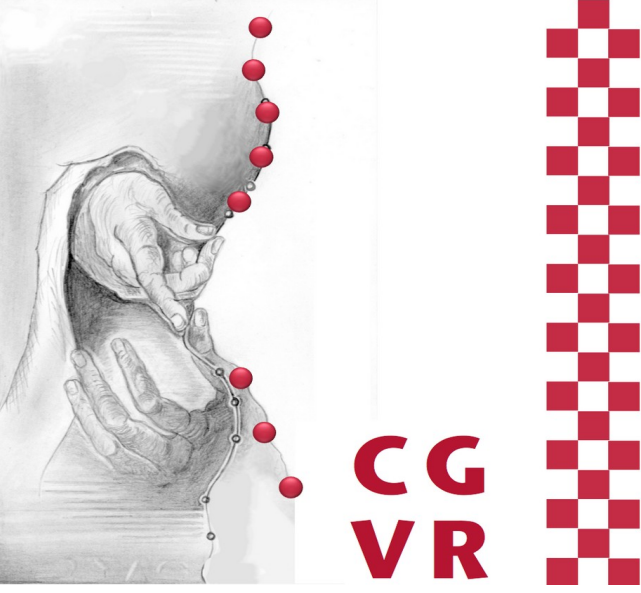




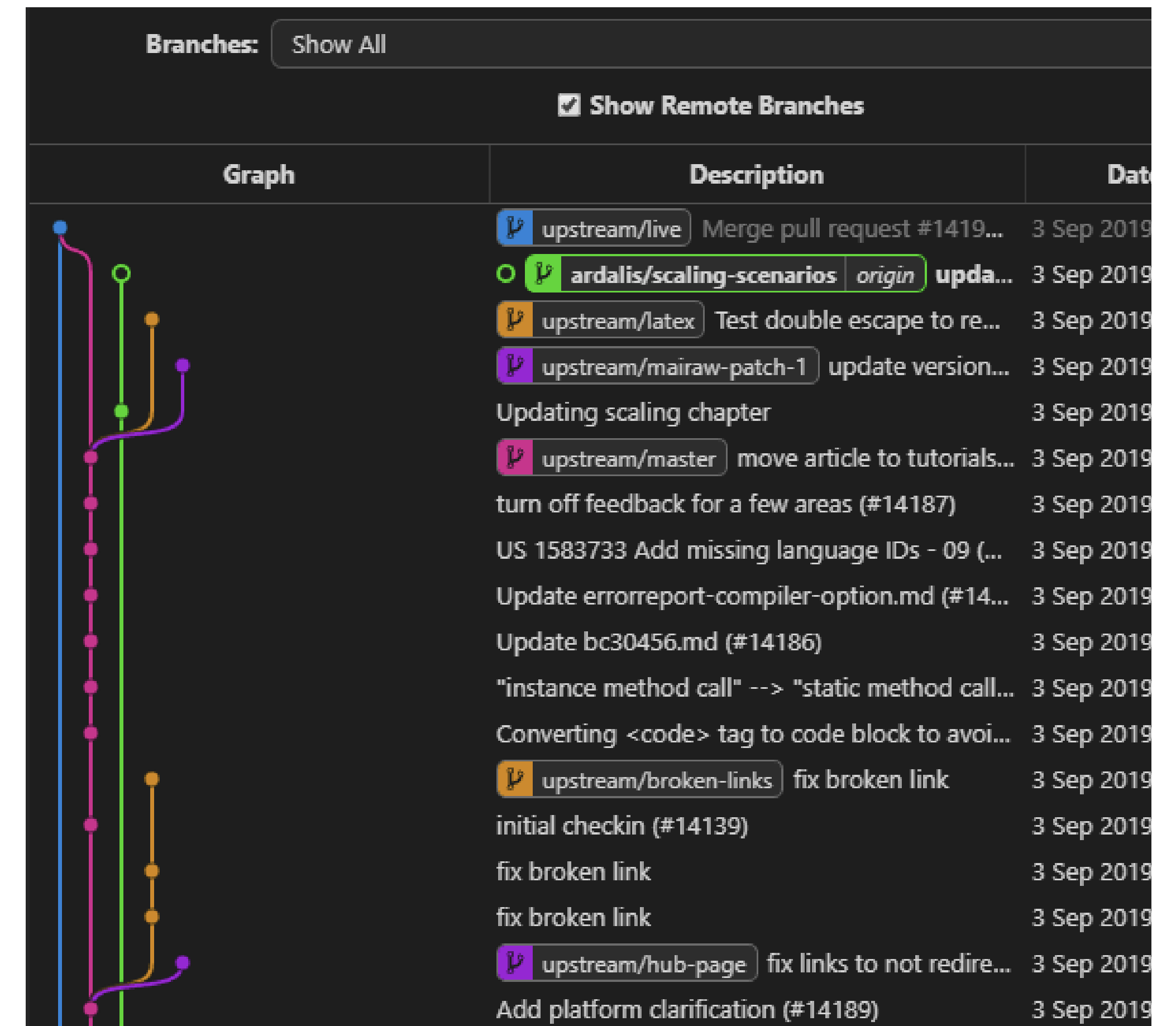
# Computergrafik 1

## Tutorium 1

### git Einführung

# Git