Git
Best practices

Jiří Jabůrek (jjaburek)
Red Hat 2012

Basic credentials setup

- Name & email
  - Use full name
  - Use valid email address
    - (or <name@something.else>)
- Example:

```
$ git config --global user.name "Jiri Jaburek"
$ git config --global user.email "jjaburek@redhat.com"
$ cat ~/.gitconfig
[user]
        name = Jiri Jaburek
        email = jjaburek@redhat.com
```

- More about `git-config`, config types
  - Global (per user) versus local (per repo)

Making a commit

Making a commit

- Descriptive header
  - Bad: "fixed the annoying bug"
  - Good: "fixed $var shell escaping"
  - Examples all over tests/ repos
- Use commit message for details
  - Repository "documentation" versus code comments
    - Git commits are tied to history (timeline)
- Commit only relevant changes
  - Eliminate secondary unrelated code changes
  - One big versus multiple smaller commits (later)

Making a commit

■ Always check diffs before commit/push

□ Uncommited changes (before commit)

– `git-diff`

– `git-diff --cached`

– `git-diff HEAD`

`git-gui`

□ Given multiple commits (before push)

– `git-log -p`

– `gitk`

– `tig`

– `git-diff origin/master`

Making a commit

- Watch for whitespace errors

  - Combined tabs and spaces

  - Tabs in a spaced file or vice versa

  - Trailing whitespaces

  - Empty lines with white characters

  - Solution?
    - `git-diff --check`
    - `git-apply --whitespace=fix`
      - has drawbacks

Workflow tips

Workflow tips

- Rebase before push
  - After committing your changes
    - `git pull --rebase`
  - With unstaged changes only
    - `git stash ; git pull ; git stash apply`

Workflow tips

- Multiple smaller commits versus one big
  - Easier to review / document / bisect
  - Easier to revert / cherry-pick / export (diff)
  - One big disadvantage; added effort / time
  - Separation
    - per-file ; per-change ; ...?
  - Not that hard to manage in git
    - `git-add --patch`

Workflow tips

- Subdirectory svn checkout versus Git
    - svn = different workflow, git unrelated
    - Management change
    - Git submodules

Questions?

Appendix: my ~/.gitconfig

```
[user]
    name = Jiri Jaburek
    email = jjaburek@redhat.com
    signingkey = 5C33C50A
[advice]
    pushNonFastForward = false
    statusHints = false
    commitBeforeMerge = false
    resolveConflict = false
    implicitIdentity = false
    detachedHead = false
[color]
    branch = auto
    diff = auto
    grep = auto
    interactive = auto
    showbranch = auto
    status = auto
    ui = auto
[color "diff"]
    whitespace = red reverse
[push]
    default = matching
```

```
[branch]
    autosetupmerge = false
[merge]
    stat = false
    tool = vimdiff
[rebase]
    stat = false
[sendemail]
    smtpserver = localhost
    smtpserverport = 25
    chainreplyto = false
[diff]
    renames = true
```

The End

Thanks for listening