

“Grok” Git

Fundamentals and Functionality

Madison Swain-Bowden



Purpose

- Radical approach to git's innards
- Workflow tips & tricks

Outline

- **Git in 30 seconds**
- **How the heck does git work**
- **Collaboration with git**
 - Commit messages
 - PRs
- **GitKraken showcase**
- **Github's CLI**

Version Control

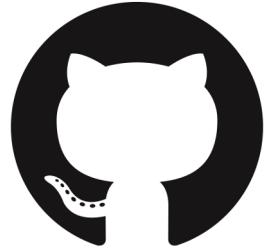
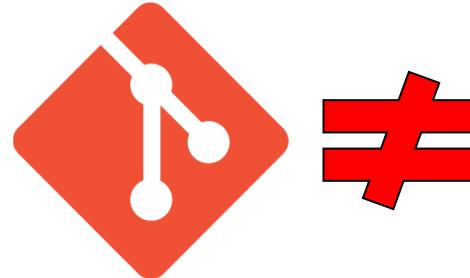
- System that tracks changes of files (generally text) over time
- Many different kinds of VC
 - New folders for each version
 - Zipped backups
 - Email backups
- Git - common VCS

| Name | Type |
|---|-------------|
| Code_version20102201 | File folder |
| Code_version20102201 - Copy | File folder |
| Code_version20102201_latest | File folder |
| Code_version20102201_latest_evening | File folder |
| Code_version20102202 | File folder |
| Code_version20102202_Jdon'trememberwhatIchanged | File folder |
| Code_version20102202_version2 | File folder |
| Code_version20102202_version2-after_version1 | File folder |
| Code_version20102202-2 | File folder |
| Code_version20102205 | File folder |
| Code_version20102206_during_dinner | File folder |
| Code_version20102207 | File folder |
| Code_version20102208_after_measurements | File folder |
| Code_version20102209 | File folder |
| Code_version2010220999 | File folder |
| Code_version20102209999 | File folder |
| Code_version201022099999 | File folder |
| Code_version2010220999999 | File folder |
| Code_version20102209999999 | File folder |

Git

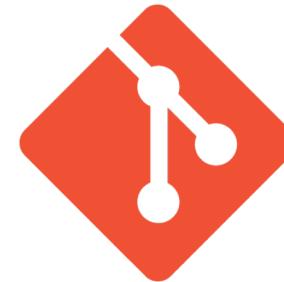
OOKLA®

- Version control software developed in 2005
- Captures “changes” across time
- Allows multiple developers to work on the same codebase simultaneously
- Not synonymous with Github!
 - Git is the software used locally
 - Github is a website/service used to host git



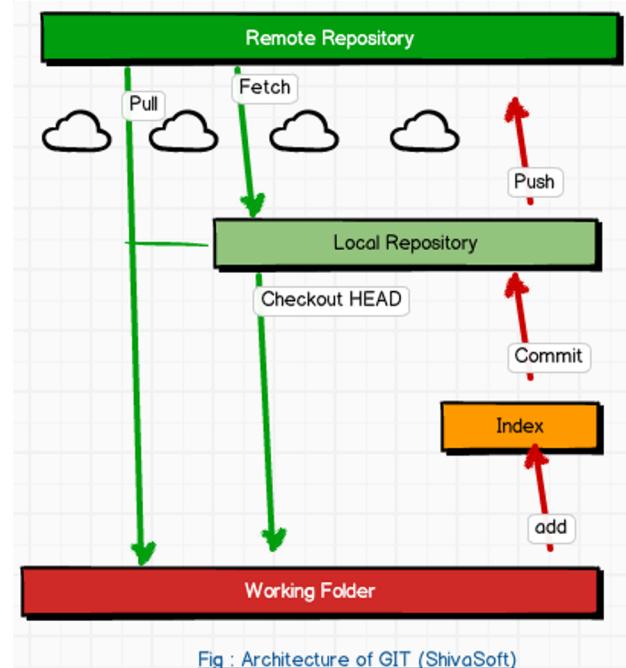
How the heck does git work??

- Directed Acyclic Graph
 - Each repository is a set of “steps” (commits)
 - Steps can only go forward
- Commit - set of changes applied from the previous snapshot (delta)
- Very (very) short workflow summary:
 - Make some changes
 - Tell Git which changes to keep
 - Commit those changes to the timeline (history)



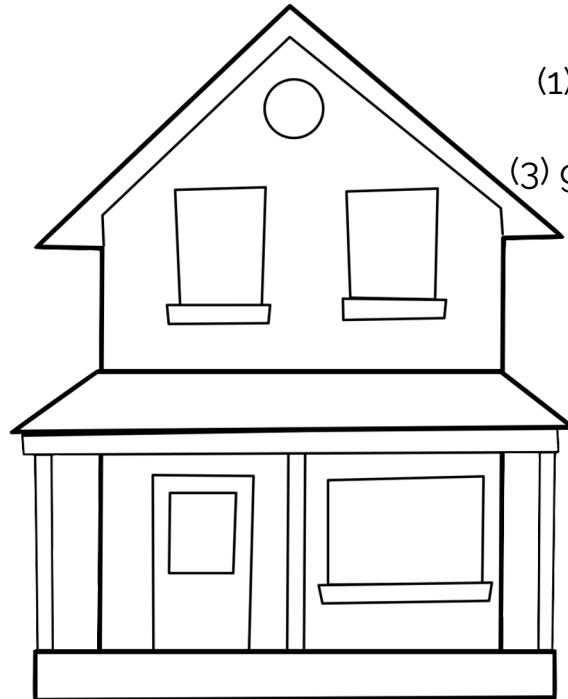
Git - Local & Remote

- Can (and usually do) have two copies of a repository - Local and Remote
- Local - where all operations are done
- Remote - where history is “synced”
- One more level - Index (or Stage)
 - Changes for a commit
 - Seems unnecessary, but is extremely useful



