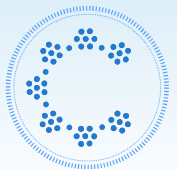


# *How To Open Source Your In-House App*

*Karl Fogel*

`kfogel@civiccommons.org`



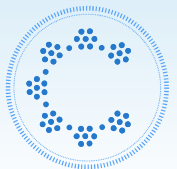
**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

# *How Open Source Projects Work*

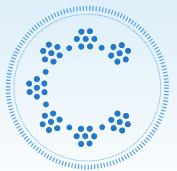
*Karl Fogel*

`kfogel@civiccommons.org`



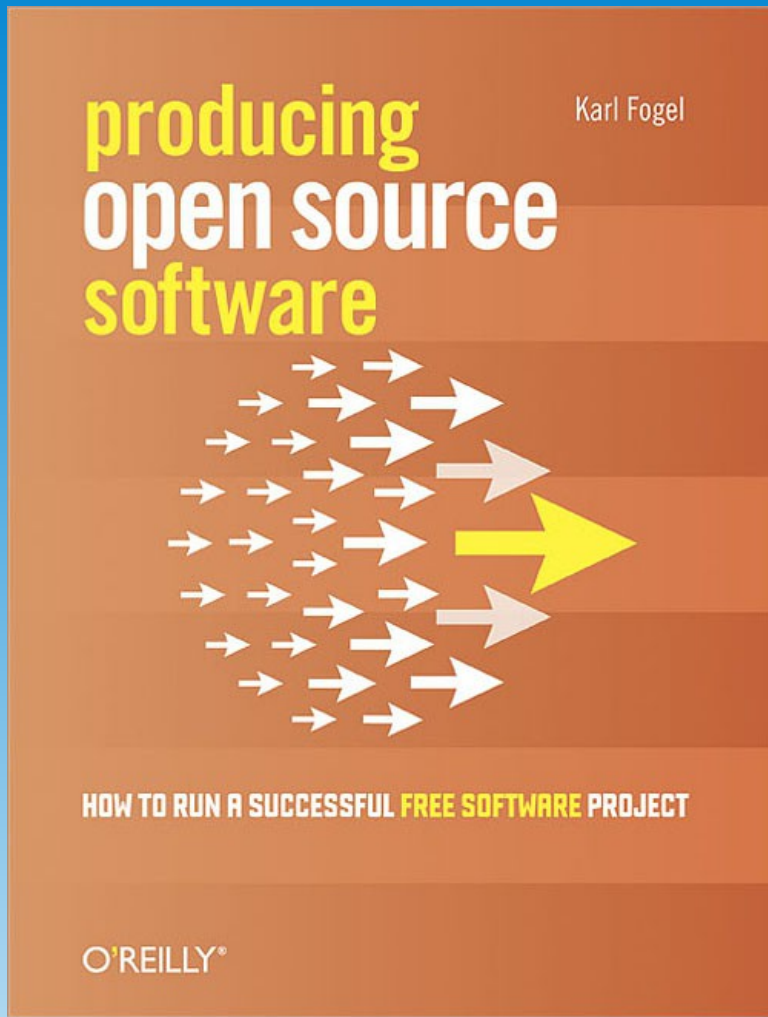
**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD



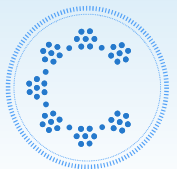
**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD



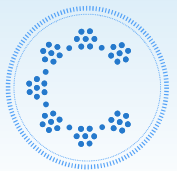
“Producing Open Source Software”

producingoss.com



**CIVIC COMMONS**

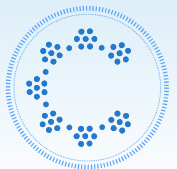
SHARING TECHNOLOGY FOR THE PUBLIC GOOD



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

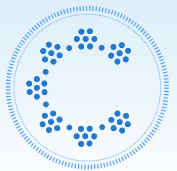
# What does “open source” make you think?



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

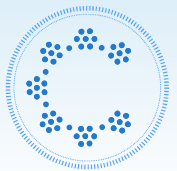
- Open source software is commercial software



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

- Open source software is commercial software
- It's just not proprietary

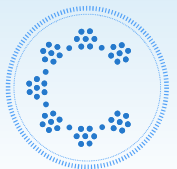


**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD



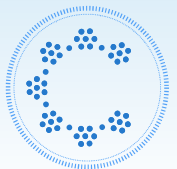
- Open source software is commercial software
- It's just not proprietary
- No such thing as academic-only OSS



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

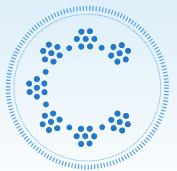
- Open source software is commercial software
- It's just not proprietary
- No such thing as academic-only OSS
- No such thing as gov't-only OSS



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

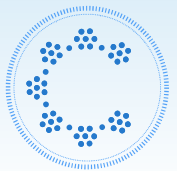
- Open source software is commercial software
- It's just not proprietary
- No such thing as academic-only OSS
- No such thing as gov't-only OSS
- “Open source” same as “free software”



