

# How did you get involved with your open source community? [closed]

Asked 15 years, 8 months ago   Modified 10 years, 5 months ago

Viewed 917 times



10



**Closed.** This question does not meet [Stack Overflow guidelines](#). It is not currently accepting answers.



This question does not appear to be about programming within the scope defined in the [help center](#).

Closed 10 years ago.

[Improve this question](#)

It seems that the normal progression to join projects is to contribute for a while, earn the trust, then get accepted as a member of the community (i.e. having commit access).

Now, I already apparently know "the best way" of how to get involved, in a manner of speaking; this is not my question; what I was hoping to attain is: How did everyone else get involved? Surely not everyone has gone down the "find a project and submit patches" route - or have they? I don't happen to know anybody in the open source community, so I'm just itching to know...

Perhaps you already knew someone in a community and just fell into it? Maybe you were getting frustrated with some bug and started contributing regularly as a result? Maybe you did just spot a project on SourceForge...

## Update:

It seems that the most common reason is simply scratching an itch, to [quote singpolyma](#): "Looking for a project to contribute to is often *not* the right way." Instead, you should join the open source community by contributing to a project that you already know and use.

## Important:

Please, please, please: Tell me about your specific experience, no general answers please. Also, answer only if you are either a project member or a patch contributor. Please do not give advice on how to join a community, this isn't the kind of answer I'm looking for. If you would like to give advice on joining a community, [please answer in this other thread](#).

## Great Answers:

- [Mark Harrison](#) talks about [Tcl](#), [cx\\_Oracle](#), [kap](#) and [orapig](#)
- [singpolyma](#) talks about [DiSo](#) and [Greasemonkey](#)
- [Pax](#) talks about contributing to [GnuCash](#) because of his wife

## Related:

- [How to get involved in an open source project](#)
- [How Open Source Projects Survive Poisonous People \(And You Can Too\)](#)

open-source

Share

edited Jun 20, 2020 at 9:12

Improve this question

Follow

community wiki

11 revs, 2 users 93%

Nick Bolton

---

5 This question appears to be off-topic because it isn't about a specific programming problem. – [Wladimir Palant](#) Jul 17, 2014 at 19:01

---

6 Answers

Sorted by:

Highest score (default)



9



### My personal anecdotes:

1. I got involved with the Tcl community when it was first starting out in 1991 or so. The mailing list and later the usenet newsgroup were pretty important to connect with people. I specialized in user evangelism and teaching, and eventually ended up writing two



books about the subject. One of them is still in print after ten years:



<http://www.amazon.com/dp/0201634740>

2. Now I use a lot of Python, and really like the cx\_Oracle package. Again I was active in the mailing list, and contributed a few patches.
3. I've made a couple of software packages available that I had written for work. By making them open source, I was able to get some nice contributions back, and since they were not the "secret sauce" of my employers at the time, they didn't mind sharing the code. The two most popular packages were  
<http://sourceforge.net/projects/kap/> The Kinetic Application Processor -- this was built when I was working on the China Internet backbone.  
<http://code.google.com/p/orapig/> - OraPIG, the Oracle Python Interface Generator -- it generated Python code to call APIs defined in the database, and includes an XML-RPC database interface.

## **Advice:**

Instead of looking for projects to join, try contributing to projects you already use.

It's often difficult to jump into the "core" development, because (a) on a big project, that might be a pretty big chunk of code to understand, and (b) there are probably a core group of people already working on it.

So, suppose you like a certain piece of software and want to start contributing, you can start working around the edges. Here's a couple of concrete tasks that will help you to become integrated with the group.

- write some test cases for bugs to add to the regression test suite.
- browse through the bug database and find a bug to work on. This might be the best way to get into the core development.
- look at the feature request database and see if there's a small task you can work on.
- look for "user doc" requests... a lot of them involve writing example code which you can provide.

Good luck!

Share Improve this answer

edited May 17, 2010 at 8:33

Follow

community wiki

8 revs, 3 users 55%

Mark Harrison

---

@Mark Harrison: Thanks for the tip. I'm looking for anecdotes, do you have any? – [Nick Bolton](#) Apr 5, 2009 at 1:21

---

That's the first time I've seen a sentence with "Oracle" and "pig" in it that wasn't "Oracle runs like a pig" :-)) – [paxdiablo](#) Apr 5, 2009 at 5:33

---



6



The way people normally get involved is:

- you use the FOSS product in your day to day work
- you notice a problem or a missing feature
- you mail the maintainer to ask if this bug/missing feature is real
- the maintainer says yes, this is a bug/missing feature
- you decide to try to fix/add the bug/feature
- you code like mad
- you submit a patch to the maintainer
- the maintainer laughs in or face or says "thanks very much!"