It's a git...house

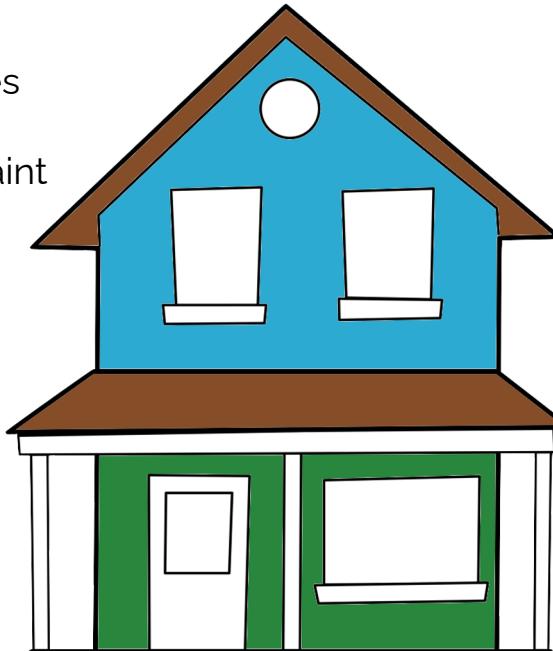
Repository



- (1) Make all changes
(2) git add .
(3) git commit -m "Paint and roof house"



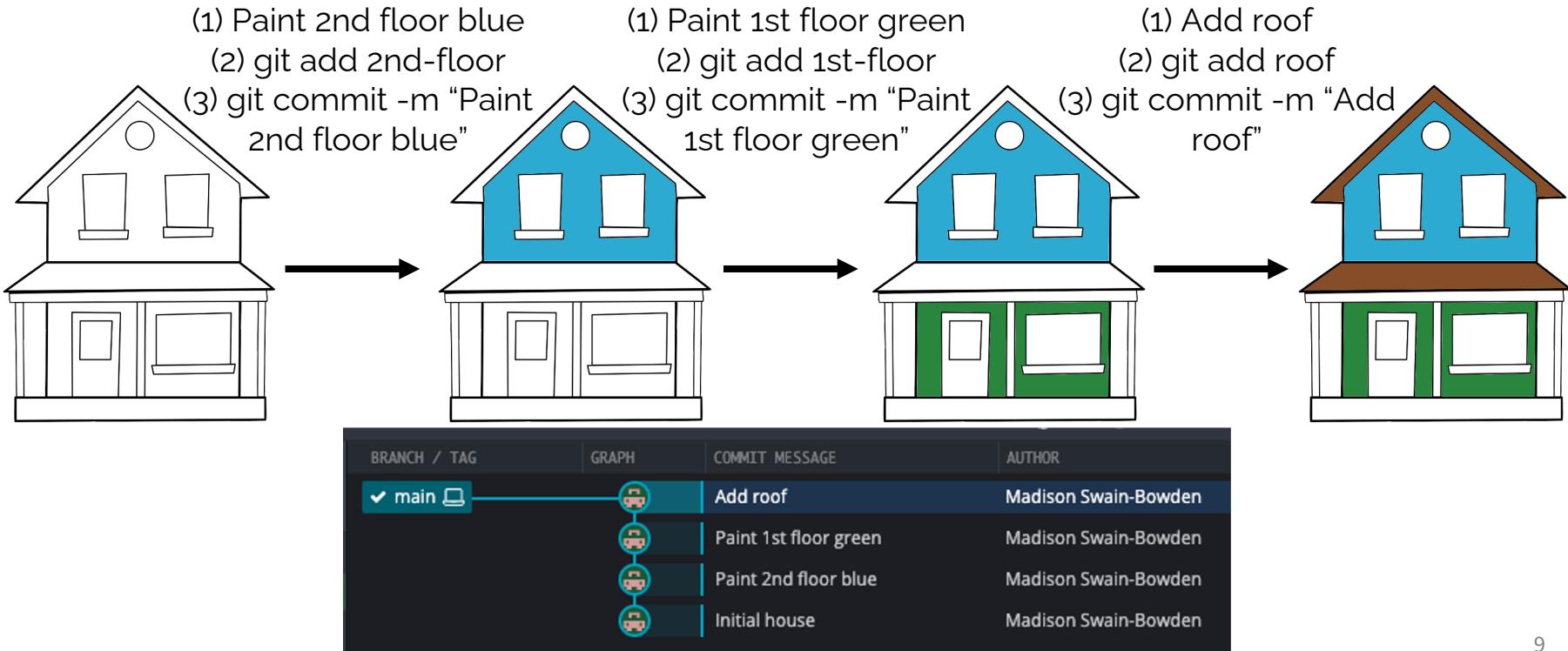
(Local Only)



| BRANCH / TAG | GRAPH | COMMIT MESSAGE | AUTHOR |
|--------------|-------|----------------------|----------------------|
| ✓ main | | Paint and roof house | Madison Swain-Bowden |
| | | Initial house | Madison Swain-Bowden |

Committing

(Local Only)



Local stuff

- Very (very) short workflow summary:
 - Make some changes
 - Tell Git which changes to keep
 - Commit those changes to the timeline (history)