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

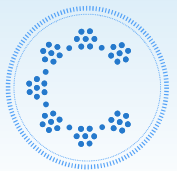
- Open source software is commercial software
- It's just not proprietary
- No such thing as academic-only OSS
- No such thing as gov't-only OSS
- “Open source” same as “free software”
- Copyleft vs non-copyleft (both qualify)



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

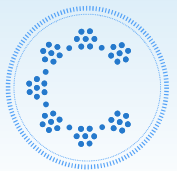
- Open source software is commercial software
- It's just not proprietary
- No such thing as academic-only OSS
- No such thing as gov't-only OSS
- “Open source” same as “free software”
- Copyleft vs non-copyleft (both qualify)
- Lots of religious wars in open source (e.g., AGPL)



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

- Open source software is commercial software
- It's just not proprietary
- No such thing as academic-only OSS
- No such thing as gov't-only OSS
- “Open source” same as “free software”
- Copyleft vs non-copyleft (both qualify)
- Lots of religious wars in open source (e.g., AGPL)
- But not about the definition.
- You open source your code, not your time.

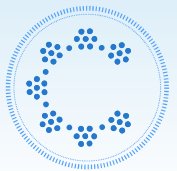


**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

# What we've got.

- NWS weather feed parser (Ron)

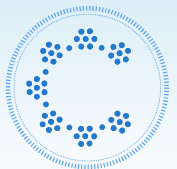


**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

# Group breakouts.

- **demo**
- **procure**
- **install**
- **use**
- **ask a question**
- **report a bug :-)**
- **contribute a patch**



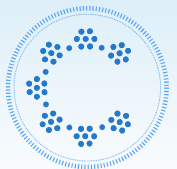
**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD



# Principles of Participation

- Participants are people, not organizations
- Make it *rewarding* to do things right
- Make it *easy* to do things right
- Make decisions in public
- Document the culture
- Don't work – amplify other people's work
- No obstacles: act now, audit later

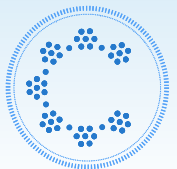


**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

# Principles of Participation

- Participants are people, not organizations
- Make it *rewarding* to do things right
- Make it *easy* to do things right
- Make decisions in public
- Document the culture
- Don't work – amplify other people's work
- No obstacles: act now, audit later

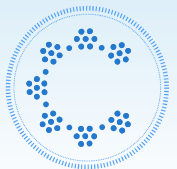


**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

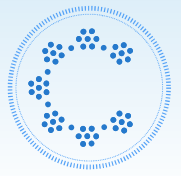
# Principles of Participation

- Participants are people, not organizations
- *Make it rewarding to do things right*
- *Make it easy to do things right*
- Make decisions in public
- Document the culture
- Don't work – amplify other people's work
- No obstacles: act now, audit later



↑  
R  
E  
W  
A  
R  
D

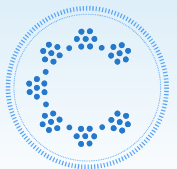
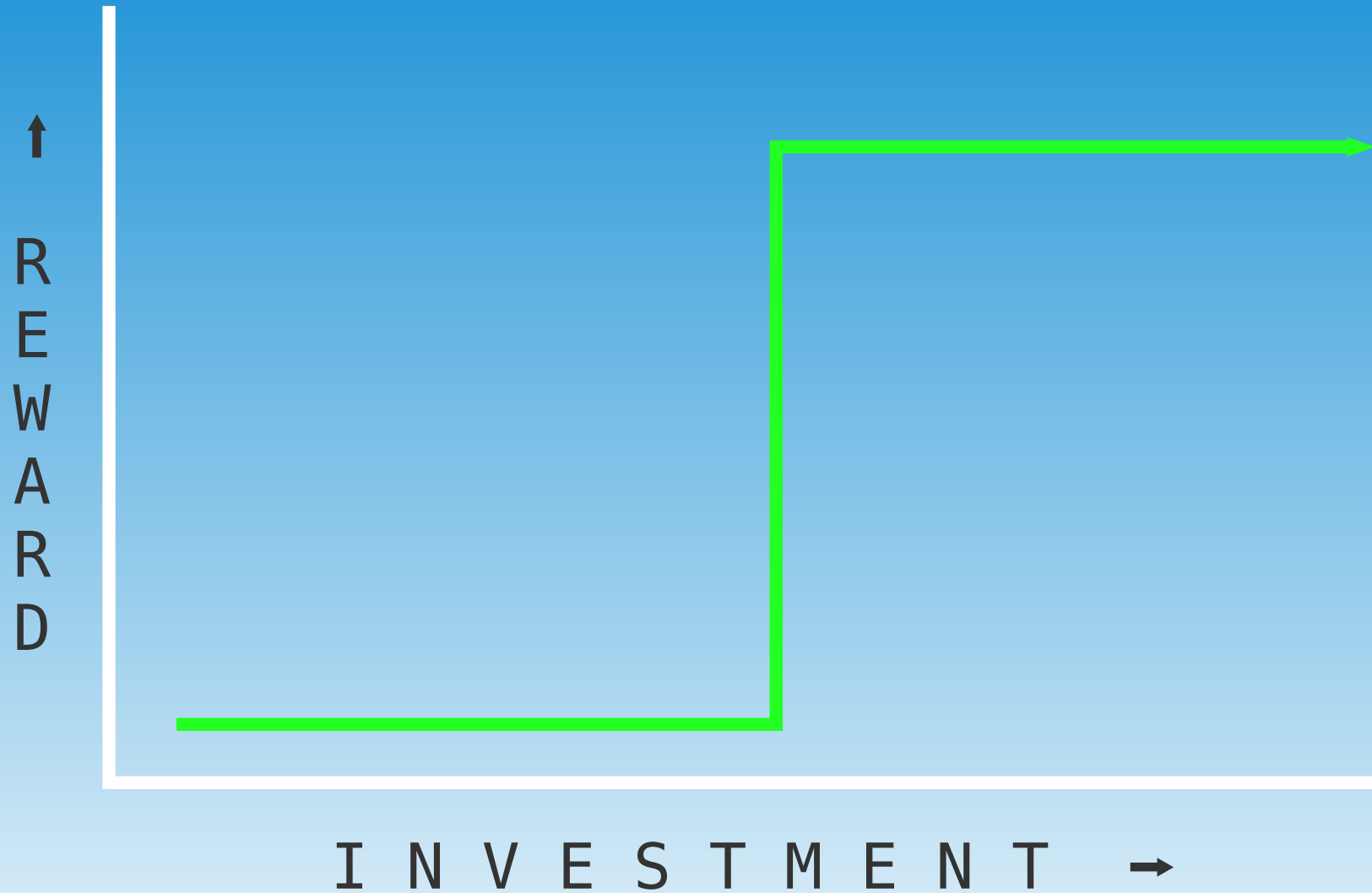
I N V E S T M E N T →



