Free/Open Source Software Development as an Approach to Global Software Engineering

Walt Scacchi

Institute for Software Research

University of California, Irvine

6th. International Conference on Global Software Engineering

Helsinki, Finland

16 August 2011

http://www.ics.uci.edu/~wscacchi/Presentations/ICGSE2011.pdf

Overview

- Background
- Individual participation
- Resources supporting activities
- Cooperation, coordination and control
- Alliances and social networking across projects
- FOSS as multi-project software ecosystems
- FOSS as social movement
- Discussion and limitations
- Research opportunities

Background

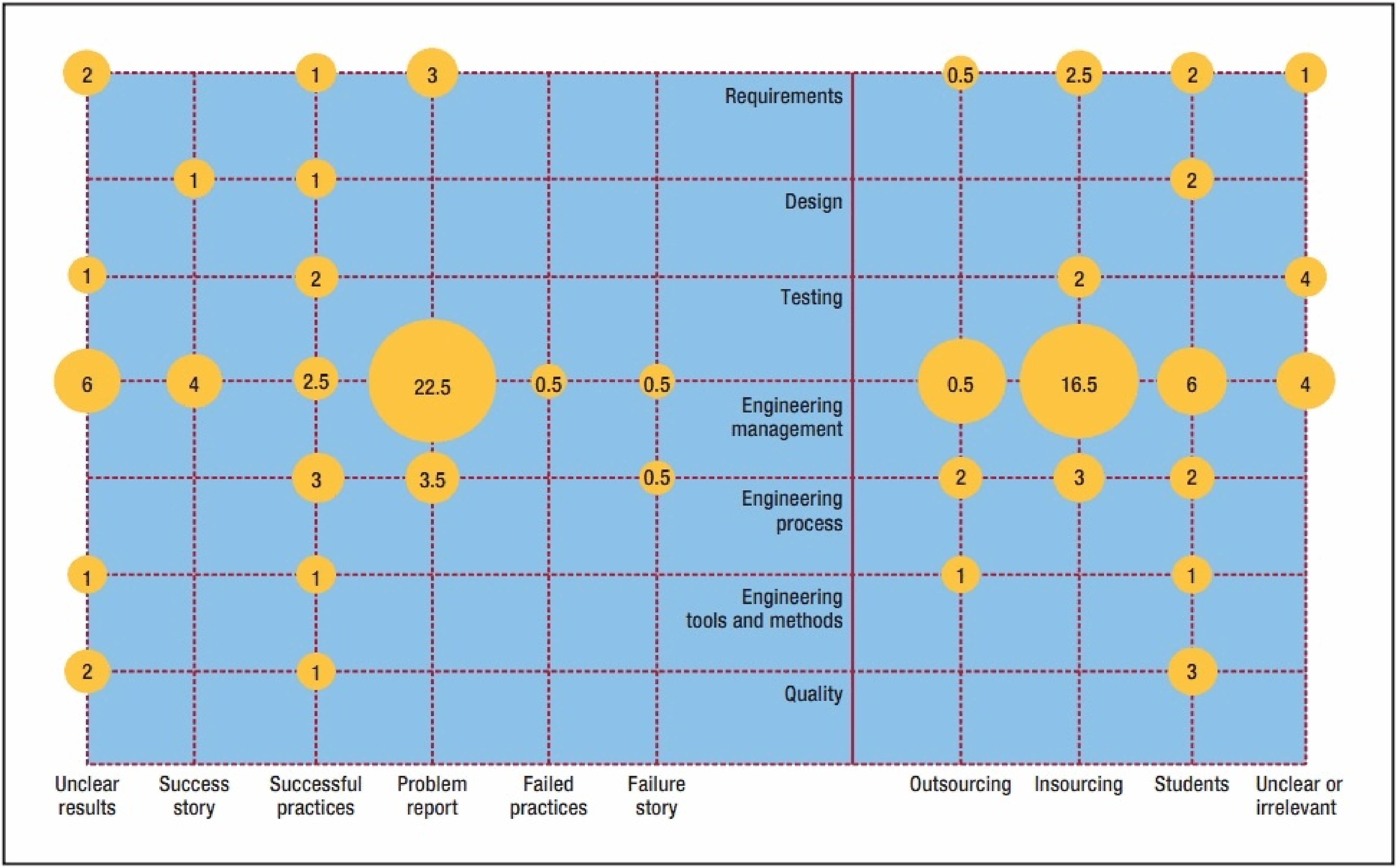
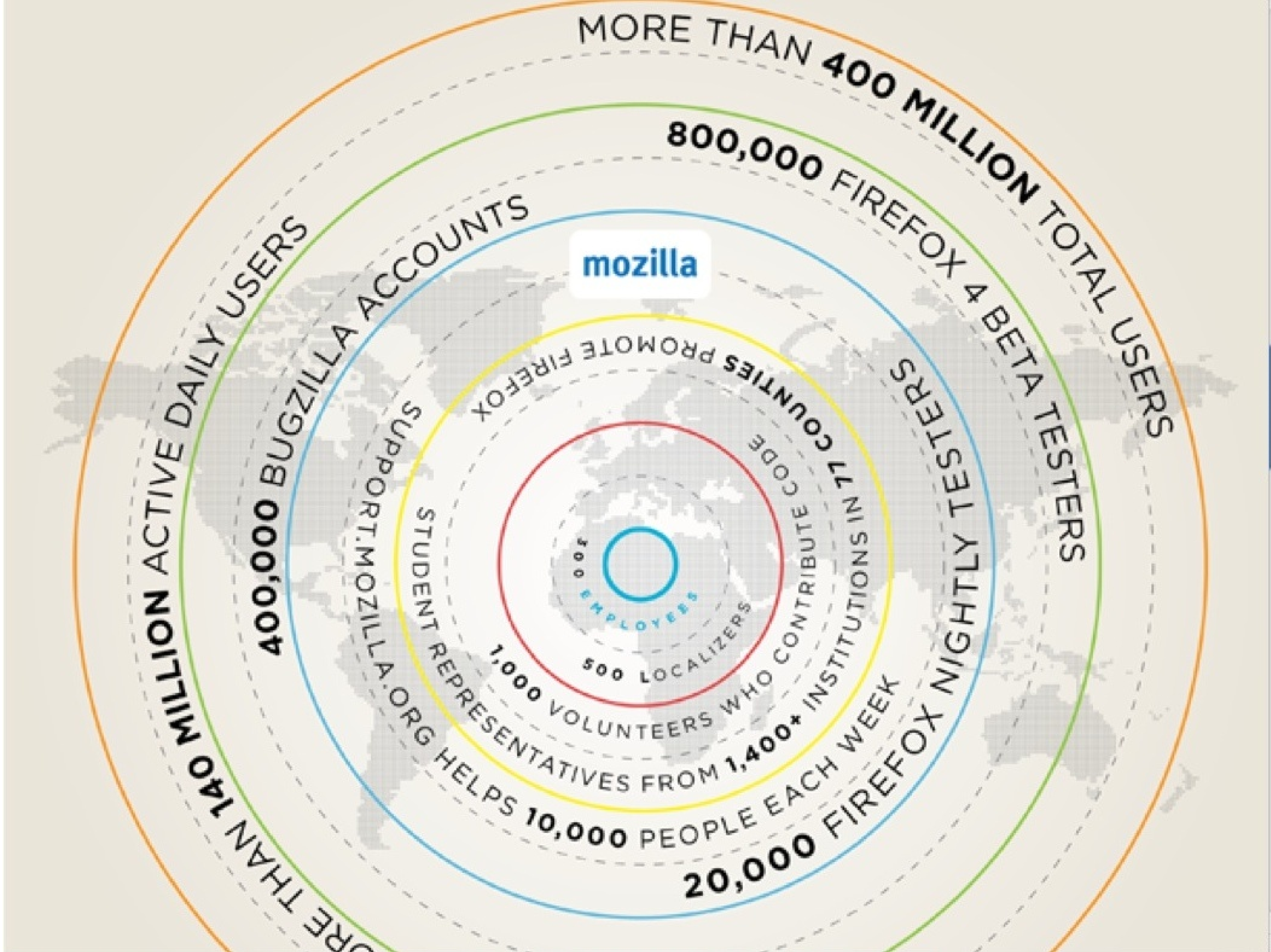


FIGURE 1. Bubble-plot overview of what we know about global software engineering (GSE). Results are based on a systematic review of the GSE literature available from 2000 to 2007.[4] The left side classifies the 59 relevant studies thematically in terms of success or failure, and the right side classifies them according to globalization type.

Smite, D. and Wohlin, . (2011). A Whisper of Evidence in Global Software Engineering, *IEEE Software*, 28(4), 15-18, July-August.

What is free/open source software development?

- Free (as in “freedom” or liberty) vs. open source
  - Freedom to access, browse/view, study, modify and redistribute the source code
  - Free is always open, but open source is not always free
- FOSSD is not “software engineering”
  - *Different*: FOSSD can be faster, better, and cheaper than SE in some circumstances
  - FOSSD teams use 10-500+ OSSD tools (versions) and communications applications to support their development work



MORE THAN **400 MILLION** TOTAL USERS

ORE THAN **140 MILLION** ACTIVE DAILY USERS

800,000 FIREFOX 4 BETA TESTERS

400,000 BUGZILLA ACCOUNTS

20,000 FIREFOX NIGHTLY TESTERS

mozilla

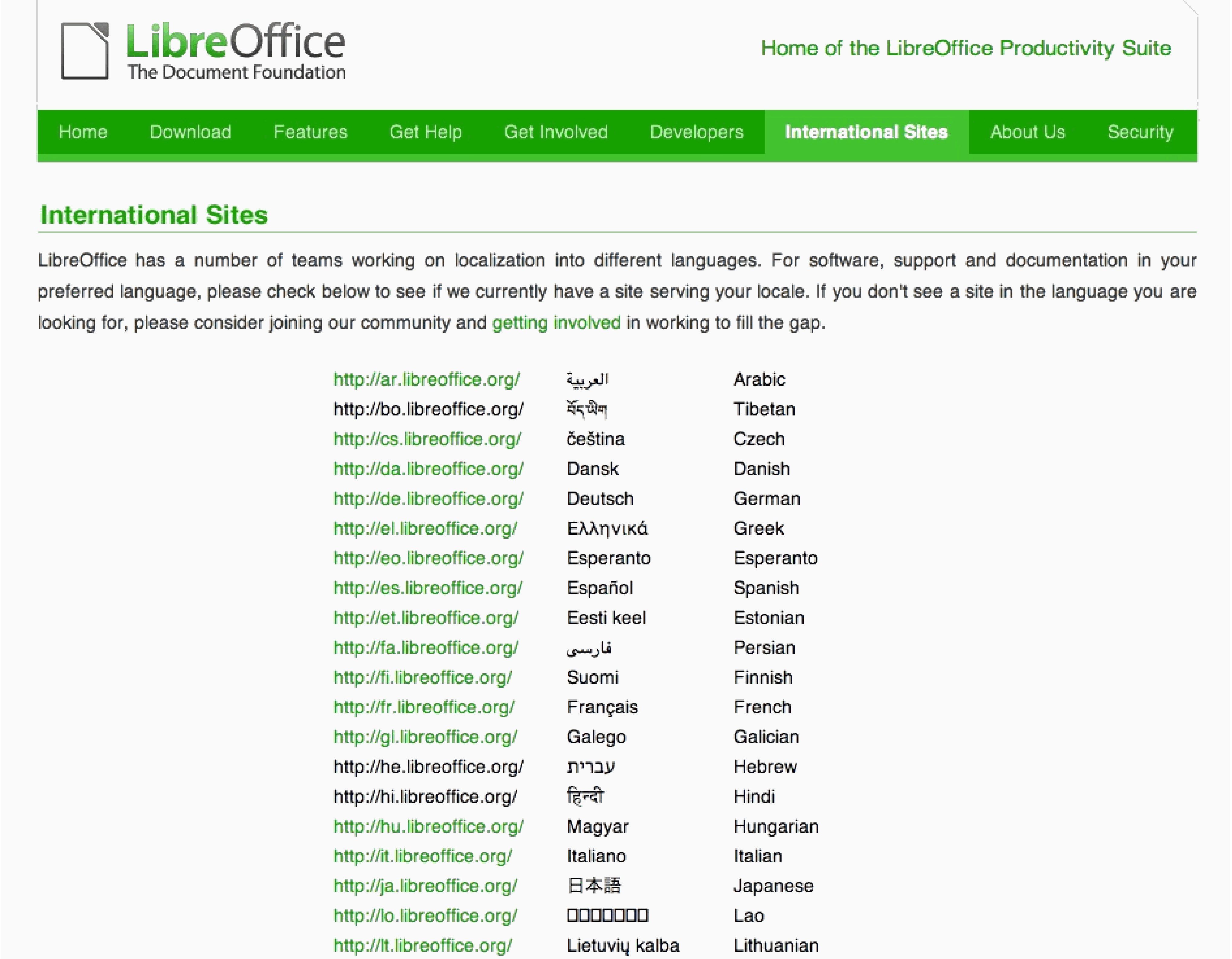
SUPPORT.MOZILLA.ORG HELPS **10,000** PEOPLE EACH WEEK

