

Introduction to VCS and Workflows

John Dickinson
Rackspace Hosting
john.dickinson@rackspace.com
<http://programmerthoughts.com>
[@notmyname](https://twitter.com/notmyname)

What we will cover

- What is a Version Control System?
- Why should I use one?
- What sort of workflows does VCS enable?

In the beginning...

In the beginning...

```
$ ls  
archive/  
archived_widgets.20100215/  
archive.20100416/  
widgets.c  
widgets.c.20100314  
widgets.c.20100226  
widgets.c.old  
widgets.c.proto  
widgets.c.with_flages.old  
widgets.c.with_flanges.new  
widgets.c_works  
widgets_and_flanges.c
```

Major Concepts of Version Control Systems

Major Concepts of Version Control Systems

- Repository (repo)
 - The history of a project
- Working directory
 - A specific version of a project
- Commit
 - To save a specific project changeset
 - A specific changeset

Major Concepts of Version Control Systems

- Checkout
 - To get a specific version of a project
- Blame/Annotate
 - show who did what in a file
- Log
 - See the history

Major Concepts of Version Control Systems

- Merge
 - Combine two or more revisions (commits)
- Trunk
 - The main flow of history
- Branch
 - Start a new path in history

Centralized vs. Decentralized

- Centralized
 - Simple concepts
 - Easy (to understand) backups and management
 - One place of accountability (permissions, auditing, etc)
 - Hard for devs
 - Must have access to do work
 - Conflict resolution is generally hard
 - Commit access is hard to manage
 - One mistake with commit access and everyone hurts

Centralized vs. Decentralized

- Distributed
 - Great for devs
 - Less barriers to “get stuff done”
 - branching and merging is easy(er)
 - Harder for people to grok
 - no central repo, how is it managed?
 - most systems are single entities

A (Very) Brief Overview of (Some) Major Version Control Systems

- CVS
 - Granddaddy of VCS
 - File-based
- svn
 - CVS, but better
 - Changeset-based
 - Established the CVS commands as a de-facto standard

A (Very) Brief Overview of (Some) Major Version Control Systems

- git
 - widely used (Linux kernel)
 - github
- hg
 - Python
 - bitbucket
- bZR
 - Canonical (Ubuntu)
 - launchpad

My Experience

None → CVS → (svn) → git → bzt

Workflow Examples

- Single dev
 - something new to work on!
 - (maybe) switch to dev branch
 - work, commit, work, commit
 - (maybe) merge with prod, when convenient
 - push to remote repo for backups

Workflow Examples

- `git init`
- Do some work
- `git add .`
- `git commit`

Workflow Examples

widgets (branch: master)

View Branch Create Branch Search

Subject	Author	Date
master better handling of usage m...	John Dickinson	2010-08-25 13:34:41
added error handling for the wid...	John Dickinson	2010-08-25 13:30:06
initial commit	John Dickinson	2010-08-25 13:28:24

