

Issues, Challenges, and Opportunities for Open Source Software Development

Walt Scacchi

Institute for Software Research

University of California, Irvine

7 January 2010

Background

- What is Free/Open Source Software Development?
- FOSSD project characteristics and practices
- Open Architectures and secure computing

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 10-50+ OSSD tools and communications applications to support their development work

Collaborative OSS tools

Product Development	Technical Communications	Project Management	Project Management
Web-based source code access	Web-based file and content management	Incremental or partial project planning	Project/task status tracking
Bug and issue-tracking	Mailing list management	Process/workflow support	Update tracking
Configuration and version mgmt.	Discussion forums	Role-based access control	Audit logs and history
Search/index across source code and documents	Project document (Web page) templates	Enterprise or project branding	

Browse by Category

- [Topic \(383297\)](#)
 - [Communications \(31635\)](#)
 - [Database \(13279\)](#)
 - [Desktop Environment \(6955\)](#)
 - [Education \(12853\)](#)
 - [Formats and Protocols \(9422\)](#)
 - [Games/Entertainment \(38523\)](#)
 - [Internet \(49513\)](#)
 - [Mobile \(1617\)](#)
 - [Multimedia \(31884\)](#)
 - [Office/Business \(24519\)](#)
 - [Other/Nonlisted Topic \(8858\)](#)
 - [Printing \(1058\)](#)
 - [Religion and Philosophy \(796\)](#)
 - [Scientific/Engineering \(35027\)](#)
 - [Security \(6858\)](#)
 - [Social sciences \(906\)](#)
 - [Software Development \(61696\)](#)
 - [System \(40610\)](#)
 - [Terminals \(1237\)](#)
 - [Text Editors \(6069\)](#)

Ads by Google

Acquia Drupal CMS

Acquia Makes Drupal SimpleDownload Software Directly Online

[Acquia.com](#)

Find Topic Software

Results 1 – 10 of 159944

Display: [Detail](#) [Images](#) [Filters](#) View: [10](#)

Project Name	Rank	Activity	Registered	Latest File	Downloads
eMule	614	99.78%	2002-05-13	2009-02-22	515,918,421
eMule is a filesharing client which is based on the eDonkey2000 network but offers more features than the standard client					
Topic: File Sharing					
					
Download Now!					
Azureus / Vuze	58	99.98%	2003-06-24	2009-07-09	475,051,215
Vuze (formerly Azureus) is a P2P file sharing client using the bittorrent protocol. Search and download torrent files. Play, convert and transcode videos and music for playing on many devices such as PSP, TiVo, Xbox, PS3, iTunes (iPhone, iPod, Apple TV).					
Topic: Internet , File Sharing					
					
Download Now!					
Ares Galaxy	46	99.98%	2004-06-18	2009-02-03	212,788,001
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.					
Topic: File Sharing , Chat					
					
Download Now!					

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 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
 - Even for mission-critical systems

