An intro to GIT

Orlando Code Camp 2017

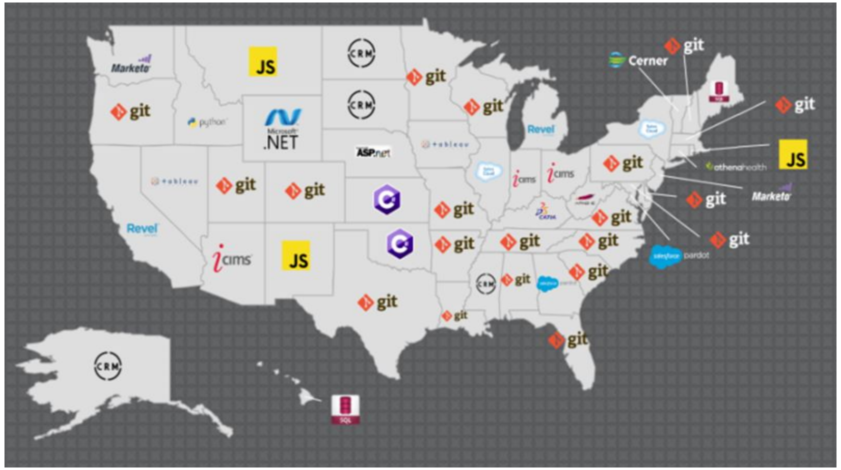
Misnomers about GIT?

- It's too hard to use.
- It's only command line based.
- My project is too big and complicated for that.
- Takes too long to use.

THIS IS GIT. IT TRACKS COLLABORATIVE WORK ON PROJECTS THROUGH A BEAUTIFUL DISTRIBUTED GRAPH THEORY TREE MODEL.
COOL. HOW DO WE USE IT?
NO IDEA. JUST MEMORIZE THESE SHELL COMMANDS AND TYPE THEM TO SYNC UP. IF YOU GET ERRORS, SAVE YOUR WORK ELSEWHERE, DELETE THE PROJECT, AND DOWNLOAD A FRESH COPY.

Why use this tool?

- Multiple backups fast
- Work in parallel on the same file
- Develop features in different 'versions' or branches
- Fast rollback and feature switching
- It's a skill in <u>high</u> demand



My Path to Version Control

- Lots of information.
- What's the 'best'?
- How do I do 'X'?
- GIT, SVN, etc?
- What do I want out of version control?

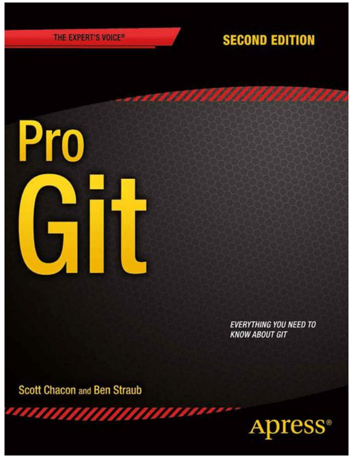
Quick answers to common problems

Git Version Control Cookbook

90 hands-on recipes that will increase your productivity when using Git as a version control system

Aske Olsson
Rasmus Voss

[PACKT] open source


THE EXPERT'S VOICE®

SECOND EDITION

Pro Git

EVERYTHING YOU NEED TO KNOW ABOUT GIT

Scott Chacon and Ben Straub

Apress®

GIT Words

- add
- am
- apply
- archive
- bisect
- blame
- branch
- bundle
- cat-file
- checkout
- cherry-pick
- clean
- clone
- commit
- commit-tree
- config
- count-objects
- daemon
- describe
- diff
- diff-index
- fast-import
- fetch
- filter-branch
- for-each-ref
- format-patch
- fsck
- gc
- gitk
- grep
- hash-object
- help
- init
- instaweb
- log
- ls-files
- ls-tree
- merge
- merge-base
- mergetool
- mv
- pop
- prune
- pull
- push
- read-tree
- rebase
- reflog
- remote
- request-pull
- reset
- revert
- rev-list
- rev-parse
- rm
- send-email
- shortlog
- show
- show-ref
- stage
- stash
- status
- submodule
- svn
- symbolic-ref
- tag
- update-index
- update-ref
- update-server-info
- verify-pack
- write-tree

GIT WORDS

add	**commit**	fsck	mergetool	rev-list	tag
am	commit-tree	gc	mv	rev-parse	update-index
apply	config	gitk	pop	rm	update-ref
archive	count-objects	grep	prune	send-email	update-server-info
bisect	daemon	hash-object	pull	shortlog	verify-pack
blame	describe	help	**push**	show	write-tree
branch	diff	**init**	read-tree	show-ref	
bundle	diff-index	instaweb	rebase	**stage**	
cat-file	fast-import	log	reflog	stash	
checkout	**fetch**	ls-files	**remote**	status	
cherry-pick	filter-branch	ls-tree	request-pull	submodule	
clean	for-each-ref	**merge**	**reset**	svn	
clone	format-patch	merge-base	revert	symbolic-ref	

Tool that we'll use

Atlassian

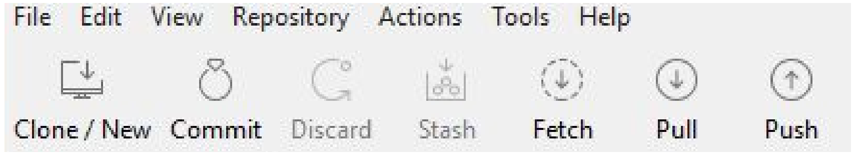
SourceTree

Let's Start !