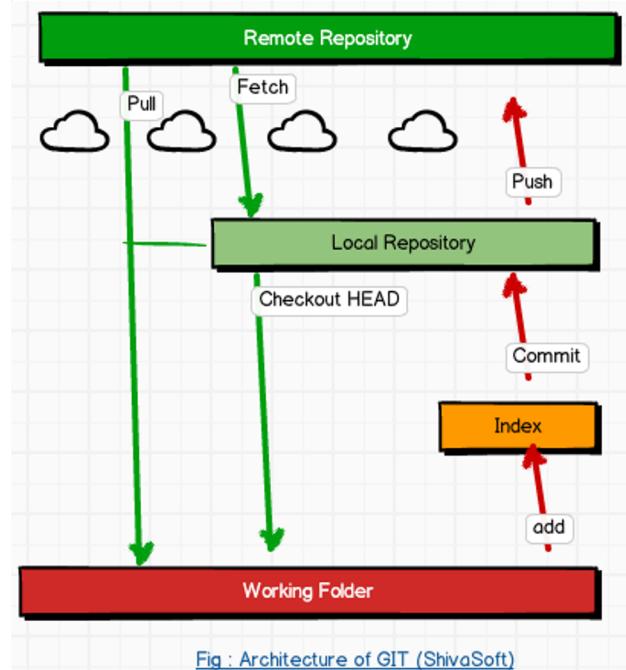


Fig : Architecture of GIT (ShivaSoft)

Local stuff

- Very (very) short workflow summary:
 - Make some changes
 - **Tell Git which changes to keep**
 - Commit those changes to the timeline (history)

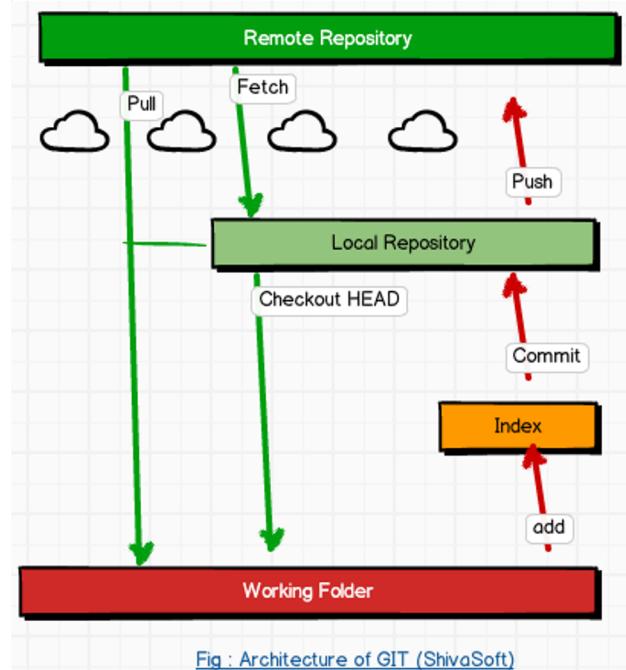


Fig : Architecture of GIT (ShivaSoft)

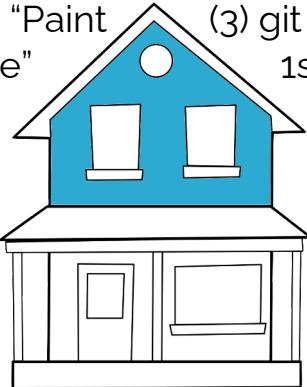
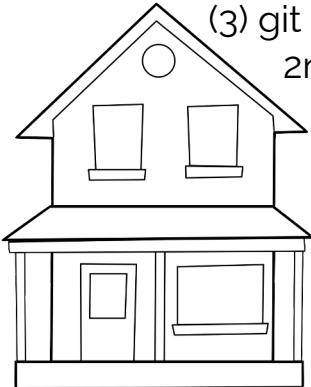
Committing



Local Repository

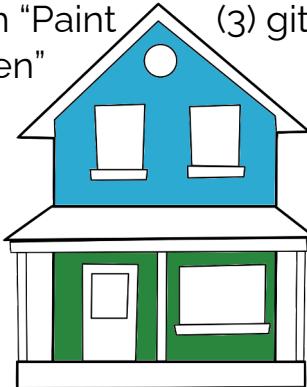
(1) Make all changes

- (2) git add 2nd-floor
- (3) git commit -m "Paint 2nd floor blue"



(1) Paint 1st floor green

- (2) git add 1st-floor
- (3) git commit -m "Paint 1st floor green"



(1) Add roof

- (2) git add roof
- (3) git commit -m "Add roof"



Working Folder



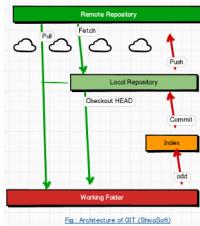
2nd floor



1st floor



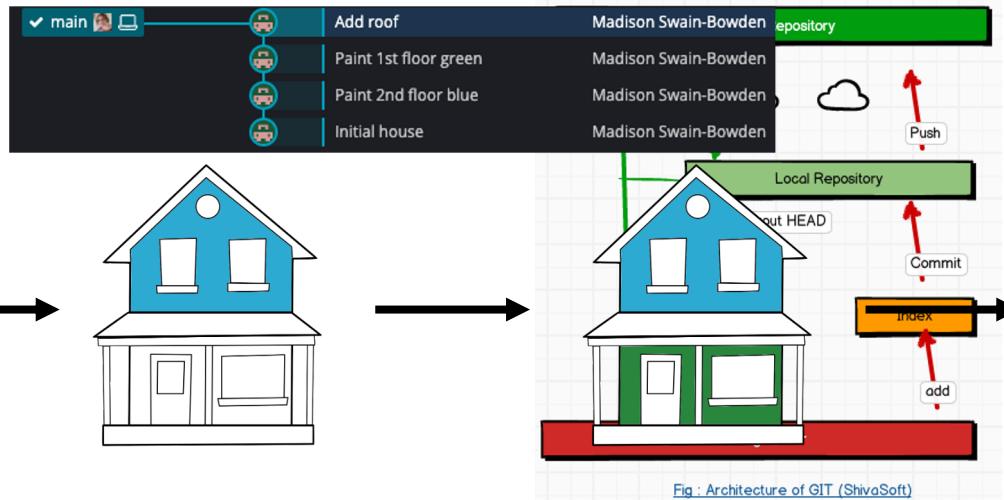
Roof



| BRANCH / TAG | GRAPH | COMMIT MESSAGE | AUTHOR |
|--------------|-------|-----------------------|----------------------|
| ✓ main | | Add roof | Madison Swain-Bowden |
| | | Paint 1st floor green | Madison Swain-Bowden |
| | | Paint 2nd floor blue | Madison Swain-Bowden |
| | | Initial house | Madison Swain-Bowden |