If you repeat the last few steps a few times, the maintainer will probably give you commit access to the project's RCS repository, and then you can really become dangerous. But the bottom line is that it is up to you to do something i.e. write some code - merely being "interested" in a project is not enough.

Share Improve this answer

[edited Jul 17, 2014 at 18:20](#)

Follow

3 revs, 3 users 85%  
anon

---

Thanks, but do you have any specific stories about how you got involved? – [Nick Bolton](#) Apr 5, 2009 at 0:53

---

I did exactly what I've described on a couple of projects - I don't know what you mean by "stories". – anon Apr 5, 2009 at 0:57

---

Pax's answer was pretty close to what I was thinking about; more anecdotal. – [Nick Bolton](#) Apr 5, 2009 at 1:01

---



5



I joined DiSo and Greasemonkey.

The best way I've found to get involved is to get in early in the life of the project, or just be very interested. With DiSo or the various github projects I'm on, it was the former, with my Greasemonkey contributions, the latter.

Looking for a project to contribute to is often not the right way. Use stuff and find out what you want to build/fix, then do that.

Share Improve this answer

edited Apr 5, 2009 at 1:14

Follow

community wiki  
2 revs, 2 users 67%  
[singpolyma](#)

---

Interesting comment about not looking for a project specifically to contribute to, I think I'm tending toward



4

I did a little bit of patch work on GnuCash since my wife restarted work part-time recently after our kids were a little more grown up.



I would've rather had my eyes ripped out with a hot poker than re-install Windows but GnuCash was missing something that [a certain other accounting package] had so I told her I'd get it added.

As it turns out, they took my patch and made it a lot better before putting it in (to the point where maybe 1% of the final patch was my stuff) but at least we can now use GnuCash instead of that proprietary stuff. They were also incredibly responsive - from patch submission to patch availability was only a week or so and it was in the product three weeks later.

I also once investigated getting a patch into the process accounting in the Linux kernel but the effort required far outweighed my needs :-)

I don't contribute on a regular basis, more as-needed (find your itch and scratch it). There are some who make a hobby of it but I'd rather be spending my spare time with the kids and, unfortunately, my employer won't pay me to contribute elsewhere.

That last bit particularly galled me since:



- the Linux patch would have greatly assisted our product (and a lot of others).
- it was change in behavior of another of our products that degraded the usefulness of our product.
- the solution was fairly simple, conceptually (the effort required was testing since a problem would have been high-impact [task switching] and very pervasive [everyone using Linux]).
- it would have been quicker to code up the patch than the workaround we eventually implemented.
- the workaround is a kludge (p'tooee).
- now, nobody in the world has the benefit of our patch (including us).

Share Improve this answer

edited Apr 5, 2009 at 1:00

Follow

answered Apr 5, 2009 at 0:45



[paxdiablo](#)

880k ● 241 ● 1.6k ● 2k

---

In response to the last bit, so, your company was using some open source software, you had the opportunity to fix it, and in short, they didn't let you? And spent time on a workaround instead... Yikes! hehe – [Nick Bolton](#) Apr 5, 2009 at 1:07

---

Well, they didn't let me on their dime. I could have done it on my own time but, as I said, I have other priorities. And I don't mind benefiting the world at large if it also serves me (a la GnuCash) but I would have received no benefit for the kernel



What I did was pretty simple; I opened one.

2



I have been joined by one permanent developer, and other two who donate code behind the scenes. The project is in very early stages, so not many users have downloaded it.



Share Improve this answer

[edited Apr 6, 2009 at 19:39](#)

Follow

community wiki

[4 revs, 2 users 63%](#)

[nbolton](#)

---



What really helps an open source project is having a plugin architecture.

0



It's much easier to contribute a simple plugin for eg. a file format than to try to add something to the Linux kernel.

This makes it a lot quicker and easier to build a community.



**TODO:** Please supply an anecdote.

Share Improve this answer

[edited Apr 6, 2009 at 16:26](#)

Follow

community wiki  
2 revs, 2 users 92%  
[Martin Beckett](#)

---

Building Linux Kernel Modules is really not that hard. LKMs are just like plugins. – [Zifre](#) Apr 6, 2009 at 16:23

---