pkgsrc with Git

Benny Siegert
‹bsiegert@NetBSD.org›
EuroBSDCon 2025
NetBSD dev summit

Why?

CVS sucks.

Yet, we are still on it.

How?

I have two trees:

- `~/pkgsrc` is a git clone (from GitHub)
- `~/pkgsrc-cvs` is a CVS checkout (directly from cvs.n.o)
    - Never any uncommitted changes in CVS

Remember: **Uncommitted changes are liabilities.**

Committing

I use `git cvsexportcommit` from devel/git-perlscripts
(git commands written in Perl but part of the upstream release).

NOT devel/git-cvs (yes, it's confusing).

Branches

Using a DVCS means you can have as many *local* branches
As you'd like.

- `trunk` is upstream only, update whenever needed
- `local/${hostname}` is *from when I last did a rolling replace*
  - I work in this branch, or create an update branch based on it.
  - I use `git cherry-pick` often, to copy a commit to another branch.

Merge before commit

If the change is in an update branch:
```
git rebase -i trunk
```

If the change is in the local branch:

- `git checkout trunk`
- `git checkout -b update/${pkgname}`
- `git cherry-pick ${hash}` and resolve merge conflicts

Commit

- `cd ~/pkgsrc-cvs`

- `GIT_DIR=~/pkgsrc/.git git cvsexportcommit -v -c ${hash}`

- `cd x/y`

- `make changes-entry && make changes-entry-commit`

Discussion

In CVS, just checking modified files takes ages.

- Also, what is "cvs update: move away `x/y'; it is in the way"??

In git, ~all operations are fast.

- I use GitHub because TNF doesn't offer a git remote (??).

Mercurial: in the past, clones were unacceptably slow.

Being able to switch between trunk and a work tree is a game changer.

One more thought on git

"I just want to say that we all finally are using mostly the same VCS and that is actually an undervalued, beautiful thing. An actual "standard" that is almost universal.

It used to be such a mess with all the VCS systems and having to know all of them to commit on various dependencies from different projects."

– Hacker News comment

Another DVCS on the Horizon: jj

This seems to be what the cool kids are using these days.

Can use Mercurial or git (but not CVS) as backends.

If you hate the git UI, maybe give it a try!

devel/jj