/cloud/category_home.do?&c=SA</p>
<p>March 29, 2011 5:59 PM</p>	<p>I agree with casey about discovering and finding the projects.</p>

March 29, 2011 6:32 PM	The Federal Communication Commission recently worked out Terms of Service with GitHub for government use. I'll get a hold of more information. And we have a process in place to move our ToS agreements to GSA.
March 29, 2011 9:11 PM	I believe that NASA has already signed a ToS agreement with GitHub. But to answer the original question, I believe that government agencies should embrace the following:(1) have their own app store that is the front-end of all software projects that are either available for licensing, download, and contribution. The app store should be there to help people search and discover software to use and projects to be part of. The app store should use an API from the code repositories to have it be the most up-to-date summary of the project, as it should be a one time form for the PM of the software to fill out. In general, you don't want duplication of information, as this is how it gets stoic and out of date. (2) Don't roll your own subversion or git repositories. You want to go to where the community is. NASA should have ToS signed with 3rd party vendors (GitHub, Sourceforge, etc.) that is based on the community they want to engage. On this list, projects should be able to abide by the ToS and not have to sign their own.
March 29, 2011 11:36 PM	It might be good to create a webpage on NASA.gov that lists all of the open source projects that NASA has started, uses, and/or contributes to.

25 Votes	What areas within NASA should be targeted as likely "early-adopters" of open source development?
	As we move forward with new policy for open source at NASA, some areas will likely benefit from using open source with less institutional resistance, and in so doing be very useful as trailblazers for verifying that the process as a whole is sound. Brainstorming which areas/projects could help in this way, and what specific issues they might face, may help us keep the "end-user" in mind while drafting policy.
Comments:	
March 28, 2011 2:17 PM	Here at MSFC, simulation software such as MAVERIC and ARTEMIS could probably benefit from being open source, but NASA has contracted with private 3rd parties who control close-source copyright over portions of the code. As these projects are jointly developed by civil service and private contractors, any efforts to unencumber the code would probably only be able to be done by the CS developers. I'd be interested to hear how JPL SPICE has fared, versus MSFC's proprietary software. Was SPICE developed from the very beginning as open source, or was there an effort to open it up after it had begun?
March 29, 2011	At Glenn, we make large use of the NASA NRA process. In the last round of NRA's we placed a very heavy emphasis on the IP statements in proposals. We

4:00 PM	actually requested proposed that included open source IP statements. I think that the very first place to start with finding OpenSource projects is places where NASA contracts for software to be developed. If NASA pays for it, there is no reason the contracting agency should retain IP control over the Product. The NRA process is a great place to enforce this, as there is a lot of funding that heads that way from the Aeronautics program side of things.
March 29, 2011 8:28 PM	Code developed for research purposes (rather than operational or flight software) should, in theory, have less institutional resistance.

22 Votes	Limitations on contributing to external open source projects
	Comments:
March 29, 2011 2:47 PM	If NASA begins to more heavily leverage the open source community for certain types of software (non-safety critical, etc), then NASA's policies should certainly allow for contributions back to the community. For example, if a NASA project decides it is of value to utilize an open source 3D graphics library, then there should be no restrictions on NASA employees contributing bug fixes and/or enhancements to that library so that the community benefits from NASA's use.
March 29, 2011 5:55 PM	I've already found -- and fixed -- bugs in numerous OSS projects we use. I do send patches back, or if I have commit rights, just commit 'em. Is there a policy that forbids fixing bugs, and publishing the fixes?

