Workflow With Git

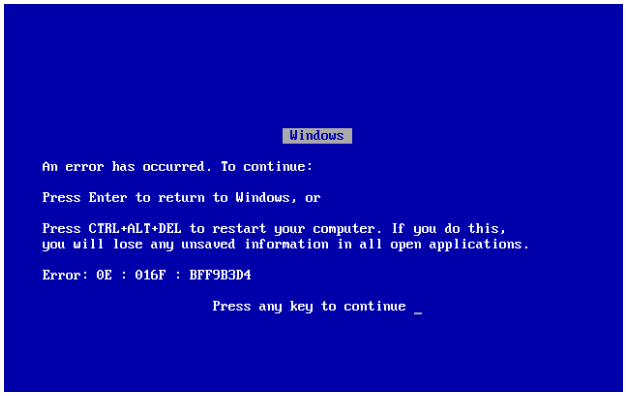
Kevin W. Gisi

ECRuby Meeting—April 8, 2010

Summary

Regardless of what language you prefer, version-control is an essential tool for getting things done - especially when collaborating with others. Git, an open-source tool, has quickly become one of the most widely-used versioning systems, mainly thanks to the ability to branch and merge with relative ease. We'll take a look at how you can begin to integrate Git into your current workflow.

Remember Windows 95?



Archives

Diff

```
*** /path/to/original ''timestamp''
--- /path/to/new      ''timestamp''
***************
*** 1,3 ****
--- 1,9 ----
+ This is an important
+ notice! It should
+ therefore be located at
+ the beginning of this
+ document!
...
```

Primitive Version Control

- Bitkeeper
- CVS
- Subversion

Beginnings of Git

Centralized Versioning

Distributed Versioning

Let's Create a Repository!

```
:$> ls
about.html  contact.html  index.html  style.css

:$> git init
Initialized empty Git repository in /home/gisikw/project
```

Adding initial files

```
:$> git add .

:$> git commit -m "Initial Commit"
Created initial commit ed3ec5b: Initial
 0 files changed, 0 insertions(+), 0 del
 create mode 100644 about.html
 create mode 100644 contact.html
 create mode 100644 index.html
 create mode 100644 style.css
```

Ignoring files

```
:$> ls
about.html ~about.html contact.html inde
```

.gitignore

```
~*.html
test.html
```

Reverting Changes

Tagging a Release

Feature Branches

Merging Branches

Additional Resources

Thanks!