Two Ways to Share Git Hooks with Your Team

Git hooks are a useful feature that can be used to manage the integrity of your source repository. On my current project, I wanted to ensure that all my Go source files were formatted correctly before allowing them to be committed. Fortunately for me, there is a simple hook available that I can save as `.git/hooks/pre-commit` to make this happen.

This works well for my purposes, but I wanted to make it as simple as possible when sharing with the rest of the team. This Stack Overflow post gives a couple of possibilities that I'll go into more depth about.

Create Your Managed Hooks Directory

Since the `.git` directory isn't versioned, I created `.githooks` at the root where all these hooks live. You can choose whatever makes the most sense for your project. Remember that when adding hooks they must be executable, so make sure you `chmod +x` each of them to make that happen. Otherwise, you'll tear your hair out when you think they should run but they don't.

Choose Your Sharing Strategy

If you're using Git version 2.9 or greater, this is as simple as setting the `core.hooksPath` configuration variable to your managed hooks directory:

```
$ git config core.hooksPath .githooks
```

If you're using an earlier version, you need to ensure that your managed hooks make it into the `.git/hooks` directory. I think symlinking is a good way to go, just make sure to clear the old ones out first:

```
$ find .git/hooks -type l -exec rm {} \; && find .githooks -type f -exec ln -sf ../../{} .git/hooks/ \;
```

Share With Your Team

That takes care of your local configuration, but each team member will need to ensure the hooks are in the right place in their local repository each time they do a new checkout. I like to just put this into my `Makefile` and include it into my general project setup. Here are the two variations expressed as make targets:

```
init:
  git config core.hooksPath .githooks
```

```
init:
  find .git/hooks -type l -exec rm {} \;
  find .githooks -type f -exec ln -sf ../../{} .git/hooks/ \;
```