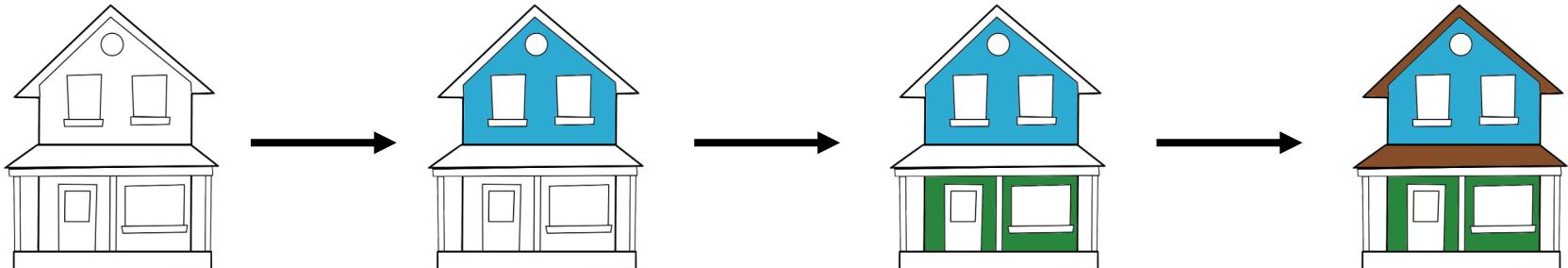
only)

Pushing Remote Repository



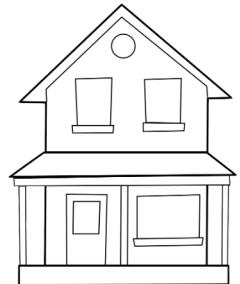
(Main only)

Local Repository



Branches

Main Branch

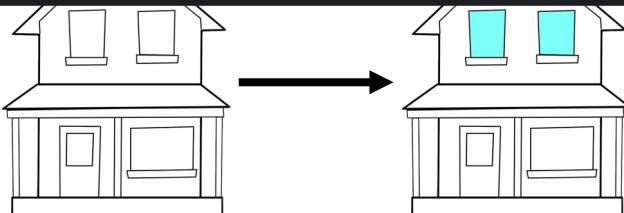


| BRANCH / TAG | GRAPH | COMMIT MESSAGE | AUTHOR |
|--------------|-------|-------------------------------|----------------------|
| ✓ branch 🏠 | | Add 2nd floor square windows | Madison Swain-Bowden |
| main 🚙 | | Add second floor round window | Madison Swain-Bowden |
| | | Add roof | Madison Swain-Bowden |
| | | Paint 1st floor green | Madison Swain-Bowden |
| | | Paint 2nd floor blue | Madison Swain-Bowden |
| | | Initial house | Madison Swain-Bowden |



Feature branch

- (1) git checkout -b <branch>
- (2) Commit like normal
- (3) git push -u origin <branch>



What next??

Combining Branches

Madison's Preference

- | | |
|--|--|
| <ol style="list-style-type: none">1. Merge commit2. Rebase3. Squash and rebase | <ol style="list-style-type: none">1. Squash and rebase2. Rebase3. Merge commit |
|--|--|

Merge commit

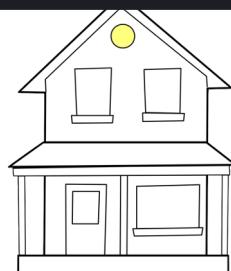
Main Branch



| BRANCH / TAG | GRAPH | COMMIT MESSAGE | AUTHOR |
|--------------|-------|----------------------------------|----------------------|
| ✓ main [] | [] | Merge branch 'feature' into m... | Madison Swain-Bowden |
| | [] | Add second floor square wind... | Madison Swain-Bowden |
| | [] | Add round window | Madison Swain-Bowden |
| | [] | Add roof | Madison Swain-Bowden |
| | [] | Paint 1st floor green | Madison Swain-Bowden |
| | [] | Paint 2nd floor blue | Madison Swain-Bowden |
| | [] | Initial house | Madison Swain-Bowden |