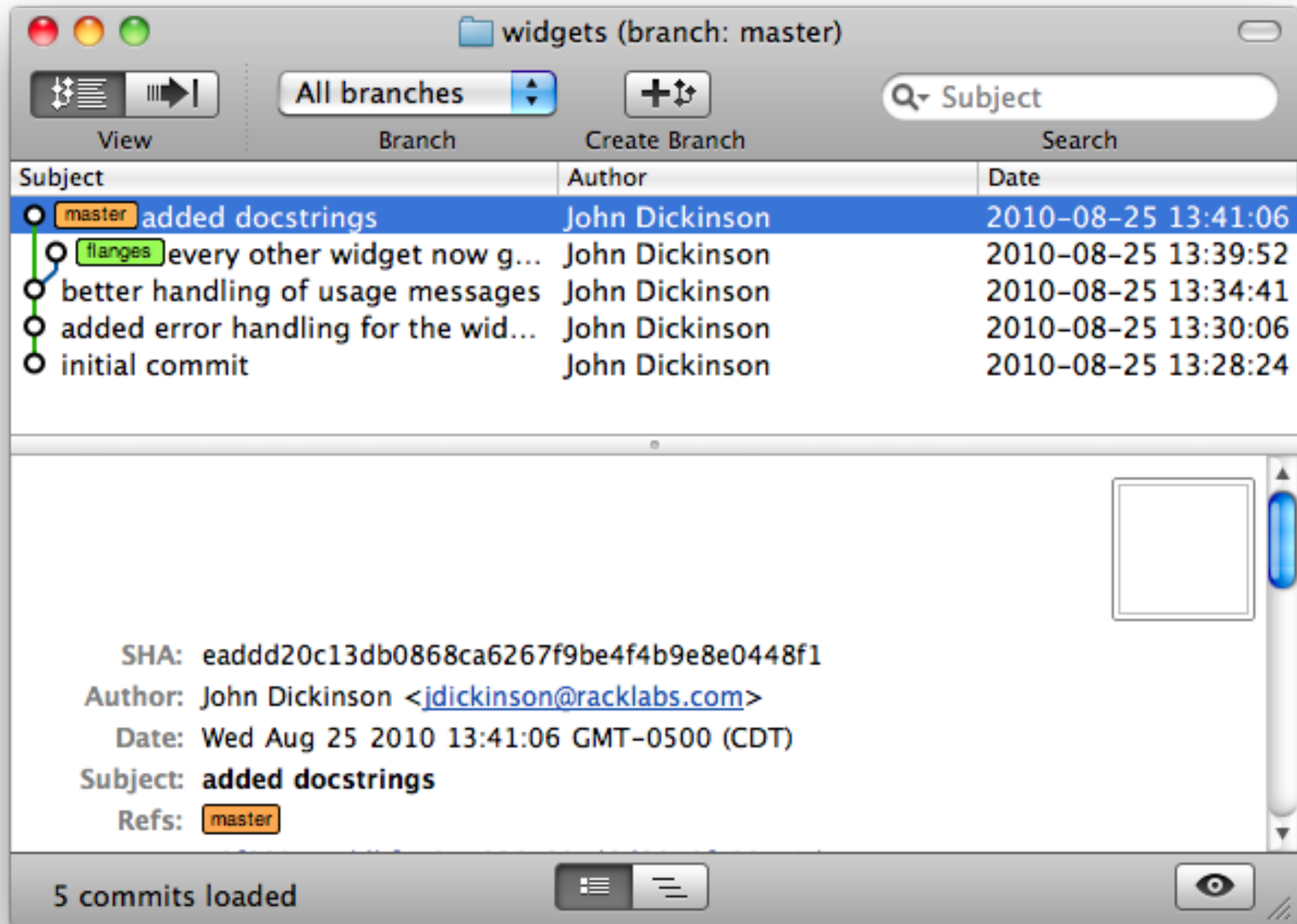
SHA: 70f682c77ddbfa7877029c60cd6d8840fe82ec1d
Author: John Dickinson <jdickinson@racklabs.com>
Date: Wed Aug 25 2010 13:34:41 GMT-0500 (CDT)
Subject: **better handling of usage messages**
Refs: **master**

3 commits loaded

Workflow Examples

- Widgets with flanges
 - `git checkout -b flanges`
 - `work, commit`
- Work continues on master

Workflow Examples



widgets (branch: master)

View Branch Create Branch Search Subject

Subject	Author	Date
master added docstrings	John Dickinson	2010-08-25 13:41:06
flanges every other widget now g...	John Dickinson	2010-08-25 13:39:52
better handling of usage messages	John Dickinson	2010-08-25 13:34:41
added error handling for the wid...	John Dickinson	2010-08-25 13:30:06
initial commit	John Dickinson	2010-08-25 13:28:24

