About git-repo and Submodules

Stefan Beller <sbeller@google.com> • Nov. 12th 2016
Overview

What is git-repo?

Why do we want to migrate off of it?

Current state of Gerrit/Git
What is git-repo?

- An open source tool to manage many Gerrit projects
- Parallel operations such as fetching, checkout.
- UI to hide Gerrit’s command line interface.
Life cycle so far
Reasons to replace git-repo

Does it solve the right problem?
- No synchronization between projects

On its Software quality
- Tests do not exist
- Documentation is very sparse (and stale?)
Synchronization is hard

- Gerrit’s current state
- Client side synchronisation via superprojects
- Gerrit support for submodules
- Git improvements for submodules
Gerrit’s “submitWholeTopic” feature

- Presented at the Gerrit Summit 2015
- Server side synchronization between Gerrit projects
- The actual problem: **client side** synchronization
Designing from scratch

- Git is the building block!
  Synchronization exists on the repository level!
- For large projects:
  - Make Git work at scale? Use subtrees?
  - Use submodules!
Submodules

● Not just performance
● ACLs
● Keeps the original projects intact!
  ○ e.g. `git format-patch` in linux.git
  ○ Merge from upstream
● The problem of submodules:
  ○ Performance
  ○ UX
Superprojects

- Only track Submodules (no other content)
- JGit can convert repo manifests to superprojects
- A submission in Gerrit can trigger the update of the submodule pointer in the superproject
  - Improved coupling via topics
  - Added ACLs
Gerrit Rest API

- Submitted-together (released)
  - List of changes
- Preview-submit (new)
  - archive of git-bundles for all affected repositories
Git submodule improvements, upstream

- Parallel network access (git-clone, git-fetch)
- `git clone --reference --recursive`
- Arbitrary bug fixes as reported by the mailing list, such as `git mv <submodule>`, and more path issues
Git submodule improvements, pending

- Grouping submodules via Git attributes
- A default group of submodules
- Integrate submodules into current workflows, e.g.
  git ls-files, grep, checkout
So long
and thanks for all the fish.

Stefan Beller <sbeller@google.com> • Nov. 12th 2016