STUDENT REPRESENTATIVES FROM **1,400+** INSTITUTIONS IN **77 COUNTIES** PROMOTE FIREFOX

1,000 VOLUNTEERS WHO CONTRIBUTE CODE

500 LOCALIZERS

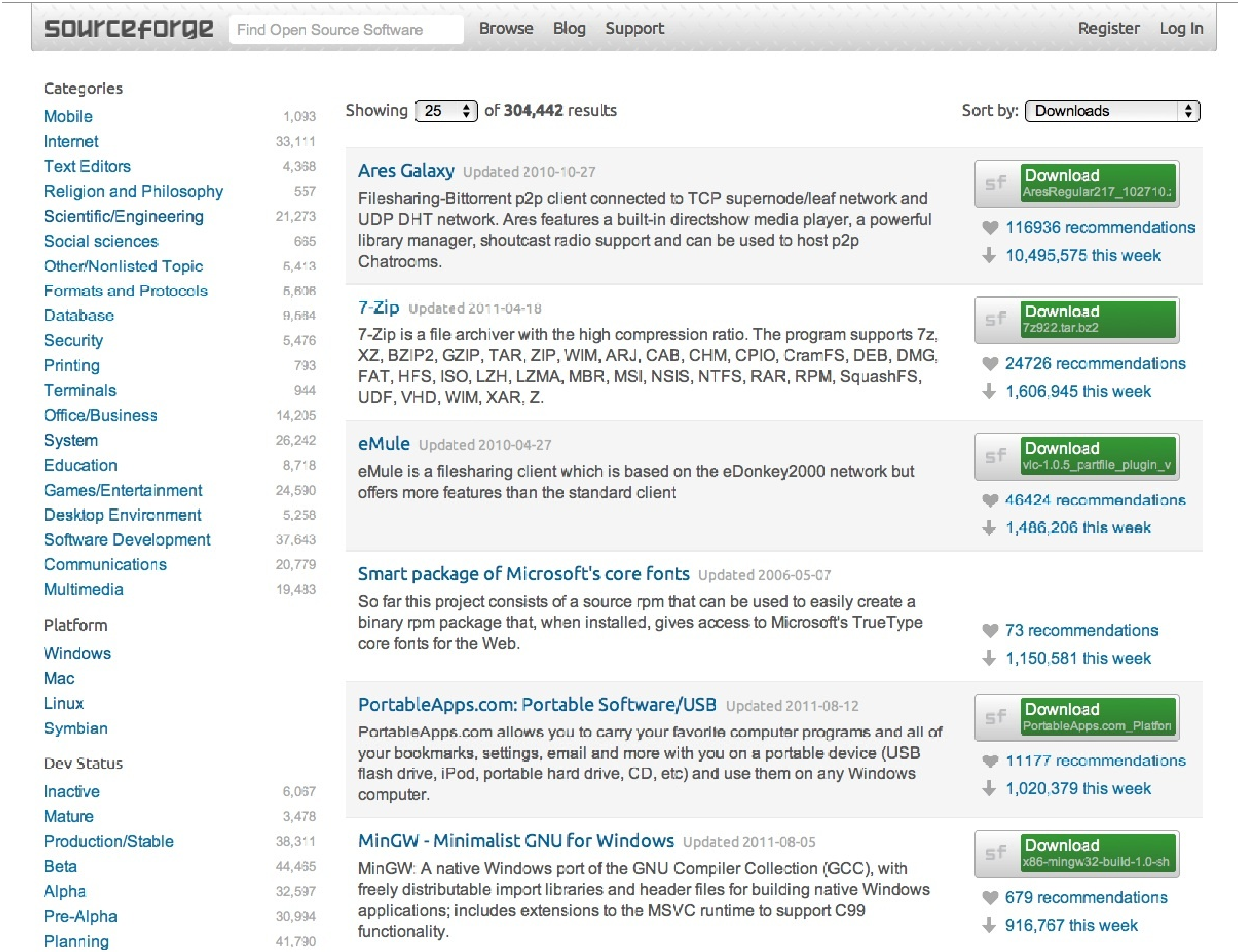
300 EMPLOYEES



International Sites

LibreOffice has a number of teams working on localization into different languages. For software, support and documentation in your preferred language, please check below to see if we currently have a site serving your locale. If you don't see a site in the language you are looking for, please consider joining our community and getting involved in working to fill the gap.

http://ar.libreoffice.org/	العربية	Arabic
http://bo.libreoffice.org/	བོད་ཡིག	Tibetan
http://cs.libreoffice.org/	čeština	Czech
http://da.libreoffice.org/	Dansk	Danish
http://de.libreoffice.org/	Deutsch	German
http://el.libreoffice.org/	Ελληνικά	Greek
http://eo.libreoffice.org/	Esperanto	Esperanto
http://es.libreoffice.org/	Español	Spanish
http://et.libreoffice.org/	Eesti keel	Estonian
http://fa.libreoffice.org/	فارسی	Persian
http://fi.libreoffice.org/	Suomi	Finnish
http://fr.libreoffice.org/	Français	French
http://gl.libreoffice.org/	Galego	Galician
http://he.libreoffice.org/	עברית	Hebrew
http://hi.libreoffice.org/	हिन्दी	Hindi
http://hu.libreoffice.org/	Magyar	Hungarian
http://it.libreoffice.org/	Italiano	Italian
http://ja.libreoffice.org/	日本語	Japanese
http://lo.libreoffice.org/	□□□□□□□	Lao
http://lt.libreoffice.org/	Lietuvių kalba	Lithuanian
]



Categories

Mobile	1,093
Internet	33,111
Text Editors	4,368
Religion and Philosophy	557
Scientific/Engineering	21,273
Social sciences	665
Other/Nonlisted Topic	5,413
Formats and Protocols	5,606
Database	9,564
Security	5,476
Printing	793
Terminals	944
Office/Business	14,205
System	26,242
Education	8,718
Games/Entertainment	24,590
Desktop Environment	5,258
Software Development	37,643
Communications	20,779
Multimedia	19,483

Platform

- Windows
- Mac
- Linux
- Symbian

Dev Status

Inactive	6,067
Mature	3,478
Production/Stable	38,311
Beta	44,465
Alpha	32,597
Pre-Alpha	30,994
Planning	41,790

Showing 25 of **304,442** results

Sort by: Downloads

Ares Galaxy Updated 2010-10-27

Filesharing-Bittorrent p2p client connected to TCP supernode/leaf network and UDP DHT network. Ares features a built-in directshow media player, a powerful library manager, shoutcast radio support and can be used to host p2p Chatrooms.

Download AresRegular217_102710.:

116936 recommendations

10,495,575 this week

7-Zip Updated 2011-04-18

7-Zip is a file archiver with the high compression ratio. The program supports 7z, XZ, BZIP2, GZIP, TAR, ZIP, WIM, ARJ, CAB, CHM, CPIO, CramFS, DEB, DMG, FAT, HFS, ISO, LZH, LZMA, MBR, MSI, NSIS, NTFS, RAR, RPM, SquashFS, UDF, VHD, WIM, XAR, Z.

Download 7z922.tar.bz2

24726 recommendations

1,606,945 this week

eMule Updated 2010-04-27

eMule is a filesharing client which is based on the eDonkey2000 network but offers more features than the standard client

Download vlc-1.0.5_partfile_plugin_v

46424 recommendations

1,486,206 this week

Smart package of Microsoft's core fonts Updated 2006-05-07

So far this project consists of a source rpm that can be used to easily create a binary rpm package that, when installed, gives access to Microsoft's TrueType core fonts for the Web.

73 recommendations

1,150,581 this week

PortableApps.com: Portable Software/USB Updated 2011-08-12

PortableApps.com allows you to carry your favorite computer programs and all of your bookmarks, settings, email and more with you on a portable device (USB flash drive, iPod, portable hard drive, CD, etc) and use them on any Windows computer.

Download PortableApps.com_Platforı

11177 recommendations

1,020,379 this week

MinGW - Minimalist GNU for Windows Updated 2011-08-05

MinGW: A native Windows port of the GNU Compiler Collection (GCC), with freely distributable import libraries and header files for building native Windows applications; includes extensions to the MSVC runtime to support C99 functionality.

Download x86-mingw32-build-1.0-sh

679 recommendations

916,767 this week

FOSSD Project Characteristics

- Operational code early and often--actively improved and continuously adapted
  - Short-cycle (FOSS) vs. long-cycle (SLC) time processes
- *Post-facto* software system requirements and design
  - FOSSD has its own “-ilities” which differ from those for SE
- *Caution*: the vast majority (>90%) of FOSSD projects fail to grow or to produce a viable, sustained software release.