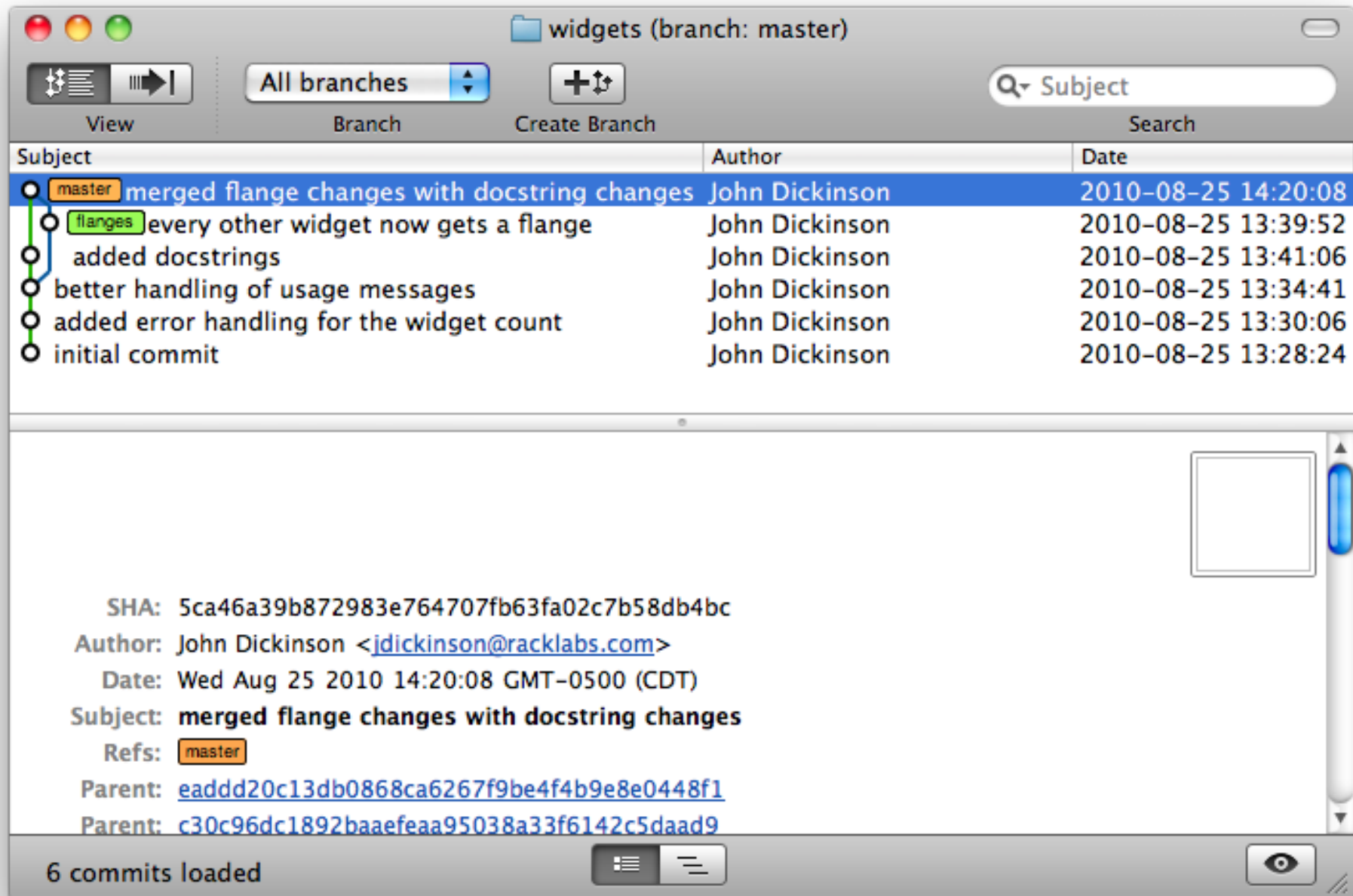
SHA: eadd20c13db0868ca6267f9be4f4b9e8e0448f1
Author: John Dickinson <jdickinson@racklabs.com>
Date: Wed Aug 25 2010 13:41:06 GMT-0500 (CDT)
Subject: **added docstrings**
Refs: **master**

5 commits loaded

Workflow Examples

- Merge flanges
 - Oh no! Conflicts!
 - git commit

Workflow Examples



widgets (branch: master)

View Branch Create Branch Search Subject

Subject	Author	Date
merged flange changes with docstring changes	John Dickinson	2010-08-25 14:20:08
every other widget now gets a flange	John Dickinson	2010-08-25 13:39:52
added docstrings	John Dickinson	2010-08-25 13:41:06
better handling of usage messages	John Dickinson	2010-08-25 13:34:41
added error handling for the widget count	John Dickinson	2010-08-25 13:30:06
initial commit	John Dickinson	2010-08-25 13:28:24