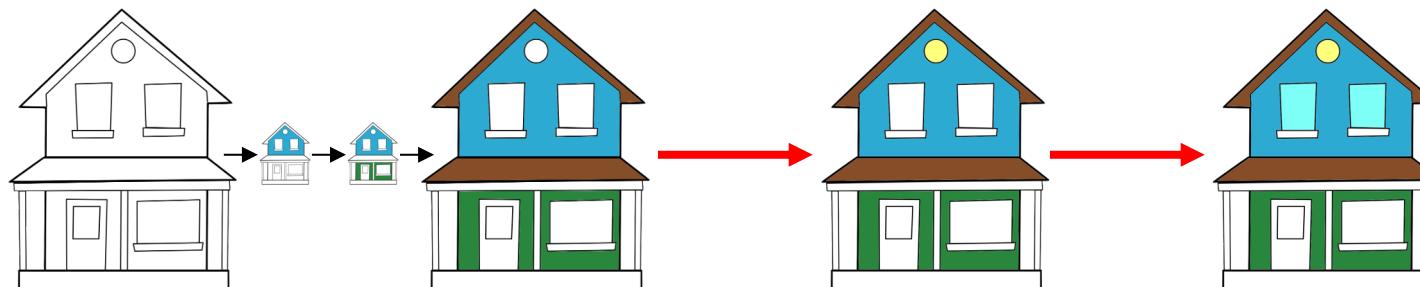
Feature branch



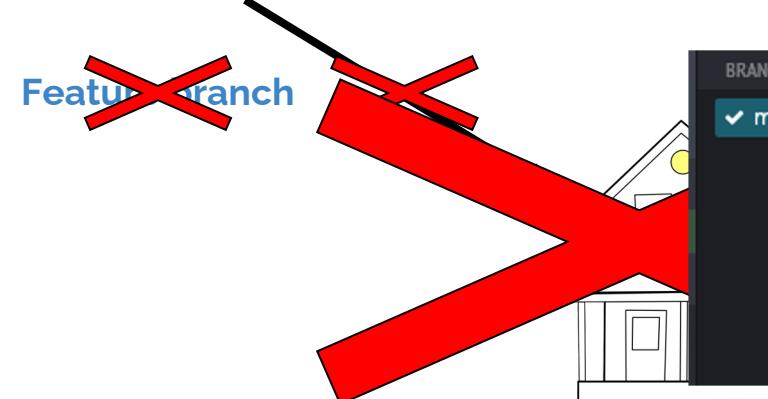
(2) Select "merge commit"

Rebase

Main Branch



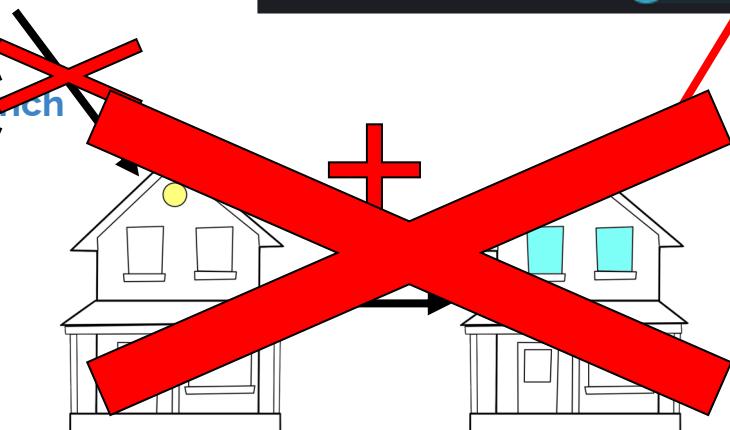
Feature branch



| BRANCH / TAG | GRAPH | COMMIT MESSAGE | AUTHOR |
|--------------|-------|------------------------------|----------------------|
| ✓ main | | Add 2nd floor square windows | Madison Swain-Bowden |
| | | Add 2nd floor round window | Madison Swain-Bowden |
| | | Add roof | Madison Swain-Bowden |
| | | Paint 1st floor green | Madison Swain-Bowden |
| | | Paint 2nd floor blue | Madison Swain-Bowden |
| | | Initial house | Madison Swain-Bowden |

Squash and Rebase

Main Branch



| BRANCH / TAG | GRAPH | COMMIT MESSAGE | AUTHOR |
|--------------|-------|-----------------------|----------------------|
| ✓ main | | Add 2nd floor windows | Madison Swain-Bowden |
| | | Add roof | Madison Swain-Bowden |
| | | Paint 1st floor green | Madison Swain-Bowden |
| | | Paint 2nd floor blue | Madison Swain-Bowden |
| | | Initial house | Madison Swain-Bowden |

| BRANCH / TAG | GRAPH | COMMIT MESSAGE |
|--------------|-------|-----------------------|
| ✓ main | | Add 2nd floor windows |
| | | Paint house, add roof |
| | | Initial house |

Integrating Code: A Comparison



Merge commit

| BRANCH / TAG | GRAPH | COMMIT MESSAGE | AUTHOR |
|--------------|-------|----------------------------------|----------------------|
| ✓ main | └─> | Merge branch 'feature' into m... | Madison Swain-Bowden |
| | └─> | Add second floor square wind... | Madison Swain-Bowden |
| | └─> | Add round window | Madison Swain-Bowden |
| | └─> | Add roof | Madison Swain-Bowden |
| | └─> | Paint 1st floor green | Madison Swain-Bowden |
| | └─> | Paint 2nd floor blue | Madison Swain-Bowden |
| | └─> | Initial house | Madison Swain-Bowden |

Rebase

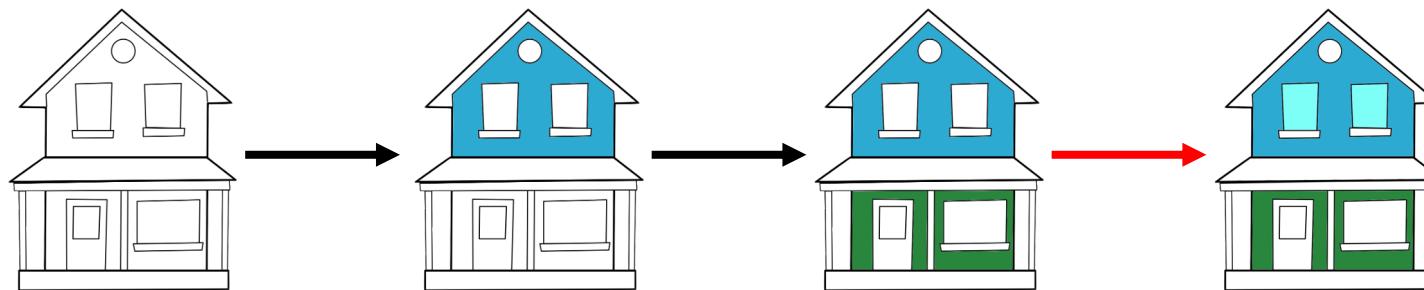
| BRANCH / TAG | GRAPH | COMMIT MESSAGE | AUTHOR |
|--------------|-------|------------------------------|----------------------|
| ✓ main | └─> | Add 2nd floor square windows | Madison Swain-Bowden |
| | └─> | Add 2nd floor round window | Madison Swain-Bowden |
| | └─> | Add roof | Madison Swain-Bowden |
| | └─> | Paint 1st floor green | Madison Swain-Bowden |
| | └─> | Paint 2nd floor blue | Madison Swain-Bowden |
| | └─> | Initial house | Madison Swain-Bowden |

