Git: (Distributed) Version Control

Computer Science and Engineering
College of Engineering
The Ohio State University

Lecture 2

The Need for Version Control

Computer Science and Engineering
The Ohio State University

□ Track evolution of a software artifact

- Development is often non-linear
 - Older versions need to be supported
 - Newer versions need to be developed

Development is non-monotonic

- May need to undo some work, go back to an older version, or track down when a mistake was introduced
- Facilitate team-based development
 - Multiple developers working on a common code base
 - How can project be edited simultaneously?

Key Idea: A Repository

Computer Science and Engineering
The Ohio State University

□ *Repository* = working tree + store + index

- Warning: "Repo" often used (incorrectly) to mean just the store or just the working tree
- □ *Working tree* = project itself
 - Ordinary directory with files & subdirectories
- □ *Store* = history of project
 - Hidden directory: don't touch!
- □ *Index* = virtual snapshot
 - Gateway for moving changes in the working tree into the store (aka stage, cache)
- □ *History* = DAG of *commits*
 - Each node in graph corresponds to a complete snapshot of the entire project

File Structure of a Repository

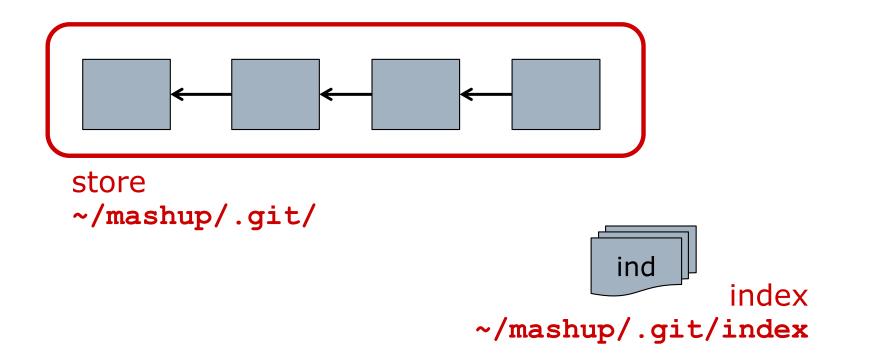
~/mashup/
- css/
buckeye-alert-resp.css
demo.css
— demo-js.html
Gemfile
HEAD
index
├── .gitignore
README.md
└──etc

Conceptual Structure

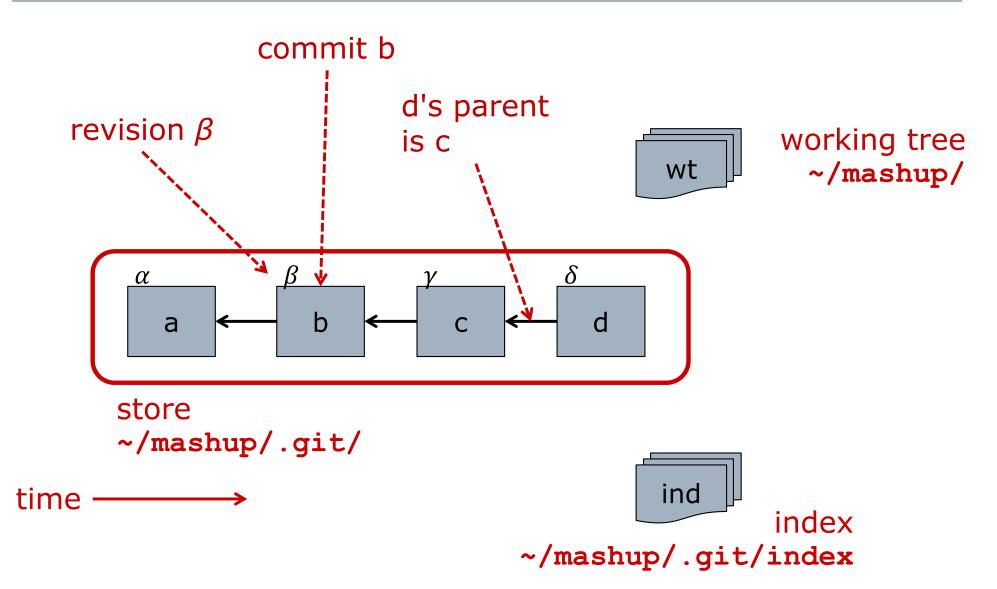
Computer Science and Engineering
The Ohio State University



working tree
~/mashup/



A History of Commits



Commit: Snapshot or Delta?

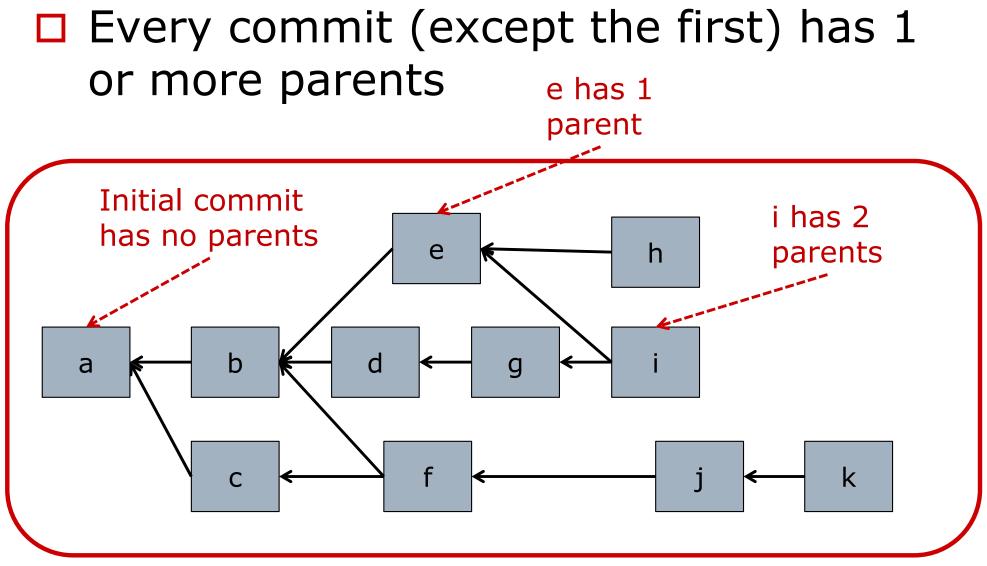
Computer Science and Engineering
The Ohio State University

□ A commit can viewed as *both*

- 1. A complete snapshot of the project at that point in time (a revision), and
- 2. A delta of the changes made (a patch); that is, the diff between revisions
- Different git actions use different views of a commit
 - View as a snapshot is most common
 - But both are useful
- Clever data structures make both views available efficiently
 - Git objects (trees, blobs...) and references
 - Merkle tree