20 Votes	How to tie into existing NASA Policy from the Office of the Chief Engineer (OCE)?
	NASA already has policy that anyone funded by the agency developing software has to follow. This policy is from the Office of Chief Engineer and the document is NPR 7150.2A, which is the NASA Procedural Requirement which guides the development of all NASA software. Link here: http://nodis3.GSFC.NASA.gov/displayDir.cfm?t=NPR&c=7150&s=2 Open source software is only mentioned very briefly in this document, in Section 1.3.2 and 2.3.1. Needless to say, this policy document does not mention much about open source.
Comments:	
March 29, 2011 5:47 PM	Also, for software release, http://nodis3.GSFC.NASA.gov/displayDir.cfm?t=NPR&c=2210&s=1C

March 29, 2011 5:58 PM	Also mentioned in section 3.2.2 (for software release): http://nodis3.GSFC.NASA.gov/displayDir.cfm?Internal_ID=N_PR_2210_001C_&page_name=Chapter3
---------------------------	--

20 Votes	Identifying current roadblocks
	What are the impediments to releasing open source code today? What are the pain points for people working on this right now? We can't move forward on improving the process if we don't have a concrete grasp of what's wrong and communicate that list back out to the public and to NASA's leadership.
Comments:	

20 Votes	There is no release process to support brand new projects that have no code base yet.
	Comments:
March 29, 2011 5:13 PM	can you get a release decision based on the software requirements specification? It seems like the list of requirements the code can meet should be sufficient to determine what type of license and/or export control issues should apply to it. In particular, the SRS would let you decide the line between the software and what it can do vs. the models that would be analyzed with the software which likely would have export control issues.
March 29, 2011 8:30 PM	This type of process would be of tremendous help, both speeding up the initial process and allowing for future development without repeatedly cycling through the entire release process again.

13 Votes	The low-hanging fruit
	As an outsider from the open source community, but with experience in the US federal government, I'm interested in talking about the low-hanging fruit. Actionable, small steps. Keeping it simple, and chalking up a wide variety of small victories. Agile development applied to project ideas. What are the small ideas that NASA can work on today using OSS that can potentially grow big in the future?
Comments:	
March 29, 2011 5:54 PM	One thing that helped kickstart at NASA HQ was deploying SVN and Trac for code repos and issue trackers. It's now a Center-supported resource for our AppDev and even Graphic Design groups. I don't think anyone cares

<p>March 29, 2011 6:41 PM</p>	<p>particularly that it's open source software -- it just works for them. But we can point to it as a successful OSS tool they already use, a camel's nose under the tent. I think Agile tools and techniques can extend the example even further.</p>
	<p>This brings to mind Vivek Kundra's idea of IT efficiency through use of "commodity services" in his "25-Point Plan Implementation Plan to Reform Federal IT" and the idea of generic API in Patrick Hogan's talk. If the software can be adapted to provide "commodity services" (well, as close to "commodity" as possible) then more will see the benefits of leveraging this past work. So an idea is when creating the marketing story for the software try to present it as a possible commodity service.</p>

<p>12 Votes</p>	<p>How can NASA Open Source everything it develops?</p>
	<p>How can it and why wouldn't it? How can this happen baring classified and export controlled technologies?</p>
<p>Comments:</p>	
<p>March 29, 2011 5:05 PM</p>	<p>Add "What are the risks in doing this and NOT doing this?"</p>
<p>March 29, 2011 6:48 PM</p>	<p>I agree that this is probably worth discussing. Export control and such are reasons to avoid open source, as you noted, but in my understanding, NASA will also avoid open sourcing software that they think may be patentable or can sell. Things like copyrights or license restrictions could limit the release options, too. As the talk noted, moving to open source as a default does have advantages, but might be a difficult move for NASA to make since it basically reverses the current process. At least the discussion could try to figure out how to make it easier to get open source releases approved.</p>

<p>11 Votes</p>	<p>Default Open Source. You should have to get approval to restrict something, not vice versa.</p>
	<p>All code should be open source by default. If you want to release it then you just fill out the web form mentioned in the 'level approval in one week...' idea. The onus should be on the part of anyone (export, IP/copyright lawyers etc) to explain why it should not be released. Else it is default open within a week. The key point is that anyone not wanting it to be released has to make that case, not as per the system at present, the people wanting to release it having to make the case that it should be released.</p>
<p>Comments:</p>	