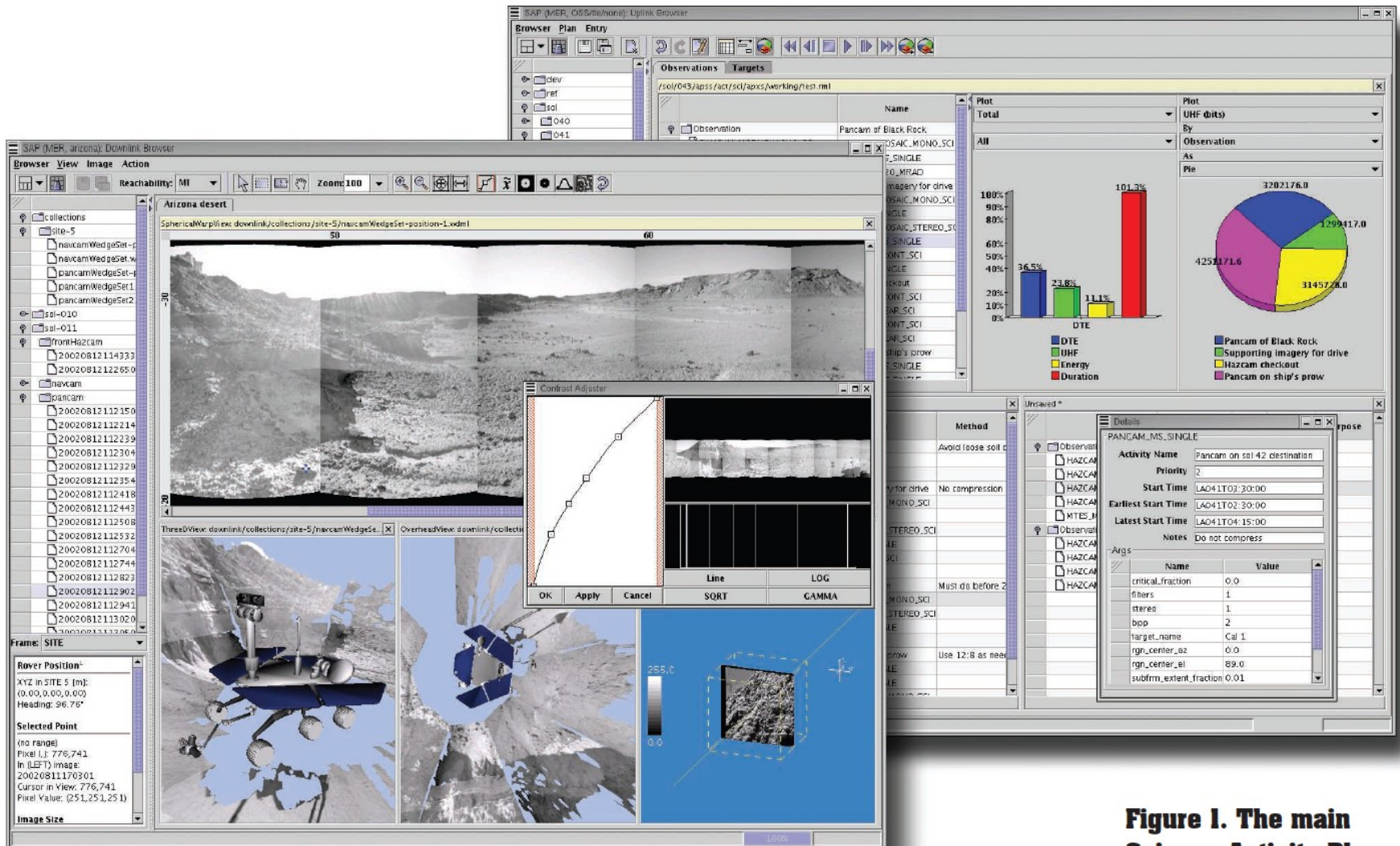


Figure 1. The main Science Activity Planner interface. With the downlink browser

Source: J.S. Norris, Mission Critical Development of Open Source Software: Lessons Learned, *IEEE Software*, 21(1), 42-49, Jan 2004.

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 RSS

« 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."

Empirical findings from OSSD practices

- **Individual participation**
- **Resources supporting FOSSD activities**
- Governance: cooperation, coordination, and control in FOSSD projects
- Alliances, social networking and community development
- **Multi-project software ecosystems**
- FOSS as a social movement

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

FOSSD resources/capabilities

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

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
- Informalisms are the media for OSS requirements, design, testing, etc.

GNUe Traffic #122 For 17 Jul

KT <http://www.kerneltraffic.org/GNUe/latest.html>

- 3. 26 Jun [Object IDs used in GNUe where there is no primary key](#)
- 4. 28 Jun - 29 Jun [Problem with dropdowns validation fixed](#)
- 5. 29 Jun - 6 Jul [Displaying grids in GNUe Forms with wx 2.6](#)
- 6. 6 Jul [Current status of GNUe Reports](#)

Introduction

This newsletter mainly covers the the #gnuenterprise IRC channel, with occasional coverage of the three main mailing lists (gnue-announce, gnue and gnue-dev) for the [GNU Enterprise](#) project.