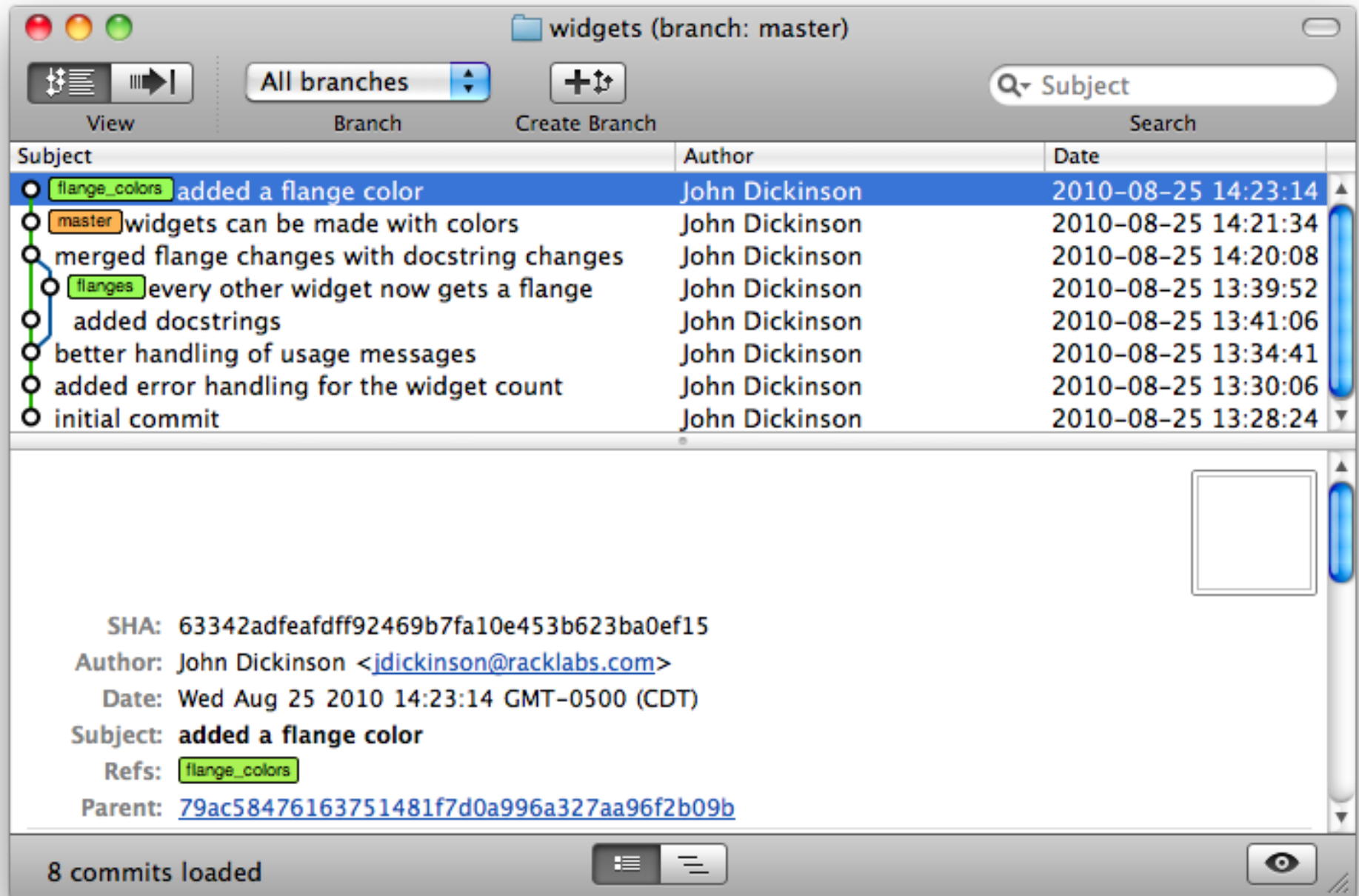
SHA: 5ca46a39b872983e764707fb63fa02c7b58db4bc
Author: John Dickinson <jdickinson@racklabs.com>
Date: Wed Aug 25 2010 14:20:08 GMT-0500 (CDT)
Subject: **merged flange changes with docstring changes**
Refs: **master**
Parent: [eadd20c13db0868ca6267f9be4f4b9e8e0448f1](#)
Parent: [c30c96dc1892baafeaa95038a33f6142c5daad9](#)

6 commits loaded

Workflow Examples

- New feature: Colored flanges
 - `git checkout -b flange_colors`
- Simple merge
 - `git checkout master`
 - `git merge flange_colors`

Workflow Examples



widgets (branch: master)