<p>March 30, 2011 1:38 PM</p>	<p>I love this: "default allow" vs the more firewall-style "default deny". It makes so much sense and is so easy to understand... that it'll never be accepted. :-(But I'm with you on it, it's a great idea.</p>
-----------------------------------	--

9	Software Development Standards
Votes	Internal vs. external and the complexities of aligning the two.
Comments:	

9	Barriers to facilitating real two-way open source
Votes	There are just too many, and the world will move on ahead without NASA
Comments:	

8	Incorporating legacy software into a NASA project
Votes	<p>How does NASA incorporate legacy software into a project and then release it as open source?</p> <p>For example, how does NASA incorporate scientific models that have been openly developed and openly passed around the scientific community into a NASA project and then release that project as open source when it is not possible to identify all potential copyright holders and therefore acquire all necessary copyright assignment.</p> <p>The scientific community is not concerned about such issues and move forward with its research and development, but these issues hold NASA projects back and make it difficult to release and collaborate with others.</p>
Comments:	

7	Reducing the scope of the software release process so that it is project-size appropriate
Votes	The SRA is too onerous for small projects.
Comments:	

7 Votes	Governance
Comments:	
March 29, 2011 4:24 PM	Do you mean governance of individual projects, or policies within NASA to cover all NASA-originated projects?
March 29, 2011 8:00 PM	Governance of individual projects would (IMHO) be particularly cumbersome. There should be an over-arching governance policy specifically for OSS projects.

5 Votes	How does the maintenance, packaging, and upkeep of the OSS get funded?
	While providing public access to an Open Source revision control is fairly simple, a good Open Source project needs continual upkeep. Integrating public submissions, bug tracking, web site maintenance, publicity, etc, all take dedicated efforts.
	Who sponsors this work? How does it get funded?
Comments:	

5 Votes	Established path for OSS Projects created in govt labs to move to better governance groups
	Although more government labs are beginning to release their code as OSS, they are generally poorly situated to run major OSS projects due to their missions, funding limitations (both quantity and types), and temperament. Many projects begun in labs, such as SimDIS, BRL-CAD, Delta3D are ready for such a transition and would benefit by such.
	There should be an easy and straightforward way to transition an OSS project from a lab to a body more suited for long-term governance.
Comments:	

5 Votes	Whatever happened to Cosmocode?
	Wasn't it supposed to be NASA's own version of sourceforge / berlios / GitHub? It seems to have vanished without explanation, and now some spammer is squatting cosmocode.org.

Comments:	
March 29, 2011 6:40 PM	My understanding of CosmoCode was to create a community of developers for open development of NASA software. The initiative realized that the internal policies weren't in place yet to utilize the development of this community. Moreover, as we are hearing here today regarding rolling an entirely new system, I wouldn't be an advocate for NASA to create its own version of GitHub. NASA should create the policy to allow open development of software then allow the performers to host it where ever they see fit (based on the community they are engaging, and appropriate approval). I am a fan, however, of a NASA app store where people can see all the software under development.

5 Votes	Sharing software
	Including what technologies can be used
Comments:	
March 24, 2011 4:51 PM	Can you be more specific about this topic? Are we talking about how we publicize open-sourced projects? What platforms (GitHub, etc.) we use to make open sourced code available? Need more info...
March 29, 2011 5:42 PM	Maybe we can frame this to be about how the open source software can be packaged such that people that are interested in the product can try it with very little effort. For example, the idea of setting up server environments, writing security plans, adapting interfaces can be daunting and make the investigation of open source solutions a non-starter. I think use of the cloud, published virtual machine images, and templates for required documentation (e.g. for FISMA compliance) would be beneficial here.