Einführung

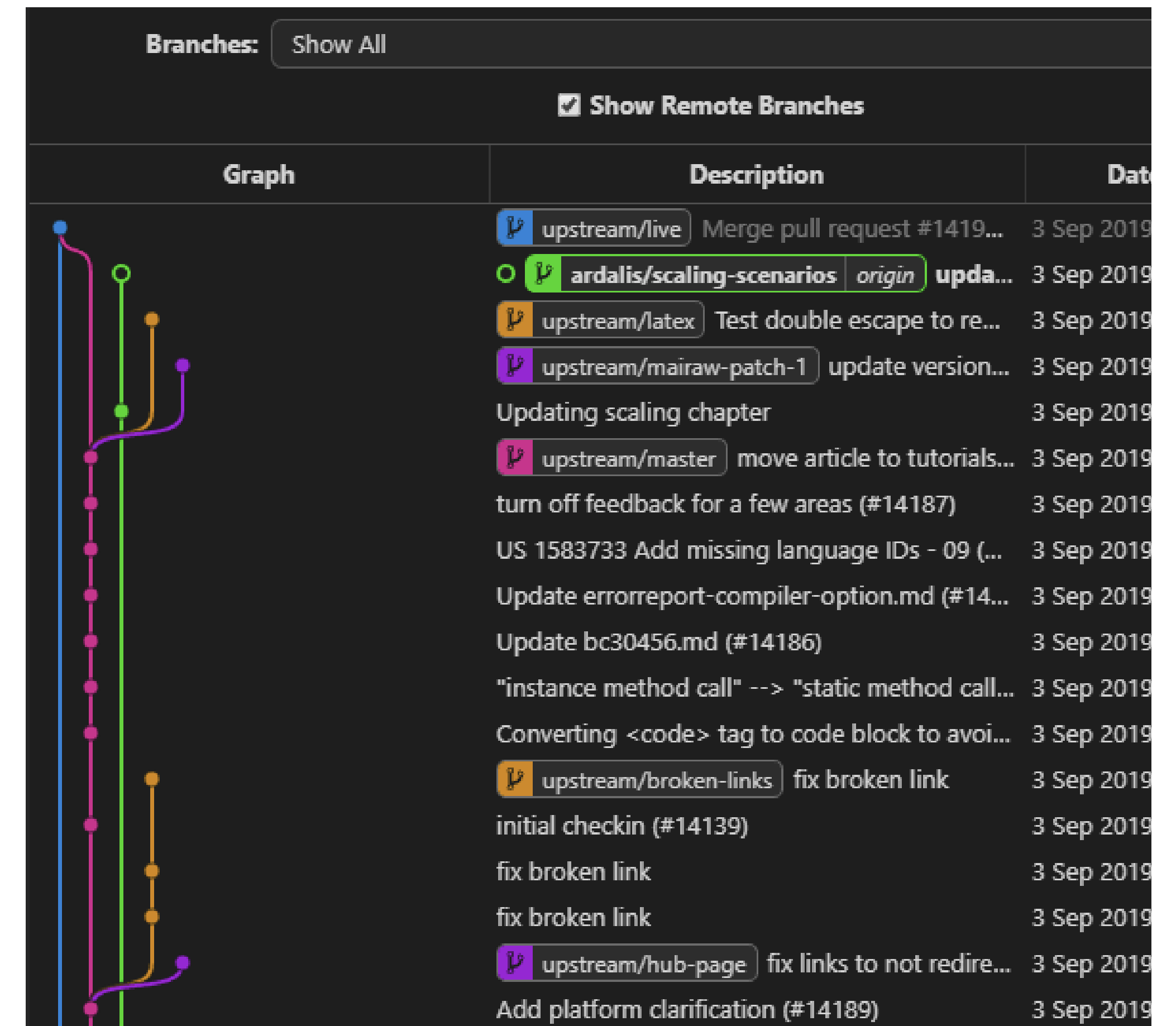
- Was ist git?
  - Versionskontrolltool (VCS) zum kooperativen Arbeiten an Software / Code
  - Codeänderungen können lokal oder auf einem Server / der Cloud gespeichert werden.
  - Änderungen können dann synchronisiert und integriert werden.