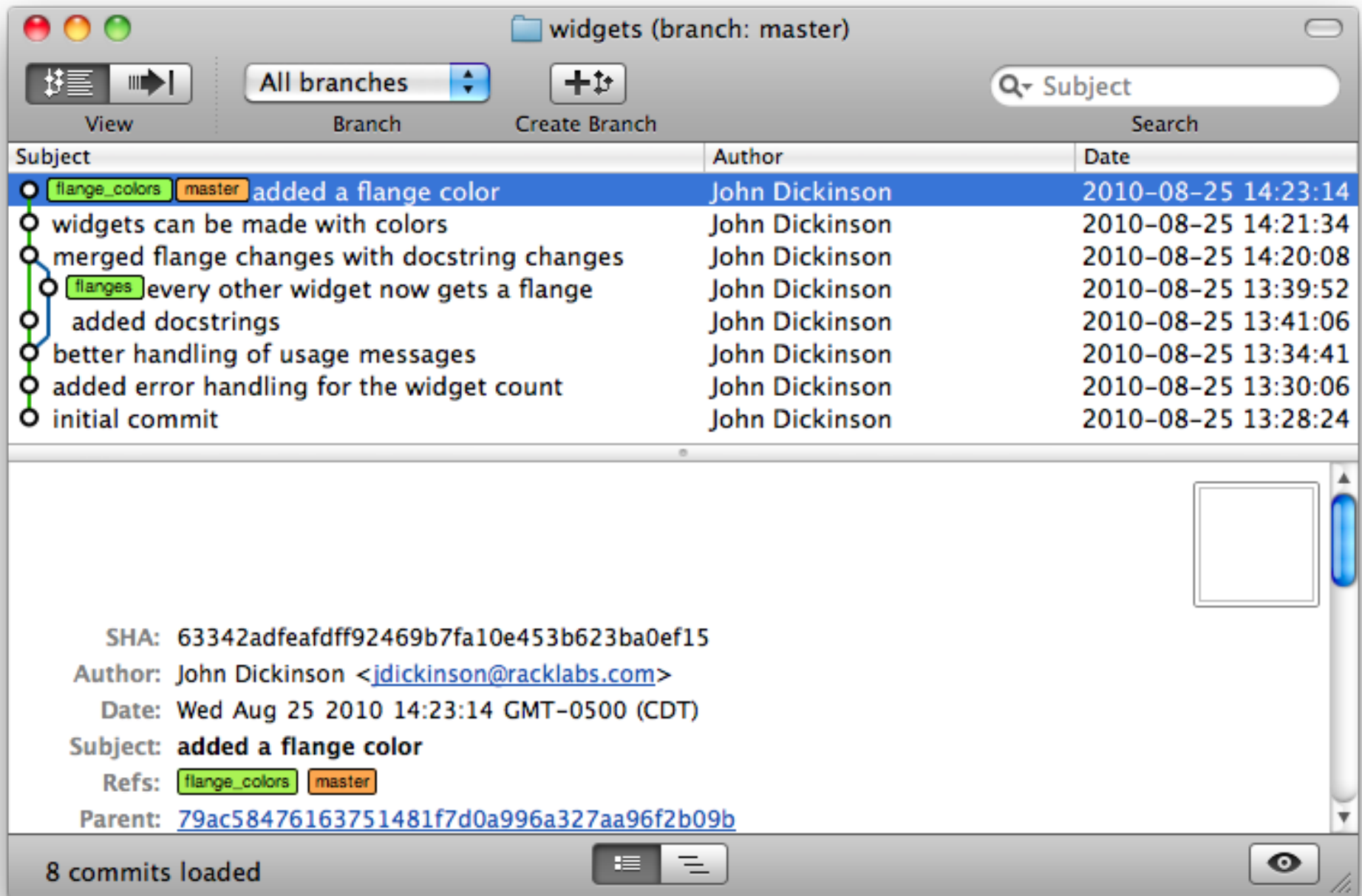
View Branch Create Branch Search Subject

Subject	Author	Date
flange_colors added a flange color	John Dickinson	2010-08-25 14:23:14
master widgets can be made with colors	John Dickinson	2010-08-25 14:21:34
merged flange changes with docstring changes	John Dickinson	2010-08-25 14:20:08
flanges every other widget now gets a flange	John Dickinson	2010-08-25 13:39:52
added docstrings	John Dickinson	2010-08-25 13:41:06
better handling of usage messages	John Dickinson	2010-08-25 13:34:41
added error handling for the widget count	John Dickinson	2010-08-25 13:30:06
initial commit	John Dickinson	2010-08-25 13:28:24

SHA: 63342adfeafdf92469b7fa10e453b623ba0ef15
Author: John Dickinson <jdickinson@racklabs.com>
Date: Wed Aug 25 2010 14:23:14 GMT-0500 (CDT)
Subject: **added a flange color**
Refs: **flange_colors**
Parent: [79ac58476163751481f7d0a996a327aa96f2b09b](#)

8 commits loaded

Workflow Examples



widgets (branch: master)

View Branch Create Branch Search

Subject	Author	Date
flange_colors master added a flange color	John Dickinson	2010-08-25 14:23:14
widgets can be made with colors	John Dickinson	2010-08-25 14:21:34
merged flange changes with docstring changes	John Dickinson	2010-08-25 14:20:08
flanges every other widget now gets a flange	John Dickinson	2010-08-25 13:39:52
added docstrings	John Dickinson	2010-08-25 13:41:06
better handling of usage messages	John Dickinson	2010-08-25 13:34:41
added error handling for the widget count	John Dickinson	2010-08-25 13:30:06
initial commit	John Dickinson	2010-08-25 13:28:24

SHA: 63342adfeafdf92469b7fa10e453b623ba0ef15
Author: John Dickinson <jdickinson@racklabs.com>
Date: Wed Aug 25 2010 14:23:14 GMT-0500 (CDT)
Subject: **added a flange color**
Refs: **flange_colors** **master**
Parent: [79ac58476163751481f7d0a996a327aa96f2b09b](#)

8 commits loaded

Workflow Examples

- Small dev team
 - something new to work on!
 - make new topic branch
 - work, commit, work, commit
 - pull and merge from other devs
 - (maybe) squash commits
 - code review
 - push to canonical

Workflow Examples

- New feature (color limitations)
 - `git checkout -b color_limits`
 - work, commit, work, commit

Workflow Examples

The screenshot shows a Git GUI window titled "widgets (branch: color_limits)". The interface includes a toolbar with "View", "Branch", and "Create Branch" buttons, and a search bar labeled "Subject". The main area displays a list of commits with columns for "Subject", "Author", and "Date". The selected commit is "set up widget and flange color pairings" by John Dickinson, dated 2010-08-25 14:34:13. Below the list, the commit details are shown, including the SHA hash, author information, date, subject, refs, and parent commit.