5 Votes	Is the NASA culture ready to OSS?
	Cultures are complicated, with systems in place to maintain the status quo. What indicators reveals that the culture is ready? How do you promote, use, develop, etc. OSS within the current culture? What is does the future culture at NASA look like that will continue to adopt/use/dev OSS? Bottom line - is the culture REALLY ready?
Comments:	
March 29, 2011 7:47 PM	I feel that NASA culture is slowly but surely transitioning to one of "openness". Consider its current use of open technologies, like social media. We now are federally mandated to collaborate on human spaceflight ventures with commercial entities. That said, NASA is more than ever quite accepting of open collaboration on thoughts, ideas, and activities, so the groundwork for collaborative software development has been laid.

<p>March 30, 2011 1:37 PM</p>	<p>It starts small, but I imagine that as NASA brings on younger software developers, they will have grown up with public code repos, open source licenses, and all the rest. And they'll expect and demand to use those approaches, as they may well find close-source tools and methods to be huge barriers. If you deter your brightest, most motivated developers, they'll move on. The bureaucracy required to even deploy a project has caused a number of excellent developers I know to leave the NASA mothership, and that's a loss to the Agency; we used to be a place that attracted the best developers.</p>
-----------------------------------	---

<p>4 Votes</p>	<p>Imagination: great data but where are the ideas</p>
<p>Hold an open ideas competition: the NASA Data Challenge, and award prizes to projects that use NASA data / open source code in ways that are able to excite and mobilize the open source community. Use a crowd sourcing model (for the early stages), and a peer reviewed model for the latter stages of the competition. Offer various categories of award to attract a wide range of participants and reward bold and imaginative projects.</p>	
<p>Comments:</p>	
<p>March 29, 2011 7:24 PM</p>	<p>Take a look at JSC's Inclusion and Innovation Initiative. JSC has been doing this type of thing for the past 4 years or so.</p>
<p>March 29, 2011 10:18 PM</p>	<p>Any good links? http://www.NASAhackspace.org/johnson_space_center/Not found anything where, for example, a great app or game came out of a competition?</p>

<p>4 Votes</p>	<p>NASA use of public source code repositories</p>
<p>What, if any, impediments are there to the use of public repositories?</p>	
<p>Comments:</p>	
<p>March 29, 2011 7:57 PM</p>	<p>Code validation is an issue. Attackers are beginning to "post" malicious code in repositories, "app stores", and various other resources. It would be advantageous for NASA to develop a shared (but trusted) repository for source code, utilizing AAA and a code review process. Or, if the use of a public repository has been properly vetted, NASA could host a mirror of that repo.</p>
<p>March 29, 2011 8:15 PM</p>	<p>To clarify ... AAA - authentication, authorization and accounting.</p>
<p>March 30, 2011 1:44 PM</p>	<p>One could use a public repo like GitHub, with a "NASA" organization. The org would accept contributors who had direct commit access. If external contributors cannot (by policy) be trusted (AAA, whatever) then code can be</p>

	pulled from them and vetted by the NASA project maintainers. I don't think it's a technical problem but a political one.
March 30, 2011 4:28 PM	Good point! What do you think about mirror sites at NASA?

4 Votes	USG employees as part of their official duties cannot be copyrighted in the US.
	If USG employees work is solely public domain, how does that work with OSS licenses? Is it a non-issue?
Comments:	
March 29, 2011 5:38 PM	(1) Much of the USG SW development is done by contractors and not public servants (as per the info given in W's talk), and as such isn't subject to automatic entry into the PD.(2) I would imagine that the incorporation of PD code software wouldn't be an issue with many of the standard OSS licenses.