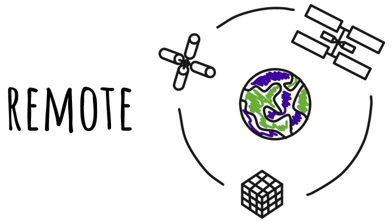
Clone/ Init

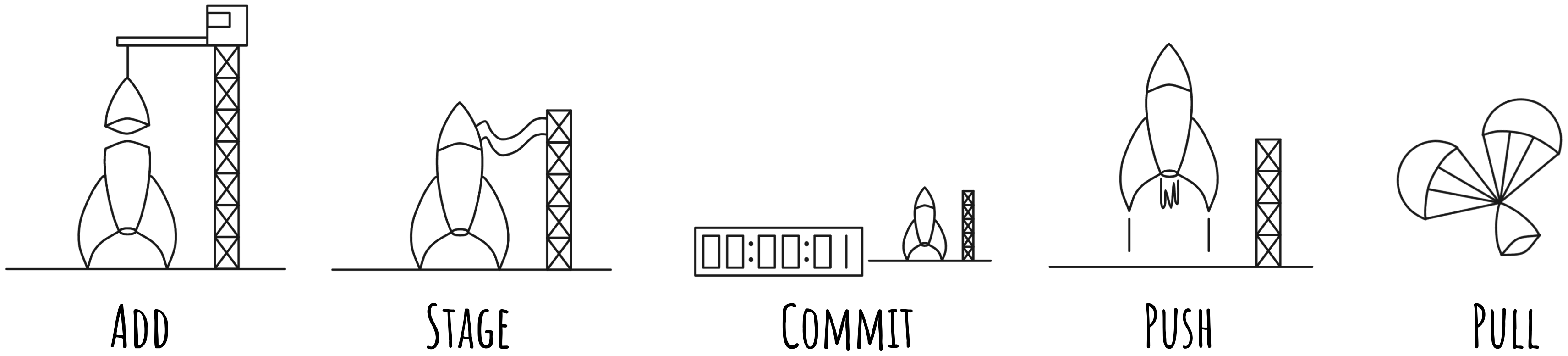
Creates a new repo or copies an existing one

You can always add ‘remotes’ but it's easier to start from there.



Typical Workflow

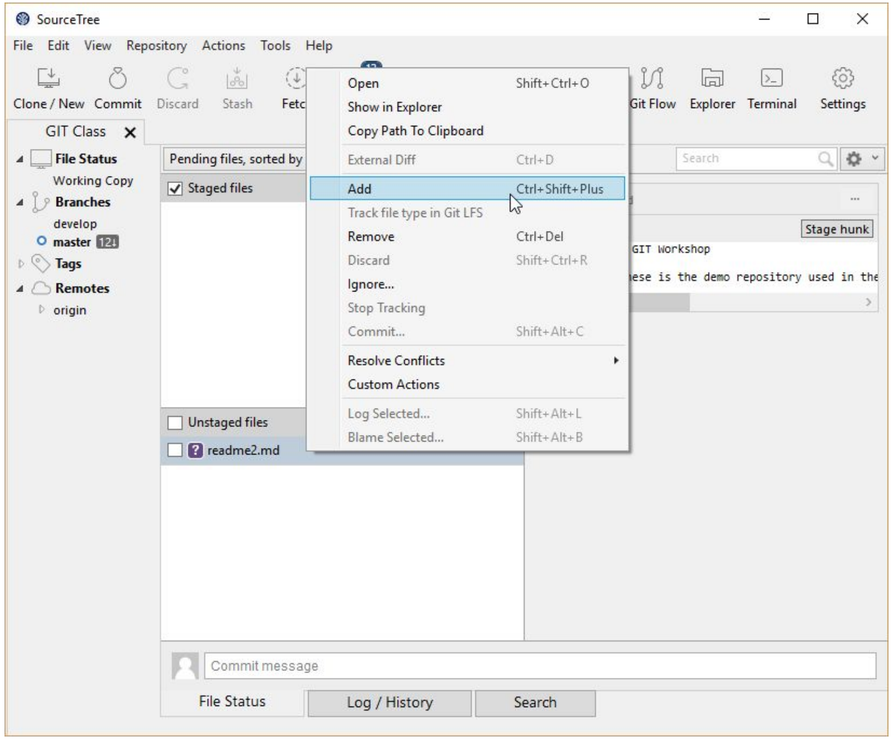




ADD

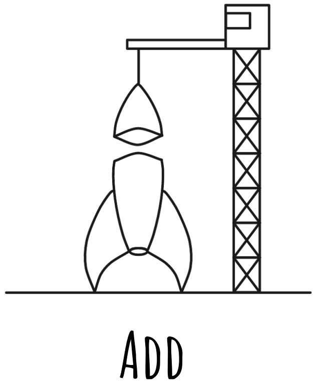
Tells GIT to start tracking the files.

Ignore allows you to not track files in a folder - compiled files or temp files typically.



GIT Workshop

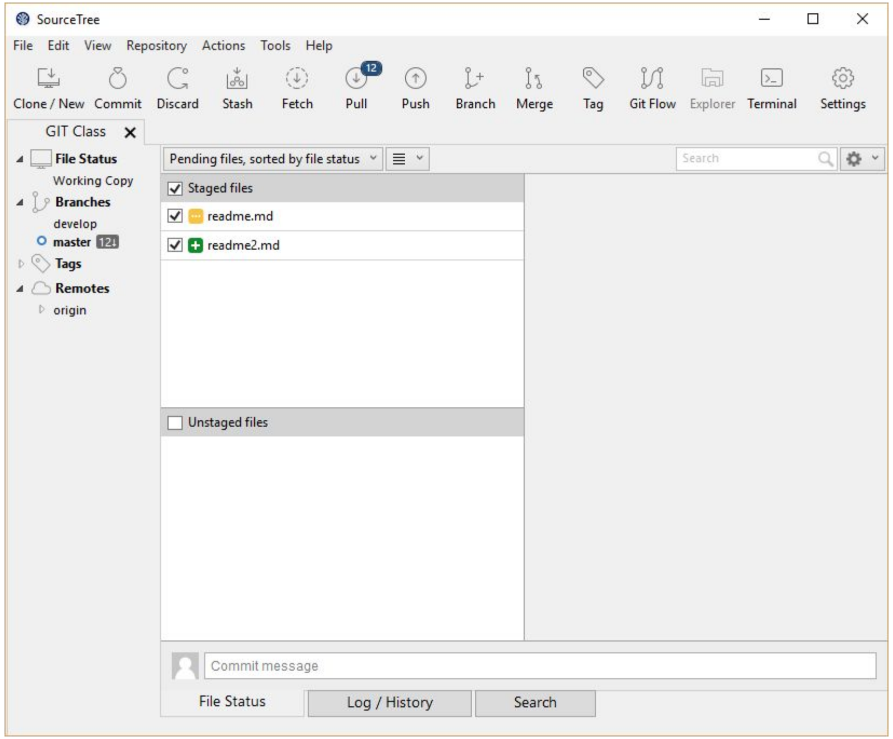
ese is the demo repository used in the

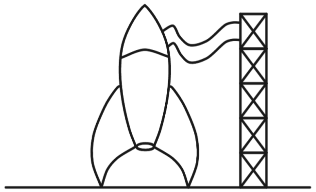


STAGE

Lets you get your commit where you want it.

TIP:
You can split your changes into multiple commits for tracking.