Subject	Author	Date
color_limits set up widget and flange color pairings	John Dickinson	2010-08-25 14:34:13
added color limitations	John Dickinson	2010-08-25 14:30:31
flange_colors master added a flange color	John Dickinson	2010-08-25 14:23:14
widgets can be made with colors	John Dickinson	2010-08-25 14:21:34
merged flange changes with docstring changes	John Dickinson	2010-08-25 14:20:08
flanges every other widget now gets a flange	John Dickinson	2010-08-25 13:39:52
added docstrings	John Dickinson	2010-08-25 13:41:06
better handling of usage messages	John Dickinson	2010-08-25 13:34:41
added error handling for the widget count	John Dickinson	2010-08-25 13:20:06

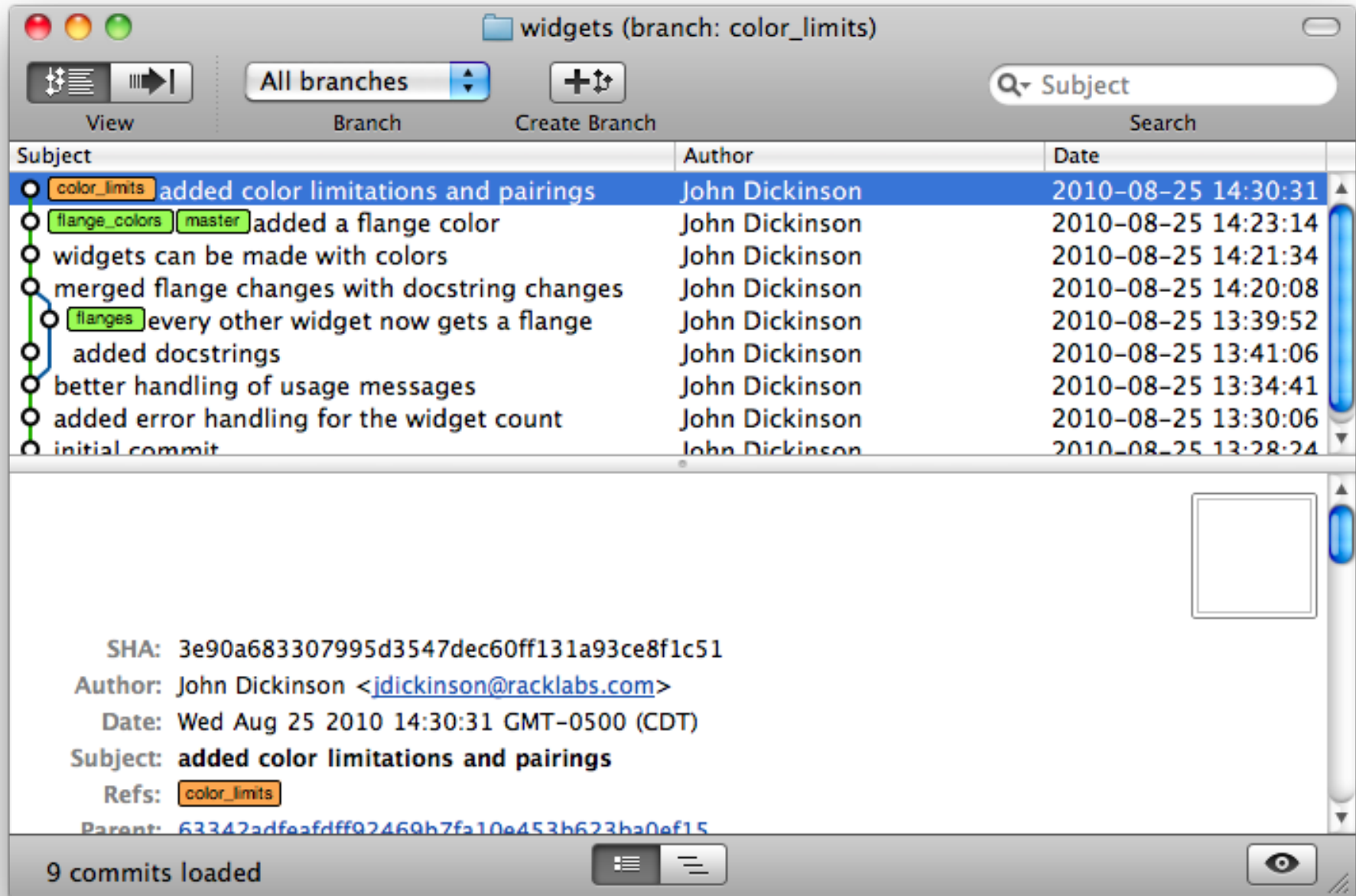
SHA: afd2e8b078a38727ceeb692829b7a175c0339cfd
Author: John Dickinson <jdickinson@racklabs.com>
Date: Wed Aug 25 2010 14:34:13 GMT-0500 (CDT)
Subject: **set up widget and flange color pairings**
Refs: color_limits
Parent: [4f07f174b2955fecef2dbffc0e11d0cee254b21b](#)