FOSSD Project Characteristics

- FOSS developers are typically users of what they build, while FOSS users (~1%) are also FOSS developers
- Requires "*critical mass*" of contributors and FOSS components connected through socio-technical interaction networks
- FOSSD projects can emerge/evolve via *bricolage*
  - Unanticipated architectural (de)compositions
  - Multi-project component integrations

OSS Development Models

- Free Software (GPL)
- Permissive Open Source (BSD/MIT, FreeBSD)
- Corporate/Inner Source (Hewlett-Packard)
- Consortium/Alliance (OSDL, SugarCRM)
- Non-profit foundations (Apache, Mozilla, Gnome, Perl)
- Corporate-Sponsored (Google, HP, IBM, Microsoft, Nokia, Oracle)
- Open Modding Extensions to Closed Source (many game companies)
- Community Source (Sakai, Westwood)

----------------------------------- not OSSD models below -----------------------

- Shared Source with Non-Disclosure (Microsoft)
- Open Systems (open APIs, closed components)

Research methodology

- Comparative (case) studies
  - Multiple open software development projects
    - Scaling: individual projects; interrelated projects; project domain clusters; global project populations (1K-50K+ projects)
- Qualitative, quantitative, hybrid techniques
- Analyzing and modeling
  - development processes, work practices and roles, project forms and community networks
  - development artifacts and tools
  - FOSS evolution
  - FOSS ecosystems

Individual participation

Individual participation in FOSSD projects: motives and consequences

- FOSS developers want to:
  - learn about new tools, techniques, skills, etc.
  - have fun building software
  - exercise their technical skill
  - try out new kinds of systems to develop
  - interconnect multiple FOSSD projects
- FOSS developers frequently:
  - build trust and reputation with one another
  - achieve “geek fame” (for project leaders)
  - spend more time reading online documents and communicating with one another than writing code

Resources supporting FOSS activities

FOSSD resources/capabilities

- Personal software development resources
- Beliefs supporting FOSSD
- FOSSD informalisms
- Skilled, self-organizing developers
- Discretionary time and effort
- Trust and social accountability

Personal software development resources

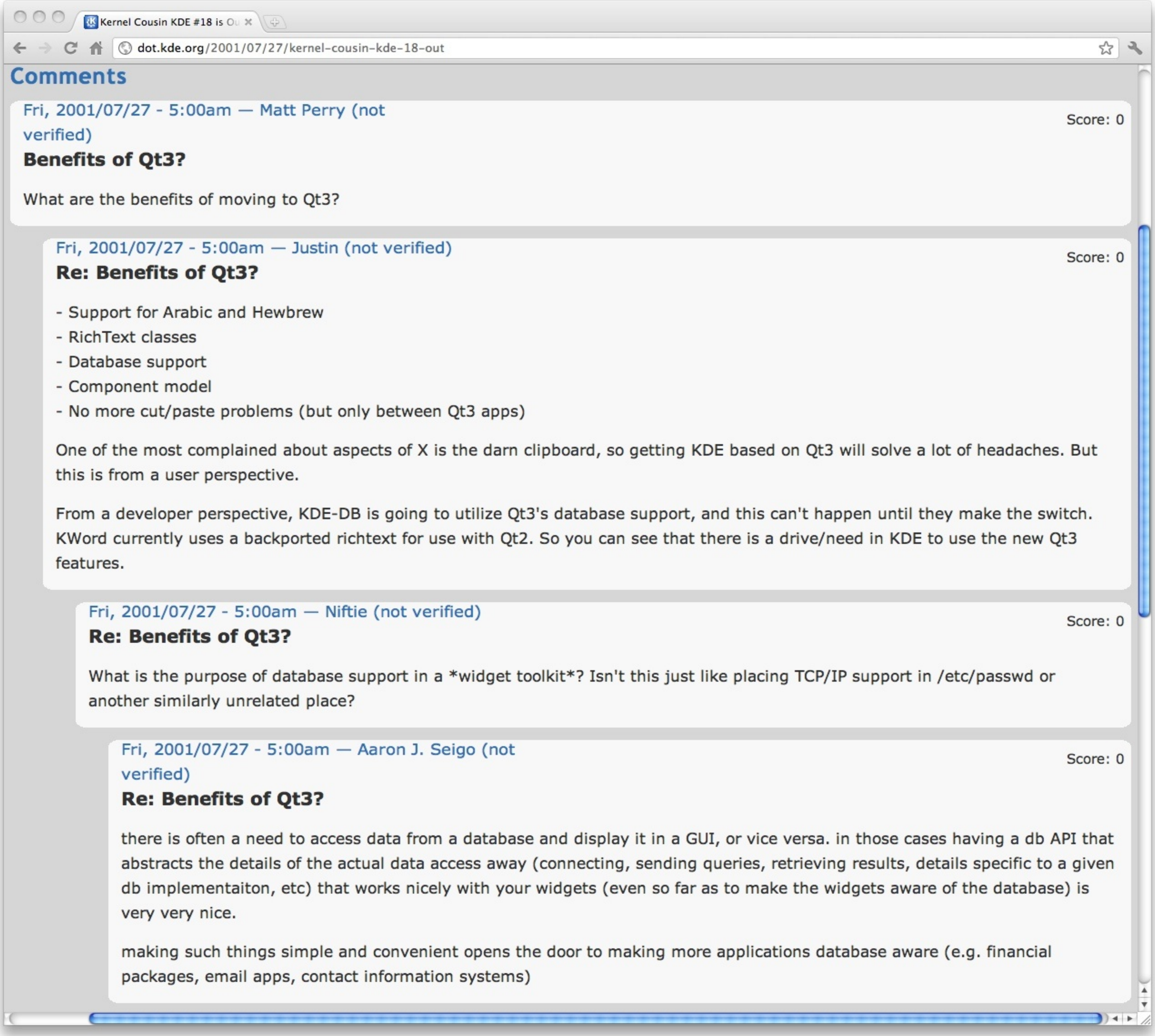
- Sustained commitment of personal resources helps *subsidize* FOSSD projects
  - Personal computer(s)
  - Internet access
  - Hosting personal Web site
  - Hosting project repositories
  - Personal choice of software development tools or tool set

Beliefs supporting FOSSD

- *Freedom of expression*
  - What to develop or work on
  - How to develop it
  - What tools to employ
- *Freedom of choice*
  - When to release work products
  - Expressing what can be said to whom with or without reservation
- Observation: *Beliefs shape architecture*

FOSSD Informalisms

- Software *informalisms*--artifacts participants use to describe, proscribe, or prescribe what's happening in a project
- Informalisms capture detailed rationale and debates for what changes were made in particular development activities, artifacts, or source code files


Kernel Cousin KDE #18 is Ou
dot.kde.org/2001/07/27/kernel-cousin-kde-18-out

Comments

Fri, 2001/07/27 - 5:00am — Matt Perry (not verified)

Score: 0

Benefits of Qt3?

What are the benefits of moving to Qt3?

Fri, 2001/07/27 - 5:00am — Justin (not verified)

Score: 0

Re: Benefits of Qt3?

- Support for Arabic and Hewbrew
- RichText classes
- Database support
- Component model
- No more cut/paste problems (but only between Qt3 apps)

One of the most complained about aspects of X is the darn clipboard, so getting KDE based on Qt3 will solve a lot of headaches. But this is from a user perspective.

From a developer perspective, KDE-DB is going to utilize Qt3's database support, and this can't happen until they make the switch. KWord currently uses a backported richtext for use with Qt2. So you can see that there is a drive/need in KDE to use the new Qt3 features.

Fri, 2001/07/27 - 5:00am — Niftie (not verified)

Score: 0

Re: Benefits of Qt3?

What is the purpose of database support in a *widget toolkit*? Isn't this just like placing TCP/IP support in /etc/passwd or another similarly unrelated place?

Fri, 2001/07/27 - 5:00am — Aaron J. Seigo (not verified)

Score: 0

Re: Benefits of Qt3?

there is often a need to access data from a database and display it in a GUI, or vice versa. in those cases having a db API that abstracts the details of the actual data access away (connecting, sending queries, retrieving results, details specific to a given db implementaiton, etc) that works nicely with your widgets (even so far as to make the widgets aware of the database) is very very nice.

making such things simple and convenient opens the door to making more applications database aware (e.g. financial packages, email apps, contact information systems)

FOSSD informalisms

- Email lists
- Discussion forum
- News postings
- Project digests
- IM/Internet Relay Chat
- Scenarios of usage
- How-to guides
- Screenshots
- FAQs; to-do lists: item lists
- Project Wikis
- System documentation
- External publications
- Copyright licenses
- Architecture diagrams
- Intra-app scripting
- Plug-ins
- Code from other projects
- Project Web site
- Multi-project portals
- Project source code
- Project repositories
- Software bug reports
- Issue tracking databases
- Blogs, videos, photos, etc.

Skilled, self-organizing developers

- Successfully developing an open architecture system requires prior experience
- Organizing project work as a *virtual organization*
  - Skill-based meritocracy
  - Informal rules of governance and control, but rules are readily recognized by participants
  - Social control incorporated into software and informalisms
    - How, where, and when to access data via APIs, UIs, and other architectural features

Discretionary time and effort

- Self-determination
  - work on what’s interesting
- Peer recognition
  - becoming a social gateway
- Project affiliation or identification
- Self-promotion
  - How to realize career advancement
- Belief in inherent value of FOSS

Trust and social accountability

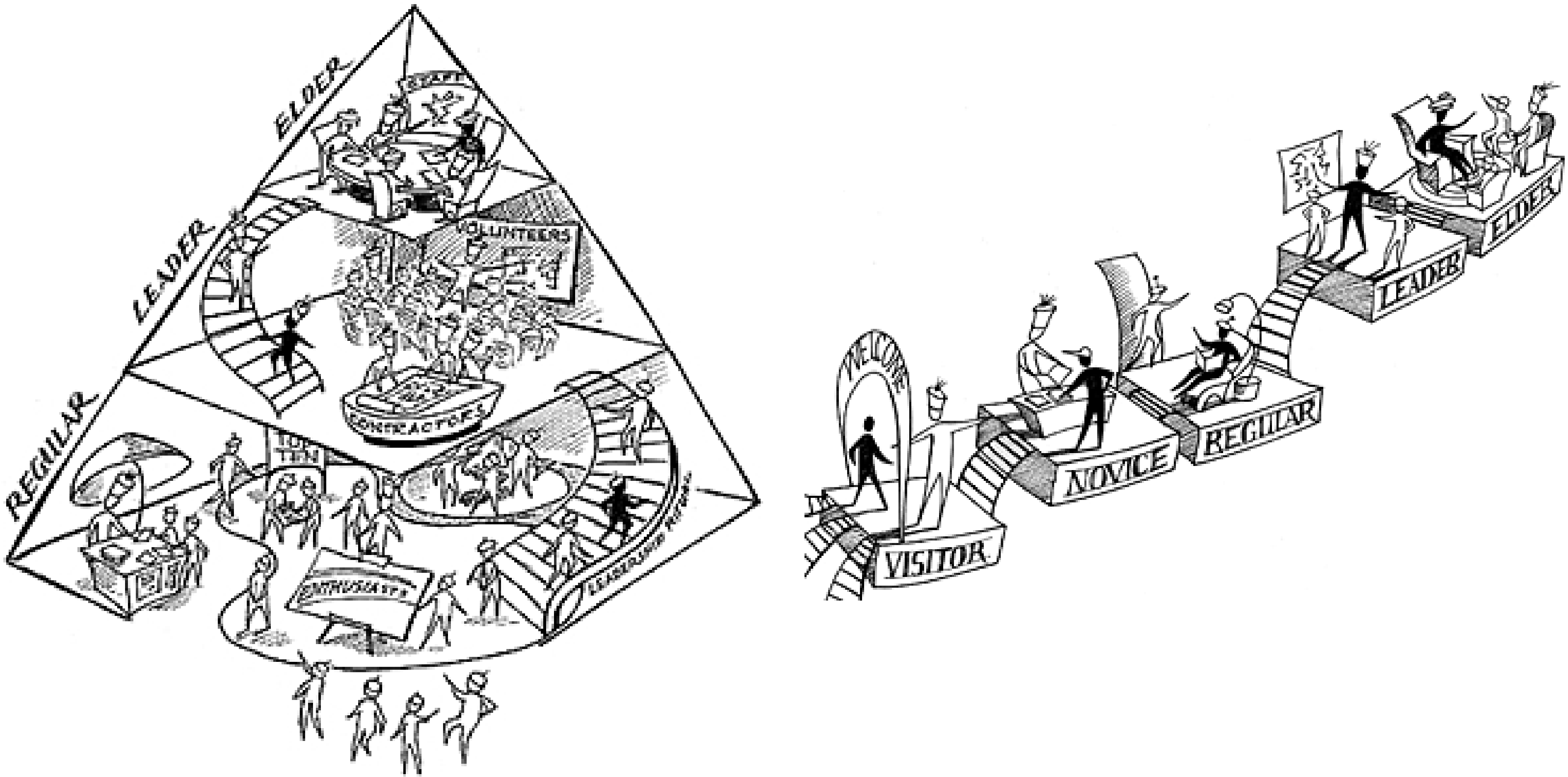
- *Social capital* accrues via:
  - Assuming ownership of a FOSS module
  - Voting on approval of other’s actions
  - Shared peer reviewing
  - Contributing “gifts” that are reusable
- Accrued social capital is used to mitigate conflicts and accommodate resolutions
- Sustained social capital enables social networking externalities
- Shared investment of social capital as basis for trust