1. Further trouble-shooting with the wx 2.6 drivers

20 Jun - 21 Jun Archive Link: "[\[IRC\] 20 Jun 2006](#)"
 Summary By [Peter Sullivan](#)
 Topics: [Forms](#), [Common](#)
 People: [Reinhard Müller](#), [James Thompson](#), [Johannes Vetter](#), [Peter Sullivan](#)

Further to [Issue #117, Section #2](#) (22 May : Layout in GNUe Forms with wx 2.6 driver) , Reinhard Müller (reinhard) suggested to James Thompson (jamest) **"if you are bored, you can try again the wx26 uidriver"** , as Johannes Vetter (johannesV) had done **"some massive changes and it might be that your issues with fscking up the boxes are solved"** . James said that, although he was busy, **"i really need to get that tested, as the dropdown box issues in 2.4 are preventing some selections from being allowed"** . So he was keen to have a version of GNUe Forms that worked with the user interface driver for wx 2.6 as soon as possible.

Trying Johannes' new code for GNUe Forms with his existing GNUe Forms Definitions, James found problems - **"none of which are due to anything wrong with what you've done - it's all in my forms"** , where he had been relying on 'features' (such as overlapping text boxes) that Johannes had treated as 'bugs' and now fixed. Johannes confirmed that **"overlapping is now being checked ... not only for boxes but for all widgets"** . He added, **"if you click the detail-button you'll see the offending line in your XML-File - this makes debugging"** a GNUe Form Definition (gfd) **"a lot easier"** . James reported that all five of his existing GNUe Form Definitions were not working with the new code - but **"i would still imagine it's something funky I'm doing in the form"** rather than a problem with Johannes' code. He noted that, on the last one, the problem that he had been having with the dropdown menu had been fixed, but the form now **"aborts on query"** .

(ed. [Peter Sullivan] Note that the lack of any guarantees on backward compatability, even with 'features'/'bugs' is one of the reasons why GNUe Forms remains at a version number below 1.0 as of time of writing, as discussed further in [Issue #112, Section #4](#) (13 Apr : Forms approaching version 1.0?) .)

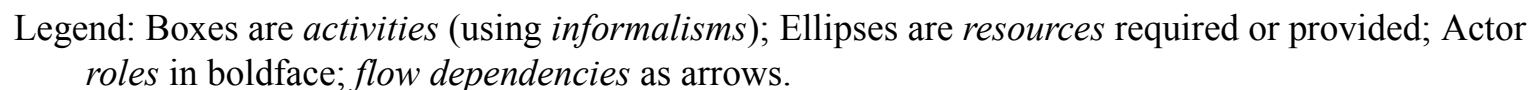
Done

FOSSD informalisms

Email lists	Discussion forums	News postings	Project digests
IM/Internet Relay Chat	Scenarios of usage	How-to guides	To-do lists
FAQ's and 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 Web sites	Project source code web
Project repositories	Software bug reports	Issue tracking databases	etc.