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

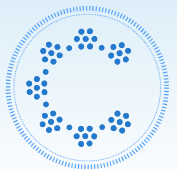
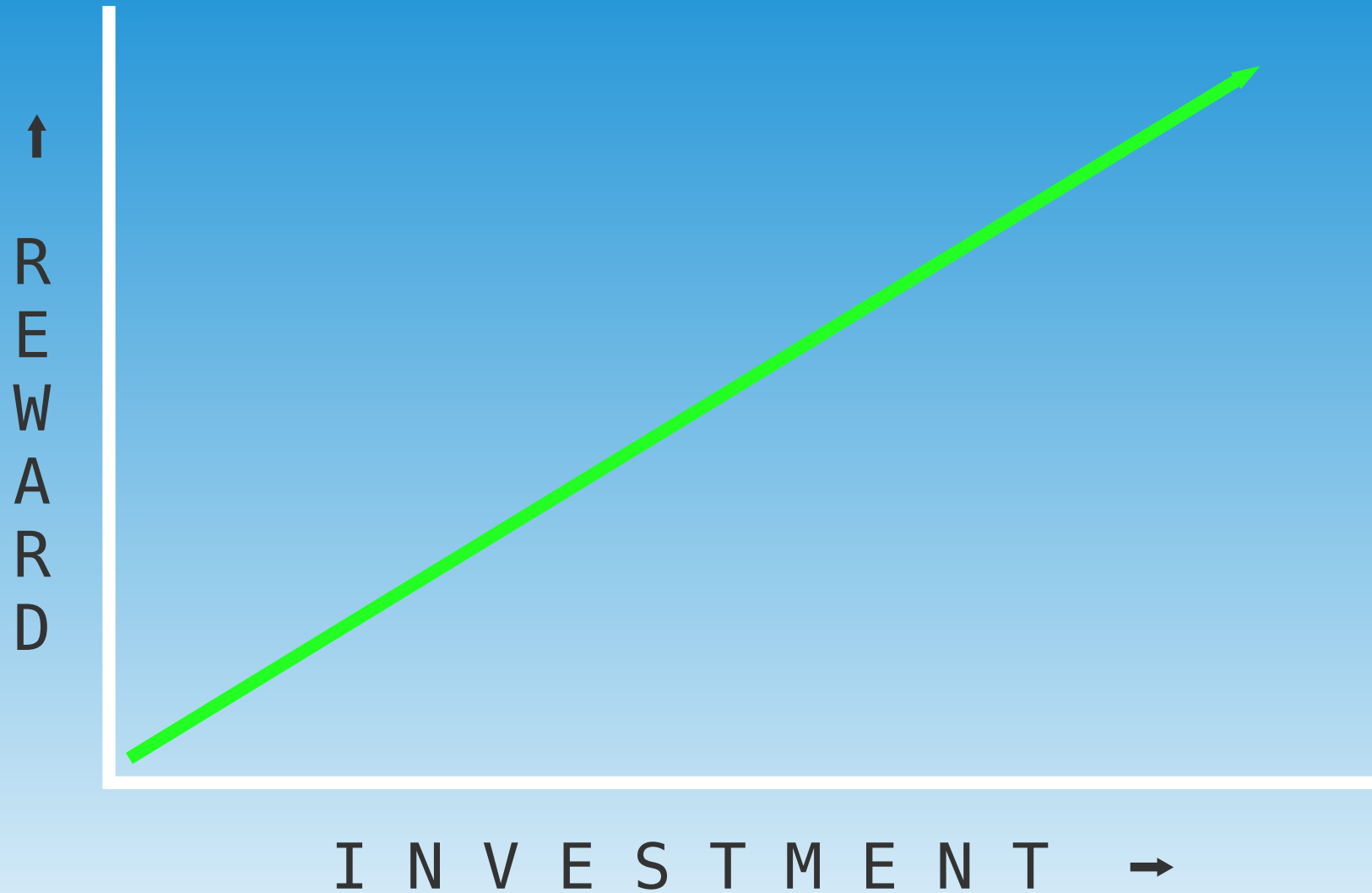
# *Step Function = High-Barrier Community*