Quelle: [ardalis.com](https://ardalis.com)

# Git Einführung

- Wie nutzt man git?
  - Kann über das Terminal bedient werden.
  - Heutzutage haben alle Entwicklungsumgebungen git-Einbindung, also auch VS Code.
- Installation
  - Auf Linux & Mac i.d.R. vorinstalliert.
  - Auf Windows:
   
<https://git-scm.com/downloads/win>

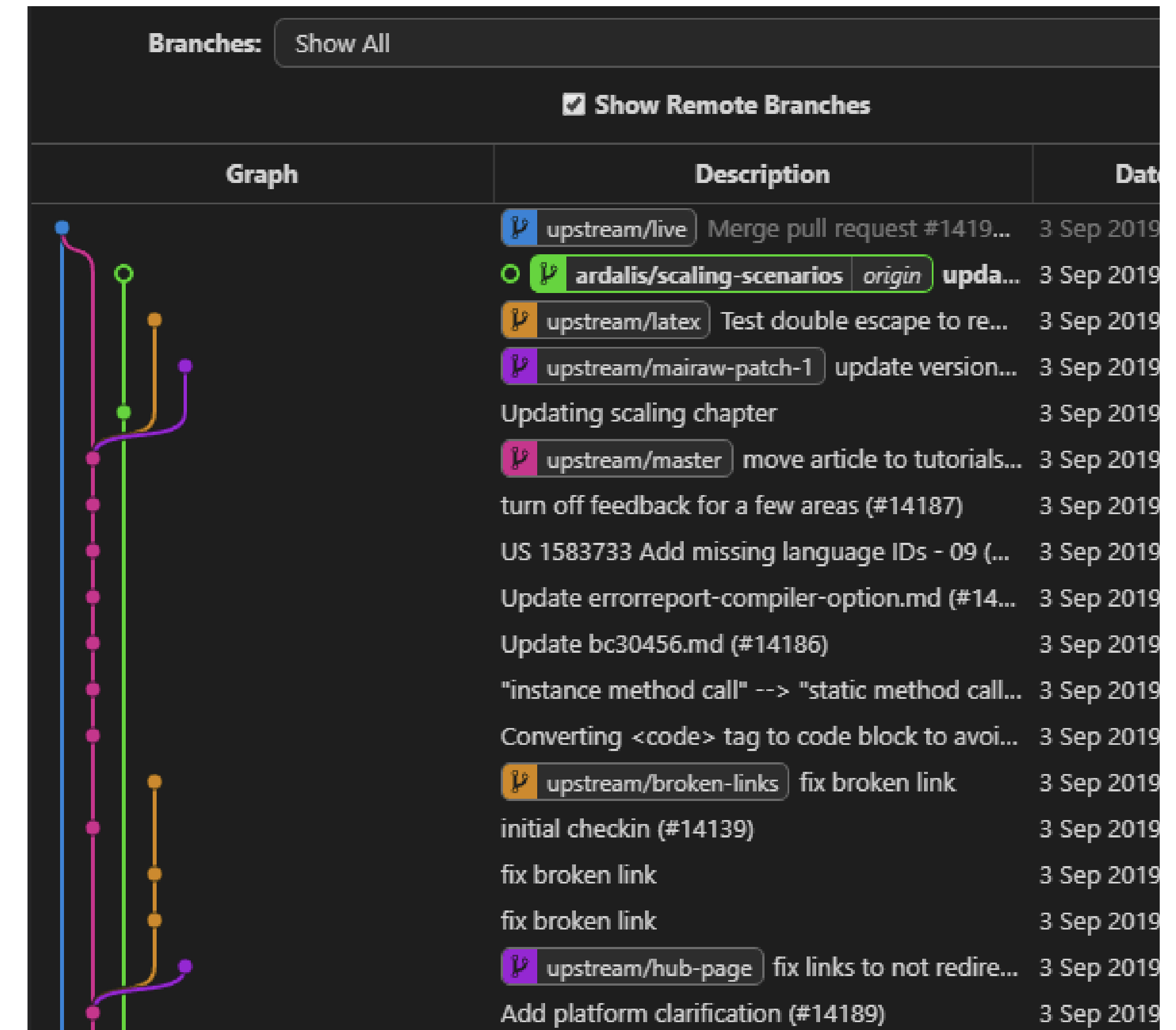


Quelle: [ardalis.com](https://ardalis.com)



# Git Einführung

- Grundlegende Begriffe
  - Repository
  - Commit
  - Branch
  - Staged Changes
  - Pull / Push
  - Clone
  - Merge