SourceTree
File Edit View Repository Actions Tools Help
Clone / New Commit Discard Stash Fetch Pull 12 Push Branch Merge Tag Git Flow Explorer Terminal Settings
GIT Class
File Status
Working Copy
Branches
develop
master 12↓
Tags
Remotes
origin
Pending files, sorted by file status
Search
Staged files
readme.md
readme2.md
Unstaged files
Commit message
File Status
Log / History
Search

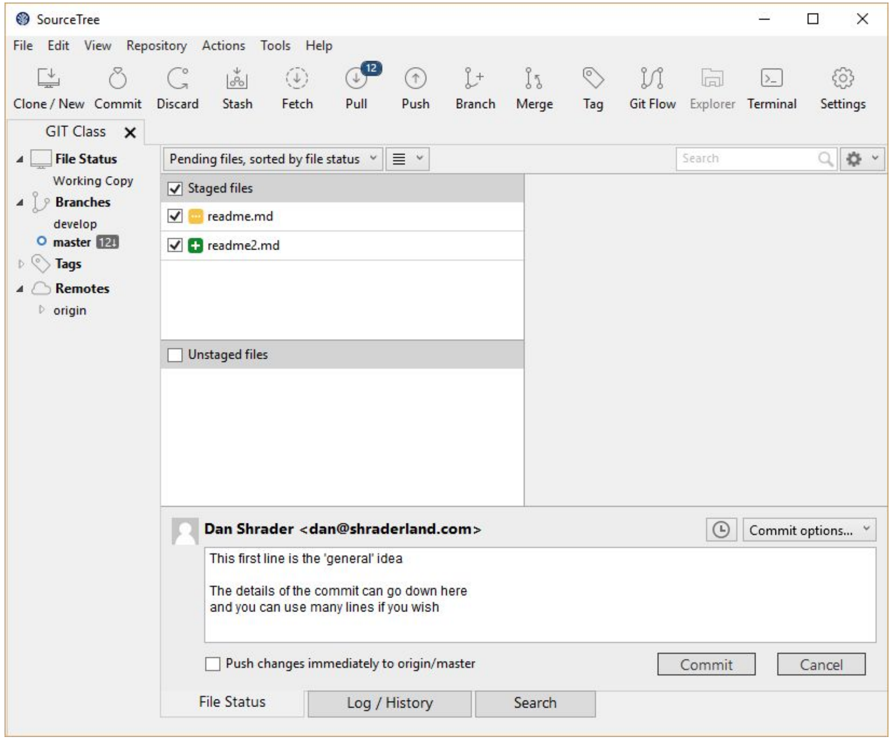


STAGE

COMMIT

Snapshot in time of your project.

Be detailed on your descriptions. You'll thank yourself later and so will the other contributors to the project.

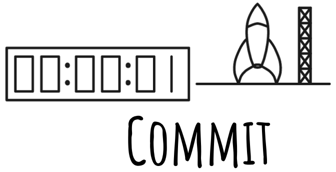


Dan Shrader <dan@shraderland.com>

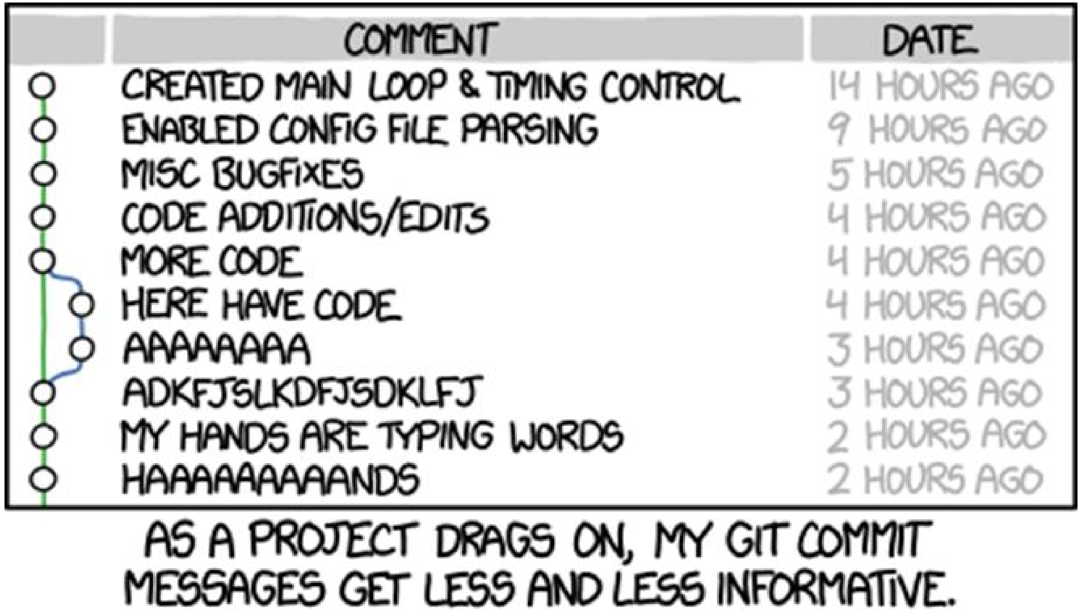
This first line is the 'general' idea

The details of the commit can go down here
and you can use many lines if you wish

☐ Push changes immediately to origin/master



COMMIT



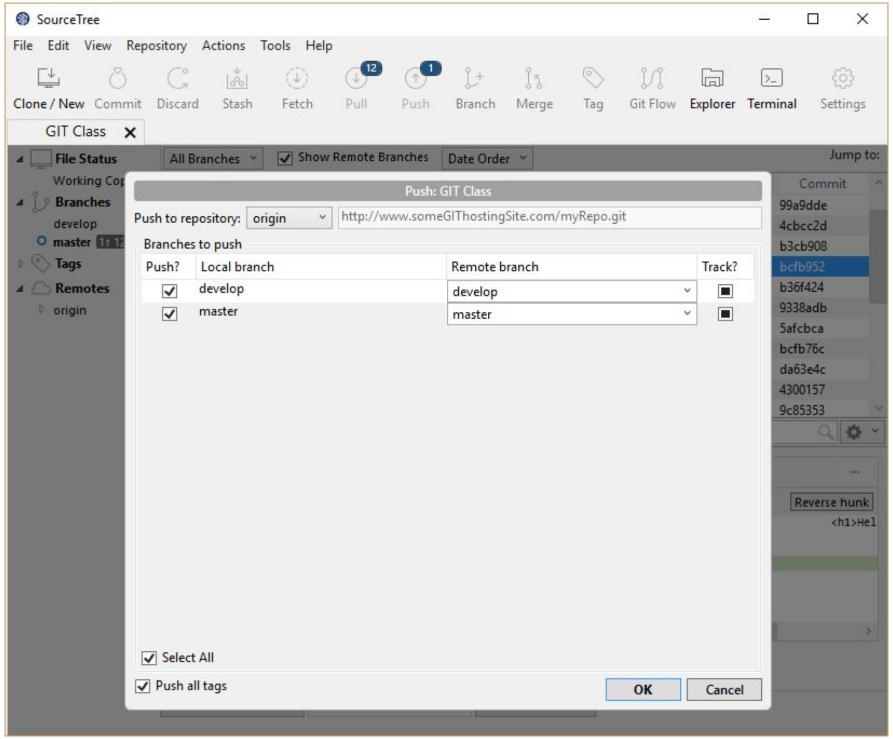
	COMMENT	DATE
○	CREATED MAIN LOOP & TIMING CONTROL	14 HOURS AGO
○	ENABLED CONFIG FILE PARSING	9 HOURS AGO
○	MISC BUGFIXES	5 HOURS AGO
○	CODE ADDITIONS/EDITS	4 HOURS AGO
○	MORE CODE	4 HOURS AGO
○	HERE HAVE CODE	4 HOURS AGO
○	AAAAAAAA	3 HOURS AGO
○	ADKFJSLKDFJSDKLFJ	3 HOURS AGO
○	MY HANDS ARE TYPING WORDS	2 HOURS AGO
○	HAAAAAAAANDS	2 HOURS AGO