4 Votes	Determination of risk (for software use)
Comments:	
March 29, 2011 8:13 PM	S.T.R.I.D.E.

4 Votes	Community Development
Comments:	

3 Votes	security of software
	How does NASA prevent viruses and trojan horses from being downloaded and installed, transmitted by piggyback on open source? E.g., build and test open source standard loads that are swept for viruses (as we do for proprietary loads before pushing updates)? As a Windows user downloading open source to evaluate it, I don't attempt to review or even recompile the code before installing it. Again, the solution seems to be institutionalizing

support for open source.	
Comments:	
March 29, 2011 6:32 PM	On the contract I work, all OSS software is security scanned prior to being put on any environment that is not considered "destructible." Most mission-critical platforms have not easy way to pull external source (e.g., from the internet) without yards of paperwork and time tables and whatnot. But for the regular production environments, I would imagine that the software has been security/virus scanned as part of its integration into the environments.
March 29, 2011 7:08 PM	We have to use a risk-based approach ... Security should be integrated into each level of the development life cycle. That includes code pulled from external sources. Code review should occur at multiple levels, and applications should be "staged" and scanned prior to being placed into production.
March 30, 2011 1:41 PM	How does NASA prevent virii, trojans, et al from being downloaded in closed source? At the risk of beating the "given enough eyeballs, all bugs are shallow" mantra to death, I find security issues get fixed much quicker in open source than closed. OSS seems to be going thru a very healthy change to heavily depend on automated build and test tools. There's no reason security auditing software can't be made part of that, just as you would with Section 508 conformance testing. And with the source, you have a chance to find insecure code -- something not possible with closed source.

3 Votes	Limitations on the use of open source for mission, flight, human rated systems
	Although this seems like an easy question, the current use of proprietary COTS in these areas is well established. Should that be extended to non-proprietary as well, so that the source code is available, or at lease the unmodified source code? Open source communications stacks are currently used and there are efforts to use full stack advanced protocols to extend the internet to orbital space craft. Should those be Open Source?
Comments:	

2 Votes	Patents
Comments:	
March 28, 2011 1:30 PM	In our current environment, software patents for open source projects seem somewhat hypocritical at first glance, but acquiring a patent for an open source innovation may protect it from being unethically appropriated and patented by

	another. I'm no patent lawyer, though, and may be naive in my assessment (and somewhat paranoid).
March 28, 2011 2:28 PM	Patents and open source are not mutually exclusive. Not at all.
March 29, 2011 7:17 PM	It's generally accepted that one would patent an "idea" - a [somewhat] unique concept ... Whereas, the software itself is copyrighted to prevent its unauthorized or unlawful use (or copying) by others. This has been an ongoing debate (See https://secure.wikimedia.org/wikipedia/en/wiki/Software_patent_debate).

2 Votes	What's the HASHTAG?
	Can someone please post the Twitter HASHTag
Comments:	
March 29, 2011 4:22 PM	Community, I can't log into the chat site? Signup right there doesn't seem to work either. Someone?
March 29, 2011 4:26 PM	We're using #NASAoss

2 Votes	Culture change & hiring practices: open source participation as an explicit desirable on resumes?
	Organizational culture is partly about the attitudes & habits people bring in with them. Could experience participating in open source projects be a checkbox NASA routinely looks for when hiring software engineers? Not necessarily decisive, just worth N out of M points or something like that.
Comments:	
March 29, 2011 6:42 PM	Realization: this is even more important when hiring *managers*.
March 30, 2011 1:50 PM	Great point-- especially on hiring managers. We've sought out folks who've contributed to OSS projects when hiring developers. People who do this code because they enjoy it, and happy coders are more productive (and I expect, "better") than galley-slaves who only code for the paycheck.

1 Votes	Identification of open source possibilities
	There are capabilities that are useful across NASA centers that would likely also have potential for uses outside NASA. However, currently it is difficult to determine ownership and reuse potential. How can projects identify potential open source candidates (at a scope below the list of the entire project)
Comments:	

1 Votes	What are the real-time standards issues for mission-critical projects?
	Submitted on behalf of a physical participant at #NASAoss concerned about realistically addressing the mission ops environment.
Comments:	

1 Votes	Engagement of Remote Participants: Wish you were here?
	Remote (virtual) participation is evolving every day, as new tools and techniques hit the market and people become more comfortable with exchanging ideas via cyberspace and telecon. We'd love to know what really worked during the OSS. What could have been better? We're looking for constructive criticism from the end-users so that engagement can be brought to a new level every time we extend the community beyond the walls of the physical meeting. Since we won't have time to bring this up during the sessions at OSS, let's start the conversation below in the comments.
Comments:	
March 29, 2011 8:44 PM	To save time on introductions, how about having ALL participants (local and remote) provide a head shot, contact info, and a brief bio/resume - info could probably be copied from existing profiles on various Social Networking Services. Brainstorming primarily by voice seemed tedious. Suggest using an environment like Group Systems' ThinkTank, which can support simultaneous idea generation, commenting, and multiple types of voting/ranking.