Cooperation, coordination, and control in FOSSD projects

Software version control

- Enables stabilization and synchronization of dispersed, invisible FOSSD work
- SVC tools (CVS, SVN, Git, etc.) used as:
    - Central mechanism coordinating development
    - Online venue for mediating control over what changes will be accommodated
    - Gentle but sufficient social control mechanism that constrains overall project complexity

A meritocractic role hierarchy and role migration paths for FOSSD



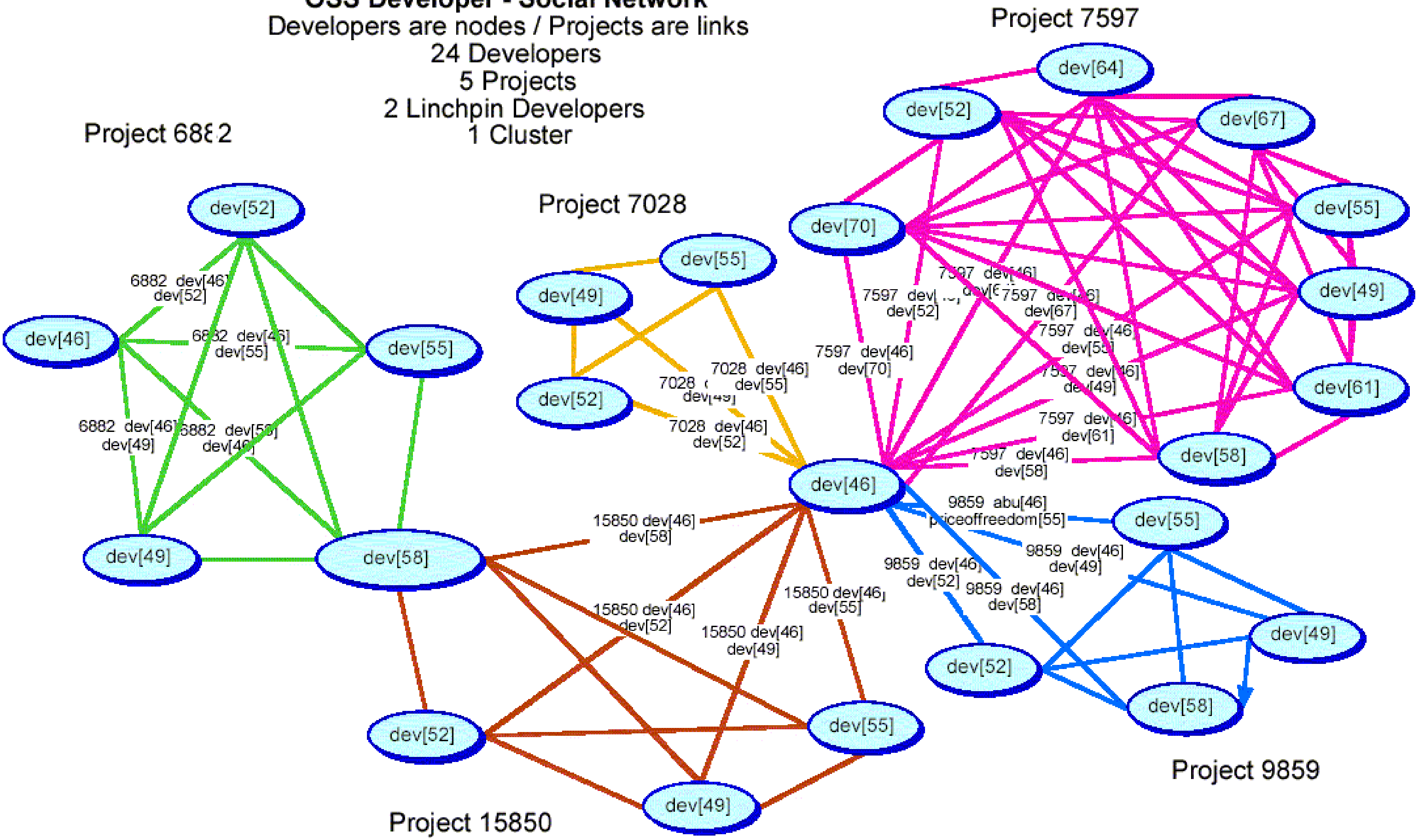
Implicit project management

- FOSSD projects self-organize as a *meritocractic role-hierarchy* and *virtual project management*
    - Meritocracies embrace incremental innovations over radical innovations
    - VPM requires people to act in leadership roles based on skill, availability, and belief in project community
- Reliance on evolving web of software informalism content constrains collective action within FOSSD project

Alliances, social networking, and community development

OSS Developer - Social Network

Developers are nodes / Projects are links
24 Developers
5 Projects
2 Linchpin Developers
1 Cluster



Source: G. Madey, *et al.,* 2005

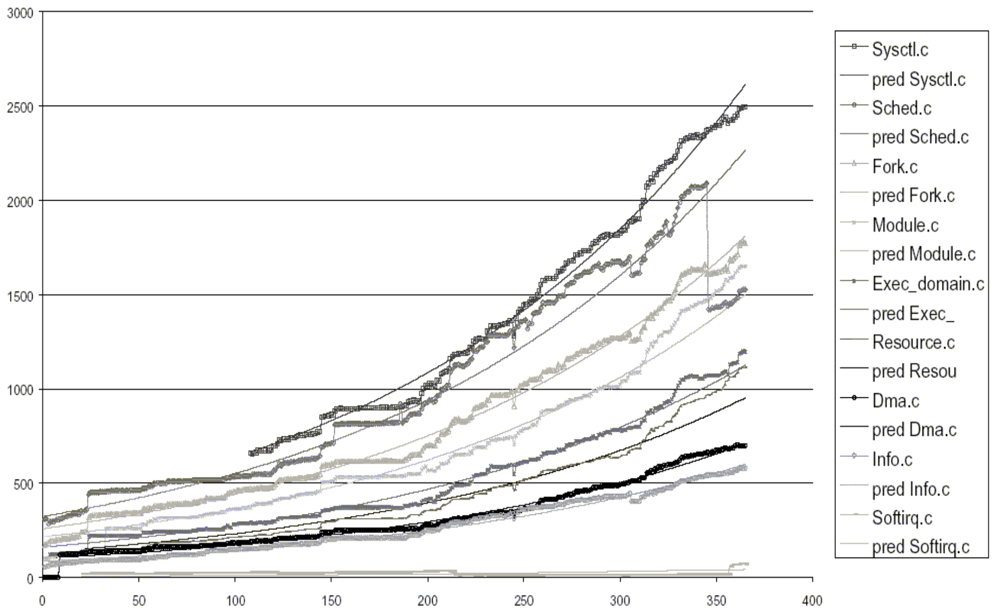
Community networking

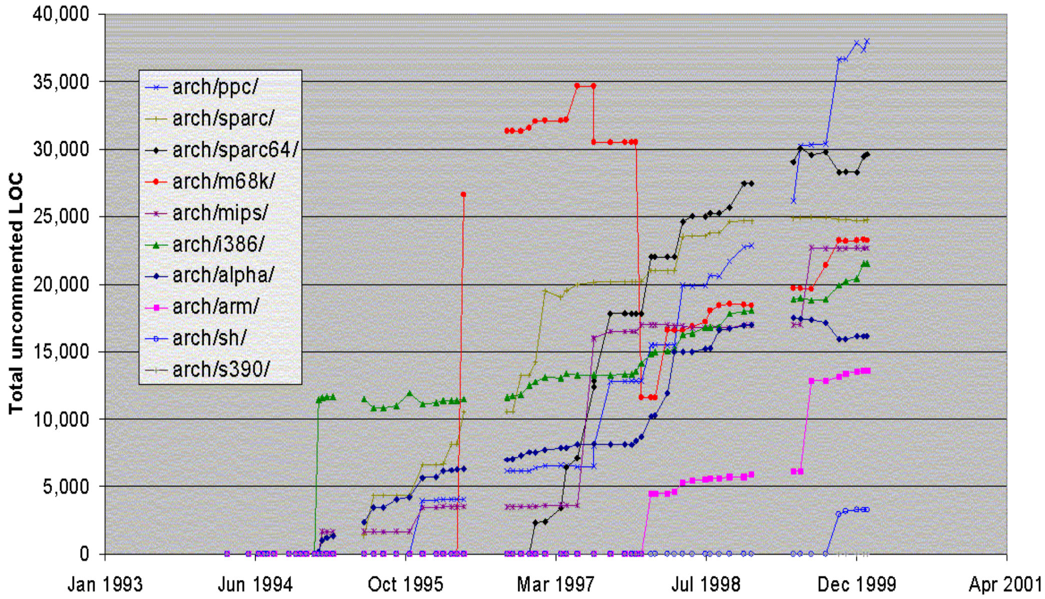
- Becoming a central node in a network of FOSS developers increases social capital
  - *Linchpin developers* as social gateways
  - Sharing beliefs, tools, artifacts enables shared experience, camaraderie, collective learning
- Multi-project clustering enables small projects to merge into sustainable projects
- Intellectual property regime fosters alignment and alliance with other projects and organizations

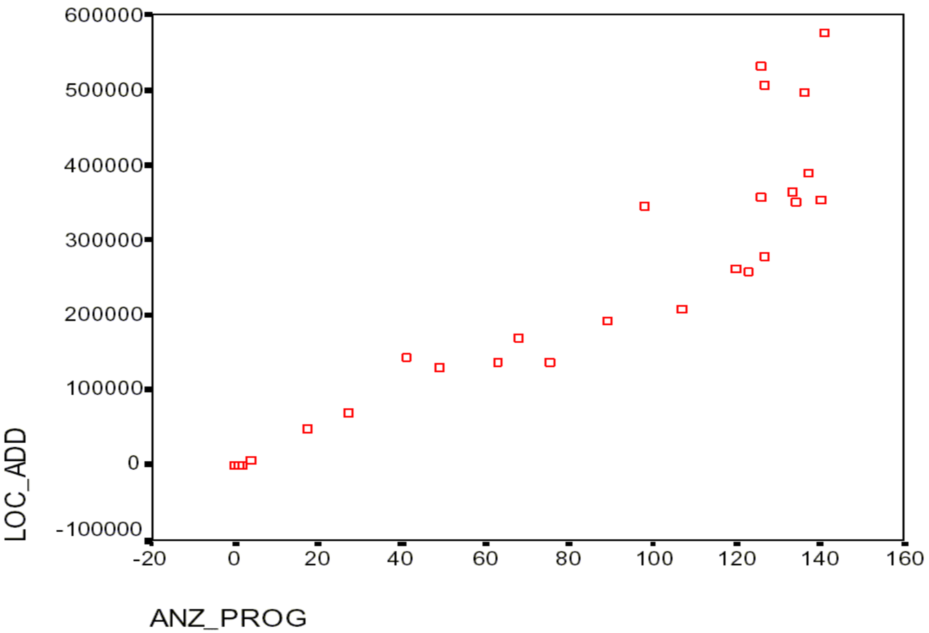
FOSS as multi-project software ecosystems