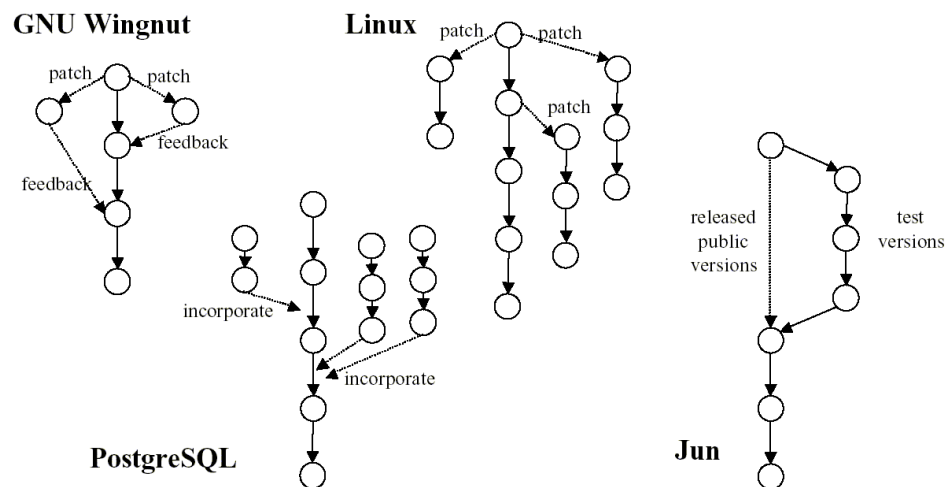
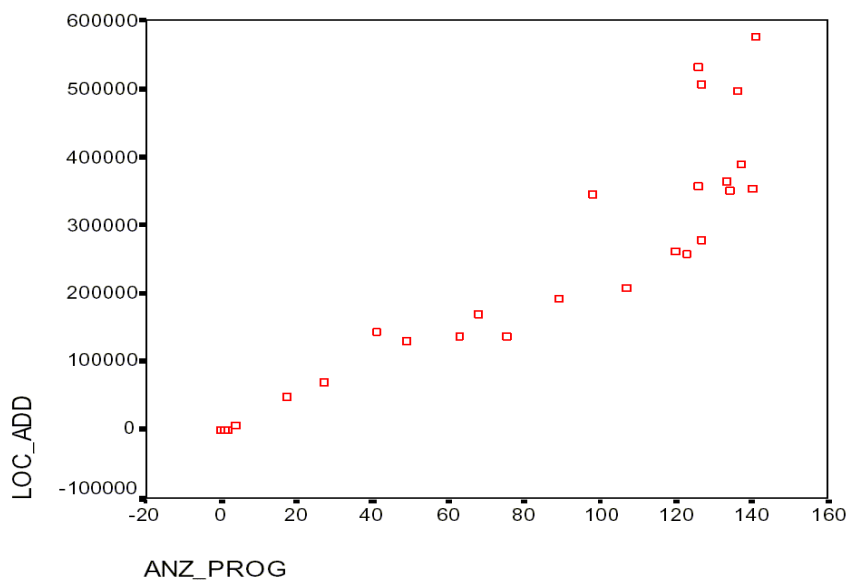
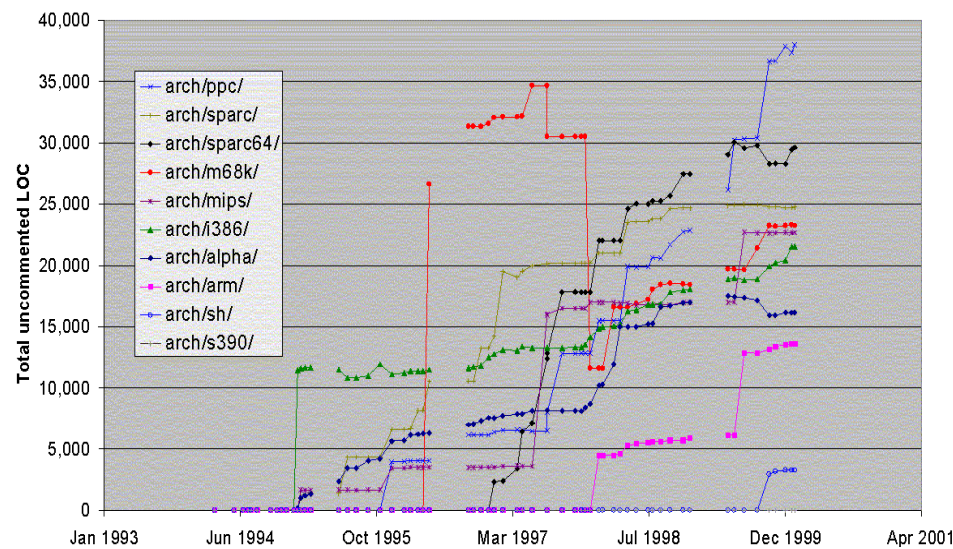
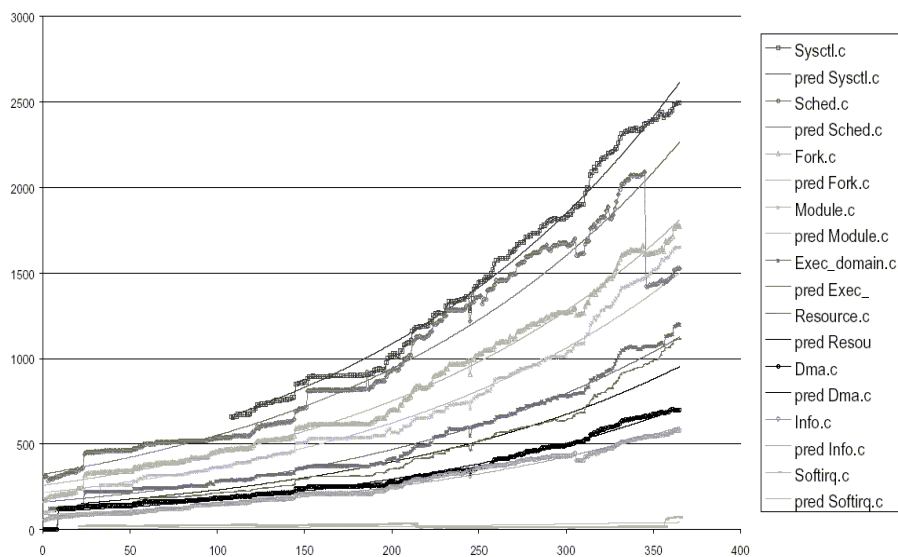
Implicit project management