History is a DAG

Computer Science and Engineering
The Ohio State University



store

Example View of DAG

Computer Science and Engineering
The Ohio State University

	5	
update eleventy config	12 Aug 2020 17:04	930e56c3
update tech task 1	12 Aug 2020 16:35	7a2c28e2
add two CSS related resource links	5 Aug 2020 01:37	d26a42c8
update project 3	4 Aug 2020 20:52	0851f5ca
Merge remote-tracking branch 'origin/master' into	4 Aug 2020 12:29	2e87942c
deploy: 1dde4c26e5abaf073bf5b0bc46052e58c3c4c	3 Aug 2020 12:35	ba0f7989
watch .eleventy.js for changes	3 Aug 2020 12:34	1dde4c26
😢 origin/header-anchor-links add links to h2, h3, h	3 Aug 2020 12:34	684f3125
deploy: fb1e3f6a886c51fcb7743ec37ddb203f86d2b	31 Jul 2020 23:50	d0963fb7
Merge pull request #68 from cse3901-osu/project-2	31 Jul 2020 23:49	fb1e3f6a
clean up description	31 Jul 2020 23:45	6c7a2cbe
deploy: b2c9b033bcb20aaee9834263884c8fc2012e	31 Jul 2020 16:26	2aa47d18
assignments helper macro	31 Jul 2020 16:25	b2c9b033
reorganize table includes	31 Jul 2020 16:22	0f0959dc
use template inheritance for assignment tables	31 Jul 2020 15:32	924517e6
draft assignments	31 Jul 2020 15:31	77ba7f68
origin/ejs convert homepage to ejs	30 Jul 2020 15:58	52d6b95f
find some interesting APIs	29 Jul 2020 16:09	b882519f
update optional functionality list	29 Jul 2020 16:08	1429d5b2

time

Example View of DAG

```
$ git log --oneline --no-decorate --graph
```

```
* 1618849 clean up css
* d579fa2 merge in improvements from master
|\
| * 0f10869 replace image-url helper in css
* | b595b10 add buckeye alert notes
* | a6e8eb3 add raw buckeye alert download
|/
* b4e201c wrap osu layout around content
* e9d3686 add Rakefile and refactor schedule loop
* 515aaa3 create README.md
* eb26605 initial commit
```

Commit

Computer Science and Engineering
The Ohio State University

Each commit is identified by a hash

- 160 bits (i.e., 40 hex digits)
- Practically guaranteed to be unique
- Can use short prefix of hash if unique

\$ git show --name-only --no-decorate commit <u>16188493c252f6924baa17c9b84a4c1baaed438b</u> Author: Brutus Buckeye <brutus@users.noreply.github.com> Date: Mon Mar 29 15:30:50 2023 +0200

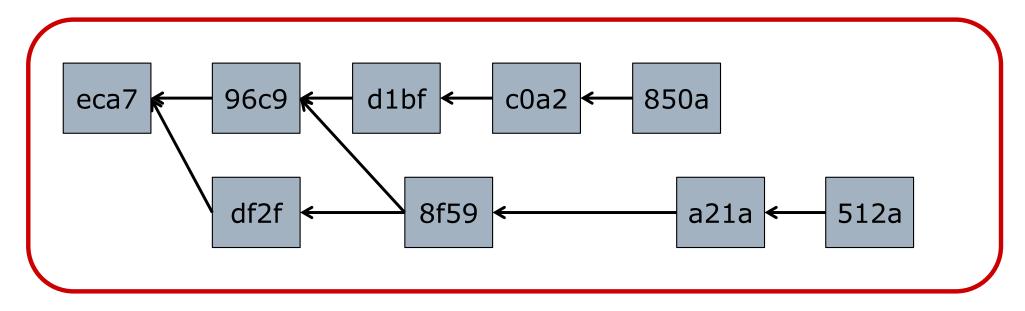
clean up css

source/stylesheets/_site.css

History is a DAG

Computer Science and Engineering
The Ohio State University

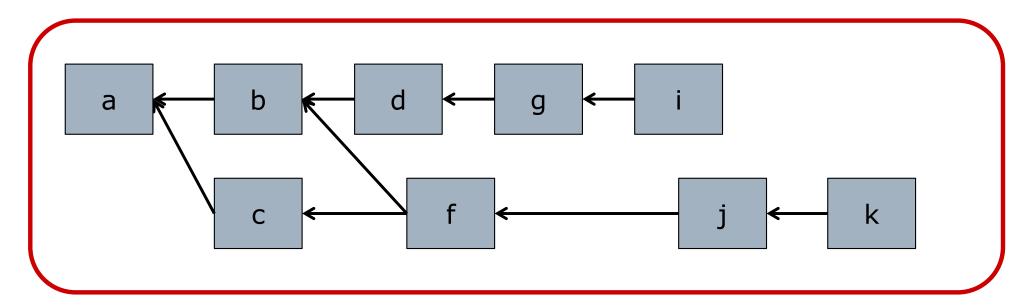
A better picture would label each commit with its hash (prefix)



History is a DAG

Computer Science and Engineering
The Ohio State University

A better picture would label each commit with its hash (prefix)

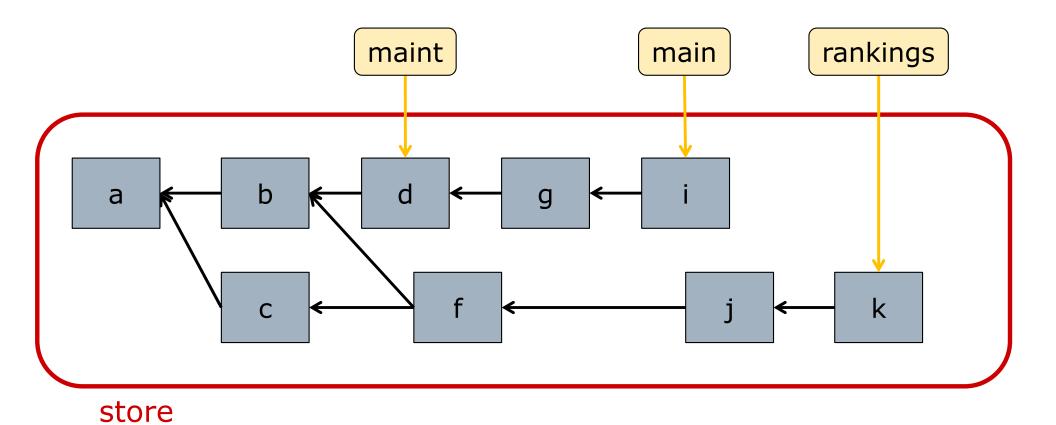


But in these slides, we abbreviate the hash id's as just: 'a', 'b', 'c'...