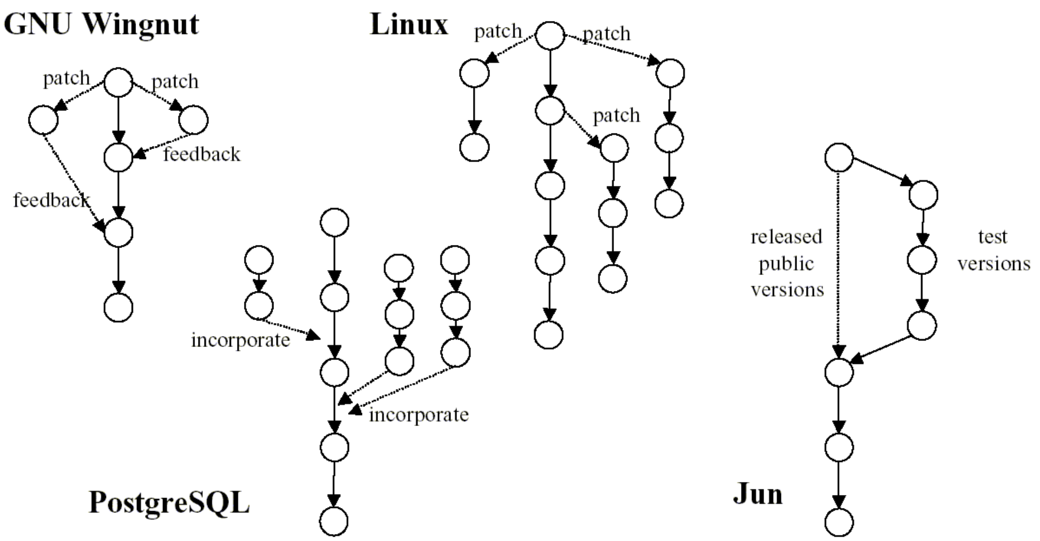
Multi-project software ecosystem

- Mutually dependent FOSS development and evolution propagate architectural styles, dependencies, and vulnerabilities
- *Architectural bricolage* arises when autonomous FOSSD projects, artifacts, tools, and systems co-mingle or merge
  - Enables discontinuous or exponential growth of FOSS source code, functionality, complexity, contributions


3000
2500
2000
1500
1000
500
0
0
50
100
150
200
250
300
350
400
Sysctl.c
pred Sysctl.c
Sched.c
pred Sched.c
Fork.c
pred Fork.c
Module.c
pred Module.c
Exec_domain.c
pred Exec_
Resource.c
pred Resou
Dma.c
pred Dma.c
Info.c
pred Info.c
Softirq.c
pred Softirq.c


40,000
35,000
30,000
25,000
20,000
15,000
10,000
5,000
0
Total uncommented LOC
Jan 1993
Jun 1994
Oct 1995
Mar 1997
Jul 1998
Dec 1999
Apr 2001
arch/ppc/
arch/sparc/
arch/sparc64/
arch/m68k/
arch/mips/
arch/i386/
arch/alpha/
arch/arm/
arch/sh/
arch/s390/


600000
500000
400000
300000
200000
100000
0
-100000
LOC_ADD
-20
0
20
40
60
80
100
120
140
160
ANZ_PROG


GNU Wingnut
patch
patch
feedback
feedback
Linux
patch
patch
patch
incorporate
incorporate
PostgreSQL
released
public
versions
test
versions
Jun

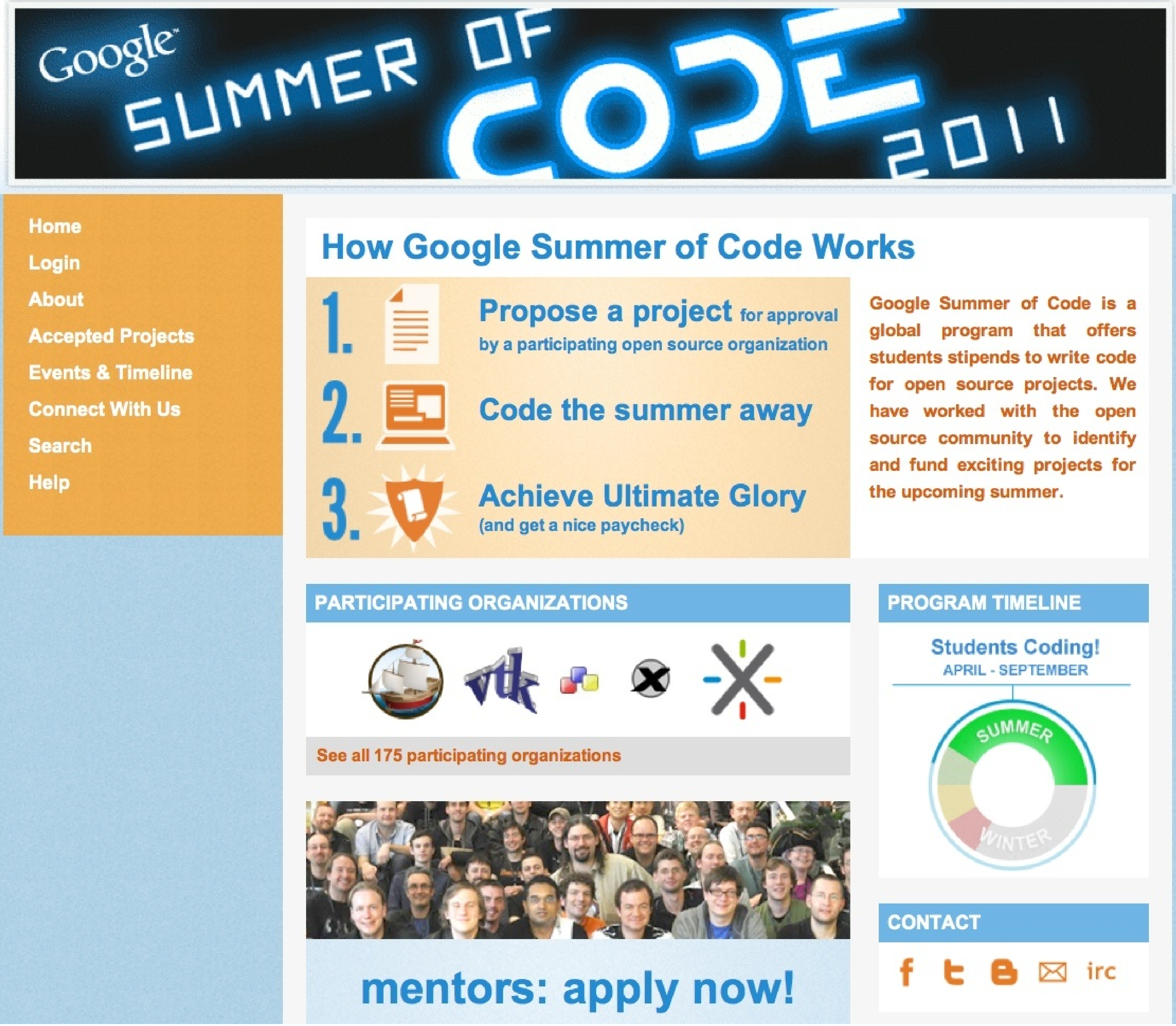
Evolutionary redevelopment, reinvention, and redistribution

- Overall evolutionary dynamic of many FOSSD projects is *reinvention and redevelopment*
  - Reinvention enables continuous improvement and collective learning
- FOSS evolve through minor mutations
  - Expressed, recombined, redistributed via incremental releases
- FOSS systems *co-evolve* with their development community
  - Success of one depends on the success of the other

FOSS as social movement

FOSS as social movement

- Free/OSS property regimes and licenses
  - Reiterate and institutionalize FOSS *culture* (values, norms, and beliefs)
    - GNU Public License (GPL) for *free* software
    - More than 50 other open source licenses at the Open Source Initiative,
    - “Creative Commons” Project at Stanford Law School developing public license framework
- Interest in FOSS spans multiple disciplines, institutions, nations, and cultures



How Google Summer of Code Works

1. **Propose a project** for approval by a participating open source organization
2. **Code the summer away**
3. **Achieve Ultimate Glory** (and get a nice paycheck)

Google Summer of Code is a global program that offers students stipends to write code for open source projects. We have worked with the open source community to identify and fund exciting projects for the upcoming summer.

PARTICIPATING ORGANIZATIONS

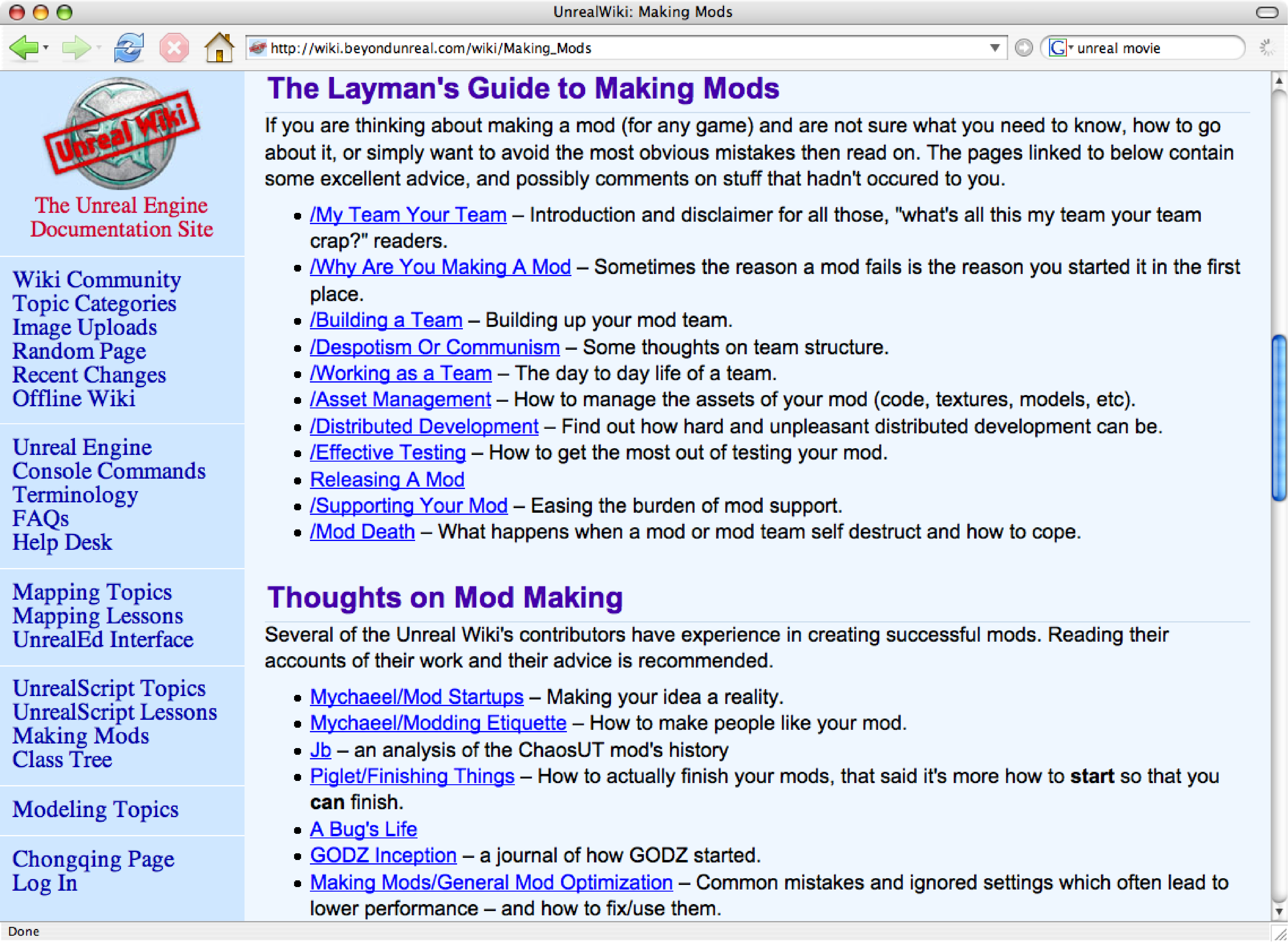
See all 175 participating organizations

PROGRAM TIMELINE

Students Coding!
APRIL - SEPTEMBER

CONTACT

mentors: apply now!