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

# *Linear Function = Low-Barrier Community*

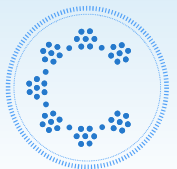


**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

# Principles of Participation

- Participants are people, not organizations
- Make it *rewarding* to do things right
- Make it *easy* to do things right
- **Make decisions in public**
- Document the culture
- Don't work – amplify other people's work
- No obstacles: act now, audit later



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

Subversion development | Please help wrap up 1.5 dev work!

```
[10:12]      --> DannyB (n=DannyB@66.17.133.19) has joined #svn-dev
[10:12]      --- ChanServ gives channel operator status to DannyB
[10:13]      <lgo> Why would svn switch require write access on the
branch I'm switching too??
[10:15]      <glasser> huh?
[10:16]      <glasser> where are you seeing that?
[10:16]      <lgo> glasser: no, don't bother, misinterpreted a result
of a test I'm writing.
[10:16]      <lgo> pfew
[10:16]      --> DannyB_ (n=DannyB@56.sub-75-197-117.myvzw.com) has
joined #svn-dev
[10:17]      <glasser> yay
[10:17]      <-- DannyB_ has quit (Client Quit)
[10:19]      <-- DannyB has quit (Read error: 104 (Connection reset
by peer))
[10:19]      <CIA-7> cmpilato * r27220 /trunk/subversion/
(libsvn_ra_neon/fetch.c libsvn_ra_serf/getlocationseg
ments.c): Clean up some subpools after we're done
with them. ...
[10:20]      <jeremyw> Not being a dav expert, why would I want to use neon
over serf or vice versa?
```



Mailing lists  
Mailing list etiquette  
Announcements

**Getting Subversion**

Downloads  
Documents & files  
Version control

**Project Management**

Development  
Road map  
Task List

**Additional Resources**

FAQ  
Subversion Book  
License

Tools & Contrib  
Links  
Testimonials

Commercial support  
Trademark policy

Security

**Search**

This project

Go

Advanced search

Powered by  
**COLLABNET**

**How do I...**

Get release notes for  
CollabNet 4.5.1?  
Get help?

**Mailing list details**

**List address** [dev@subversion.tigris.org](mailto:dev@subversion.tigris.org)

**List description** The development mailing list for Subversion. Non-development questions should be posted to 'users' instead; that includes build problems, configuration issues, etc, as well as usage questions.

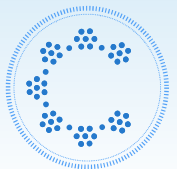
**Total messages** 131269

Messages 1 through 783 of 783 shown | [Show paged](#)

Author	Subject	Date	
Charles Acknin	<a href="#">svn commit: r27222 - branches/svnpatch-diff/subversion/libsvn_client</a>	2007-10-16	<a href="#">Delete</a>
Blair Zajac	<a href="#">svn_opt_parse_path and canonical paths</a>	2007-10-16	<a href="#">Delete</a>
David Glasser	<a href="#">svn commit: r27224 - trunk/subversion/tests/cmdline</a>	2007-10-16	<a href="#">Delete</a>
David Glasser	<a href="#">svn commit: r27222 - branches/svnpatch-diff/subversion/libsvn_client</a>	2007-10-16	<a href="#">Delete</a>
Blair Zajac	<a href="#">svn_opt_parse_path peg revision inconsistency</a>	2007-10-16	<a href="#">Delete</a>
Vlad Georgescu	<a href="#">Authz inconsistency in svn_repos_get_logs()</a>	2007-10-16	<a href="#">Delete</a>
Miller, Eric	<a href="#">[Issue 2972] - Error processing command 'set-timestamp'</a>	2007-10-16	<a href="#">Delete</a>
Vlad Georgescu	<a href="#">Authz inconsistency in svn_repos_get_logs()</a>	2007-10-16	<a href="#">Delete</a>
svnngo@mobsol.be	<a href="#">svn commit: r27190 - in trunk/subversion: libsvn_ra_neon tests/cmdline</a>	2007-10-16	<a href="#">Delete</a>
C. Michael Pilato	<a href="#">svn checkpoint: r27213 - checkpoints/relative-externals</a>	2007-10-16	<a href="#">Delete</a>
Brian W. Fitzpatrick	<a href="#">svn checkpoint: r27139 - checkpoints</a>	2007-10-16	<a href="#">Delete</a>
Erik Huelsmann	<a href="#">svn checkpoint: r27213 - checkpoints/relative-externals</a>	2007-10-16	<a href="#">Delete</a>
Blair Zajac	<a href="#">svn checkpoint: r27213 - checkpoints/relative-externals</a>	2007-10-16	<a href="#">Delete</a>
Justin Erenkrantz	<a href="#">svn checkpoint: r27213 - checkpoints/relative-externals</a>	2007-10-16	<a href="#">Delete</a>
Karl Fogel	<a href="#">svn checkpoint: r27213 - checkpoints/relative-externals</a>	2007-10-16	<a href="#">Delete</a>
Karl Fogel	<a href="#">svn checkpoint: r27213 - checkpoints/relative-externals</a>	2007-10-16	<a href="#">Delete</a>
Karl Fogel	<a href="#">[RFC] Distinguish Between Paths/File For Summarized Diff Output</a>	2007-10-16	<a href="#">Delete</a>
Blair Zajac	<a href="#">svn checkpoint: r27213 - checkpoints/relative-externals</a>	2007-10-16	<a href="#">Delete</a>
Blair Zajac	<a href="#">svn checkpoint: r27213 - checkpoints/relative-externals</a>	2007-10-16	<a href="#">Delete</a>
Ben Collins-Sussman	<a href="#">svn checkpoint: r27213 - checkpoints/relative-externals</a>	2007-10-16	<a href="#">Delete</a>
Kamesh Jayachandran	<a href="#">[PATCH] XFail test to prove issue #2969</a>	2007-10-16	<a href="#">Delete</a>
Kamesh Jayachandran	<a href="#">svn merge question</a>	2007-10-16	<a href="#">Delete</a>