Squash & Rebase

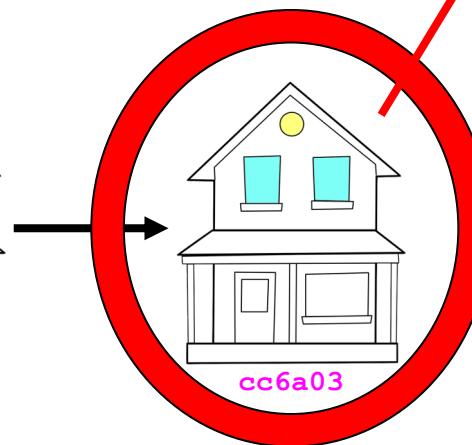
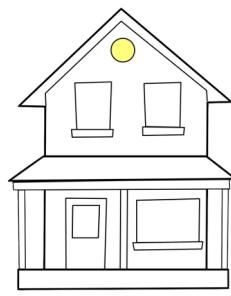
| BRANCH / TAG | GRAPH | COMMIT MESSAGE | AUTHOR |
|--------------|-------|-----------------------|----------------------|
| ✓ main | └─> | Add 2nd floor windows | Madison Swain-Bowden |
| | └─> | Add roof | Madison Swain-Bowden |
| | └─> | Paint 1st floor green | Madison Swain-Bowden |
| | └─> | Paint 2nd floor blue | Madison Swain-Bowden |
| | └─> | Initial house | Madison Swain-Bowden |

Special Case - Cherry Pick

Main Branch



Feature branch

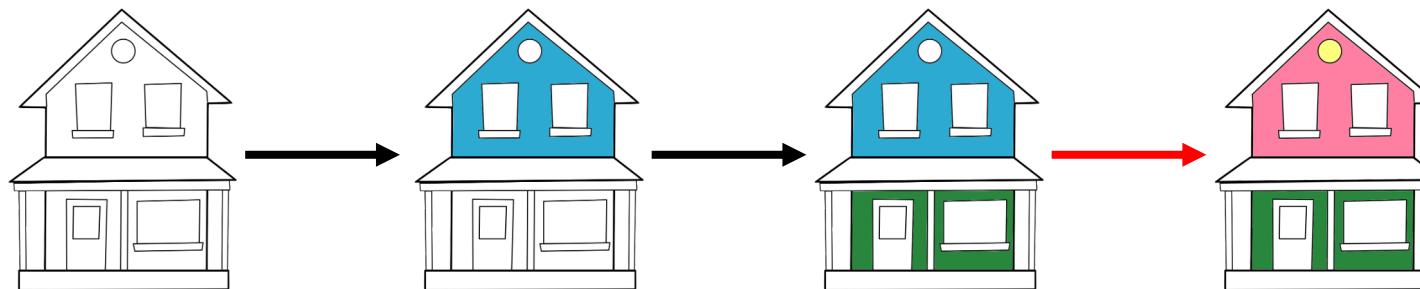


(1) git checkout <branch-to-alter>
(2) git cherry-pick <commit>

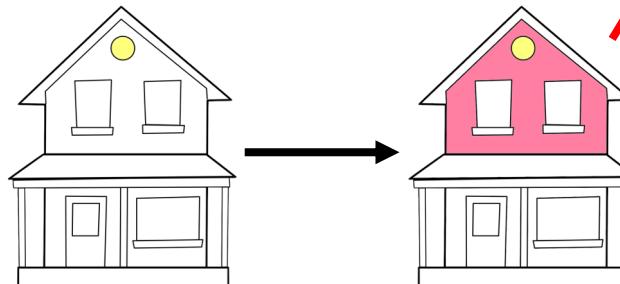
(1) git checkout `main`
(2) git cherry-pick `cc6a03`

Special Case - Conflict Resolution

Main Branch



Feature branch



Resolve the conflict locally
(usually through a UI)

Collaboration: Commit Messages

| | |
|-------------|--|
| af03989ba4e | Yet Another Useless Nitpick |
| 57289ce34a5 | Commenting this for compatibility |
| c8152e5e6a3 | these should be variables |
| 4e17fa7c36b | using better names |
| cd40f4aacbc | Using the provided debian service account for slurm |
| 2e9cd30d01a | Pretty sure this is not needed |
| 82321a70407 | Generalizing variables |
| 0f7c79616e6 | Changing permissions of spool folder |
| 6dc2cf0ff77 | Still don't need svc account home folders |
| b4db2f48c36 | Cleaning up |
| ba0a96edf2f | I am terrible at this.... |
| d136003472c | changing docker munge service user name to prevent crossover |
| 7646f0c29af | Swapping AWS images in defaults/ playbook to artifactory |
| ee5f2842527 | Removing unnecessary service user home |
| 1326ea5b3c8 | Changing UID to avoid conflicts |
| bbdc5f9a2a0 | Utilizing the aics-mariadb image |
| 39d44e22957 | Not sure why these are in this order |