The Layman's Guide to Making Mods

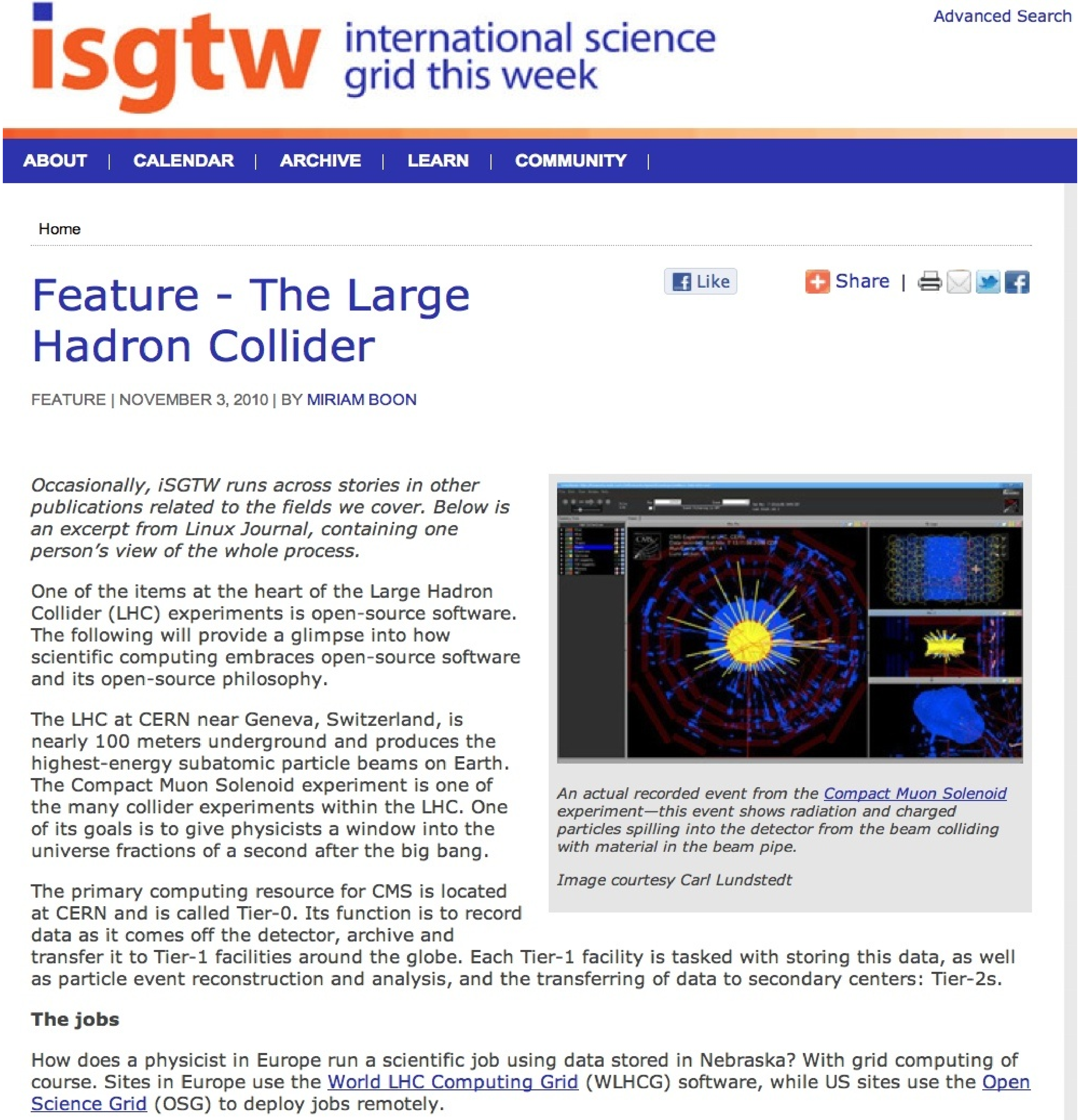
If you are thinking about making a mod (for any game) and are not sure what you need to know, how to go about it, or simply want to avoid the most obvious mistakes then read on. The pages linked to below contain some excellent advice, and possibly comments on stuff that hadn't occured to you.

- /My Team Your Team – Introduction and disclaimer for all those, "what's all this my team your team crap?" readers.
- /Why Are You Making A Mod – Sometimes the reason a mod fails is the reason you started it in the first place.
- /Building a Team – Building up your mod team.
- /Despotism Or Communism – Some thoughts on team structure.
- /Working as a Team – The day to day life of a team.
- /Asset Management – How to manage the assets of your mod (code, textures, models, etc).
- /Distributed Development – Find out how hard and unpleasant distributed development can be.
- /Effective Testing – How to get the most out of testing your mod.
- Releasing A Mod
- /Supporting Your Mod – Easing the burden of mod support.
- /Mod Death – What happens when a mod or mod team self destruct and how to cope.

Thoughts on Mod Making

Several of the Unreal Wiki's contributors have experience in creating successful mods. Reading their accounts of their work and their advice is recommended.

- Mychaeel/Mod Startups – Making your idea a reality.
- Mychaeel/Modding Etiquette – How to make people like your mod.
- Jb – an analysis of the ChaosUT mod's history
- Piglet/Finishing Things – How to actually finish your mods, that said it's more how to **start** so that you **can** finish.
- A Bug's Life
- GODZ Inception – a journal of how GODZ started.
- Making Mods/General Mod Optimization – Common mistakes and ignored settings which often lead to lower performance – and how to fix/use them.



isgtw international science grid this week

Advanced Search

ABOUT | CALENDAR | ARCHIVE | LEARN | COMMUNITY |

Home

Feature - The Large Hadron Collider

Like

Share |

FEATURE | NOVEMBER 3, 2010 | BY MIRIAM BOON

Occasionally, iSGTW runs across stories in other publications related to the fields we cover. Below is an excerpt from Linux Journal, containing one person's view of the whole process.

One of the items at the heart of the Large Hadron Collider (LHC) experiments is open-source software. The following will provide a glimpse into how scientific computing embraces open-source software and its open-source philosophy.

The LHC at CERN near Geneva, Switzerland, is nearly 100 meters underground and produces the highest-energy subatomic particle beams on Earth. The Compact Muon Solenoid experiment is one of the many collider experiments within the LHC. One of its goals is to give physicists a window into the universe fractions of a second after the big bang.

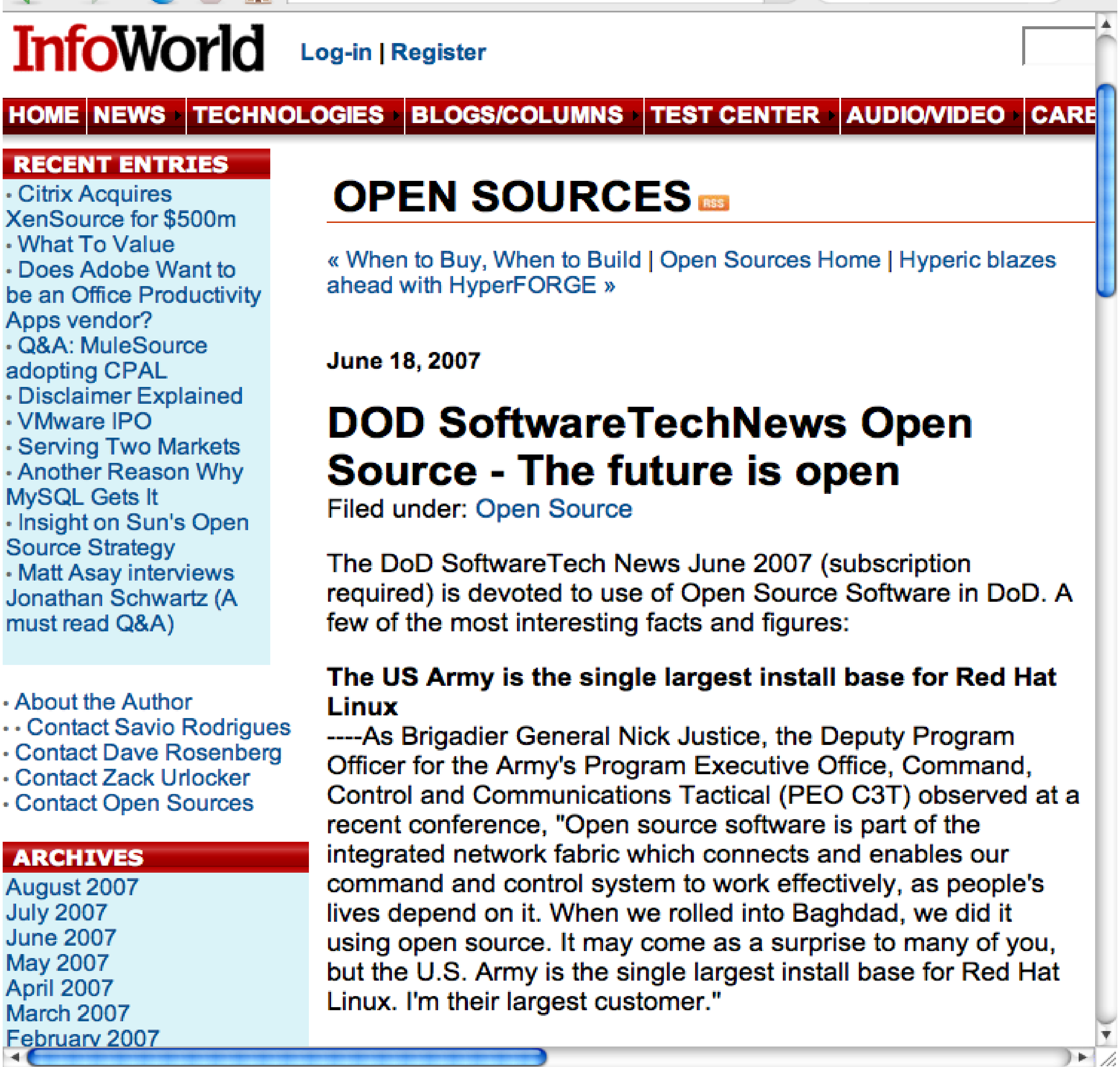
An actual recorded event from the Compact Muon Solenoid experiment—this event shows radiation and charged particles spilling into the detector from the beam colliding with material in the beam pipe.