# Principles of Participation

- Participants are people, not organizations
- Make it *rewarding* to do things right
- Make it *easy* to do things right
- Make decisions in public
- **Document the culture**
- Don't work – amplify other people's work
- No obstacles: act now, audit later



**Project tools**[Project home](#)[Membership](#)[Issue tracker](#)[Mailing lists](#)[Mailing list etiquette](#)[Announcements](#)**Getting Subversion**[Downloads](#)[Documents & files](#)[Version control](#)**Project Management**[Development](#)[Road map](#)[Task List](#)**Additional Resources**[FAQ](#)[Subversion Book](#)[License](#)[Tools & Contrib](#)[Links](#)[Testimonials](#)[Commercial support](#)[Security](#)

## Hacker's Guide to Subversion

If you are contributing code to the Subversion project, please read this first.

\$LastChangedDate: 2007-10-09 11:10:32 -0700 (Tue, 09 Oct 2007) \$

### Table of Contents

- [Participating in the community](#)
- [Theory and documentation](#)
- [Code to read](#)
- [Directory layout](#)
- [Coding style](#)
- [Secure coding guidelines](#)
- [Destruction of stacked resources](#)
- [Document everything](#)
- [Using page breaks](#)
- [Error message conventions](#)
- [Other conventions](#)
- [APR pool usage conventions](#)
- [APR status codes](#)
- [Exception handling](#)
- [Automated tests](#)
- [Writing test cases before code](#)
- [Debugging the server](#)
- [Tracking down memory leaks](#)
- [Writing log messages](#)

Security

**Search**
 This project  

Advanced search

 Powered by  
**How do I...**
 Get release notes for  
 CollabNet 4.5.1?  
 Get help?
**Category** **Featured projects**

scm	Subversion, Subclipse, TortoiseSVN, RapidSVN
issuetrack	Scarab
requirements	xmlbasedsrs
design	ArgoUML
techcomm	SubEtha, eyebrowse, midgard, cowiki
construction	antelope, scon, framework, build-interceptor, propel, phing
testing	maxq, aut
deployment	current
process	ReadySET
libraries	GEF, Axion, Style, SSTree
Over 500 <a href="#">more tools...</a>	

- [Writing log messages](#)
- [Crediting](#)
- [Patch submission guidelines](#)
- [Filing bugs / issues](#)
- [Issue triage](#)
- [Commit access](#)
- [The configuration/build system under unix](#)
- [How to release a distribution tarball](#)
- [Release numbering, compatibility, and deprecation](#)
- [Stabilizing and maintaining releases](#)
- [Signing source distribution packages](#)
- [Localization \(I10n\)](#)

**Participating in the community**

Although Subversion is originally sponsored and hosted by CollabNet (<http://www.collab.net>), it's a true open-source project under a BSD-style license. A number of developers work for CollabNet, some work for other large companies (such as RedHat), and many others are simply excellent volunteers who are interested in building a better version control system.

The community exists mainly through mailing lists and a Subversion repository. To participate:

Go to <http://subversion.tigris.org/> and

- Join the "dev", "svn", and "announce" mailing lists. The dev list, [dev@subversion.tigris.org](mailto:dev@subversion.tigris.org), is where almost all discussion takes place. All development questions should go there, though you might want to check the list archives first. The "svn" list receives automated commit emails.
- Get a copy of the latest development sources from <http://svn.collab.net/repos/svn/trunk/>. New development always takes place on trunk. Bugfixes, enhancements, and new features are backported from there to the various release branches.