1 Votes	Artificial gravity and extended sleep time
	Note: This idea was proposed by one of the participants in the NASA Open Source Summit Suggestions to this off-topic area. Please feel free to discuss it here.
Comments:	

1 Votes	Technical choices that facilitate the use of open source software
	How can technology decisions be made that lend themselves to use and creation of open source software?
Comments:	

1 Votes	software quality process
	How do we "buy off" or "certify" open source, or can we just preach about many eyes looking at it... we can run a few tests on a release, but nothing compared to all the users who are working with open source. What do I tell a manager who wants to know how I manage software quality when I compose from open source?
Comments:	

1 Votes	Encouraging NASA scientists and engineers to participate in OSS development
	NASA scientists and engineers should be encouraged to participate in the development of open source software. If a NASA employee fixes a bug in a project, he or she should be thinking about sending a patch to the project's mailing list. Even non-programmers can participate by opening bug reports, answering questions on mailing lists, and writing documentation.
Comments:	

1 Votes	Automatic sharing of all code within 6 months. Take a lesson from the scientists.
	We should take a lesson from the PDS. All science data from NASA missions is mandated to be online within 6 months. The same could be true of all NASA developed Code.
Comments:	

1 Votes	Profit Motive
	While NASA may have an interest in keeping its software open to the public, is profit motive among its contractors a significant obstacle in keeping their software development closed?
Comments:	

--

1 Votes	How do you integrate a continuous review cycle with a continuous release cycle?
	This was raised during the second breakout session regarding section 508 compliance. Restrictions and compliance must be managed by review for inclusion in a main branch of development. A parallel branch of development is for development free from compliance and restriction processes. How would we balance and integrate these cycles: software release and software review?
Comments:	
March 29, 2011 7:46 PM	Yes, why not?
March 29, 2011 8:12 PM	Of course the process would have to be tailored to software development, but we should emulate the FISMA Continuous Monitoring processes for information and information systems. Classify the software (purpose, use, ...), select and implement controls (repeatable peer review procedures), assess those controls, then authorize SW release (via a governance board).Something like that ... ;-)
March 30, 2011 1:47 PM	We've used Continuous Integration systems to check out our code on every commit, build it, run tests, check code coverage, etc (with H, no J). If you can automate testing of things like 508, just build it in. CI systems can be configured to build your main branch targetted for deployment, as well as any more developmenty branches. Quick notification of problems with the code allow developers to fix issues quickly, as the code changes are fresh in their mind.

1 Votes	who funds operations efficiency initiatives?
	Has anyone else found that there is no "customer" at NASA for efficiency initiatives, short of the usual amputation approach? That is, the initiatives that also elevate quality? It feels like swimming upstream.
Comments:	

1 Votes	Use of appropriate toolsets to enable open source collaboration for scientists / engineers
	Engineers and scientists writing code that may lend itself well to open source are often not

	familiar with toolsets that would enable their work. Establishing best practices on proper toolsets or providing feedback from IT savvy developers would help. Knowledge on toolsets for integrating diverse code-types. For example a current project has components written in SQL, Matlab, Excel, and Visual Basic for Applications. Finding toolsets to help manage development from open source collaborators would be extremely helpful.
Comments:	

1 Votes	How would NASA start a software project as open source from the very start?
	If there is no code to review because none has been written yet, how would NASA create a software project from scratch with the community?
Comments:	
March 30, 2011 1:52 PM	Just host the code repo on a public repo like GitHub from the get-go. Instead of a private repo inside a Center or Agency walled garden. Assuming we could get a policy pronouncement that this is OK. Who you allow to contribute to your project is still up to you. It just becomes available for others to review, contribute to, leverage in their own projects, etc.