10 commits loaded

Workflow Examples

- Time to merge!
 - `git rebase -i master`
- Work on master continues
 - rebase again
 - merge after rebase is simple

Workflow Examples



The screenshot shows a Git GUI window titled "widgets (branch: color_limits)". The interface includes a "View" button, a "Branch" dropdown set to "All branches", a "Create Branch" button, and a search field labeled "Subject". The commit history is displayed in a table with columns for "Subject", "Author", and "Date". The top commit is selected and highlighted in blue. Below the table, the commit details for the selected commit are shown, including the SHA, author, date, subject, refs, and parent.

Subject	Author	Date
color_limits added color limitations and pairings	John Dickinson	2010-08-25 14:30:31
flange_colors master added a flange color	John Dickinson	2010-08-25 14:23:14
widgets can be made with colors	John Dickinson	2010-08-25 14:21:34
merged flange changes with docstring changes	John Dickinson	2010-08-25 14:20:08
flanges every other widget now gets a flange	John Dickinson	2010-08-25 13:39:52
added docstrings	John Dickinson	2010-08-25 13:41:06
better handling of usage messages	John Dickinson	2010-08-25 13:34:41
added error handling for the widget count	John Dickinson	2010-08-25 13:30:06
initial commit	John Dickinson	2010-08-25 13:28:24

SHA: 3e90a683307995d3547dec60ff131a93ce8f1c51
Author: John Dickinson <jdickinson@racklabs.com>
Date: Wed Aug 25 2010 14:30:31 GMT-0500 (CDT)
Subject: **added color limitations and pairings**
Refs: color_limits
Parent: 63342adfaefdf92469b7fa10e453b623ba0ef15

9 commits loaded

Workflow Examples

widgets (branch: master)

