

Per Repo hooks

This is a little helper script that enables version controlled hooks, and per project hooks:

Examples are:

```
.hooks/pre-push      # version controlled repo hook
.hooks/pre-push.local # overrides the previous hook

.hooks.local/pre-push # will be run after the hook above
```

Files and folder ending in `.local` should be git-ignored.

Setup

Add the following helper script to your path:

```
#!/usr/bin/env bash
# name: repo-hooks.sh

#{{{ Bash settings
# abort on nonzero exitstatus
set -o errexit
# abort on unbound variable
set -o nounset
# don't hide errors within pipes
set -o pipefail
#}}}

#{{{ Variables
IFS=$'\t\n' # Split on newlines and tabs (but not on spaces)
SCRIPT_NAME=$(basename "${0}")
SCRIPT_DIR=$(cd "$(dirname "${BASH_SOURCE[0]}")" && pwd)
readonly SCRIPT_NAME SCRIPT_DIR
#}}}

if [ $# -eq 1 ]; then
```

```
hook=$1
else
  echo "Error: please provide exactly one hook"
  exit 1
fi

if test -f ".hooks/$hook"; then
  if test -f ".hooks/$hook.local"; then
    echo "Source +hl hooks/$hook.local"
    source ".hooks/$hook.local"
  else
    echo "Source +h hooks/$hook.local"
    source ".hooks/$hook"
  fi
fi

if test -f ".hooks.local/$hook"; then
  echo "Source +l hooks/$hook.local"
  source ".hooks.local/$hook"
fi
```

Add the following line to the hooks you want to enable this for in `~/.config/git/hooks/`:

```
repo-hooks.sh $(basename $0)
```

Note: Hooks run under the current user, so don't run this on untrusted repositories that might contain evil hooks!

Revision #4

Created 15 May 2023 07:58:13 by Sander

Updated 15 May 2023 08:23:15 by Sander