AS A PROJECT DRAGS ON, MY GIT COMMIT MESSAGES GET LESS AND LESS INFORMATIVE.

PUSH

Sends your commits to a centralized server.

Note: there are different types of workflows. We are referencing the 'centralized' workflow.



Push: GIT Class

Push to repository: origin http://www.someGIThostingSite.com/myRepo.git

Branches to push

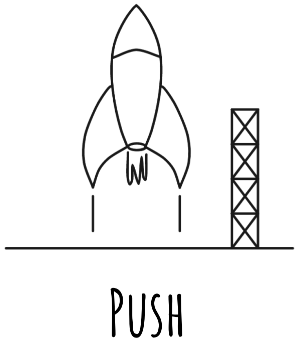
Push?	Local branch	Remote branch	Track?
✓	develop	develop	■
✓	master	master	■

✓ Select All

✓ Push all tags

OK Cancel

99a9dde
4cbcc2d
b3cb908
bcfb952
b36f424
9338adb
5afcbca
bcfb76c
da63e4c
4300157
9c85353



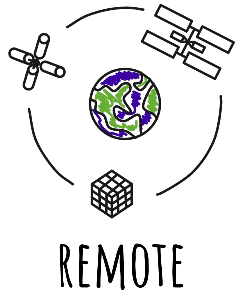
Remote

A place where your code is hosted.

Bitbucket

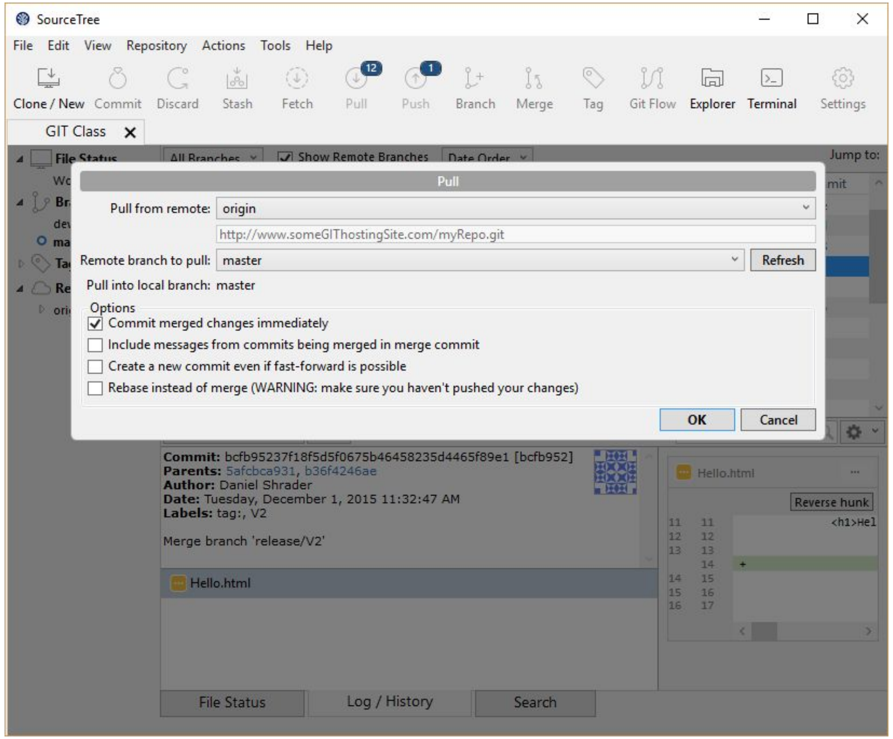
GitHub

GitLab



PULL

Brings changes committed to the remote into your branch.


SourceTree
File Edit View Repository Actions Tools Help
Clone / New Commit Discard Stash Fetch Pull Push Branch Merge Tag Git Flow Explorer Terminal Settings
GIT Class
File Status
All Branches
Show Remote Branches
Date Order
Jump to:
File Status Log / History Search

Pull

Pull from remote: origin

http://www.someGIThostingSite.com/myRepo.git

Remote branch to pull: master [Refresh]

Pull into local branch: master

Options

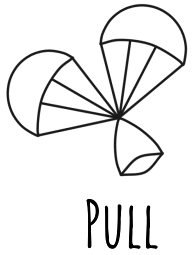
- [✓] Commit merged changes immediately
- [ ] Include messages from commits being merged in merge commit
- [ ] Create a new commit even if fast-forward is possible
- [ ] Rebase instead of merge (WARNING: make sure you haven't pushed your changes)

[OK] [Cancel]

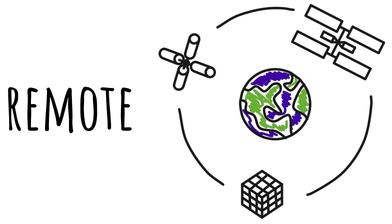
Commit: bcfb95237f18f5d5f0675b46458235d4465f89e1 [bcfb952]
Parents: 5afcbca931, b36f4246ae
Author: Daniel Shrader
Date: Tuesday, December 1, 2015 11:32:47 AM
Labels: tag:, V2

