Subject: Re: GIT essentials
Posted by Novo on Tue, 12 Jan 2010 05:07:00 GMT
View Forum Message <> Reply to Message

Just my two cents.

A couple years ago I had the same problem: my CVS server broke down and I lost one year history. Fortunately, I had backups from previous years.

After a short research I chose "monotone".

Cons:
1) It is just one executable (plus a couple of dlls). You don't really need to install it.
2) It is quite powerful. Actually, GIT borrowed a lot of ideas from monothone.
3) It is easy to use. Monotone has not that big set of commands.
4) Repository and workspace can be located in different places. (You can have different workplaces for the same branch. Actually, I'm using this for a different purpose.)

Pros:
1) No usable GUI. Nothing similar to TortoiseSVN or TortoiseGIT.
2) Not very popular. Actually, the same can be told about U++. Though, that doesn't make it bad.

The only thing I'm missing about monotone is GUI, which can be easily developed using U++, because monotone stores repository in SQLITE database.

A week ago I tried to migrate from monotone to GIT because of lack of GUI. That didn't happen because I couldn't resolve several work-flow issues.

1) GIT keeps repository and workspace in the same directory. In order to checkout another branch you need to clone repository. Basically, this means that you cannot store several projects in one repository. This is not a problem with monotone. In my case "branch" is often equal to "project".
2) U++ is a set of "packages". When I create a new project I want to assemble it from several "packages". This is possible with GIT, but in this case each package should be represented as a separate GIT repository. This seems to be way too complicated.

I'd like to get more feedback about work-flows with distributed VCS from you guys.

TIA