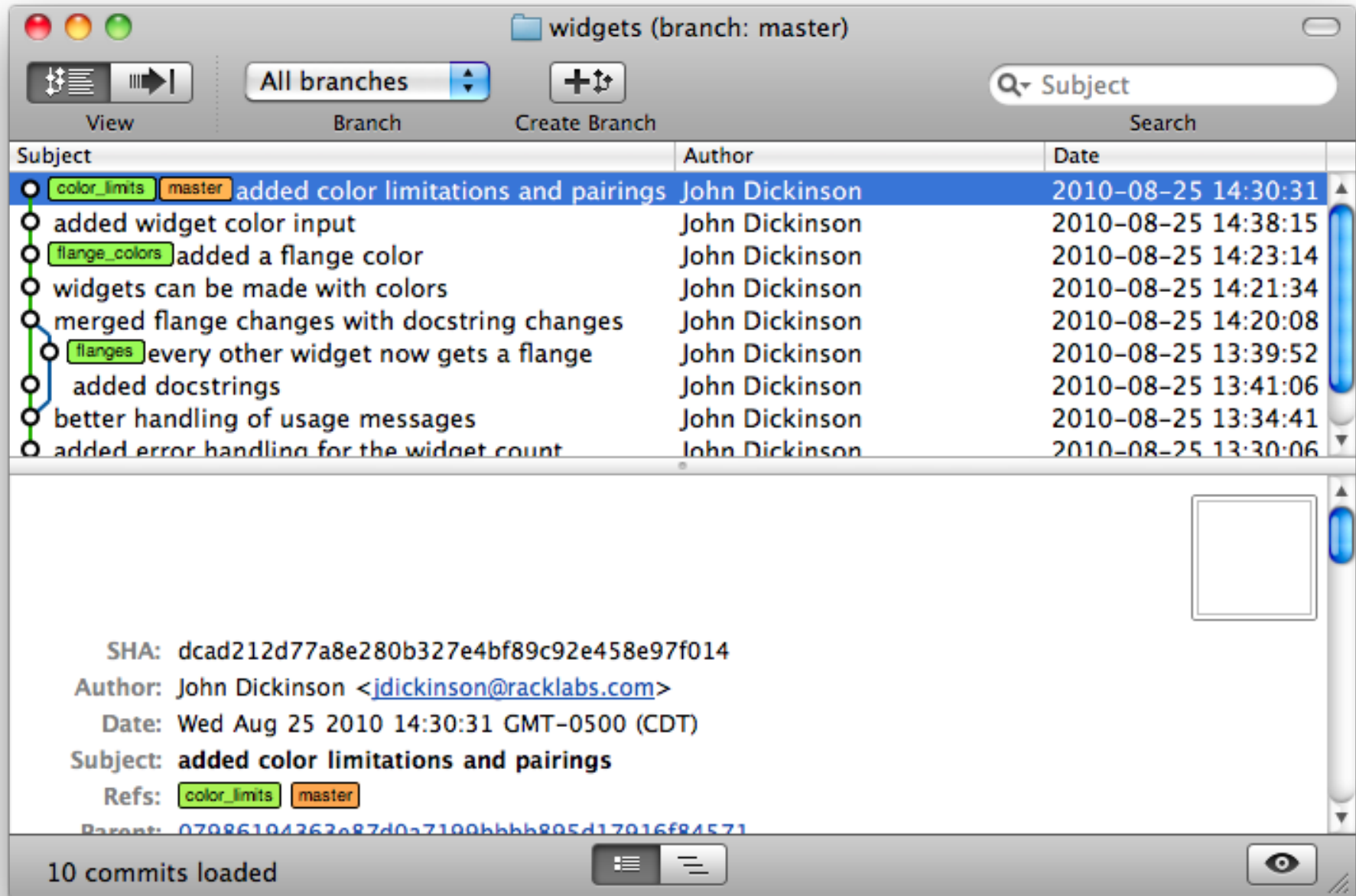
View Branch Create Branch Search Subject

Subject	Author	Date
master added widget color input	John Dickinson	2010-08-25 14:38:15
color_limits added color limitations and pairings	John Dickinson	2010-08-25 14:30:31
flange_colors added a flange color	John Dickinson	2010-08-25 14:23:14
widgets can be made with colors	John Dickinson	2010-08-25 14:21:34
merged flange changes with docstring changes	John Dickinson	2010-08-25 14:20:08
flanges every other widget now gets a flange	John Dickinson	2010-08-25 13:39:52
added docstrings	John Dickinson	2010-08-25 13:41:06
better handling of usage messages	John Dickinson	2010-08-25 13:34:41
added error handling for the widget count	John Dickinson	2010-08-25 13:30:06

SHA: 07986194363e87d0a7199bbbb895d17916f84571
Author: John Dickinson <jdickinson@racklabs.com>
Date: Wed Aug 25 2010 14:38:15 GMT-0500 (CDT)
Subject: **added widget color input**
Refs: **master**
Parent: 62242adfa9dfff02469b7fa10e452b622ba0ef15

10 commits loaded

Workflow Examples



widgets (branch: master)

All branches Create Branch Search Subject

Subject	Author	Date
color_limits master added color limitations and pairings	John Dickinson	2010-08-25 14:30:31
added widget color input	John Dickinson	2010-08-25 14:38:15
flange_colors added a flange color	John Dickinson	2010-08-25 14:23:14
widgets can be made with colors	John Dickinson	2010-08-25 14:21:34
merged flange changes with docstring changes	John Dickinson	2010-08-25 14:20:08
flanges every other widget now gets a flange	John Dickinson	2010-08-25 13:39:52
added docstrings	John Dickinson	2010-08-25 13:41:06
better handling of usage messages	John Dickinson	2010-08-25 13:34:41
added error handling for the widget count	John Dickinson	2010-08-25 13:30:06

SHA: dcad212d77a8e280b327e4bf89c92e458e97f014
Author: John Dickinson <jdickinson@racklabs.com>
Date: Wed Aug 25 2010 14:30:31 GMT-0500 (CDT)
Subject: **added color limitations and pairings**
Refs: **color_limits** **master**
Parent: 07086104262e87d0a7100bbb805d17016f84571

10 commits loaded

Workflow Examples

- Large project
 - something new to work on!
 - make a topic branch
 - work, commit, work, commit
 - merge as needed, tests, code review, etc
 - squash and send patch to gatekeeper
 - gatekeeper approves or rejects
 - gatekeeper pushes patch to canonical

Example

- Cloud Files process
- OpenStack swift process

Questions