Image courtesy Carl Lundstedt

The primary computing resource for CMS is located at CERN and is called Tier-0. Its function is to record data as it comes off the detector, archive and transfer it to Tier-1 facilities around the globe. Each Tier-1 facility is tasked with storing this data, as well as particle event reconstruction and analysis, and the transferring of data to secondary centers: Tier-2s.

The jobs

How does a physicist in Europe run a scientific job using data stored in Nebraska? With grid computing of course. Sites in Europe use the World LHC Computing Grid (WLHCG) software, while US sites use the Open Science Grid (OSG) to deploy jobs remotely.



RECENT ENTRIES

- Citrix Acquires XenSource for $500m
- What To Value
- Does Adobe Want to be an Office Productivity Apps vendor?
- Q&A: MuleSource adopting CPAL
- Disclaimer Explained
- VMware IPO
- Serving Two Markets
- Another Reason Why MySQL Gets It
- Insight on Sun's Open Source Strategy
- Matt Asay interviews Jonathan Schwartz (A must read Q&A)

- About the Author
- • Contact Savio Rodrigues
- Contact Dave Rosenberg
- Contact Zack Urlocker
- Contact Open Sources

ARCHIVES

August 2007
July 2007
June 2007
May 2007
April 2007
March 2007
February 2007

OPEN SOURCES

« When to Buy, When to Build | Open Sources Home | Hyperic blazes ahead with HyperFORGE »

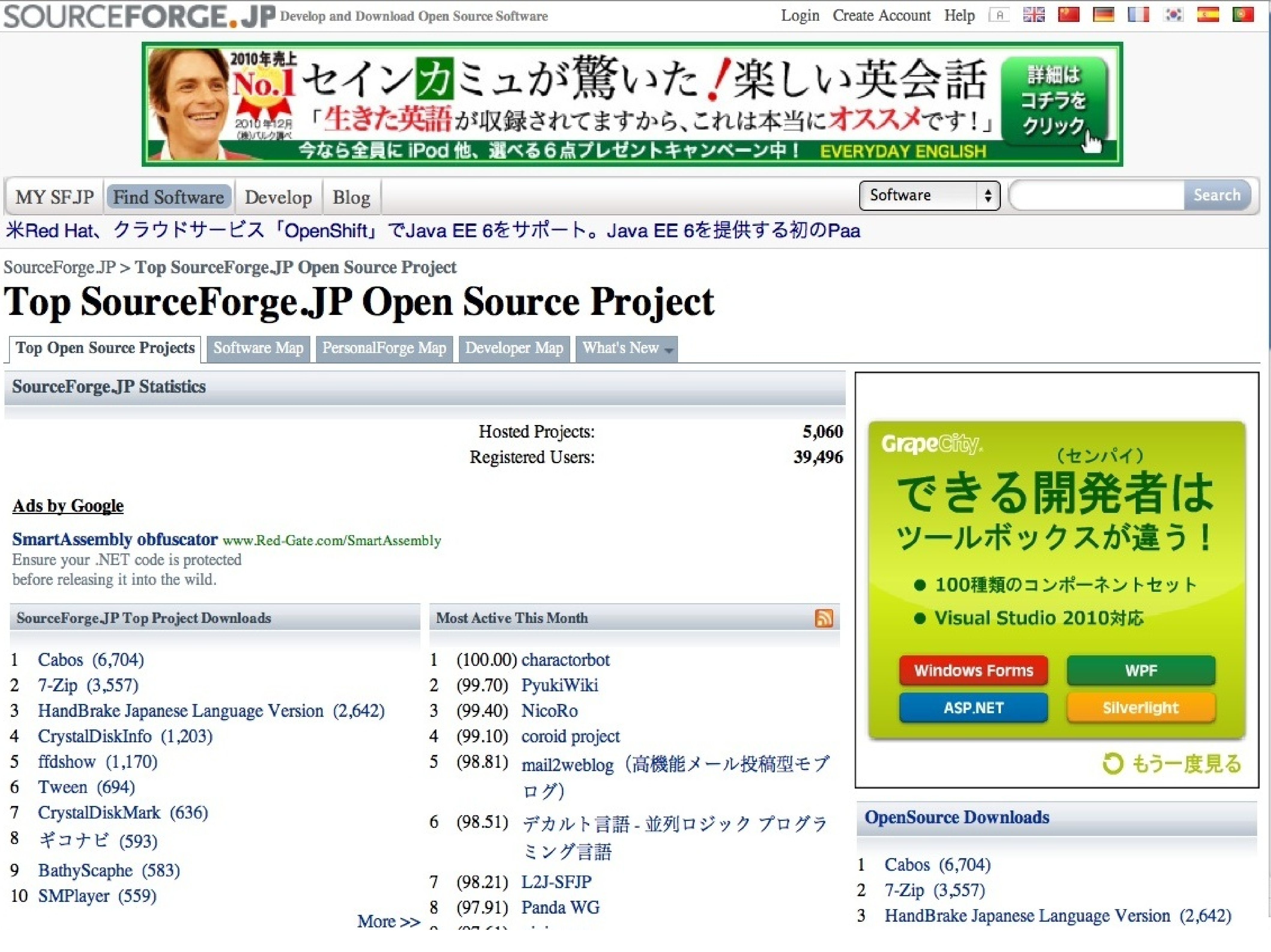
June 18, 2007

DOD SoftwareTechNews Open Source - The future is open

Filed under: Open Source

The DoD SoftwareTech News June 2007 (subscription required) is devoted to use of Open Source Software in DoD. A few of the most interesting facts and figures:

The US Army is the single largest install base for Red Hat Linux
----As Brigadier General Nick Justice, the Deputy Program Officer for the Army's Program Executive Office, Command, Control and Communications Tactical (PEO C3T) observed at a recent conference, "Open source software is part of the integrated network fabric which connects and enables our command and control system to work effectively, as people's lives depend on it. When we rolled into Baghdad, we did it using open source. It may come as a surprise to many of you, but the U.S. Army is the single largest install base for Red Hat Linux. I'm their largest customer."



米Red Hat、クラウドサービス「OpenShift」でJava EE 6をサポート。Java EE 6を提供する初のPaa

SourceForge.JP > Top SourceForge.JP Open Source Project

Top SourceForge.JP Open Source Project

SourceForge.JP Statistics

Hosted Projects:	5,060
Registered Users:	39,496

SourceForge.JP Top Project Downloads

1	Cabos (6,704)
2	7-Zip (3,557)
3	HandBrake Japanese Language Version (2,642)
4	CrystalDiskInfo (1,203)
5	ffdshow (1,170)
6	Tween (694)
7	CrystalDiskMark (636)
8	ギコナビ (593)
9	BathyScaphe (583)
10	SMPlayer (559)

More >>

Most Active This Month

1	(100.00)	charactorbot
2	(99.70)	PyukiWiki
3	(99.40)	NicoRo
4	(99.10)	coroid project
5	(98.81)	mail2weblog（高機能メール投稿型モブログ）
6	(98.51)	デカルト言語 - 並列ロジック プログラミング言語
7	(98.21)	L2J-SFJP
8	(97.91)	Panda WG

OpenSource Downloads

1	Cabos (6,704)
2	7-Zip (3,557)
3	HandBrake Japanese Language Version (2,642)

FOSS as social movement

- Emerging as a global-scale socio-technical movement that increasingly permeates society at an institutional, governmental, and international level in ways no prior software development regime has previously achieved.
- Unlikely any company/nation can inhibit FOSS in the near-term

Discussion and limitations

Defining characteristics of FOSSD projects

- Public availability of project data and artifacts
  - Collecting FOSSD process data may be more cost effective compared to proprietary (G)SE projects
  - *Prediction*: growing share of empirical SE research will be performed using FOSS data

FOSSD research limitations

- Individual participation
  - Some form of reciprocity and intrinsic, self-serving motivation is necessary
- Cooperation, coordination, and control
  - Negotiation and conflict management are <u>part of the cost</u> FOSS developers incur in order to have their believes fulfilled
  - Time, effort, and attention are spent negotiating socio-technical dependencies

FOSSD research limitations

- Alliances and community development
  - FOSSD projects give rise to *new kinds of requirements* for community building, community software, and community information sharing systems
  - Alliances and community require attention to sustain their effectiveness, and to prevent them from becoming self-serving and bureaucratic

FOSSD research limitations

- Empirical studies of FOSSD are expanding the scope of what we can observe, discover, analyze, and learn about large software systems.
  - Mining software repositories
  - Multi-modal modeling and analysis of socio-technical processes and networks found in sustained FOSSD projects

Research opportunities

FOSS systems research areas

- Development processes, practices, and project forms
- Collaboration
- Ecosystems
- Evolution
- Instrumentation and infrastructure

Development processes, practices, and project forms

- What are the
  - development processes,
  - work practices,
  - alternative project organizational forms

  that give rise to successful FOSS systems?
- What works where, when, why and how, and for whom?

Collaboration

- How does the practice of developing large or very large scale software systems depend on the collaborative work practices and communities of practice found in successful FOSS system projects?
- How do software licenses facilitate or inhibit collaboration among global software developers?

Ecosystems

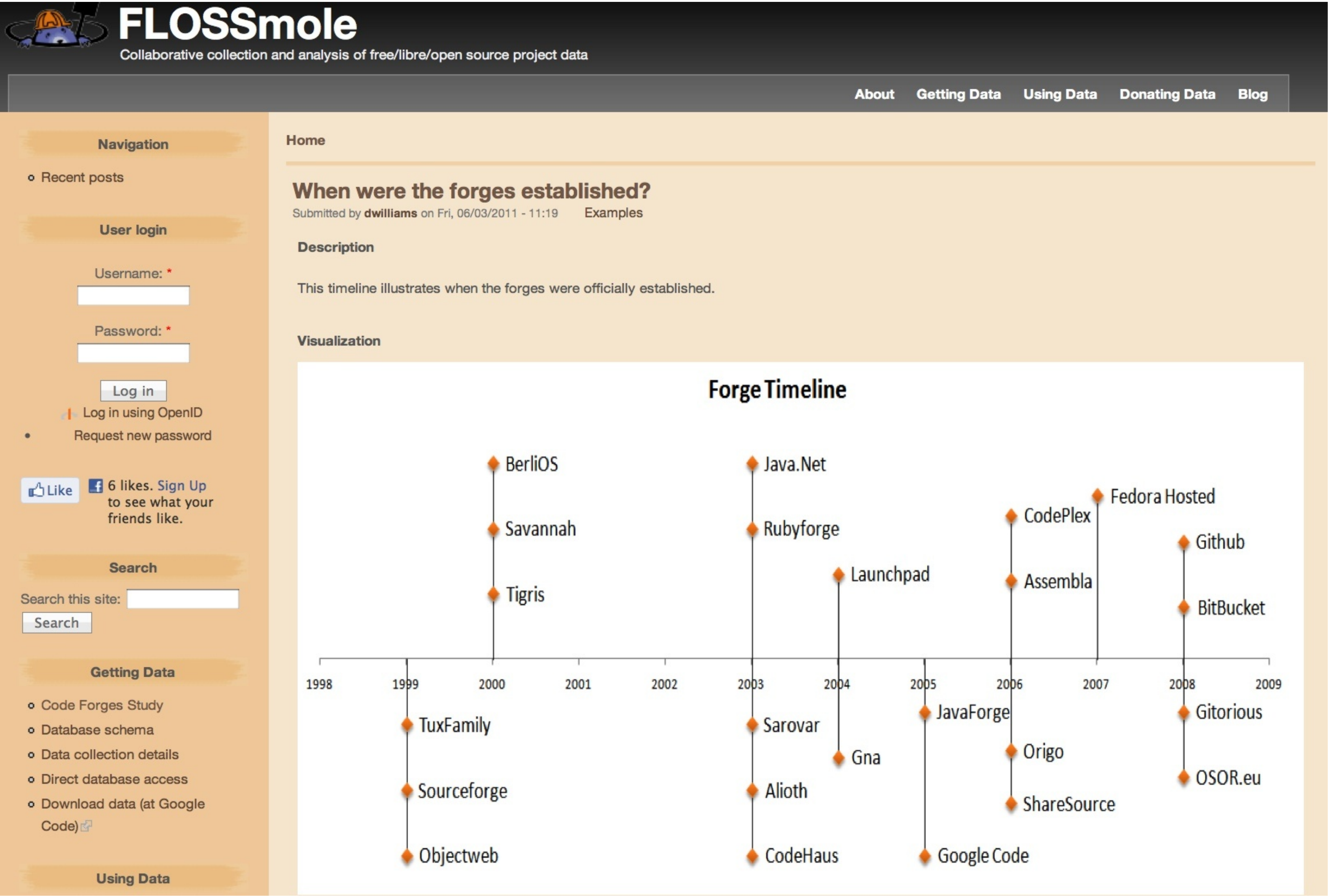
- How do FOSS systems emerge within a complex, decentralized web of people, artifacts, practices, and other infrastructural resources while most FOSS projects fail to take root and thrive?
- How do those few that do succeed become widespread and transform industry, government, or science practices?