- FOSSD projects self-organize into a *meritocratic role-hierarchy* that enables *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 via traceable and searchable information/content legacy



Multi-project software ecosystem

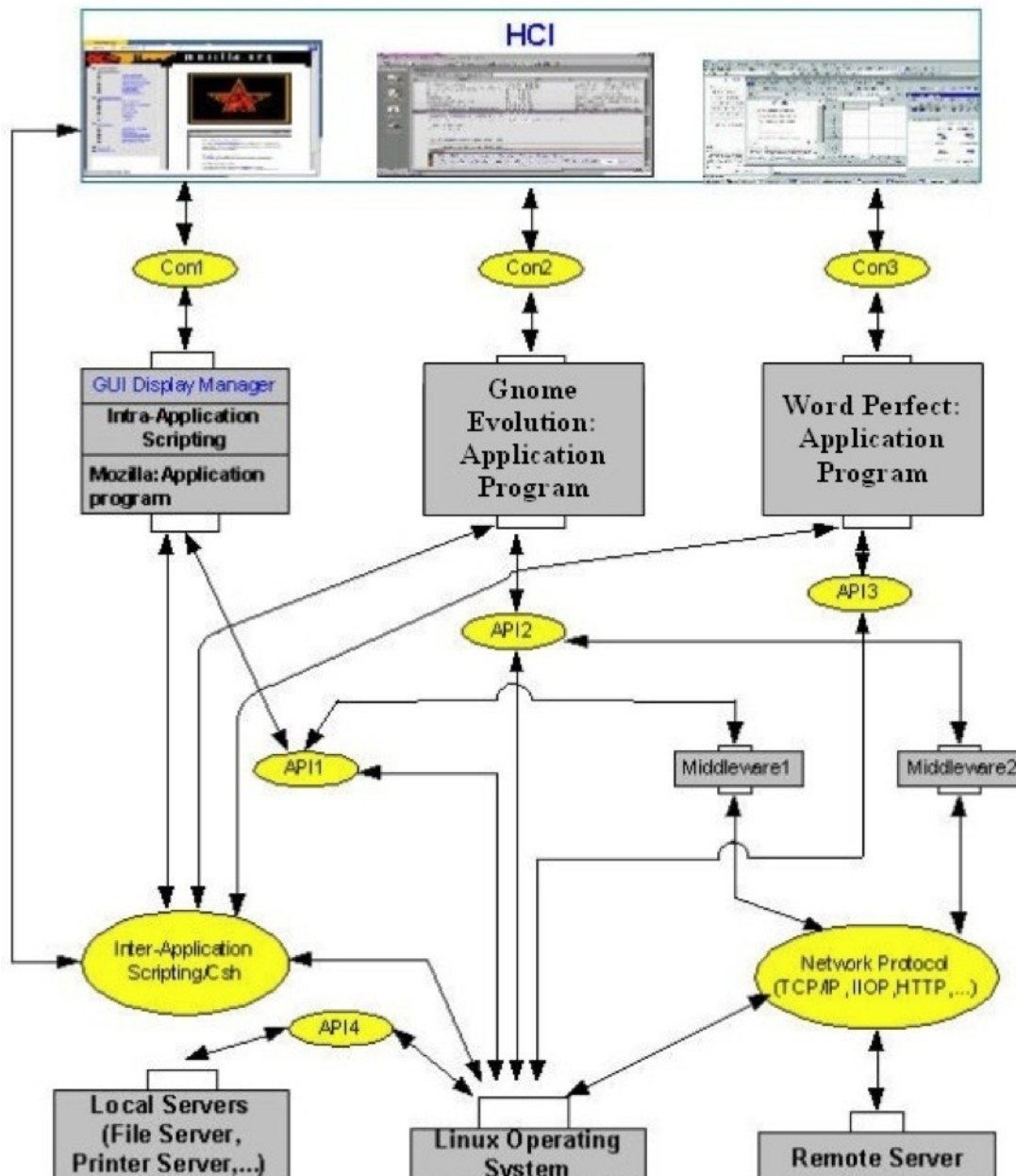
- Mutually dependent FOSS development and evolution propagate functional software cliches/idioms, cloned code, architectural styles, dependencies, and vulnerabilities
- *Architectural bricolage* arises when autonomous FOSSD projects, artifacts, tools, and systems co-mingle or merge
 - Enables discontinuous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