| | |
|--------------|--|
| 63bc7624075 | [MDA-196] Set arguments and upload metadata for structure wrapper |
| c8119a10cdd | [MDA-196] WIP - Add structure segmentation wrapper |
| aee6304e20d | [MDA-185] Minor args correction |
| 32340e03a77 | [MDA-185] Update wrapper and DAG for brightfield prediction test |
| e14e1f39dcd | [MDA-185] Add brightfield prediction file IDs |
| 0b2d2f6e443 | [MDA-185] Drop files without brightfield predictions |
| 6c043ca3c2f | [MDA-185] Add v7 test file |
| d60f6fcbb6de | [MDA-193] Be more specific about which files are exportable |
| 133740ee2fe | [MDA-193] Outputs should be pushed and pulled by group |
| acc4eb1b714 | [MDA-193] Make second contours dependent on first, flake8 |
| f337542989e | [MDA-193] New DAG with contour generation testing |
| 73eee2b161d | HOTFIX: Add init files in bin subfolders for setup.py Once the scripts were moved into s |
| 5ec190109e2 | [MDA-189] Update wrappers to use save_single_channel |
| 3b8d40e6ed6 | [MDA-180] Make population of defaults more explicit |
| a2dbc37d05e | [MDA-180] Flake8 |
| 1871b3af0d3 | [MDA-180] Use legacy python type annotations, remove p4.4 from failures |
| ca8344b9c7d | [MDA-180] Refactor pipeline 4.4 DAG to use all components at top level |

Collaboration: Pull Requests

Bugfix: CoverageRight overrides #389

Merged madison-ookla merged 3 commits into `main` from `bugfix/cr-overrides-update` on Feb 12

Conversation 2 Commits 3 Checks 4 Files changed 5



madison-ookla commented on Feb 11

...

Description:

This PR fixes an issue discovered recently wherein Airflow introspects the `python_callable`, sees that it has a `**kwargs` (in this case, `**overrides`) variable, and tries to be helpful by injecting the entire Airflow context into that variable. This behavior actually breaks the mechanism we were using for determining valid/invalid configuration override variables. I've opted to move the overrides to an explicit argument, since there doesn't seem to be a way to tell Airflow *not* to do the thing it's doing.

I don't love that we have to make these changes in two places, so maybe we'll re-assess how much of the CR rewrite/OG codebase can be shared.

Development Checklist:

I solemnly swear that I have:

For Airflow python code

- Run `black`, `flake8`, and `mypy` against the changed files (see the "Contributing" section of the [README.md](#))
- Run `pytest .` in the `airflow/dags` folder and verified all tests pass
- Run the affected DAG locally without errors

I have not run the dags but I rendered the tasks's variables using Airflow's CLI and it looks good.

Deploy Requires:

- Updating the recurring-tasks repo on the Airflow VM
- Re-deploying Airflow on the VM

Additional notes

DAT-2848 Add incremental component to rotating pipelines #820

Merged madison-ookla merged 6 commits into `master` from `task/DAT-2848/rotating-plus-incremental` on 21 days ago

Conversation 5 Commits 6 Checks 0 Files changed 6



madison-ookla commented on Feb 12 • edited

This is a PR for DAT-2848:

What does this PR do?

This PR adds an incremental component to rotating pipelines, so that no matter where the pipeline are in the table, each run will pull in the most recent data. This adds some additional complexity, but should make our rotating pipelines more robust.

These changes also warranted generalizing some of the components of the incremental pipeline for use here, specifically some of the queries used.

NOTE: Entries in the `dataload_state` table have not been made yet.



I have:

- Pulled down master or rebased and this branch is up-to-date
- Used a descriptive, *Jira*-style, branch name and up-to-date ticket (with the `airflow` tag)
- Placed DAGs in their own sub-folders (including templates, helper code, etc.)
- Verified that the DAG uses appropriate pools (`pool name(s) here`)
- Written clear, well-documented code and self-reviewed
- Documented any templates and their parameters
- Run `flake8` and `black` on all source files

Pipelines

See [CONTRIBUTING.md](#) for more information

- Regenerated the pipelines documentation if any new pipelines have been added
- Added `dataload_state` entries if any new incremental or rotating pipelines have been added
- Ran pipeline validation tool (paste output below)

Collaboration: PR tips

Claims: Separate jobs for asymptotic/bootstrap approaches #865

Draft halldc wants to merge 1 commit into `master` from `split-bootstrap-asymptotic/DAT-5000`

Conversation 2 Commits 1 Checks 0 Files changed 2

Changes from all commits ▾ File filter... ▾ Jump to... ▾

Always Unified Split

Just for now Hide whitespace changes

Apply and reload

```
@@ -55,38 +55,40 @@  
55 CLAIMS = []  
56  
57 for service in [fixed, mobile]  
58 - for service_status in [  
59 -  
60 - claim = ClaimsConfig(  
61 -     dev=DEBUG,  
62 -     schedule_interval=service_status.schedule_interval,  
63 -     ingestion_delay_hr_delta=service_status.ingestion_delay_hr_delta,  
64 -     job_path=SPARK_JOB_PATH,  
65 -     config_template_name="spark-r-claims",  
66 -     unload_template_path=UNLOAD_TEMPLATE_PATH,  
67 -     service=service.name,  
68 -     status=service_status.status,  
69 -     claim_group=CLAIM_GROUP,  
70 -     claim_group_ref=CLAIM_GROUP_REF,  
71 -     user_id_field=service.user_id_field,
```

+223 -0

0 / 1 files viewed Review changes

Viewed ...

```
s":{  
lylan.ledbetter",  
st":"DQA-287",  
_date":"2021-03-08",  
play_name":"VT"
```

GitKraken Showcase



The screenshot displays the GitKraken application interface. At the top, there are several tabs: recurring-tasks, pookla, data-catalog, airflow-dags, extracts, amundsen, ookla-intelligence-admin, test-repo (which is currently active), and a plus sign icon. Below the tabs is a navigation bar with buttons for Undo, Redo, Pull, Push, Branch, Stash, Pop, Boards, and Timelines.