There are many ways to join the project, either by writing code, or by testing and/or helping to manage the bug database. If you'd like to contribute, then look at:

- The bugs/issues database [http://subversion.tigris.org/project\\_issues.html](http://subversion.tigris.org/project_issues.html)
- The bite-sized tasks page [http://subversion.tigris.org/project\\_tasks.html](http://subversion.tigris.org/project_tasks.html)

To submit code, simply send your patches to [dev@subversion.tigris.org](mailto:dev@subversion.tigris.org). No, wait, first read the rest of this file, *then* start sending patches to [dev@subversion.tigris.org](mailto:dev@subversion.tigris.org). :-)

**Subversion and IDEs**

Eclipse

JDeveloper

NetBeans

dev@subversion.tigris.org. :-)

To help manage the issues database, read over the issue summaries, looking and testing for issues that are either invalid, or are duplicates of other issues. Both kinds are very common, the first because bugs often get unknowingly fixed as side effects of other changes in the code, and the second because people sometimes file an issue without noticing that it has already been reported. If you are not sure about an issue, post a question to dev@subversion.tigris.org. ("Subversion: We're here to help you help us!")

Another way to help is to set up automated builds and test suite runs of Subversion on some platform, and have the output sent to the svn-breakage@subversion.tigris.org mailing list. See more details at <http://subversion.tigris.org/servlets/ProjectMailingListList> in the description for the svn-breakage list.

**Theory and documentation**

## 1. Design

A [design spec](#) was written in June 2000, and is a bit out of date. But it still gives a good theoretical introduction to the inner workings of the repository, and to Subversion's various layers.

## 2. API Documentation

The [API documentation](#) is generated by Doxygen from the C header files and gives detailed information about all functions, types and so on available in the Subversion API.

## 3. Delta Editors

Karl Fogel wrote a chapter for O'Reilly's 2007 book [Beautiful Code: Leading Programmers Explain How They Think](#) covering the design and implementation of Subversion's delta editor interface.

## 4. Network Protocols

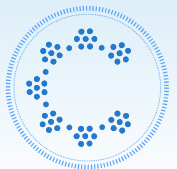
The [WebDAV Usage](#) document is an introduction to Subversion's DAV network protocol, which is an extended dialect of HTTP and uses URLs beginning with "http://" or "https://".

The [SVN Protocol](#) document contains a formal description of Subversion's ra\_svn network protocol, which is a custom protocol on port 3698 (the default), whose URLs begin with "svn://" or "svn+ssh://".

## 5. User Manual

Version Control with Subversion is a book published by O'Reilly that shows in detail how to effectively use Subversion. The text of the book is free, and is actively being revised. On-line versions are available at <http://svnbook.red-bean.com>. The XML source and translations to other languages are maintained in their own repository at <http://svn.red-bean.com/svnbook>.

...skip 43 pages...



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

announcement. Include this file in a mail, typically in reply to the announcement of the unofficial tarball.

If you've downloaded and tested a `.tar.bz2` file, it is possible to sign a `.tar.gz` file with the same contents without having to download and test it separately. The trick is to extract the `.bz2` file, and pack it using `gzip` like this:

```
bzip2 -cd subversion-1.3.0-rc4.tar.bz2 \  
| gzip -9n > subversion-1.3.0-rc4.tar.gz
```

The resulting file should be identical to the file generated by the release manager, and thus can be signed as described above. To verify that they are identical, you may use either the MD5 checksums or the release manager's signature, both of which should be provided with the tarballs.

## Localization (I10n)

Translation has been divided into two domains. First, there is the translation of server messages sent to connecting clients. This issue has been [p](#)  
[for now](#). Second there is the translation of the client and its libraries.

The `gettext` package provides services for translating messages. It uses the `xgettext` tool to extract strings from the sources for translation. This works by extracting the arguments of the `_()` and `N_()` macros. The former is used in context where function calls are allowed (typically anything except static initializers). The latter is used whenever `_()` isn't. Strings marked with `N_()` need to be passed to `gettext` translation routines whenever referenced in code. For an example, look at how the header and footer are handled in `subversion/svn/help-cmd.c`.

When using direct calls to `gettext` routines (`*gettext` or `*dgettext`), keep in mind that most of Subversion code is library code. Therefore the default domain is not necessarily Subversion's own domain. In library code you should use the `dgettext` versions of the `gettext` functions. The domain name is defined by the `PACKAGE_NAME` define.

All required setup for localization is controlled by the `ENABLE_NLS` conditional in `svn_private_config.h` (for \*nix) and `svn_private_config.hw` (for Windows). Be sure to put

```
#include "svn_private_config.h"
```

as the last include in any file which requires localization.

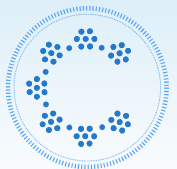
Also note that return values of `_()` and `*gettext()` calls are UTF-8 encoded; this means that they should be translated to the current locale being used as any form of program output.

The GNU `gettext` manual ([http://www.gnu.org/software/gettext/manual/html\\_node/gettext\\_toc.html](http://www.gnu.org/software/gettext/manual/html_node/gettext_toc.html)) provides additional information on writing translated programs in its section "Preparing Program Sources". Its hints mainly apply to string composition.



# Principles of Participation

- Participants are people, not organizations
- Make it *rewarding* to do things right
- Make it *easy* to do things right
- Make decisions in public
- Document the culture
- **Don't work – amplify other people's work**
- No obstacles: act now, audit later



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD



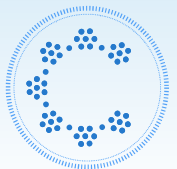
-----  
r27593 | hwright | 2007-11-05 07:10:13 -0800 (Mon, 05 Nov 2007) | 11 lines

In the command line client, don't reference a pointer to a local variable outside of its declared scope.