Nomenclature: Branch

Computer Science and Engineering
The Ohio State University

Branch: a pointer to a commit Not a "branch" in the DAG's shape



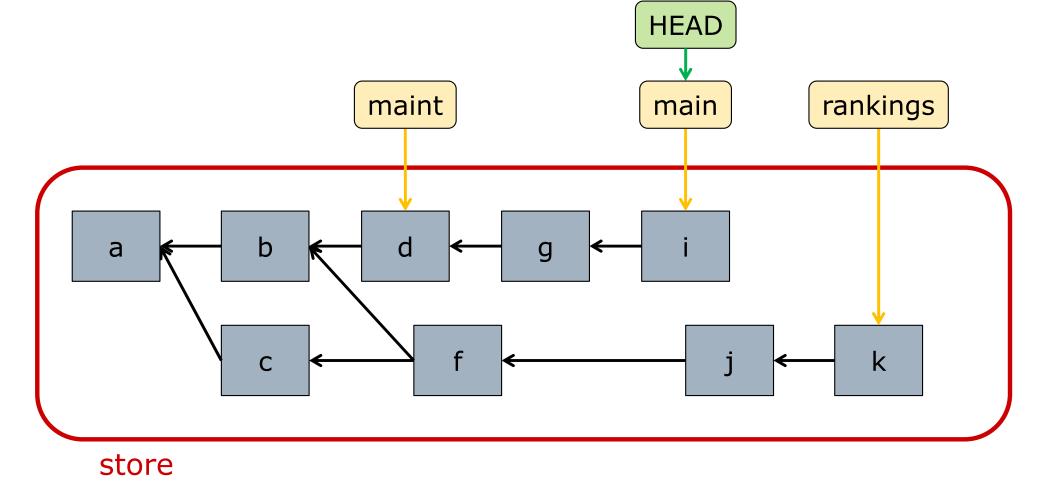
A Note on Naming Conventions

- Any name can be used for a branch
 - Typically short, but hopefully descriptive
 - Many branches, each with a unique name
- □ Initially, a repo has a single branch
 - Default branch for many git commands
 - Convention: use "main" as the name of this default branch
 - Warning: Repos created locally use an old naming convention ("master") by default
 This default is user-configurable
 init.defaultBranch main

Nomenclature: HEAD

Computer Science and Engineering
The Ohio State University

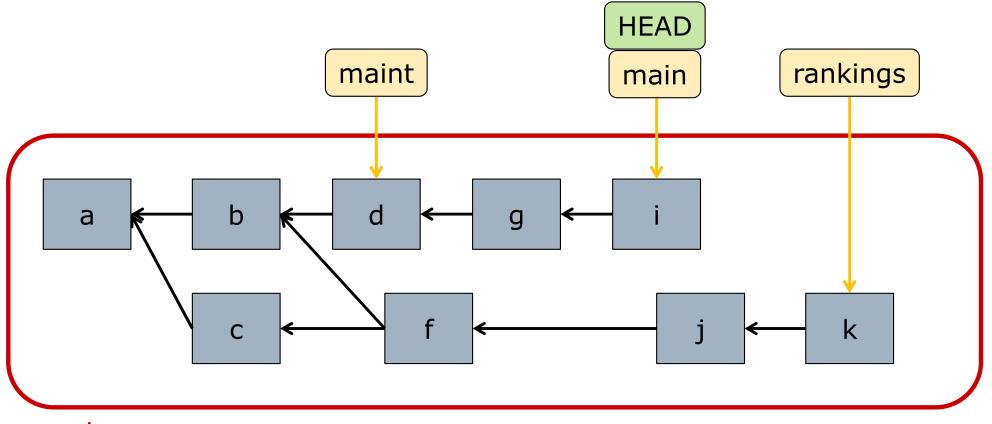
□ *HEAD*: a special reference, (usually) points to a branch



Nomenclature: HEAD

Computer Science and Engineering
The Ohio State University

Useful to think of HEAD as being "attached" to a particular branch



store

View of DAG with Branches

Computer Science and Engineering
The Ohio State University

\$ git log --oneline --graph --all

```
* 1618849 (HEAD -> main) clean up css
```

```
* d579fa2 (alert) merge in improvements from master
|\
```

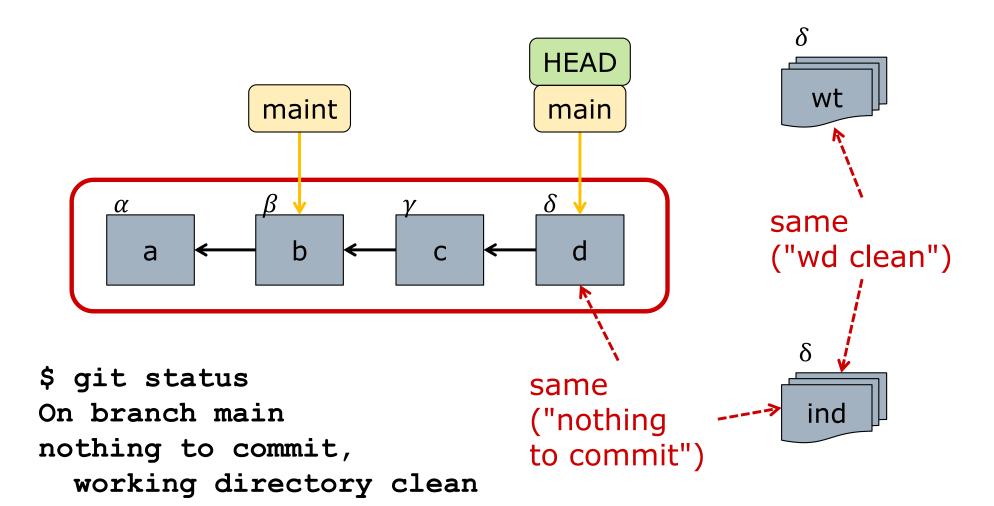
```
| * Of10869 replace image-url helper in css
```

* | b595b10 add buckeye alert notes

```
* | a6e8eb3 add raw buckeye alert download
|/
```

- * b4e201c wrap osu layout around content
- * e9d3686 add Rakefile and refactor schedule loop
- * 515aaa3 create README.md
- * eb26605 initial commit

