Is contributing internal tools to open source worth the effort? [closed]

Asked 15 years, 6 months ago Modified 14 years, 8 months ago

Viewed 1k times

22

Closed. This question is opinion-based. It is not currently accepting answers.

Want to improve this question? Update the question so it can be answered with facts and citations by editing this post.

Closed 9 years ago.

Improve this question

I know this is a general question, but I'd like to hear other people's opinion about our case:

I work in a small company. Our main development tool is PowerBuilder, which is a very limited IDE with a shrinking community. We've created some tools, which we use internally to solve a certain needs. They have neither been properly designed nor properly tested, and are not in production quality. OTOH, they do save us quite some time, and might help others as well. I'm sure other companies have the same kind of tools, and was wondering how common a practice is it to share them with others. As I see it -

The pros:

- Good karma
- More attention to our website
- Perhaps getting fixes and improvements from others

The cons:

- Without investing more development, the tools might make us look bad
- Publishing of the code requires some effort
- Some of the tools might be too specialized for our needs
- The whole effort might go unnoticed given the shrinking community

Have you or your company ever contributed such tools, or used such tools developed by others? Is it worth the effort?

EDIT:

For those how wondered, the tools I had in mind include -

- A tool that makes using SourceSafe easier, by listing objects that are checked out to the current user or others, backing up checked-out objects, and reconstructing PBGs.

- A tool that recognizes PB controls at runtime, like Spy++ does (requires some infrastructure at the target app).
- PBNI wrapper for SQLite (in-process access, no ODBC).
- An SQL client, text measurement tool etc.

open-source **powerbuilder**

Share

Improve this question

Follow

edited Jun 17, 2009 at 19:12

asked Jun 16, 2009 at 18:22

Eran

22k ● 6 ● 58 ● 91

2 If you decide to release, please link your site from this question! We are a powerbuilder shop too and I would definitely be interested. In fact I might be able to get one or two of our internal tools released too. – Colin Pickard Jun 17, 2009 at 11:52

Very much interested in the code, too. – Frerich Raabe Jun 18, 2009 at 10:54

2 This question has been eloquently answered and your question is a great one. I just wanted to say good job; it is developers thinking like you do that make an IT shop (and user community) better. Of course there are exceptions; but how can anyone argue with your sincere desire to not just think of yourself but to put extra effort in and make things

better for everyone. In my experience this always comes back around in a positive way. There is one Fortune 100 client I've worked for that is an exception; and it was just one department; they were all about slamming junk out the door. – Rich Bianco Mar 21, 2011 at 20:54

7 Answers

Sorted by: Highest score (default) ⬍

▲

12

▼

✓

"Open source" originally meant you published a tool, and you made the *source available*. Because of some projects that expected, and in some cases through licenses **demanded** that changes to the source code be resubmitted for sharing, "open source" now quite often adds the concept of *collaborative development* to the mix. I did (or attempt to do) the latter; allow me to share.

There are magnitudes of difference between the effort associated with *source available* and *collaborative development* open source.

Leadership: You need to tell people the who, what, where, when, why and how of changes. And very possibly, you'll need to diplomatically poke and prod your volunteers. You may need to define the vision and prioritize goals of the project, and then enforce them when someone tries to take things another way. And, unless you only want people to come across your tool through serendipity, you'll have to advertise, running that very thin line (even thinner on the Internet) between attention-getting and gaudy. If the project is going to implement the concept of meritocracy, as many open source proponents say should happen, then someone will have to judge people's accomplishments and dole out the rights and responsibilities appropriately.