The main workspace shows a commit graph for the 'test-repo' under the 'main' branch. The commits are listed in a table:

| BRANCH / TAG | GRAPH | COMMIT MESSAGE | AUTHOR | COMMIT DATE / TIME |
|--------------|-------|-----------------------|----------------------|----------------------|
| main | main | Add 2nd floor windows | Madison Swain-Bowden | 3/15/2021 @ 12:00 PM |
| test | test | Paint house, add roof | Madison Swain-Bowden | 3/15/2021 @ 12:00 PM |
| main | main | Add 2nd floor windows | Madison Swain-Bowden | 3/15/2021 @ 11:59 AM |
| | | Add roof | Madison Swain-Bowden | 3/12/2021 @ 3:29 PM |
| | | Paint 1st floor green | Madison Swain-Bowden | 3/12/2021 @ 3:29 PM |
| | | Paint 2nd floor blue | Madison Swain-Bowden | 3/12/2021 @ 3:29 PM |
| | main | Initial house | Madison Swain-Bowden | 3/12/2021 @ 3:12 PM |

On the left sidebar, there are sections for repository, branch, pull requests, issues, and issue tracker selection. The issue tracker section lists GitHub, GitHub Enterprise, GitKraken Boards, GitLab, GitLab Self-Managed, Jira Cloud, Jira Server, Trello, and None, with GitHub selected. Other sections include Tags (0/0), Submodules (0), and GitHub Actions (0).

The right panel provides a detailed view of the commit 'Initial house' (commit: 32f7bb). It shows the commit message 'Initial house', author 'Madison Swain-Bowden', date 'authored 3/12/2021 @ 3:12 PM', and parent commit 'Add 2nd floor windows'. It also lists '1 added' file, 'README.md', and includes Path and Tree buttons.

GitKraken Showcase (note for CLI users)



```
base | pookla: madison@Madisons-MacBook-Pro
```

```
base | airflow-dags: madison@Madisons-MacBook-Pro (-bash)  #1
```

```
base | pookla: madison@Madisons-MacBook-Pro (-bash)  #2
```

```
~/.git/pookla
```

```
madison (e) base ➜ task/rename-logging ~ > git > pookla ➜ git log --all --decorate --oneline --graph
```

```
* a09e223 (origin/main) Pookla CLI: k8slogs (#21)
* dcfbab6 Fix/aws/athena/query s3 key (#20)
* 60cc6be Rename logging modules to log_utils (#19)
| * 0c36851 (HEAD -> task/rename-logging) Rename logging modules to log_utils
|/
* 289f161 (main) CI: pre-commit (#16)
* f4640c1 Deploy: sort by version number rather than alphanumeric (#15)
* 84194b7 DBHook: ODBC Athena Support (#14)
* 43d7935 Add deployment flag, instructions in README (#13)
* ed84457 Docs: Contributing (#12)
* 332e904 Pookla writer utils (#11)
* 72e4925 Pookla: Extra Requirements (#10)
| * dee365b (origin/feature/table-structure-difference) Add pookla as an editable install in conda requirements
| * 6f85b19 Conform values by default when comparing structures
| * 3991847 Add compare structure tests
| * 88a11b8 Add test requirements to extra install option
| * 438589a Add type hinting to database structure module
| * c9156f8 Add other structure functions
| * 3a1e572 Rename structure module, add compare structure functions
| * 88f6246 Unify conda environment file with requirements file
|/
* 41a0b73 Add Dockerfile, requirements.txt (#6)
* 2a05023 Fix S3 pip index, Pookla: Set Pyarrow Version (#8)
* 6495d72 Support Multiple Packages (#7)
| * fb3da43 (origin/feature/scheduled-build, feature/scheduled-build) Change cron to daily for testing
| * 3730578 Use correct environment variable
| * c89a15b Use both arguments in tool version function
| * c43dce7 Add scheduled portion to Jenkinsfile
|/
* cfeca990 Add CI checks (#4)
* d64e2d4 Merge pull request #2 from teamookla/migrate_pookla
| \
| * aa4327b Update with github py gitignore
| * 6630b43 Migrate pookla from ookla-data
|/
* 43409bd Setup code owners
* c9a2e0a Init repo
* 54b9d6d Create .gitignore
```

```
madison (e) base ➜ task/rename-logging ~ > git > pookla ➜
```



GitHub CLI Showcase

GITHUB CLI

Take GitHub to the command line

GitHub CLI brings GitHub to your terminal. Free and open source.

`brew install gh` or [Download for Mac](#)

[View installation instructions →](#)

\$ `gh pr create`

Create a new pull request.

```
$ gh pr create
Creating pull request for feature-branch into main in ampinsk/test
? Title My new pull request
? Body [(e) to launch vim, enter to skip]
? What's next? Submit
http://github.com/owner/repo/pull/1
```

Takeaways

- Git records changes to files
 - You have great power over which changes to report and how to report them
- Git is a tool for collaboration
 - Try to make things easy and approachable for others (or even your future self!)
- Always be aware of “where” you are in the git commit graph
- If you run into a bad “gituation” - ask Madison!
 - Special thanks to Larisa for that nomenclature

Thank you!