Commit porting (cherry-picking) guide

This guide is primarily about CppNix->Lix commit ports and was written when we were doing a *lot* of those before 2.90. It also applies to Lix->Lix ports, though, those are much easier, often just taking clicking "cherry pick" in the gerrit UI.

Don't forget, using Gerrit is a bit different than other systems.

Lix -> Lix ports

We generally only backport fixes from `main` if there are severe bugs that users will hit in production, or if we had to patch the derivation in nixpkgs. The only real thing to remember about this is to use `git cherry-pick -x`. Gerrit will match up the Change-Ids and link the CLs on release branches to the other CLs on release branches or main. Then, push as if any other change: `git push origin HEAD:refs/for/release-2.92` (e.g.).

You can also use the cherry-pick button in the Gerrit UI.

CppNix -> Lix ports

Commit style:

- Remember to use `git cherry-pick -x` to generate the `(cherry-picked from abc)` line.
- It's strongly preferred to include a link to the PR as `Upstream-PR: https://github.com/nixos/nix/pulls/123` in the trailer of the commit message.

GitHub encourages messy branch history, so the cherry-picking procedure can be rather annoying in a codebase like Lix which mandates that every commit passes CI. However, this document describes how we have generally done it.

Also note that we have renamed `src/` to `lix/`, which may or may not confuse git sometimes.

Procedure (focused mostly on CppNix -> Lix)

single commits

try `git cherry-pick -x` first. if this works, excellent. if not, apply the usual cherry-picking procedures:

- track down apply failures to intermediate changes. maybe cherry-pick those first if they're not too awful, but experience shows that they usually are too awful
- anything that touches the store or contains the word `Accessor` in the vicinity probably needs to be picked about and rewritten from scratch. that stuff is evidence of the CppNix version of lazy trees which is not in Lix.
- *always* test ofc, even if something applies cleanly it will sometimes just fail to build (or worse, run)
- nix prs tend to have broken inner commits. it is often necessary to pick parts from later commits in a pr to fix ci, in that case note this down as `fixes taken from <commits...>`
- sometimes a commit may be so broken that it can't reasonably be fixed except by squasing it with some other commit, in that case just squash them and not it down somehow (either with multiple `cherry picked from` commit hashes or multiple commit message+cherry-pick-hash blocks, depending on whether the fix messages were any useful)

full prs

single-commit prs were mostly picked using `cherry-pick -x -m1` to keep the association with the upstream pr number for clarity. this implicitly squashes the pr into a single commit so it's only useful for single-commit prs. (some prs that have broken intermediate commits also benefit from this, but see above for that)

when pushing these to gerrit please set a topic like `backport-<pr-number>` using push options (`-o topic=backport-<pr-number>` in `git push`) to delineate one picked pr from a pr that depends on it

---

Revision #3
Created 2024-03-27 07:39:33 UTC by jade
Updated 2026-02-01 13:04:29 UTC by jade