Work flow: I haven't done an exhaustive search by any stretch of the imagination, but I have yet to see a collaborative development platform that did all the things I needed. Part of the point of open source collaborative development is that the quantity involved in code review will cover any potential issues in quality of code submissions; I haven't seen a free tool integrated into a collaborative development platform that helped manage that cleanly yet (e.g. counting code reviews; auto-promoting after *x* reviews). We had to handle that, hacking manual methods into the existing tools. Probably at some point you'll have to define a version and create a build. Then there's the grunt tasks like documentation. (Ever try to release a new version of something free *without* release notes? The furor!! *grin*)

PB-specific issues: PowerBuilder is a commercial tool, and while there are *cheap* versions available, there are not *free* versions. The DRM added to PB11 has probably reduced or eliminated piracy that developers were probably doing to take copies of their office PB home, and while PB11 and later have a dual license policy that would allow developers to take home a copy legally (with permission and cooperation of the original license owners to create a second license), I don't see a lot doing it. (No scientific study, that's just what I see.) That cuts down a lot of potential collaboration, even from enthusiasts.

Issues of compatibility of code between versions of PowerBuilder, plus the fact that very few people will own every version, will limit again your list of potential contributors.

Don't get me wrong. I'd love to see more *collaborative development* open source in the PowerBuilder community. I'd love to know how to work out the issues myself, and I have an effort in the works to see if I can make a new model work. (My first effort to follow the popular model failed miserably, IMHO.)

Is there a reason to feel badly about firing a ZIP file up to the web and forgetting it? I don't know. Is there any more pride or embarrassment in a 4 year old ZIP file as opposed to a SourceForge project whose last contribution 3 1/2 years ago was a post "Where the heck is everyone?" There is a reason why Sybase CodeXchange devolved from a *collaborative development* platform to a *source available* platform: next to no one was using the *collaborative development* features. If you *source available* open source your code, you'll have plenty of company.

BTW, CodeXchange may be an answer to your concern about visibility to the PowerBuilder community, although you'll lose the web site traffic. The PowerBuilder Web Ring is another, significantly less effective, method to help your visibility that keeps traffic on your web site, but it demands a navigation bar on the target page on your site. CodeXchange may also be a way to get over your concerns about code quality and narrowness of purpose of what you have to share. *grin*

What should you do? Don't underestimate the effort with a *collaborative development* sharing, but don't let it stop you from a *source available* sharing.

Good luck,

Terry.

Share Improve this answer Follow

answered Jun 17, 2009 at 3:07

Terry
6,215 ● 19 ● 16

1 Terry, thank you for the detailed answer, and thank you for years of helping other PB programmers. I wish more experts from the Team joined StackOverflow, so we could have more PB related activity on this great site (I'm familiar with the newsgroups, but like this platform better). – Eran Jun 17, 2009 at 18:59

1 Eran, if you want to see Sybase's first jab at an HTTP-based forum, sign up for the PB12 CTP (response.sybase.com/forms/WW09JUNPB12PUBLICCTP) and the forum used to support that is HTTP-based. (I don't think it's very impressive, but judge for yourself.) SO has some nice features that I like, but I'm not convinced I could live without the threading. For the questions with one answer, and one person actually has that answer, it's great. For discussions or building a solution together, I'm not so sure. JMHO – Terry Jun 18, 2009 at 18:56

3

You can probably discount one of your cons: Anyone interested enough in this kind of tool to be evaluating your offering is unlikely to be writing *Company X are teh suxors* on your feedback form; rather if they find some deficiency in what you have put out there, you are likely to get helpful bug reports or even patches.

Share Improve this answer Follow

answered Jun 17, 2009 at 11:56

Colin Pickard
46.7k ● 13 ● 104 ● 151

2

If you can get your company to buy off on contributing to the community then I would go for it. it is always worth the effort to give back a little bit and this would definitely be a good way to get some of your tools out to the public and improved upon by the community.

As far as the cons go, I wouldn't worry too much about the criticism, it can only help you guys improve the next product you deliver and people will respect you from learning from your mistakes, nobody is perfect.

Even if your effort goes unnoticed by your shrinking community, future employees and clients will see that you are contributing outside of the company and may help with your reputation with them.

I think the pros far outweigh the cons on this one.

Share Improve this answer

answered Jun 16, 2009 at 18:26

Follow

Robert Greiner
29.7k ● 9 ● 66 ● 86

1

In short: go for it. I doubt there's little to lose, but much to gain.