Found by: Barry Scott <barry@barrys-emacs.org>  
Patch by: Stefan Sperling <stsp@elego.de>

\* subversion/svn/copy-cmd.c

(svn\_cl\_\_copy): Allocate the peg revision for each copy source on the heap, instead of just using a pointer to the loop variable (which changes each time through the loop).



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

## Contributors as of r27762

---

The following list of contributors and their contributions is meant to help us keep track of whom to consider for commit access. The list was generated from "svn log" output by [contribulyze.py](#), which looks for log messages that use the [special contribution format](#).

*Please do not use this list as a generic guide to who has contributed what to Subversion!* It omits existing [full committers](#), for example, because they are irrelevant to our search for new committers. Also, it merely counts changes, it does not evaluate them. To truly understand what someone has contributed, you have to read their changes in detail. This page can only assist human judgement, not substitute for it.

1. [Madan U S <madan@collab.net> \(madanus\)](#) [51 patches, 19 non-patches] (partial committer)
2. [Bhuvaneshwaran <bhuvan@collab.net> \(bhuvan\)](#) [37 patches, 4 non-patches] (partial committer)
3. [Joe Swatosh <joe.swatosh@gmail.com> \(joeswatosh\)](#) [17 patches, 24 non-patches] (partial committer)
4. [Senthil Kumaran S <senthil@collab.net> \(styleesen\)](#) [28 patches, 2 non-patches] (partial committer)
5. [Stefan K ung <tortoisesvn@gmail.com>](#) [18 patches, 21 non-patches]
6. [Kouhei Sutou <kou@cozmixng.org> \(kou\)](#) [10 patches, 23 non-patches] (partial committer)
7. [Giovanni Bajo <rasky@develer.com> \(giovannibajo\)](#) [6 patches, 26 non-patches] (partial committer)
8. [S. Ramaswamy <srsy70@gmail.com>](#) [16 patches, 3 non-patches]
9. [Jelmer Vernooij <jelmer@samba.org> \(jrvernooij\)](#) [15 patches] (partial committer)
10. [Peter Samuelson <peter@n12n.org>](#) [11 patches, 8 non-patches]

133. [Norbert Unterberg <nepo@gmx.net>](#) [2 patches, 1 non-patch]
134. [Tim Van Holder <tim.van.holder@telenet.be>](#) [2 patches, 1 non-patch]
135. [phpxcache](#) [1 patch, 3 non-patches]
136. [Ariel Arjona <beerfrick@gmail.com>](#) ([beerfrick](#)) [2 patches] (partial committer)
137. [Art Haas <ahaas@neosoft.com>](#) [2 patches]
138. [Artem Egorkine <arteme@gmail.com>](#) [2 patches]
139. [Artem Khodush <greenkaa@mail.ru>](#) [2 patches]
140. [Augie Fackler <durin42@gmail.com>](#) [2 patches]
141. [Barry Scott <barry@barrys-emacs.org>](#) [4 non-patches]
142. [Bill Wohler <wohler@newt.com>](#) [1 patch, 2 non-patches]
143. [Bruce A. Mah <bmah@packetdesign.com>](#) [2 patches]
144. [Craig Peterein <craig@peterein.org>](#) [2 patches]
145. [Daniel Patterson <danpat@danpat.net>](#) ([danpat](#)) [2 patches] (partial committer)
146. [Daniel Westermann-Clark <dwc@tigris.org>](#) [2 patches]
147. [David Mahakian <tmoli42@gmx.net>](#) [2 patches]
148. [David Waite <mass@akuma.org>](#) ([mass](#)) [2 patches] (partial committer)
149. [Edmund Horner <ejrh00@gmail.com>](#) [2 patches]

# Barry Scott <barry@barrys-emacs.org>

Found	Review	Suggested
<a href="#">r27619, r27593</a>	<a href="#">r7349</a>	<a href="#">r27618</a>

[r27619](#) | [djames](#) | 2007-11-06 11:14:32 -0600 (Tue, 06 Nov 2007)

\* autogen.py: Add helpful note when ctypesgen subdirectory is missing.

Found by: Barry Scott <barry@barrys-emacs.org>

[r27618](#) | [djames](#) | 2007-11-06 11:10:05 -0600 (Tue, 06 Nov 2007)

\* autogen.py: Use apu-1-config to calculate include paths and link paths for apr-util. Add new '--apu-config' option for specifying path to apu-config script.

Suggested by: Barry Scott <barry@barrys-emacs.org>

[r27593](#) | [hwright](#) | 2007-11-05 09:10:13 -0600 (Mon, 05 Nov 2007)

In the command line client, don't reference a pointer to a local variable outside of its declared scope.

Found by: Barry Scott <barry@barrys-emacs.org>

Patch by: Stefan Sperling <stsp@elego.de>

\* subversion/svn/copy-cmd.c  
 (svn\_cl\_copy): Allocate the peg revision for each copy source on the heap, instead of just using a pointer to the loop variable (which changes each time

[svn]

Repository: svn

Go

# Revision 27593

Jump to revision:

Go



Author: hwright

Date: Mon Nov 5 15:10:13 2007 UTC (*6 days, 13 hours ago*)

Log Message:

In the command line client, don't reference a pointer to a local variable outside of its declared scope.

