Extra Git Tooling

git extras

Git Extras are a series of commands that composite core Git functionality that would typically be accomplished in multiple command executions.

- `git summary`
  - The number of files, commits, authors
- `git changelog`
  - The recent changes written to a CHANGELOG file
- `git commits-since`
- `git count`
  - Total number of commits to the repo
- `git delete-branch`
  - Delete a branch locally and remotely
- `git delete-submodule`
  - Completely remove a submodule
- `git delete-tag`
  - Delete a tag locally and remotely
- `git fresh-branch`
- `git graft`
- `git ignore`
- `git release`
- `git contrib`
- `git repl`
- `git undo`
  - Undo the last commit via a reset
- `git update-extras`
- `git setup`
- `git touch`
  - Touch a file and add it to the staging area

git-flow

Git flow is a set of commands that standardizes a *workflow of branching* with Git.

It leverages core Git to begin and merge branches for integrations, features, and bug fixes.

- `git-flow init`
  - Start a new flow
- `git flow release`
  - Complete a flow

Hub

Hub is a set of scripts that offer GitHub specific shortcuts to Git. These scripts can be invoked via the command `hub` or mapped on to the normal `git` command via a Git alias. In the second situation, all normal Git commands are passed through to the regular `git` command.

Typically, GitHub configuration is globally set via `git config --global github.user "yourusername"` and `git config --global github.token "tokenvalue"` such that the `hub` command knows what user to act on behalf of.

- `hub clone`
  - Clone from github, assuming the current user set in the git configuration under the github.user node.
- `hub remote add`
  - Add a remote to the current project name under a different username (e.g. a fork).
- `hub fetch`
  - Retrieve the contents of one or more forks of the current projects under the provided comma separated usernames.
- `hub cherry-pick USERNAME@SHA`
  - Cherry-pick the contents of a given SHA under a given username, even if it has never been added as a remote or fetched before.
- `hub fork`
  - Fork the current project under your username

- `hub create`
    - Create a new project on GitHub
- `hub init`
    - Initialize a new project and add it to GitHub
- `hub push`
    - Push a repo to a comma separated list of remotes and/or usernames
- `hub browse`
    - Open the web page corresponding to the current repo.
- `hub compare`
- `hub compare TAG1..TAG2`
    - Open the web UI to compare branches or SHAs within the current repo.
- `hub submodule`
    - Add a submodule from a GitHub project, assuming the local GitHub username
- `hub help`

If `hub` is mapped as an alias to `git`, you can get help by typing `git help hub`

Williams Misc Git Tools

William maintains a set of scripts that offer visualizations into the history of a Git repository.

- `git-wtf`
    - Visualize the current state of your git repository.
- `git-publish-branch`
    - Publish a branch to a remote repository and set up tracking with `--set-upstream`
- `git-rank-contributors`
    - Rank all authors by patch size.
- `git-show-merges`
    - Show which branches are merged into the current branch.