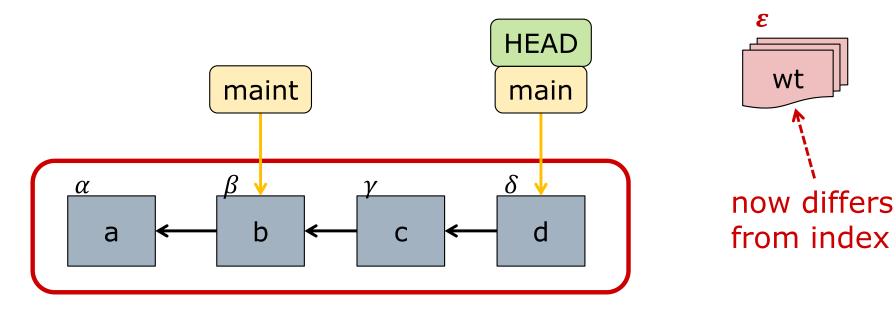
A "Clean" Repository

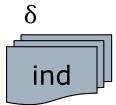


Edit Files in Working Tree

Computer Science and Engineering
The Ohio State University

□ Add files, remove files, edit files...

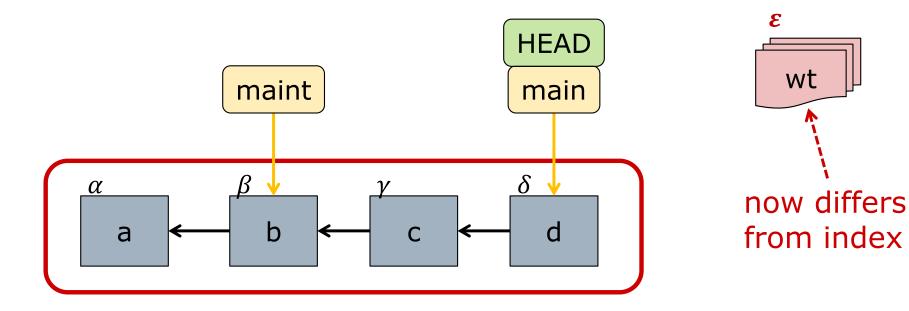




Edit Files in Working Tree

Computer Science and Engineering
The Ohio State University

□ Add files, remove files, edit files...



\$ git status
On branch main
Changes not staged for commit:
 modified: css/demo.css

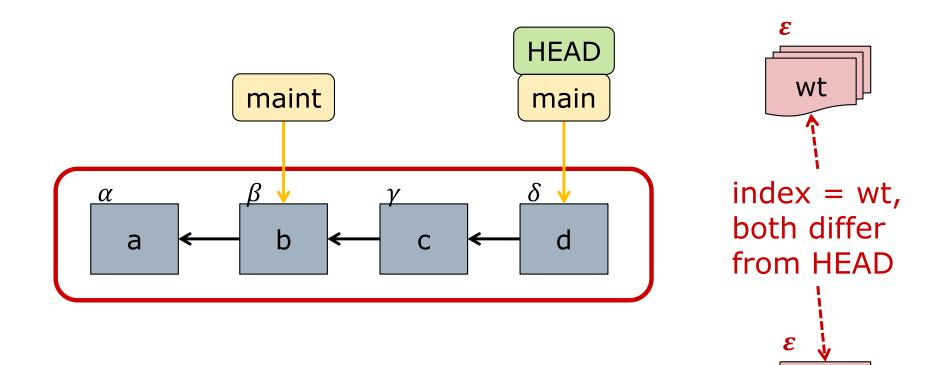
δ

Add: Working Tree \rightarrow Index

Computer Science and Engineering
The Ohio State University

ind

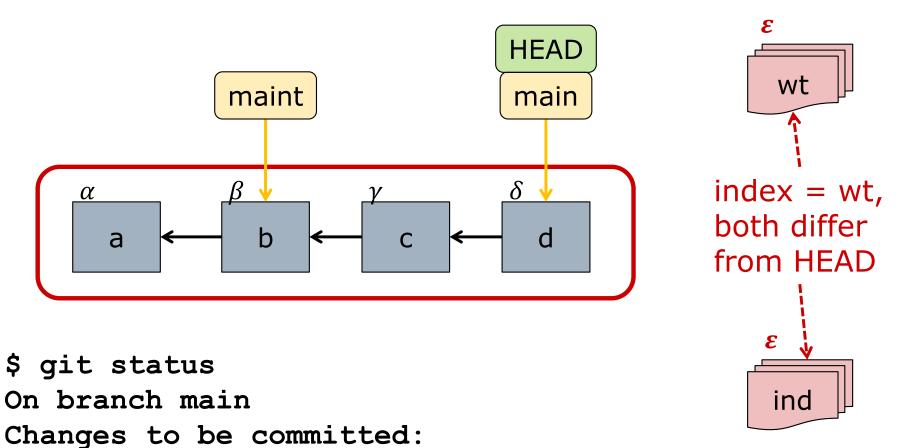
\$ git add . # current directory, and below



Add: Working Tree \rightarrow Index

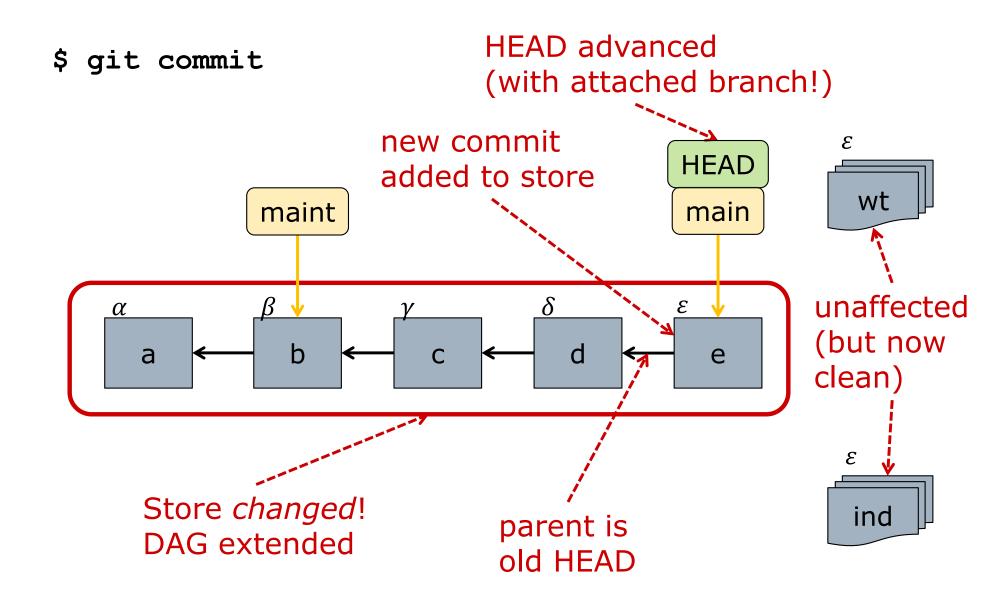
Computer Science and Engineering
The Ohio State University

