VCS and single developer "team"

Asked 15 years, 10 months ago   Modified 14 years, 8 months ago

Viewed 919 times

▲

2

▼

🔖

🕘

I am a single developer working on a project for my company. I use subversion and Trac (for bug-tracking and communication with management types). I have a staging server and a production server. Today I checked in some code and discovered that my FSFS-based svn (v1.4) repository is irreparably corrupt. While this is quite a bummer it has afforded me the opportunity to move my VCS/staging system to a more modern distro (currently on a 2-year old system). (As far as the repo is concerned I do have a non-corrupted current version of the code, so while I lose all the history and comments of the development I don't lose any code. Whew.)
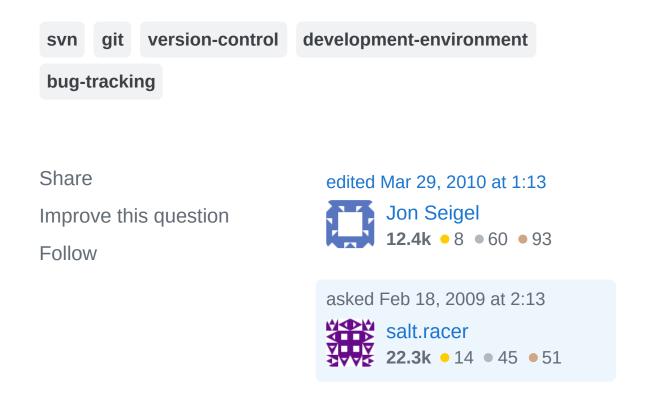
Currently I develop on Ubuntu and production runs RHEL5-64. My hardware will be staying the same, a 32-bit x86 single-core system.

I am familiar with SVN and it's constructs, but am feeling a little burned by the FSFS corruption issue. I don't know much about git except that it's rather popular. I currently use Trac to manage issues and I really like it's integration with svn. It appears that there are plugins to enable support for Git, but I'm not sure of the maturity of that development.

I'm currently thinking of building the following:

1. Ubuntu 8.10 Desktop (and then adding apache2 and other packages...the last time I tried adding a GUI to the server edition I just about pulled my hair out)

2. SVN (because I'm familiar with it and Git seems to be a bit overkill for a one person team)

3. Trac (because I'm familiar with it and it works with SVN).

I would like some suggestions and thoughts regarding my "new" vcs system. Is there a reason that I should move to Git? Is there something "better" than Trac?

svn   git   version-control   development-environment

bug-tracking

Share

Improve this question

Follow

7 Answers

Sorted by: Highest score (default)

I use git for all my personal projects. I can't imagine anything nicer for that. It's especially useful once the project outgrows you. Your development process needs to change almost not at all, except to add an occasional push to a public repo.

Share   Improve this answer

Follow

answered Feb 18, 2009 at 2:16

SingleNegationEliminati on

156k ● 35 ● 268 ● 306

> SVN (because I'm familiar with it and Git seems to be a bit overkill for a one person team)

git is pretty light weight and easy to use. The syntax is slightly different from subversion, but that's about it. I wouldn't say it is overkill. There really is nothing about it that is more or less complicated to use than subversion.

For bug tracking I use a web based system called Pivotal. It's free (for now), and it has some project management features built in. It's great for small teams because it is very simple, requiring almost no setup for use.

Share   Improve this answer

Follow

answered Feb 18, 2009 at 2:21

Himadri Choudhury

10.3k ● 6 ● 41 ● 47

Git's a fine source control system, especially if you're happy with the command line. SVN is obviously a good old workhorse, and is much better now with the 1.5 merge support.

Trac is good, but when we looked at it, it was limited to single projects and not good in the source control support department.

We use redmine now, which allows multiple projects and allows us to use different kinds of source control for each project, including git.

Oh yeah, and we use hudson for building :)

Share   Improve this answer

Follow

answered Feb 18, 2009 at 8:36

Jim T
12.4k ●5 ●31 ●43

We use svn/trac/ubuntu here.

To reduce the impact of corruption, I'd suggest that you implement an automated job (cron?) to hotcopy both the svn and trac repositories once a day and send them offsite (rsync) for backups. We use a NAS here, but something like S3 or Dreamhost works well too.

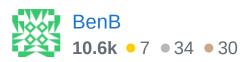That way, you will only lose one days worth of work if something goes wrong.

SVN hotcopy:

```
svnadmin hotcopy $REPOSITORY $DESTINATION
```

trac hotcopy:

```
trac-admin $REPOSITORY hotcopy $DESTINATION
```

Share   Improve this answer

Follow

answered Feb 18, 2009 at 2:29

BenB
10.6k ● 7 ● 34 ● 30

I second the recommendation for [Redmine](). While I'm primarily a Mercurial user, git is also supported (as are Darcs, svn and cvs). One of the nice things about distributed version control systems is that you basically get backups for free (you know you should have done backups of your old svn repo, right?)...

1

Share   Improve this answer

Follow

answered Feb 18, 2009 at 8:56

Keltia
14.7k ● 3 ● 31 ● 30

> Yes, I knew better. :(Redmine looks good. And considering that I am developing a Rails App... I like that as well.
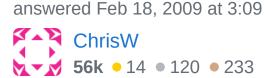> – [salt.racer]() Feb 18, 2009 at 20:26

[Perforce]() is free-as-in-beer for up to two users, and has a [good reputation]().

0

0

I'm going to step back and take a larger view. If you're familiar and comfortable with SVN and Trac, and they get the job done for you (ignore for a second the corruption issue), I would question the need to move. This is source control and bug tracking, and in general as long as they work for your team, I can't see spending a ton of effort on managing their environment, installing and evaluating new tools, etc.

My 10,000 ft. recommendation would be to use this opportunity to look at outsourcing this function entirely. There are hosts (disclaimer: my company is one) that will host your Subversion, Git, Trac, Lighthouse, etc. for you at free/low cost. Then, when something goes wrong with the disk array, or the FSFS or whatever, someone who spends 100% of his time worrying about this stuff can handle it, likely without you even becoming aware of the issue. If your company's policies allow you to use a host for this function, you might be able to save a weekend now, a weekend down the road (for the next catastrophe), and countless dollars of your productive time.

Share   Improve this answer

Follow

answered Feb 18, 2009 at 18:05

runako

6,152 ● 2 ● 26 ● 15

Unfortunately my software is developed under NDA with another third party. My employer would not support outsourcing to a code-hosting service (although I would love to). – salt.racer   Feb 18, 2009 at 20:24