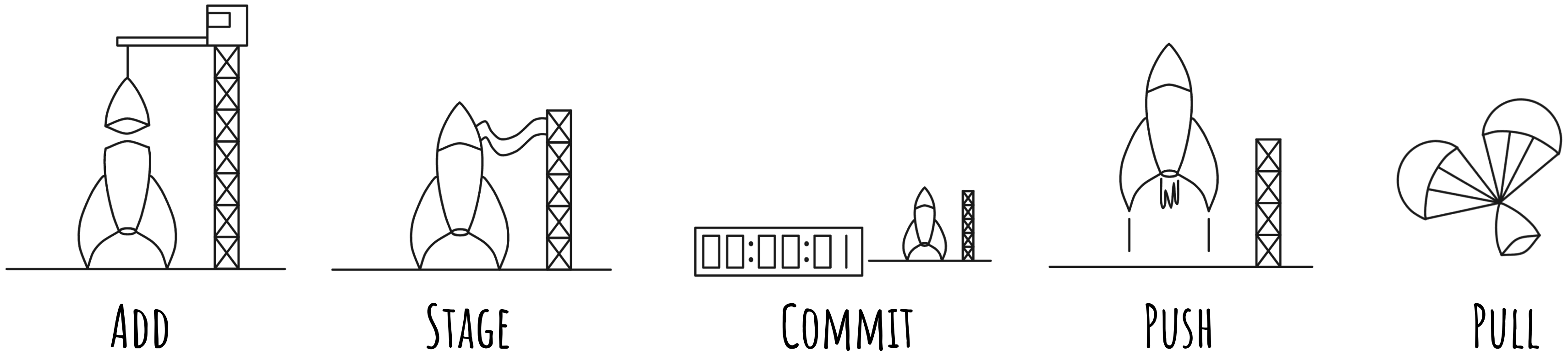
Merge branch 'release/V2'

Hello.html



Workflow Once More

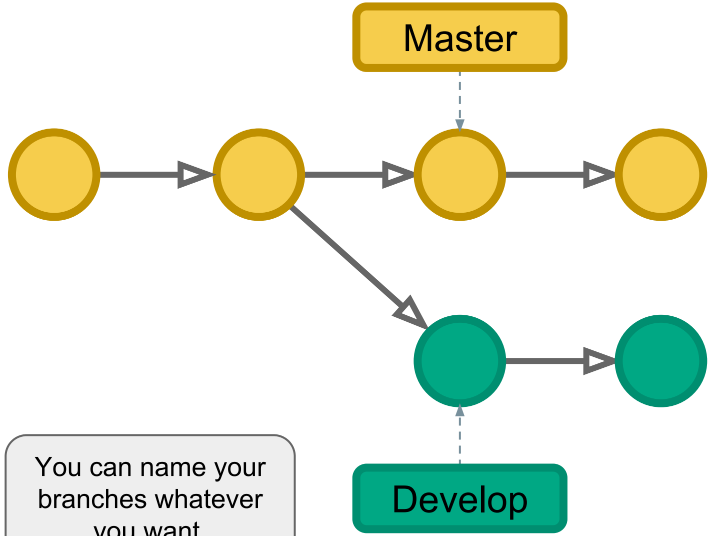




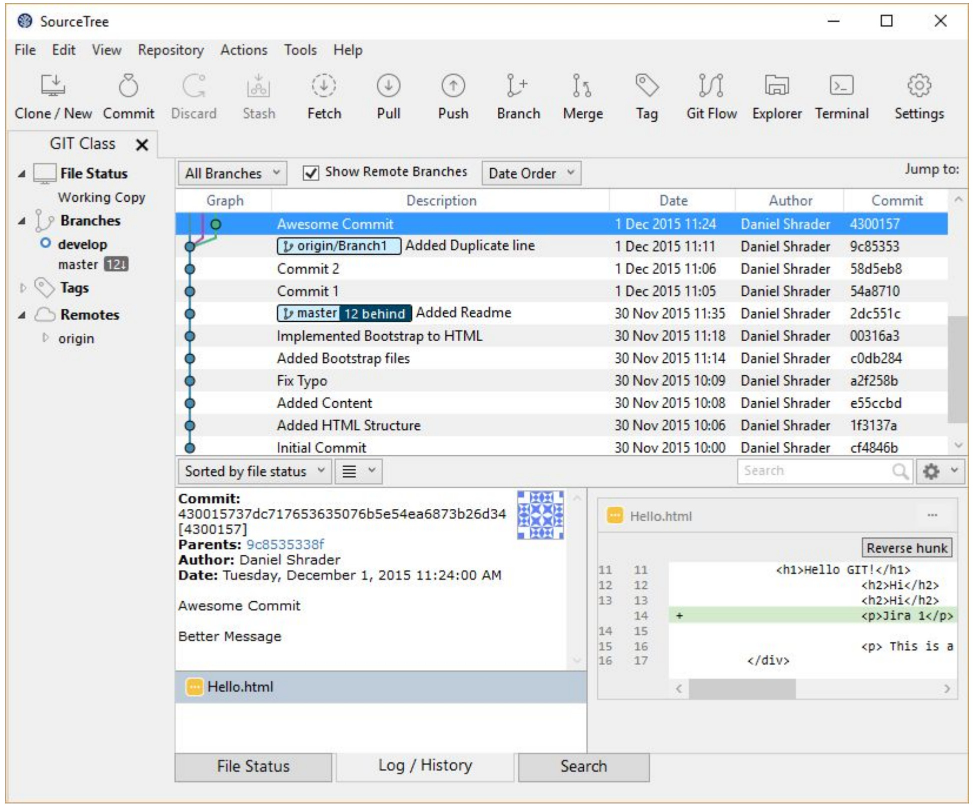
Branches!

BRANCH

A place to work which does not affect the main project.



You can name your branches whatever you want.



Description	Date	Author	Commit
Awesome Commit	1 Dec 2015 11:24	Daniel Shrader	4300157
origin/Branch1 Added Duplicate line	1 Dec 2015 11:11	Daniel Shrader	9c85353
Commit 2	1 Dec 2015 11:06	Daniel Shrader	58d5eb8
Commit 1	1 Dec 2015 11:05	Daniel Shrader	54a8710
master 12 behind Added Readme	30 Nov 2015 11:35	Daniel Shrader	2dc551c
Implemented Bootstrap to HTML	30 Nov 2015 11:18	Daniel Shrader	00316a3
Added Bootstrap files	30 Nov 2015 11:14	Daniel Shrader	c0db284
Fix Typo	30 Nov 2015 10:09	Daniel Shrader	a2f258b
Added Content	30 Nov 2015 10:08	Daniel Shrader	e55ccbd
Added HTML Structure	30 Nov 2015 10:06	Daniel Shrader	1f3137a
Initial Commit	30 Nov 2015 10:00	Daniel Shrader	cf4846b

