Subject: Re: Great (and funny) Linus' speach about GIT
Posted by mdelfede on Thu, 10 Jan 2008 11:16:06 GMT

View Forum Message <> Reply to Message

fudadmin wrote on Thu, 10 January 2008 01:31
2. Is it that bad that ... but why then you are using on a grand scale for U++ yourself?

from what I've seen on UVS2, it makes it difficult to go back through versions (maybe I'm wrong, but I think so). So, if you post wrong code, it can be time consuming to sync it again.
Quote:
3. Why no other version control system is recommended on the website?

Well, now there's svn updated quite often, if you need to get the last code. A development trunk for external contributions could be opened and kept in sync by somebody having write access to uvs2 repository. I'd keep the main svn trunk exactly in sync with uvs one, as is it now, but I see no problem to open another branch.
Of course, that'll make a bit difficult to keep all in sync, but can be done.
The only real problem I see is that, with sourceforge svn you can't open a single branch for write access (IMO), so who has write access to the development branch has write access to main branc too.
Quote:
....
1. What would be needed to improve the situation?

a miracle ?
Seriously speaking, as Mirek told, I see UVS as a mean for very *few* active developers to share code. As is it done, with many developers with write access it could be cumbersome to mantain.
Quote:
2. What if to combine uvs2(3?) (theIde?, too?) and mercurial.

I've looked inside mercurial for a while and it seems good stuff, but it has (IMHO) the weakness of all distributed versionin control systems. It relies mostly on active code shares between developers. AND, it doesn't have a merge tool embedded.
So, if 50 developers work on it, that'll be 50 different repositories with must be kept in sync. So, you'd need again a central FTP server (IMHO...) for repository sharing. OR you need the 50 developers share themselves the repo.
Linux kernel is a different stuff, it has a central developer (Linus) that join all the patches together, accepting or rejecting them. In my opinion, that's not a true decentralized system, even if it uses a decentralized versioning system.

Concluding.... I'm still thinking that a good centralized repo like svn works better for upp purpose (please, no flames about !)

Ciao

Max