Evolution

- How can successful FOSS systems continue to grow, develop (within releases), and evolve (across releases) across ever larger communities of developer-users at sustained exponential rates?
  - Via evolution/replacement of components, architecture, component licenses, project forms, communities, tools, practices, etc.
- To what end, and following what processes?

Instrumentation and infrastructure

- *Scalability*:
  - Research studies range from small-scale studies of individual FOSS projects to very-large populations of FOSS projects
- *Repositories and Meta-Repositories:*
  - Each FOSS project is an ecology of mostly informal online artifacts
  - Source code and meta-data are formal
- *Data analysis tools and analytics*
  - Supporting text and software data mining, (process) knowledge discovery, data visualization, provenance, and archiving of "executable research papers"
  - See FLOSSmole, FLOSShub, FLOSSmetrics.



FLOSSmole

Collaborative collection and analysis of free/libre/open source project data

About Getting Data Using Data Donating Data Blog

Navigation

- Recent posts

User login

Username: *

Password: *

Log in

Log in using OpenID

- Request new password

Like 6 likes. Sign Up to see what your friends like.

Search

Search this site:

Search

Getting Data

- Code Forges Study
- Database schema
- Data collection details
- Direct database access
- Download data (at Google Code)

Using Data

Home

When were the forges established?

Submitted by **dwilliams** on Fri, 06/03/2011 - 11:19 Examples

Description

This timeline illustrates when the forges were officially established.

Visualization

Forge Timeline

Research opportunities

- FOSSD is poised to alter the calculus of empirical SE (and Global SE)!
    - Software process discovery, modeling, and simulation
    - Repository mining can support software visualization, refactoring/redesign studies
    - Comparison of SE versus FOSSD approaches to software inspection and peer review

Research opportunities

- Based on results from individual motivation, participation, role migration, and turnover in FOSSD projects, (G)SE world would benefit from empirical studies that examine similar patterns in conventional software development projects
  - Is FOSSD more fun, interesting, and rewarding than (G)SE?

Research opportunities

- Conventional software cost estimation techniques (e.g., “total cost of operation”) slight/ignore social capital and socio-technical resources
  - Miscalculation of total resources and capabilities that affect predicted/actual costs of software development or FOSSD

Research opportunities

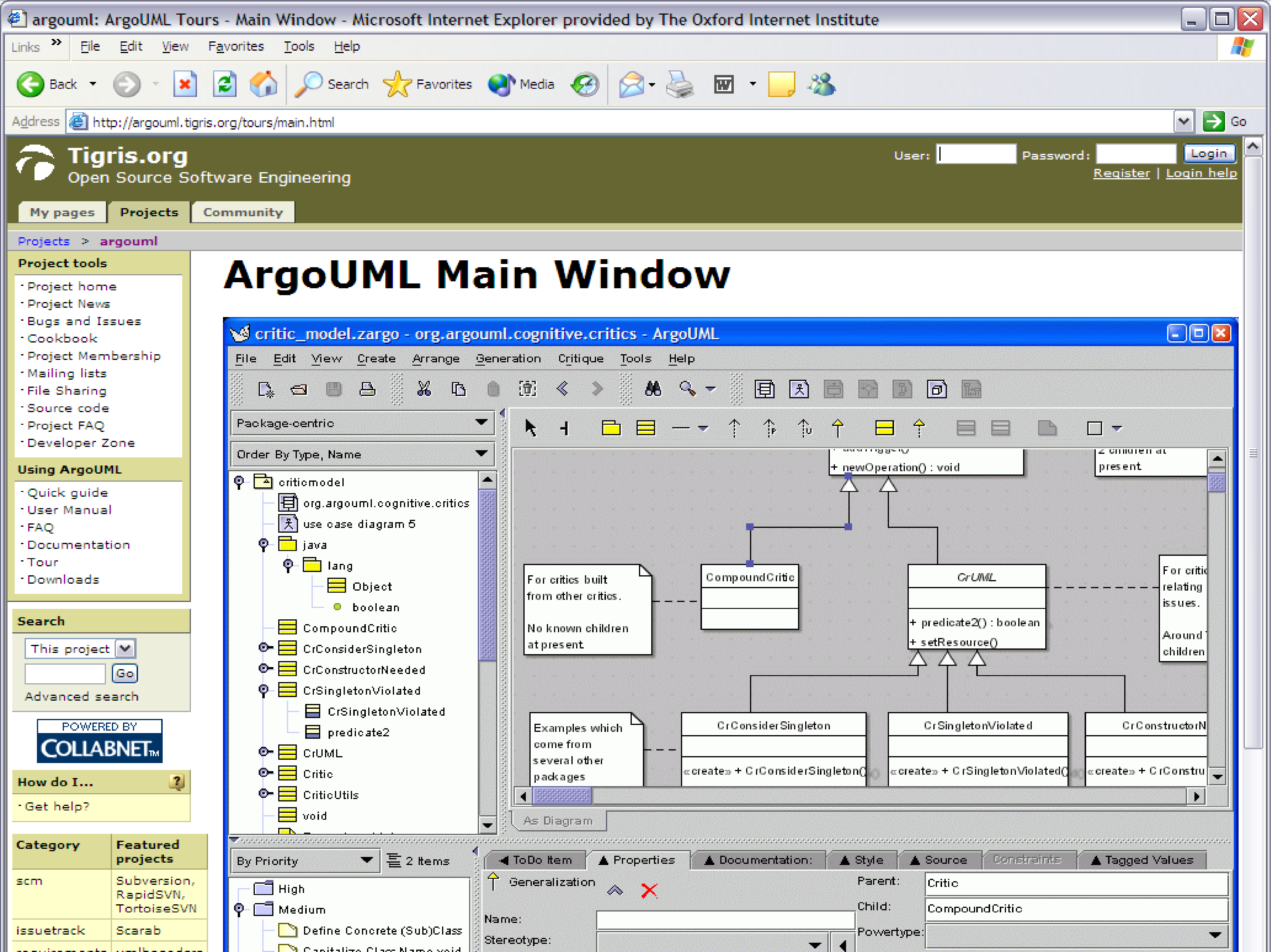
- Results from study of cooperation, coordination and control in FOSSD
    - Virtual project management and role migration can provide a lightweight approach to (G)SE project management
    - Unclear whether proprietary software projects willing to embrace VPM

Research opportunities

- Alliance formation and social networking results suggest SE projects operate at a disadvantage compared to FOSSD projects
  - SE projects tend to produce systems whose growth/evolution is limited
  - FOSSD projects can produce systems capable of sustained exponential growth/evolution of both software and developer-user community

Research opportunities

- How best to encourage the emergence of a social movement that combines best practices of FOSSD and SE
  - Consider participation or study of open source software engineering (OSSE) projects at Tigris.org, or global FOSSD
  - OSSE seeks to combine SE and FOSSD tools, techniques, and concepts


argouml: ArgoUML Tours - Main Window - Microsoft Internet Explorer provided by The Oxford Internet Institute
Links
File
Edit
View
Favorites
Tools
Help
Back
Search
Favorites
Media
Address
http://argouml.tigris.org/tours/main.html
Go

Tigris.org
Open Source Software Engineering

User: Password: Login
Register | Login help

My pages | Projects | Community

Projects > argouml

Project tools

- Project home
- Project News
- Bugs and Issues
- Cookbook
- Project Membership
- Mailing lists
- File Sharing
- Source code
- Project FAQ
- Developer Zone

Using ArgoUML

- Quick guide
- User Manual
- FAQ
- Documentation
- Tour
- Downloads

Search

This project Go
Advanced search

POWERED BY COLLABNET

How do I...

- Get help?

Category	Featured projects
scm	Subversion, RapidSVN, TortoiseSVN
issuetrack	Scarab

ArgoUML Main Window

Recent FOSS Research Surveys

- Scacchi, W. (2007). Free/Open Source Software Development: Recent Research Results and Emerging Opportunities. *Proc. 6th. ESEC/FSE*, 459–468. Also see, Free/Open Source Software Development: Recent Research Results and Methods, in M.V. Zelkowitz (ed.), *Advances in Computers*, 69, 243-295, 2007.
- Gasser, L. and Scacchi, W. (2008). Towards a Global Research Infrastructure for Multidisciplinary Study of Free/Open Source Software Development, in *Open Source Development, Community and Quality*; B. Russo, E. Damiani, S. Hissan, B. Lundell, and G. Succi (Eds.), IFIP Vol. 275, Springer, Boston, MA. 143-158.
- Hauge, O., Ayala, C. and Conradi, R. (2010). Adoption of Open Source Software in Software-Intensive Organizations - A Systematic Literature Review. *Information and Software Technology,* 52(11), 1133-1154.
- Aksulu, A. and Wade, M.R. (2010). A Comprehensive Review and Synthesis of Open Source Research, *J. Assoc. Info. Systems*, 11(11), 576-656.
- Scacchi, W., Crowston, K., Jensen, C., Madey, G., Squire, M., and others (2010). *Towards a Science of Open Source Systems*, Final Report, Computing Community Consortium, November 2010. http://foss2010.isr.uci.edu/content/foss-2010-reports/
- Höst, M., Oručević-Alagić, A. (2011). A Systematic Review of Open Source Software in Commercial Software Product Development, *Information and Software Technology*, 53(6), June, 616-624.
- Crowston, K., Wei, K., Howison, J., and Wiggins, A. (2011). Free/libre open source software development: what we know and what we do not know. *ACM Computing Surveys*, (in press).

Acknowledgements

- *Recent collaborators*:
  - Kevin Crowston, Syracuse; Les Gasser, UIllinois, Urbana-Champaign; James Howison, UTexas, Austin; Chris Jensen, Google; Greg Madey, Notre Dame University; John Noll, LERO; Megan Squire, Elon University
  - Thomas Alspaugh and others at ISR
- *Research funding support*:
  - National Science Foundation: #0083075, #0205679, #0205724, #0350754, #0534771, #0749353, #0808783.
  - Acquisition Research Program, Naval Postgraduate School (2007-2011)
  - Computing Community Consortium
  - No endorsement implied.

Thank you!