Commit: 430015737dc717653635076b5e54ea6873b26d34 [4300157]
Parents: 9c8535338f
Author: Daniel Shrader
Date: Tuesday, December 1, 2015 11:24:00 AM

Awesome Commit

Better Message

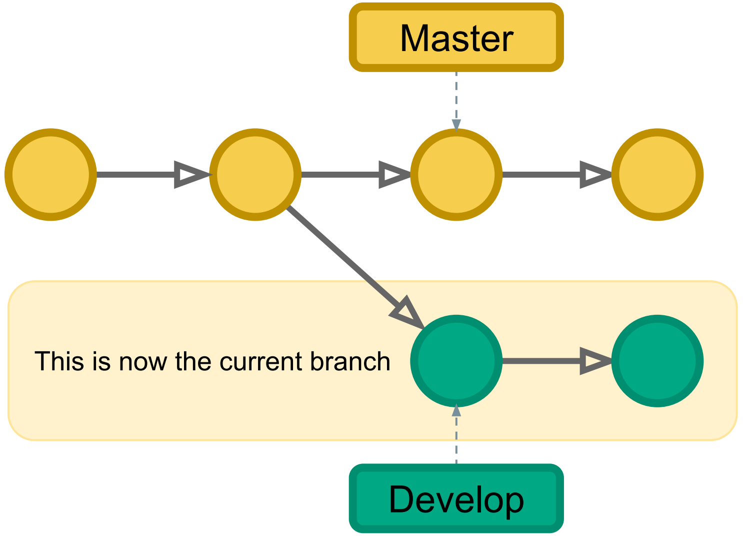
Hello.html

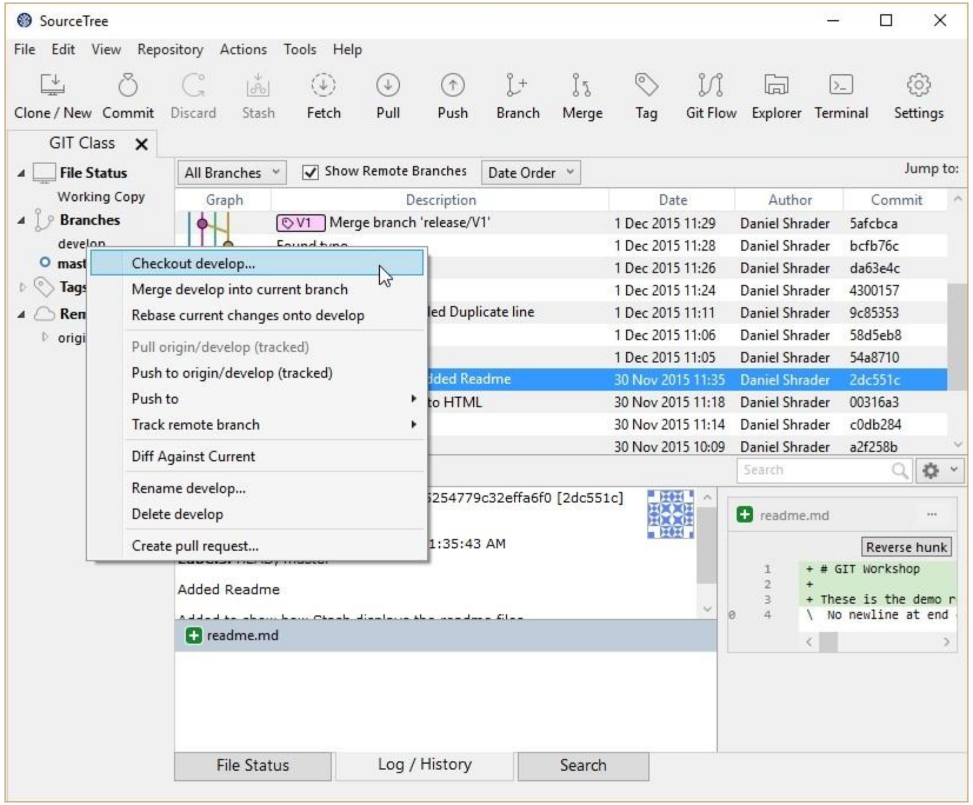
```
        <h1>Hello GIT!</h1>
            <h2>Hi</h2>
            <h2>Hi</h2>
+           <p>Jira 1</p>

            <p> This is a
    </div>
```

Checkout

Switches the branch you are on.





Graph	Description	Date	Author	Commit
	V1 Merge branch 'release/V1'	1 Dec 2015 11:29	Daniel Shrader	5afcbca
	Found typo	1 Dec 2015 11:28	Daniel Shrader	bcfb76c
		1 Dec 2015 11:26	Daniel Shrader	da63e4c
		1 Dec 2015 11:24	Daniel Shrader	4300157
	ded Duplicate line	1 Dec 2015 11:11	Daniel Shrader	9c85353
		1 Dec 2015 11:06	Daniel Shrader	58d5eb8
		1 Dec 2015 11:05	Daniel Shrader	54a8710
	dded Readme	30 Nov 2015 11:35	Daniel Shrader	2dc551c
	to HTML	30 Nov 2015 11:18	Daniel Shrader	00316a3
		30 Nov 2015 11:14	Daniel Shrader	c0db284
		30 Nov 2015 10:09	Daniel Shrader	a2f258b

5254779c32effa6f0 [2dc551c]