0 Votes	derivative works require review by Office of General Counsel of all component licenses
	Certain common scenarios (e.g. GNU tool chain, use of GCC and various libraries) can have "canned reviews" and policies for what form of license is appropriate.
Comments:	

Communication/Press Summary

News Media Attendees

- Alex Howard, O'Reilly Media
- Chris Preimesberger, eWeek
- Mike Wall, space.com
- Sean Gallagher, FedTech
- Cade Metz, The Register
- Florian Vieru, PC World France

Top News Media Coverage

- "NASA hosts its first Open Source Summit"
<http://www.eweek.com/c/a/Government-IT/NASA-Hosts-Its-First-Open-Source-Summit-326695/>
- "NASA concludes first Open Source Summit, aims to make openness the default"
<http://opensource.com/life/11/3/NASA-concludes-first-open-source-summit-aims-make-openness-default>
- "NASA Open Source Summit"
<http://NASAwatch.com/archives/2011/03/NASA-open-sourc.html>
- "2011 NASA Open Source Summit convenes innovators and technologists"
<http://gov20.govfresh.com/2011-NASA-open-source-summit-convenes-innovators-and-technologists/>
- "Space 2.0: NASA's Open Source Summit"
<http://shareable.net/blog/space-20-NASAs-open-source-summit>
- "NASA's Inaugural Open Source Summit"
<http://harishpillay.wordpress.com/2011/03/31/NASAs-inaugural-open-source-summit/>
- "NASA set to host open source summit"
<http://www.webpronews.com/open-source-NASA-2011-03>

Snapshot

Press Release: issued Friday, March 11 ~2:30pm EST

Media Links on Google (as of 3/31): ~106,000

Tweets & Re-Tweets (as of 3/14): ~1,255

Top Twitter Users:

- @NASAWatch

- @digiphile
- @opensourceway
- @FOSSwiki
- @RedHatGov
- @govfresh

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE September 2012	3. REPORT TYPE AND DATES COVERED NASA Conference Publication		
4. TITLE AND SUBTITLE Open Source Summit Proceedings March 29-30, 2011			5. FUNDING NUMBERS	
6. AUTHOR(S) Nicholas Skytland				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Lyndon B. Johnson Space Center Houston, Texas 77058			8. PERFORMING ORGANIZATION REPORT NUMBERS S-1128	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, DC 20546-0001			10. SPONSORING/MONITORING AGENCY REPORT NUMBER CP-2012-217366	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Available from the NASA Center for AeroSpace Information (CASI) 7121 Standard Hanover, MD 21076-1320 Category: 61			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) On March 29 & 30, 2011, NASA hosted its first Open Source Summit (OSS) at Ames Research Center in Mountain View, California. The event brought engineers and policy makers from across NASA together with well-respected members of the open source community to discuss current challenges with NASA's open source policy framework, and propose modifications that would make it easier for NASA to develop, release, and use open source software. Open source brings numerous benefits to NASA software projects, including increased software quality, reduced development costs, faster development cycles, and reduced barriers to public-private collaboration through new opportunities to commercialize NASA technology. Although open source release has already provided some potential benefits, the full benefits can only be realized if NASA is able to establish processes, policies, and culture needed to encourage and support open source development. This will require expanding open source activities beyond releasing software only after completion and finding new ways to support two-way collaboration with an open development community throughout the software lifecycle. This document provides a snapshot of the activities during the OSS and summarizes the major issues and recommendations received from in-person attendees, as well as through the various online venues used during the event.				
14. SUBJECT TERMS open source licensing; software development tools; internet resources			15. NUMBER OF PAGES 92	16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT Unlimited	