\$ git add . # current directory, and below



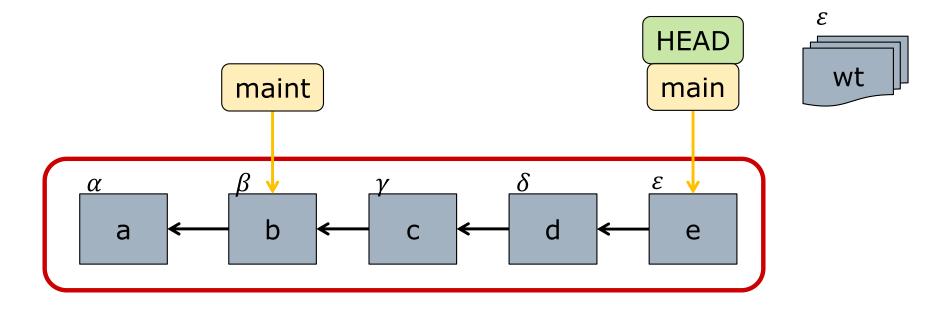
modified: css/demo.css

Commit: Index \rightarrow Store

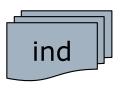


The (New) State of Repository

Computer Science and Engineering
The Ohio State University



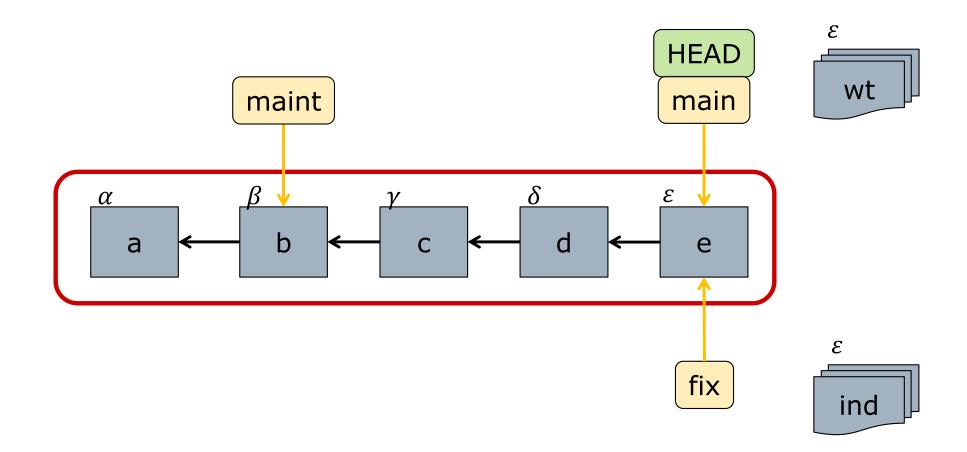
ε



Creating a New Branch

Computer Science and Engineering
The Ohio State University

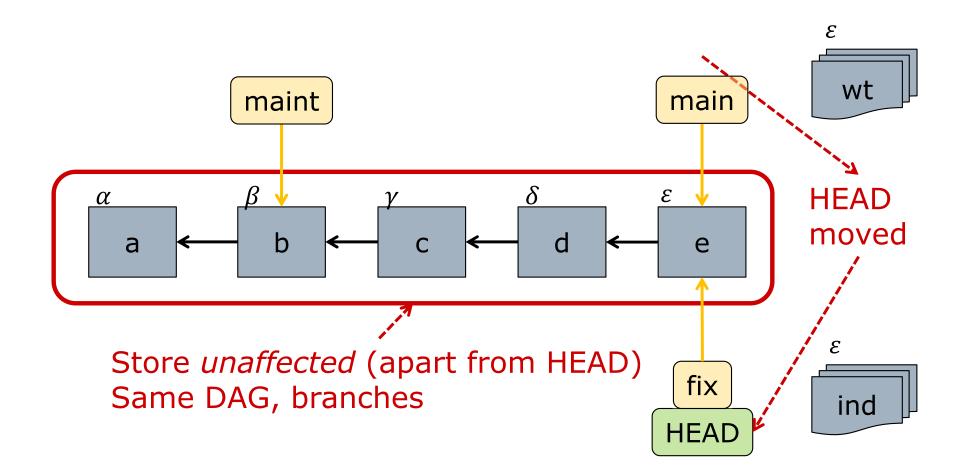
\$ git branch fix



Checkout: Changing Branch

Computer Science and Engineering
The Ohio State University

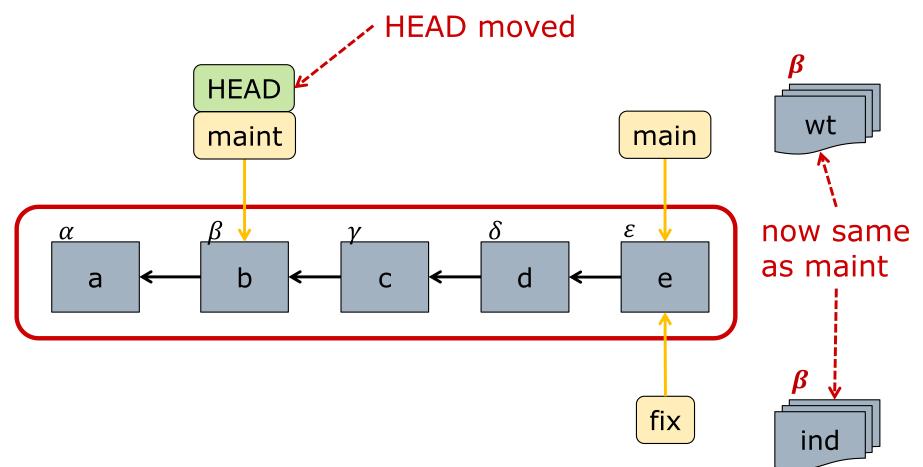
\$ git checkout fix



Checkout: Changing Branch

Computer Science and Engineering