1:35:43 AM

Added Readme

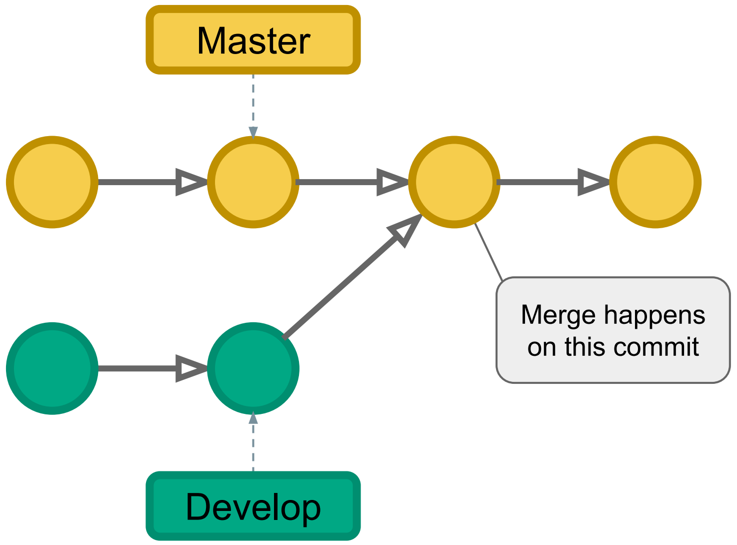
readme.md

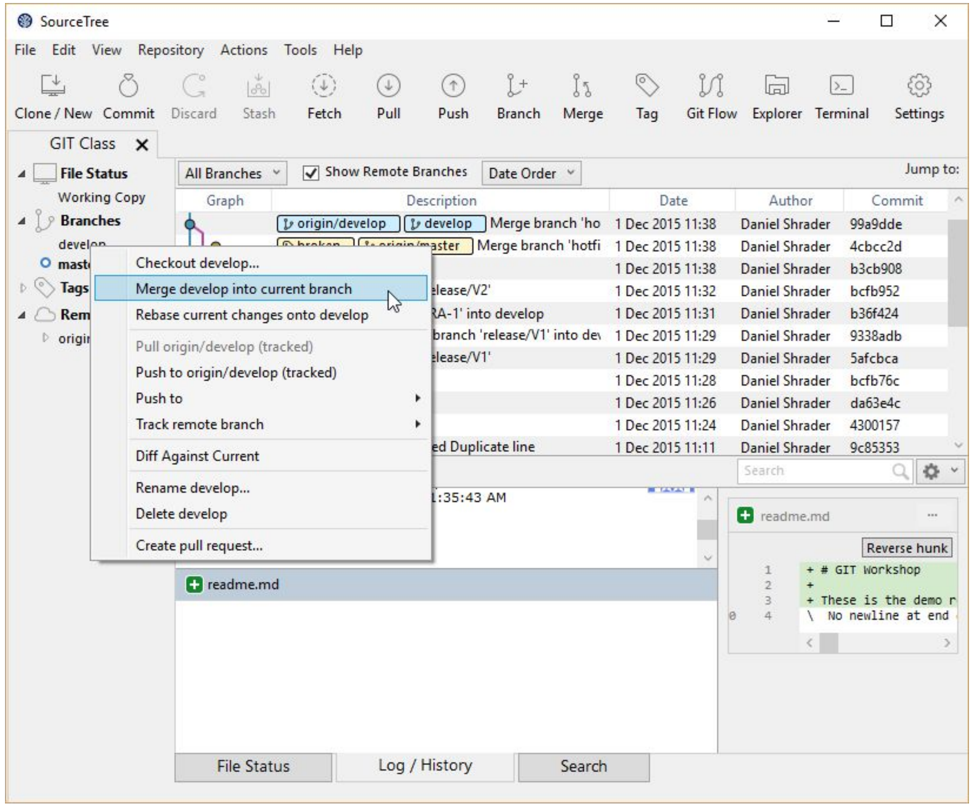
readme.md

```
+ # GIT Workshop
+
+ These is the demo r
\ No newline at end
```

Merge

Brings a branch into another. Be cautious of conflicts.





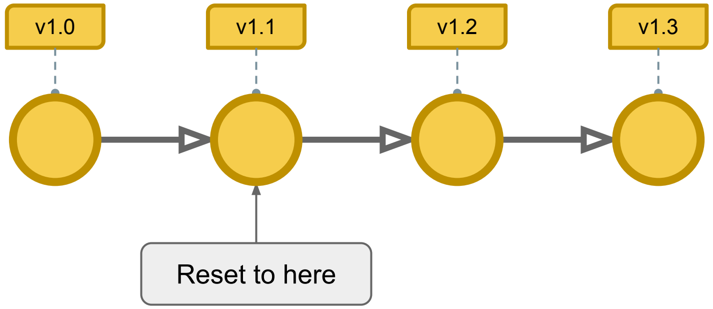
Graph	Description	Date	Author	Commit
	origin/develop develop Merge branch 'ho	1 Dec 2015 11:38	Daniel Shrader	99a9dde
	broken origin/master Merge branch 'hotfi	1 Dec 2015 11:38	Daniel Shrader	4cbcc2d
		1 Dec 2015 11:38	Daniel Shrader	b3cb908
	elease/V2'	1 Dec 2015 11:32	Daniel Shrader	bcfb952
	RA-1' into develop	1 Dec 2015 11:31	Daniel Shrader	b36f424
	branch 'release/V1' into dev	1 Dec 2015 11:29	Daniel Shrader	9338adb
	elease/V1'	1 Dec 2015 11:29	Daniel Shrader	5afcbca
		1 Dec 2015 11:28	Daniel Shrader	bcfb76c
		1 Dec 2015 11:26	Daniel Shrader	da63e4c
		1 Dec 2015 11:24	Daniel Shrader	4300157
	ed Duplicate line	1 Dec 2015 11:11	Daniel Shrader	9c85353

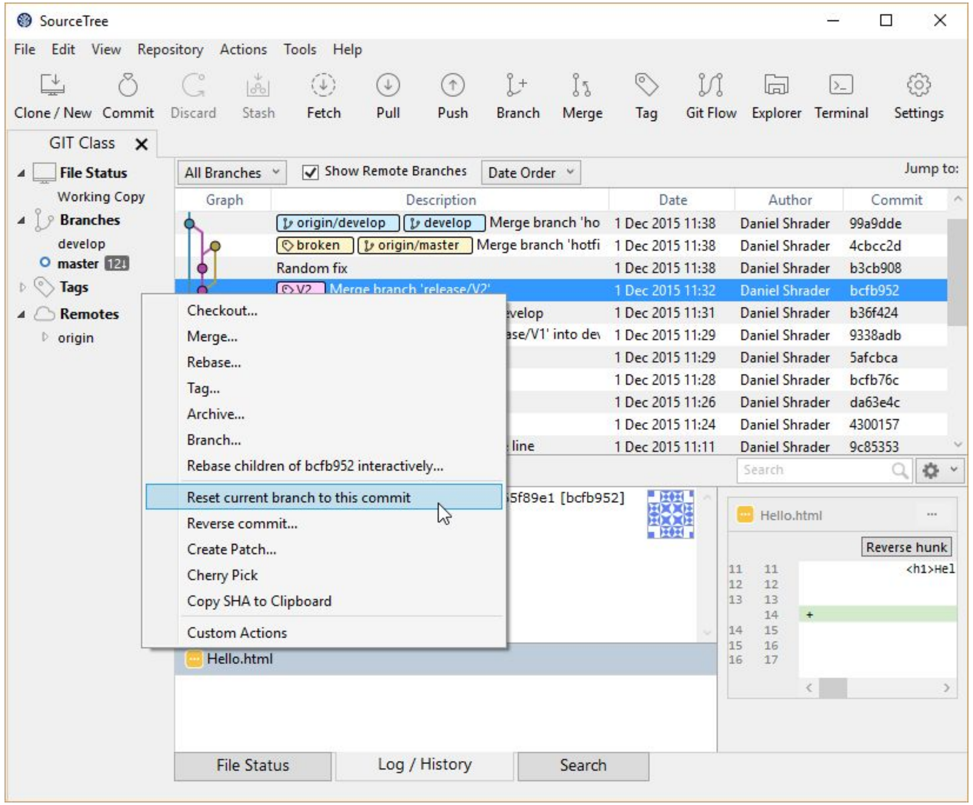
:35:43 AM

```
+ # GIT Workshop
+
+ These is the demo r
\  No newline at end
```

RESET

Lets you move around in time on your project. Really useful when looking for bug introductions.





