

DVCS Round-Up: One System to Rule Them All?--Part 1

By *srlinuxx*

Created 21/01/2009 - 12:52am

Submitted by srlinuxx on Wednesday 21st of January 2009 12:52:24 AM Filed under [Software](#) [1]

A recent development in the field of version (or revision) control software is the emergence of distributed systems. These discard the notion of a central repository where all project history is kept, and replace it by a network of several repositories that are synchronised regularly. In this model, a developer has a complete repository on his hard disk, which is synchronised with the 'outside' whenever deemed necessary. This way one can commit early and often, without fear of breaking the others' code, and it also reduces network load as it does not need a network connection for every small change. In practise, of course there will be a central repository after all, but this is a decision made by the developers and not dictated by the system.

Working distributed has other advantages, too. For example, backups become a no-brainer, since every developer has the complete project history. The classic dilemma of who should have write access to the main project repository becomes significantly less problematic, since the maintainers simply pull the changes from a few persons they trust, which in turn (hopefully) have done likewise beforehand. Linus Torvalds referred to this as a 'network of trust,' which is actually the same principle for any security and encryption software.

So, while distributed version control has several advantages, one problem remains: we have to choose one that (hopefully) fits our needs. At the time of this writing, at least half a dozen systems compete for the developers' affections, and this series will hopefully provide you with a basis for your decision.

In this review, we will take a look at six different revision control systems. Namely these are git, Mercurial, darcs, Monotone, Bazaar (which is used by the Ubuntu project), and SVK (which is based upon Subversion). All six systems are distributed, and we will take a look at the different workflows supported (or enforced) by them.

[More Here](#) [2]

[Software](#)

Source URL: <http://www.tuxmachines.org/node/33851>

Links:

[1] <http://www.tuxmachines.org/taxonomy/term/38>

[2] <http://ldn.linuxfoundation.org/article/dvcs-round-one-system-rule-them-all-part-1>