The Ohio State University

\$ git checkout maint

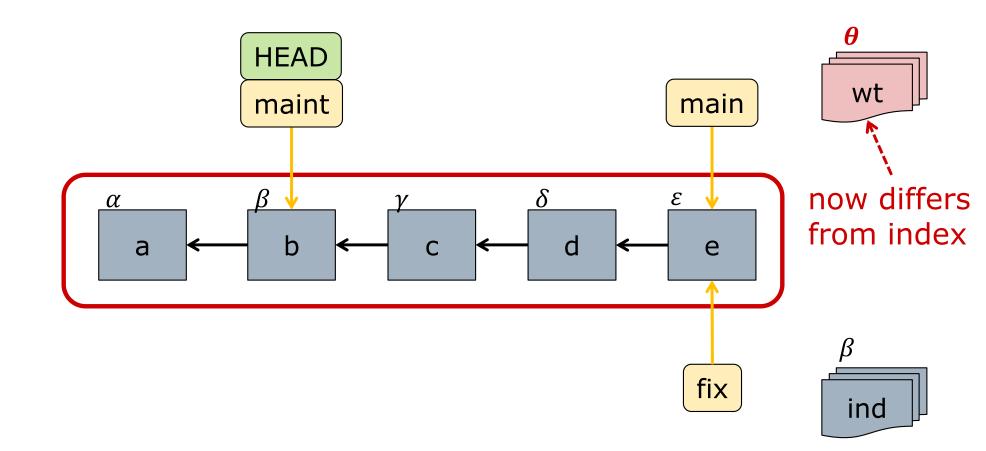


Advice: checkout <branch> only when wt is clean

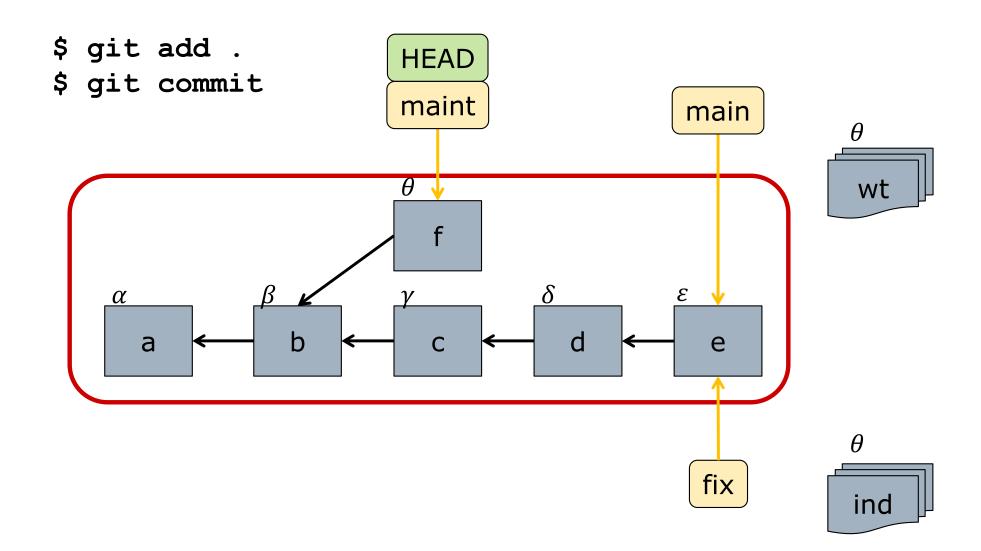
Edit Files in Working Tree

Computer Science and Engineering
The Ohio State University

□ Add files, remove files, edit files...



Add & Commit: Update Store

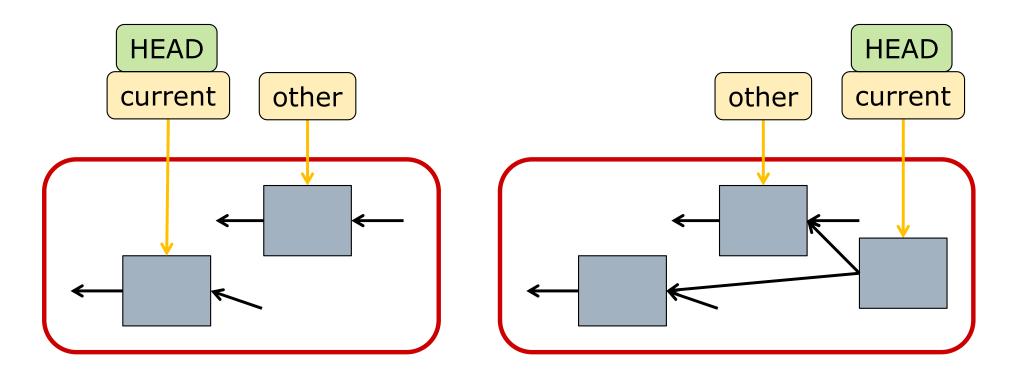


Merge: Bringing History together

Computer Science and Engineering
The Ohio State University

Bring work from another branch into current branch

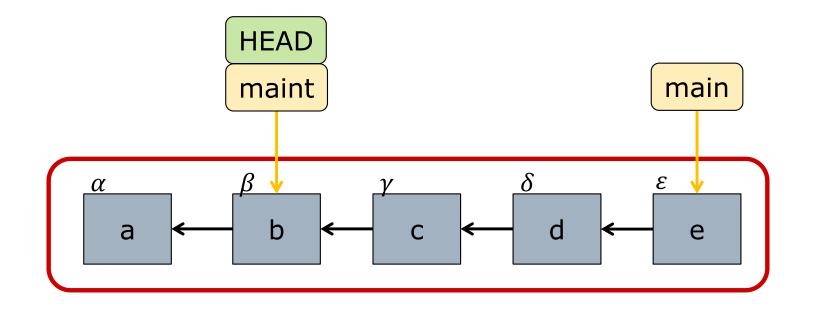
- Implemented features, fixed bugs, etc.
- Updates current branch, not other