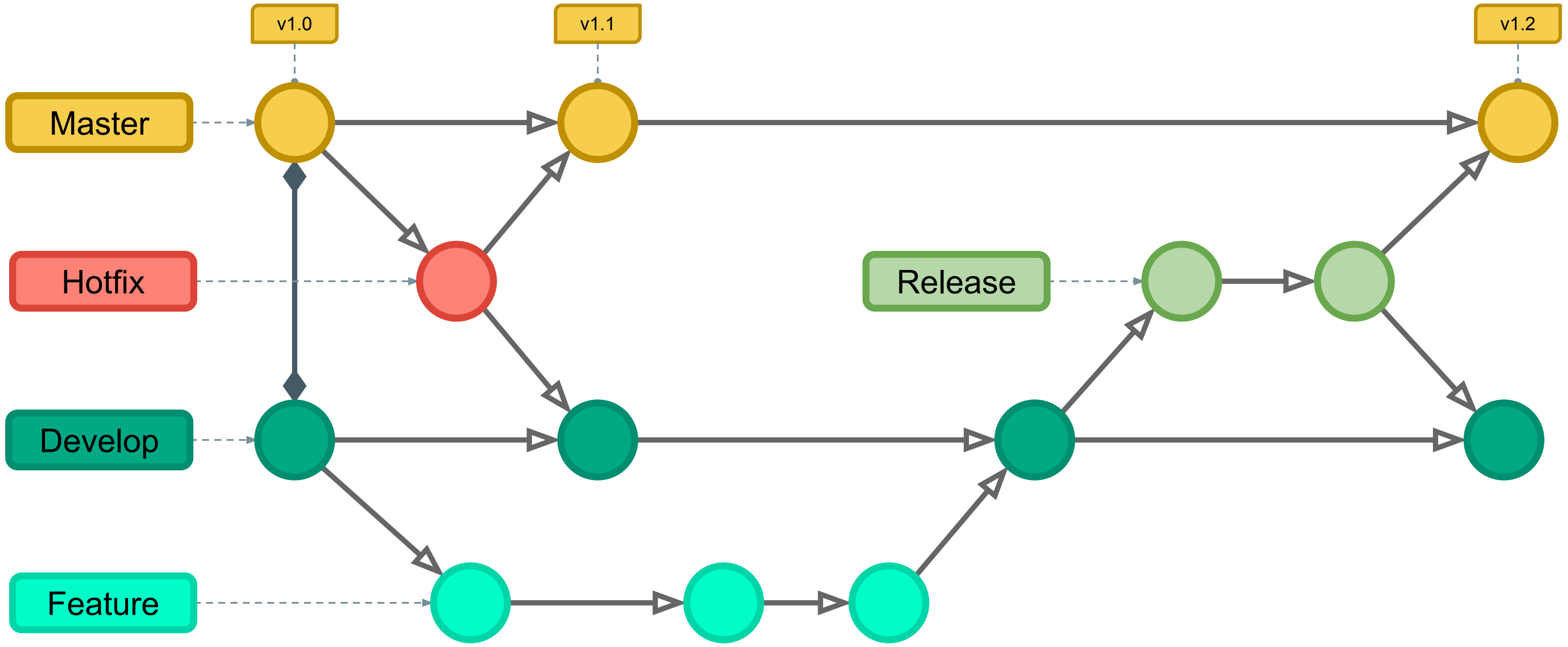
Graph	Description	Date	Author	Commit
	origin/develop develop Merge branch 'ho	1 Dec 2015 11:38	Daniel Shrader	99a9dde
	broken origin/master Merge branch 'hotfi	1 Dec 2015 11:38	Daniel Shrader	4cbcc2d
	Random fix	1 Dec 2015 11:38	Daniel Shrader	b3cb908
	V2 Merge branch 'release/V2'	1 Dec 2015 11:32	Daniel Shrader	bcfb952
	velop	1 Dec 2015 11:31	Daniel Shrader	b36f424
	ase/V1' into dev	1 Dec 2015 11:29	Daniel Shrader	9338adb
		1 Dec 2015 11:29	Daniel Shrader	5afcbca
		1 Dec 2015 11:28	Daniel Shrader	bcfb76c
		1 Dec 2015 11:26	Daniel Shrader	da63e4c
		1 Dec 2015 11:24	Daniel Shrader	4300157
	line	1 Dec 2015 11:11	Daniel Shrader	9c85353

- Checkout...
- Merge...
- Rebase...
- Tag...
- Archive...
- Branch...
- Rebase children of bcfb952 interactively...
- Reset current branch to this commit
- Reverse commit...
- Create Patch...
- Cherry Pick
- Copy SHA to Clipboard
- Custom Actions

5f89e1 [bcfb952]

```
11 11   <h1>Hel
12 12
13 13
   14  +
14 15
15 16
16 17
```

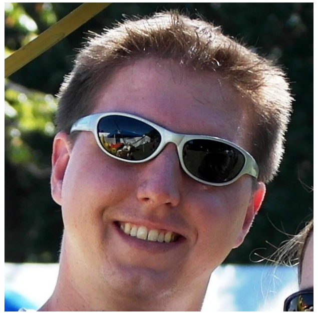
GIT FLOW WORKFLOW



Appendix

- https://www.atlassian.com
- http://www.mnu.edu/business/software-skills-demand
- https://xkcd.com/1296/
- https://xkcd.com/1597/
- http://danielkummer.github.io/git-flow-cheatsheet/
- https://training.github.com/classes/essentials/
- https://www.gitlab.com/
- https://github.com/
- https://www.sourcetreeapp.com/

About Me



I've been dabbling with code since the late 90's and currently work with Microsoft Business Intelligence products. When given the chance, I write single page applications in JavaScript. I'm also a stickler for automating as much of the day to day tasks as I can. When I'm not coding I enjoy camping and hanging out with my wife and sons.

Email: Dan@ShraderLand.com