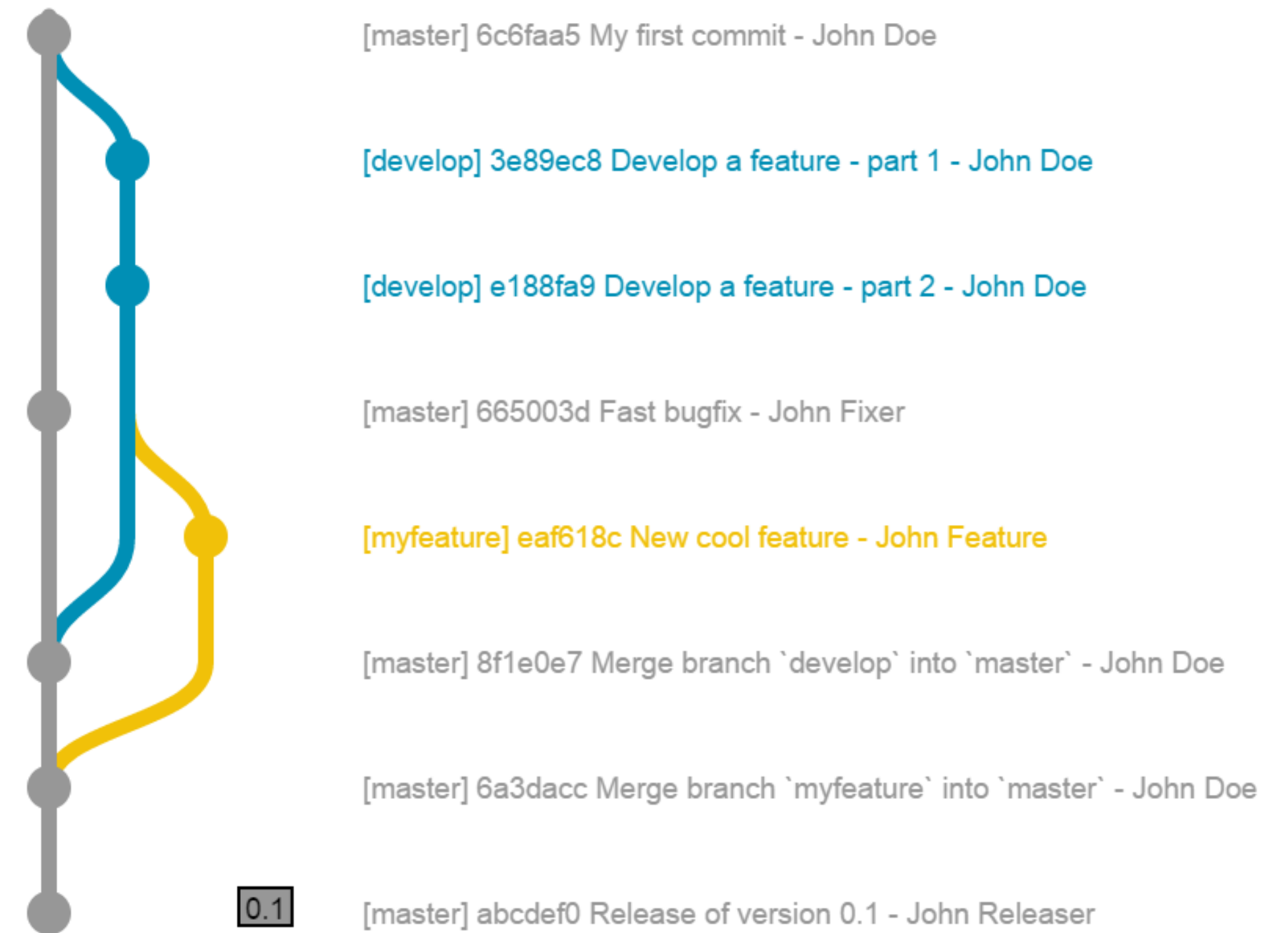
Quelle: [ardalis.com](https://ardalis.com)

# Interaktive Demonstration mit VS Code

- 1) Repository für Übungsaufgaben auf GitLab anlegen
- 2) Repository klonen (entweder per HTTPS oder SSH)
- 3) Vorgegebene Dateien dem Repo hinzufügen (eine Person per Gruppe)
- 4) Namen in `gruppenmitglieder.md` eintragen (eine Person per Gruppe)
- 5) Änderungen „stagen“ und mit einer Nachricht „committen“
- 6) Änderungen „pushen“
- 7) Restlichen Gruppenmitglieder sollen Änderungen „pullen“
- 8) Voila!

# Branches

- “Abzweigungen” vom aktuellen Stand
  - Warum? Um eigenständig an etwas zu arbeiten, ohne ständig externe Änderungen integrieren zu müssen.
- Rückintegration mit einem „Merge“
  - Erzeugt einen Merge-Commit.
  - Ggf. muss man Anpassungen vornehmen, im Falle eines...



Quelle: [GitHub/frappe/charts](https://github.com/frappe/charts)



# Merge-Konflikte

- Was wenn mehrere Personen an derselben Datei gearbeitet haben und in denselben Branch pushen / mergen?
- Es entsteht ein „Merge-Konflikt“.
  - Jetzt muss entschieden werden, welche Codeänderungen behalten werden, und welche nicht.
  - Im Zweifelsfall kann man den Merge immer abbrechen (`git merge --abort`).