Open Architectures and OSS (and some secure computing topics)

Open Software Architecture Concepts

- Software source code components
 - Standalone programs
 - Libraries, frameworks, or middleware
 - Inter-application script code (e.g., for mash-ups)
 - Intra-application script code (e.g., for Rich Internet Apps.)
- Executable software components (binaries)
- Application program interfaces (APIs)
- Software connectors
- Configured sub-system or system



Legend: Grey boxes are *components*; ellipses are *connectors*; white boxes are *interfaces*; arrows are data or control *flow paths*; complete figure is architectural design *configuration*

OSS elements subject to different IP/Security licenses

- Intellectual Property/Security licenses stipulate rights and obligations regarding use of the licensed software components/systems
 - GPL (Gnu Public License) stipulate right to access, study, modify, and *reciprocal* obligation to redistribute modified source
 - Mozilla now offers a “tri-license” for its software like Firefox:
 - GPL, MPL (lightweight), or Restricted (accommodating proprietary services)
 - Other OSS covered by different rights and obligations
- How to determine which rights and obligations will apply to a component-based configured system?
 - At *design-time* (maximum flexibility)
 - At *build-time* (may/not be able to redistribute components at hand)
 - At *run-time* (may/not need to install/link-to components from other sources)

Source: T. Alspaugh, H. Asuncion, and W. Scacchi, Intellectual Property Rights Requirements for Heterogeneously Licensed Systems, in *Proc. 17th. Intern. Conf. Requirements Engineering (RE09)*, Atlanta, GA, 24-33, September 2009.

ArchStudio 4 - architectures/hci.xml - Eclipse SDK - C:\Documents and Settings\Hazel\ArchStudio4

File Edit Navigate Search Project Run Window Help

Outline

/architectures/hci.xml

Activity Diagrams

Statecharts

Structures

main

MoZilla SubArch

Types

Component Types

MoZillaComp

Connector Types

Interface Types

*hci.xml - Archipelago

70%

```
graph TD; Mozilla --- GnomeEvolution; Mozilla --- WordPerfect; Mozilla --- CSH[CSH/Application Scripting]; GnomeEvolution --- API2; WordPerfect --- API8; CSH --- LocalServer; CSH --- API4; CSH --- LinuxOS[Linux OS]; LocalServer --- API4; LinuxOS --- API2; LinuxOS --- API8; LinuxOS --- ORB1; LinuxOS --- ORB2; LinuxOS --- NetworkProtocol[Network Protocol TCP/IP, IOP, HTTP...]; ORB1 --- NetworkProtocol; ORB2 --- NetworkProtocol; NetworkProtocol --- RemoteServer;
```