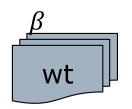


Merge – Case 1: Ancestor

Computer Science and Engineering
The Ohio State University

HEAD is an ancestor of other branch

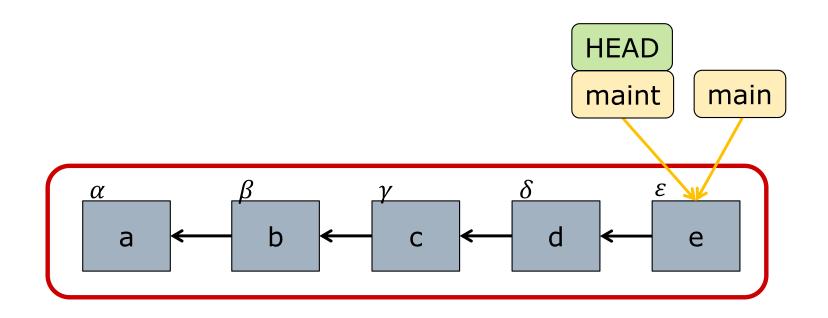


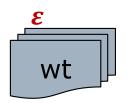


Fast-Forward Merge

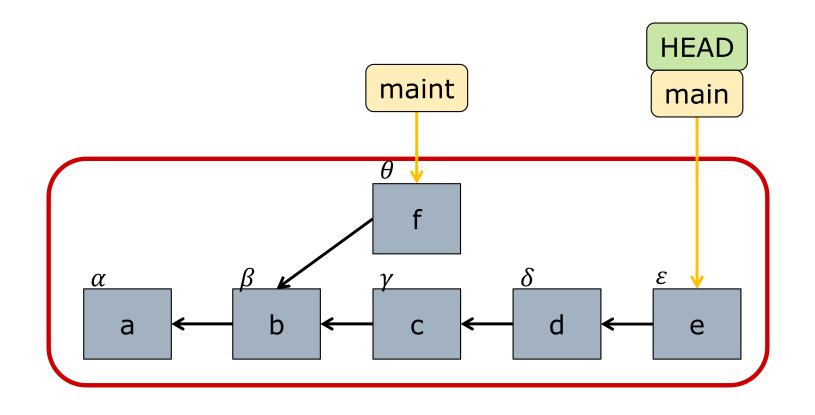
Computer Science and Engineering
The Ohio State University

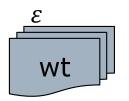
\$ git merge main



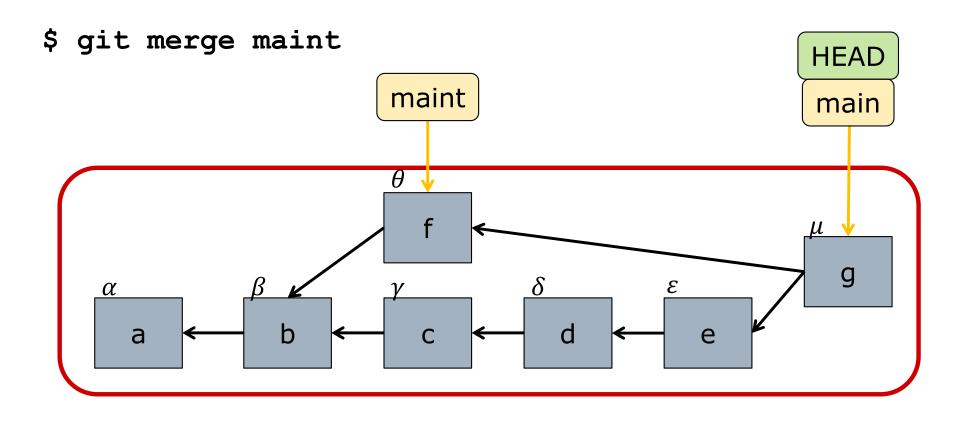


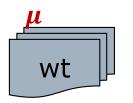
Merge – Case 2: No Conflicts





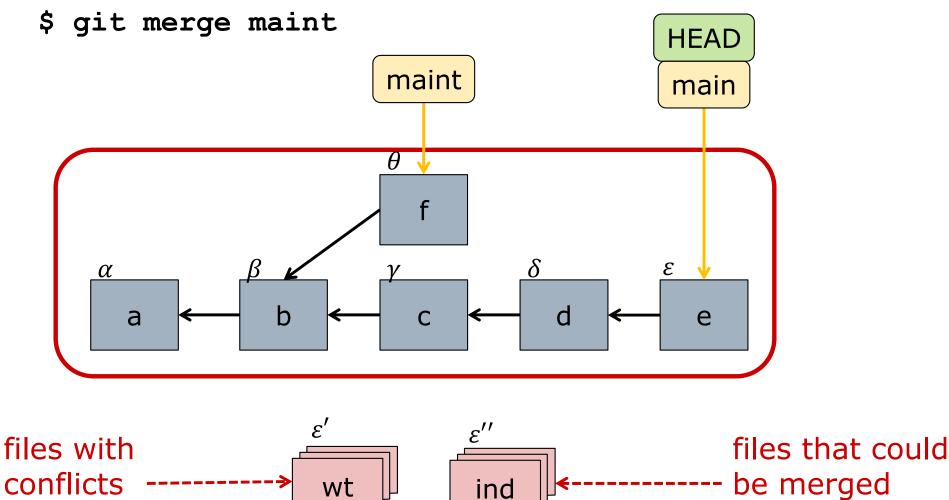
Merge Automatically Commits





Merge – Case 3: Conflicts Exist

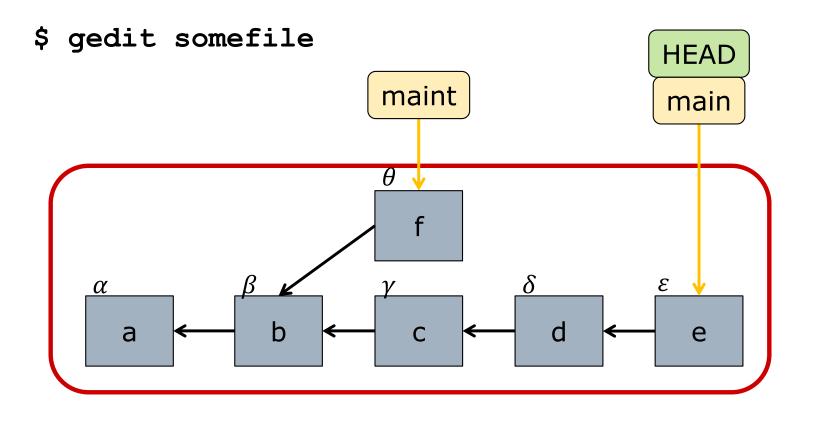
Computer Science and Engineering
The Ohio State University