The pros:

**Good karma*

never a bad thing to have.

**More attention to our website*

possibly a con if your code is *really* bad :)

**Perhaps getting fixes and improvements from others*

this is possibly the best thing you get from open-sourcing your code. Its all about sharing and helping each other, you get to use other's code, they get to use yours and everyone's gained from the trade.

The cons:

**Without investing more development, the tools might make us look bad*

I'd search through to remove dodgy/rude/stupid comments, tidy up the formatting etc.

**Publishing of the code requires some effort* requires barely any effort - set up an account in Sourceforge, create a SVN repo there and import your code. Then create a binary package (a zip file will do) and release it using the website. Might take you an hour, if you stop to read all the documentation.

**Some of the tools might be too specialized for our needs*

You could set the whole lot up as a group - eg PowerBuilder Tools, then people who see the really specialised tools won't have wasted their time getting them, they'll still have the 'more readily useful' tools.

**The whole effort might go unnoticed given the shrinking community*

Possibly, but then there's really no reason not to release the code. If you don't it may get completely lost to everyone when/if you change development tools.

Share Improve this answer Follow

edited Jun 16, 2009 at 18:52

answered Jun 16, 2009 at 18:45

gbjbaanb

52.6k ● 12 ● 110 ● 154

Publishing your source is a great way to get feedback. If you look bad because of it, that's ok. Just be willing to fix

1

the problem. If you want help with your improvements I can't think of a better way than asking for help.

By the way, plenty of open source projects can be credited with the growth of communities that were previously shrinking.

Share Improve this answer Follow

answered Jun 16, 2009 at 20:32

jjclarkson

5,974 ● 7 ● 42 ● 63

1

I think you've done a good job of identifying the pros and cons. And it's probably true that the pros will outweigh the cons. If no one likes the utilities and does nothing to or with them, then you've lost nothing really; bad code shouldn't scare experienced developers (most experienced developers, especially PB ones, have seen their share of legacy code). If even one person benefits, then you get the karma, eh?

If you proceed to submit your tools to the open source community, do as you have here, and admit up front that the tools are not polished. This may deter some from even looking at them, however, if they are at least functional and can be easily modified, then they still represent a head-start for any prospective beneficiaries. As a PB user myself, I would be curious to know more about free tools that can give us an edge in productivity.

Have you looked into Sybase CodeExchange? They have some open-source PB things there, including the

PowerBuilder Foundation Class framework.

Share Improve this answer Follow

answered Jun 16, 2009 at 20:53

Bernard Dy
1,992 ● 2 ● 29 ● 39

0

I just saw your response to my question - amazing that you have developed something similiar already. :-)

Regarding your question: the company I work for has a specific section on the web site where tools which we used internally and/or simple solutions (or code snippets) which customers frequently ask for are published. The license of these offerings is very liberal as well, I think it qualifies as open source.

In your particular case, I'm fairly interested in the Spy++-like application you talked about since I was looking for (and/or trying to develop) something like that myself.

I'm aiming for something which doesn't require any infrastructure in the target application, but so far I'd be happy to play with anything which works, even if it requires modifications to the applications. I'm just not familiar enough with the PowerBuilder API yet to make a judgement on whether this is possible without modifiying the target application.

As I mentioned, I already developed similiar Spy-like applications for ordinary Windows applications as well as managed code applications (which require interaction with the VM to query the state of the object tree), so my hope is that I'll be able to find a solution which does not require any target infrastructure.

Do you have the source code up somewhere already? It doesn't need to be compileable, I'd just be happy to look how you did it in principle so that I can (hopefully) derive something from it which solves my particular problem. In case you didn't upload the source code yet, maybe you can provide some email address which I can use to contact you privately? I tried looking for something on your profile, but so far - no luck. :-)

Share Improve this answer Follow

edited May 23, 2017 at 11:44

Community Bot

1 ● 1

answered Jun 18, 2009 at 10:52

Frerich Raabe

94.1k ● 19 ● 121 ● 216