Tracelink View

New Tracelink

Start Recording Recover

Tracelink Details

Import Links Export Links

Tracelink Options

Trace Analysis Trace License

ArchStudio 4 Launcher

File Tracker View

Archlight Issues

Archlight Notices

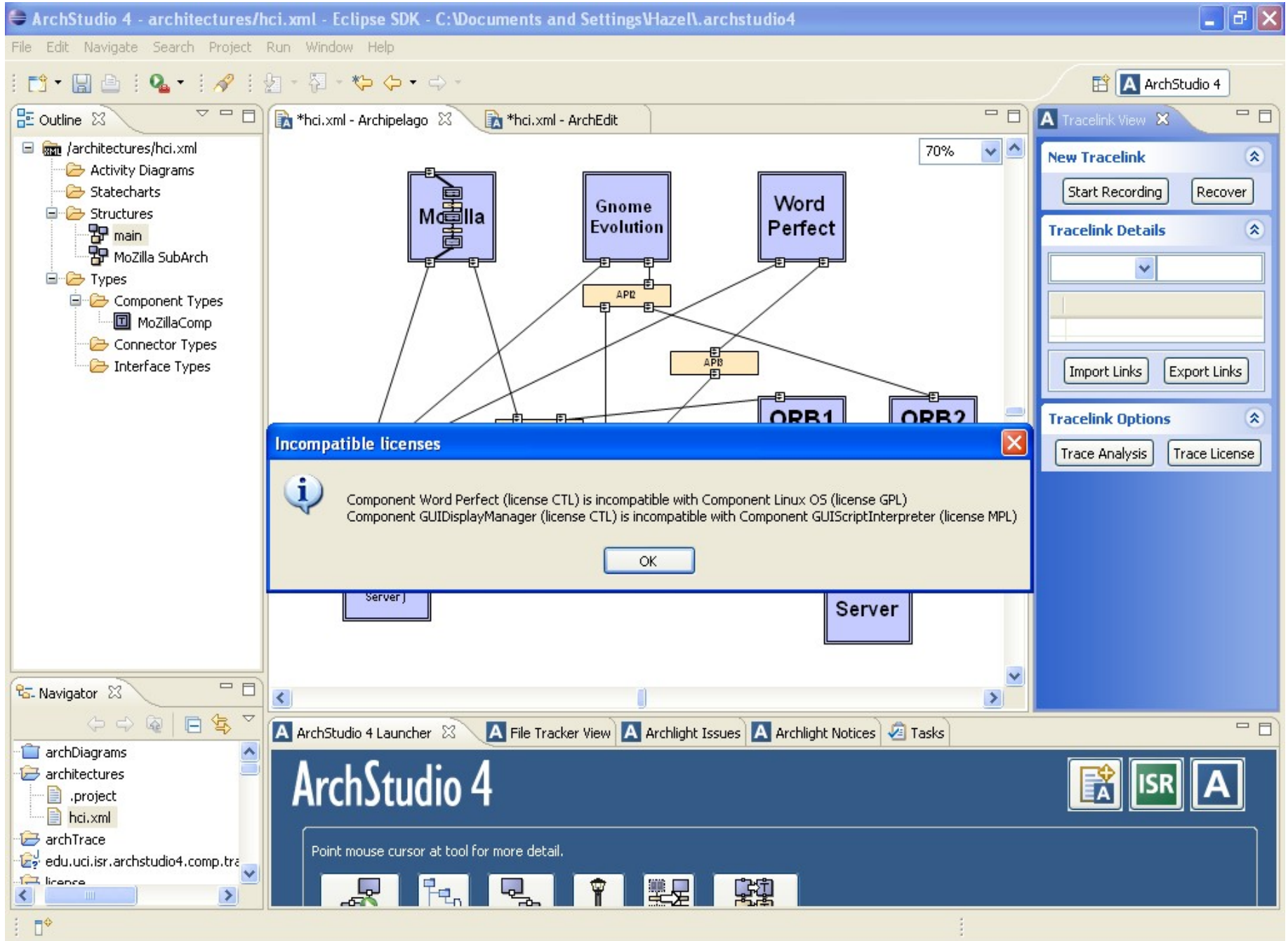
Tasks

ArchStudio 4

Point mouse cursor at tool for more detail.

ISR

A



Acknowledgements

- *Project collaborators:*
 - Mark Ackerman, UMichigan, Ann Arbor
 - Les Gasser, UIllinois, Urbana-Champaign
 - John Noll, LERO, Irish Software Engineering Research Center
 - Thomas Alspaugh, Hazel Asuncion, Margaret Elliot, Chris Jensen and others at the UCI ISR
- *Funding support:*
 - National Science Foundation: #0534771, #0749353, #0808783
 - Digital Industry Promotion (DIP) Agency, Global R&D Collaboration Center, Daegu, South Korea
 - Naval Postgraduate School: Acquisition Research Program
 - *No endorsement implied.*