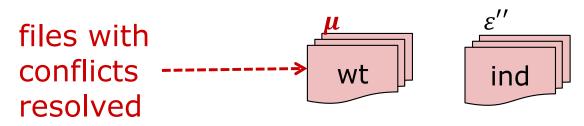


marked

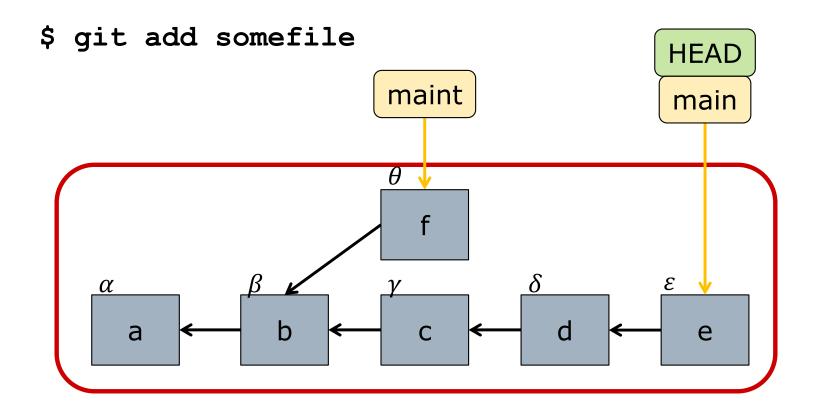
automatically

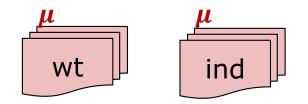
Merge: Resolve Conflicts



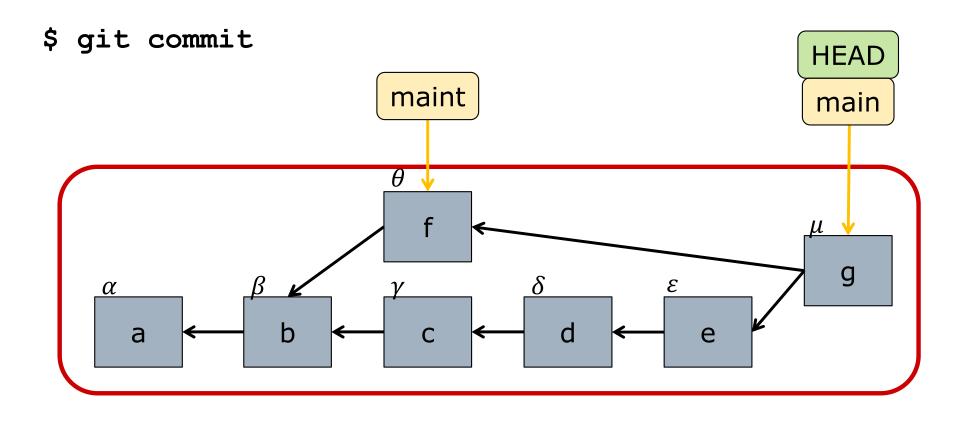


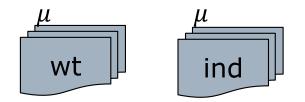
Merge with Conflicts: Add





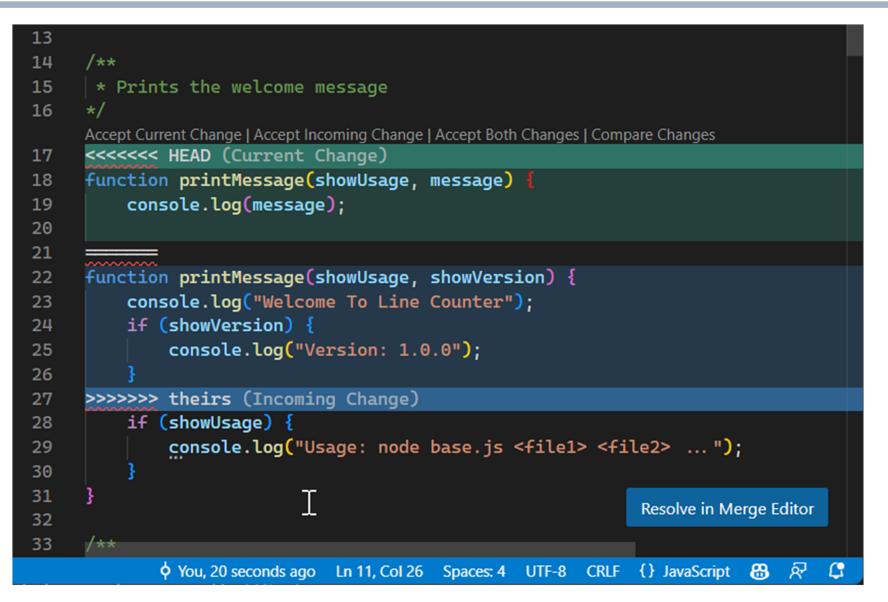
Merge with Conflicts: Commit





Merge: Edit to Resolve Conflicts

Computer Science and Engineering
The Ohio State University



https://code.visualstudio.com/docs/sourcecontrol/overview

Merge: 3-way Merge Editor

Computer Science and Engineering
The Ohio State University

Js target	.js ! • Js Merging: target.js ! •		⁴"↑↓↔⊙⊹⊛["…
merge-gi	it-playground > 🗾 target.js > 🏵 printMessage		
Incoming	∲7b18bdb • theirs ····	Current 🔶	b7bd9b1 • main ••••
15 16	<pre>/** * Prints the welcome message */ Accept Incoming Accept Combination (Incoming First) Ignore function printMessage(showUsage, showVersion) { console.log("Welcome To Line Counter"); if (showVersion) { } </pre>	15 16	<pre>/** * Prints the welcome message */ Accept Current Accept Combination (Current First) Ignore function printMessage(showUsage, message) { console.log(message); </pre>
20 21 22 23 24	<pre>console.log("Version: 1.0.0"); if (showUsage) { console.log("Usage: node base.js <file1 pre="" }<=""></file1></pre>	19 20 > 21 22	<pre>if (showUsage) { console.log("Usage: node base.js <file1> } </file1></pre>
13	rge-git-playground\target.js		1 Conflict Remaining •••
14 15 16	* Prints the welcome message */		
17 18 19	No Changes Accepted function printMessage(showUsage) { console.log("Welcome To Line Counter");		
20 21 22	<pre>if (showUsage) { console.log("Usage: node base.js <file pre="" }<=""></file></pre>	1> <file2< td=""><td>Complete Merge</td></file2<>	Complete Merge
ዖ main! 🖙	>		🛛 🛛 Ln 17, Col 31 Spaces: 4 CRLF - {} JavaScript 🔀 🔊 🗘

MS demo resolving merge conflicts: youtube.com/watch?v=HosPml1qkrg

Summary

Computer Science and Engineering
The Ohio State University

Repository = working tree + store

- Store contains history
 - History is a DAG of commits
- References, tags, and HEAD
- Commit/checkout are local operations
 - Former changes store, latter working tree
- Merge
 - Directional (merge other "into" HEAD)