Found by: Barry Scott <[barry@barrys-emacs.org](mailto:barry@barrys-emacs.org)>

Patch by: Stefan Sperling <[stsp@elego.de](mailto:stsp@elego.de)>

\* subversion/svn/copy-cmd.c  
 (svn\_cl\_copy): Allocate the peg revision for each copy source on the heap, instead of just using a pointer to the loop variable (which changes each time through the loop).

Changed paths:

Path

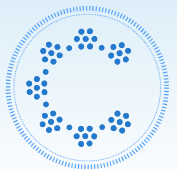
[trunk/subversion/svn/copy-cmd.c](#)

Details

[modified](#) , [text changed](#)
[svnadmin@svn.collab.net](mailto:svnadmin@svn.collab.net)
Powered by [ViewVC 1.0.4](#)[ViewVC Help](#)

# Principles of Participation

- Participants are people, not organizations
- Make it *rewarding* to do things right
- Make it *easy* to do things right
- Make decisions in public
- Document the culture
- Don't work – amplify other people's work
- **No obstacles: act now, audit later**



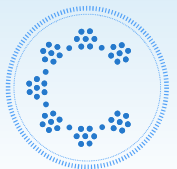
**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

# Optimize Toward Action

- Commit access for *anyone*
- Shared mailing list
- Visible results quickly

**“Audit trail” means “No regrets”.**

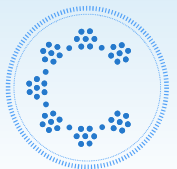


**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD



Recovery is better  
than protection.

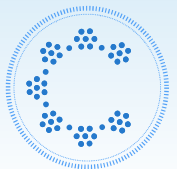


**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD

# Principles of Participation

- Participants are people, not organizations
- Make it *rewarding* to do things right
- Make it *easy* to do things right
- Make decisions in public
- Document the culture
- Don't work – amplify other people's work
- No obstacles: act now, audit later



**CIVIC COMMONS**

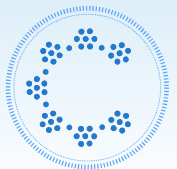
SHARING TECHNOLOGY FOR THE PUBLIC GOOD

# Your Volunteer Community?



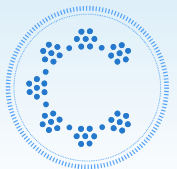
# References

- [wiki.civiccommons.org/Releasing\\_Open\\_Source](http://wiki.civiccommons.org/Releasing_Open_Source)
- [wiki.civiccommons.org/Open\\_Source\\_Development\\_Guidelines](http://wiki.civiccommons.org/Open_Source_Development_Guidelines)
- [wiki.civiccommons.org/Choosing\\_a\\_License](http://wiki.civiccommons.org/Choosing_a_License)
- [wiki.civiccommons.org/Legal\\_Policy](http://wiki.civiccommons.org/Legal_Policy) (*big*)
- [civiccommons.org/2011/05/it-dashboard-six-weeks-in](http://civiccommons.org/2011/05/it-dashboard-six-weeks-in)
- [civiccommons.org/2011/01/be-open-from-day-one](http://civiccommons.org/2011/01/be-open-from-day-one)
- *See the rest of CivicCommons.org too!*
- [cio-nii.defense.gov/sites/oss/index.shtml](http://cio-nii.defense.gov/sites/oss/index.shtml)



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD



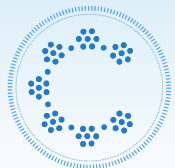
*Ben Collins-Sussman and Brian Fitzpatrick:*

“How Open Source Projects Survive  
Poisonous People (And You Can Too)”

<http://video.google.com/videoplay?docid=-4216011961522818645>

“What's In It For Me? Benefits  
from Open Sourcing Code”

<http://youtube.com/watch?v=ZtYJoatnHb8>



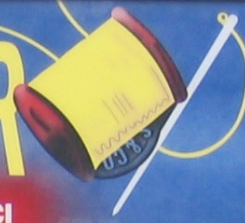
**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD



# PANTALONS SUPERIEUR

LES JEANS LEVI'S SONT DISPONIBLES ICI



PARASUCO

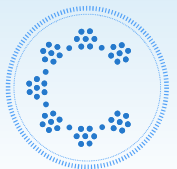
69

Levi's  
44<sup>ES</sup>  
44<sup>ES</sup>

4  
296

PTT  
4  
297

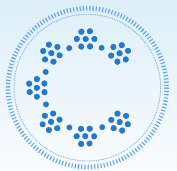




**CIVIC COMMONS**

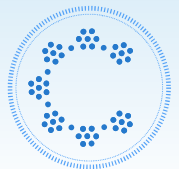
SHARING TECHNOLOGY FOR THE PUBLIC GOOD





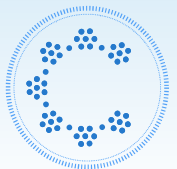
**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD



**CIVIC COMMONS**

SHARING TECHNOLOGY FOR THE PUBLIC GOOD