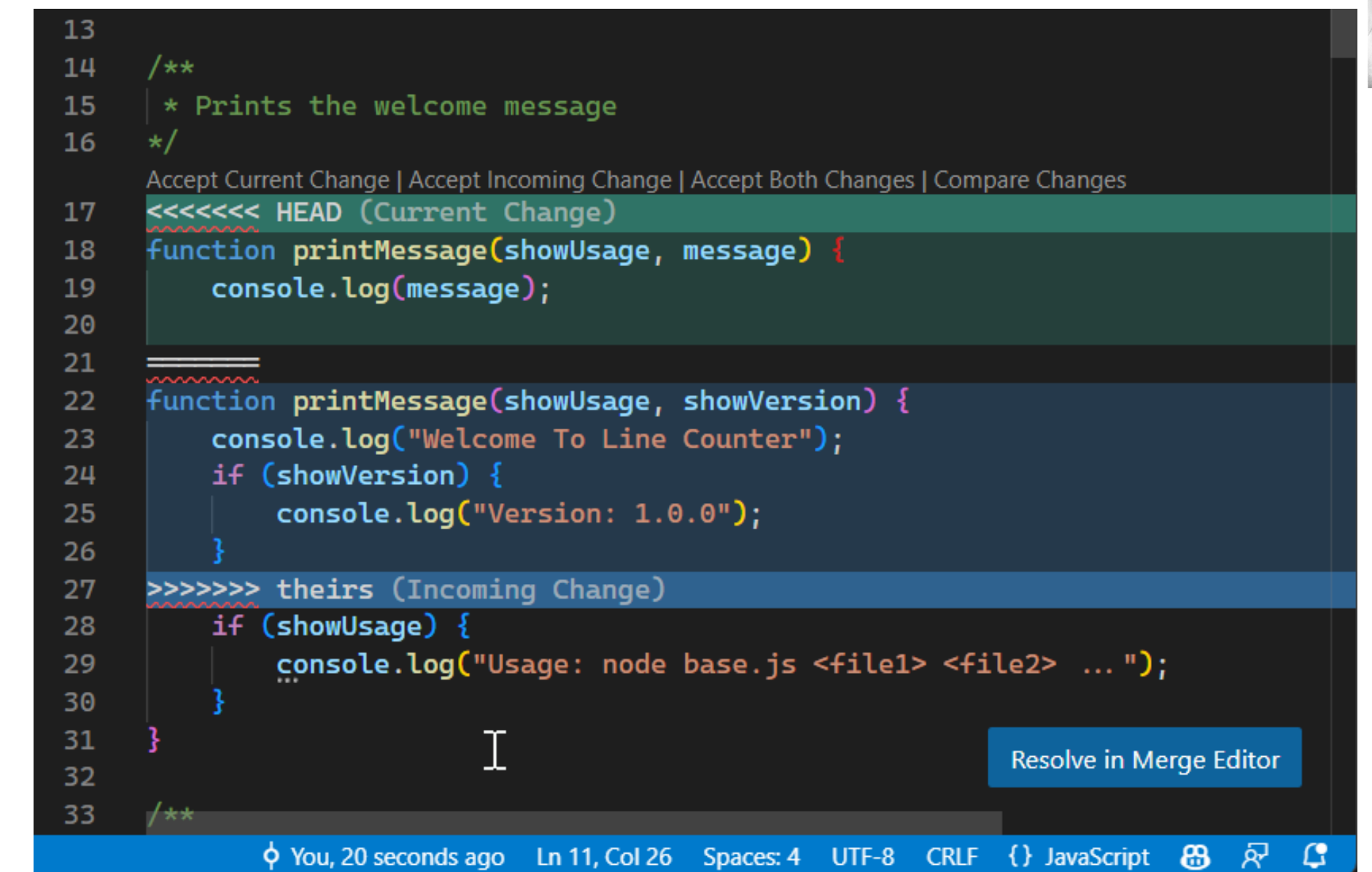
```
src > colors.txt
1 red
   Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
2 <<<<<< HEAD (Current Change)
3 green
4 =====
5 white
6 >>>>>> his-branch (Incoming Change)
7 blue

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
react-app-demo my-branch git merge his-branch
Auto-merging src/colors.txt
CONFLICT (content): Merge conflict in src/colors.txt
Automatic merge failed; fix conflicts and then commit the result.
x react-app-demo my-branch >M<
```

Quelle: [ihatetomatoes.net](https://ihatetomatoes.net)

# Merge-Konflikte

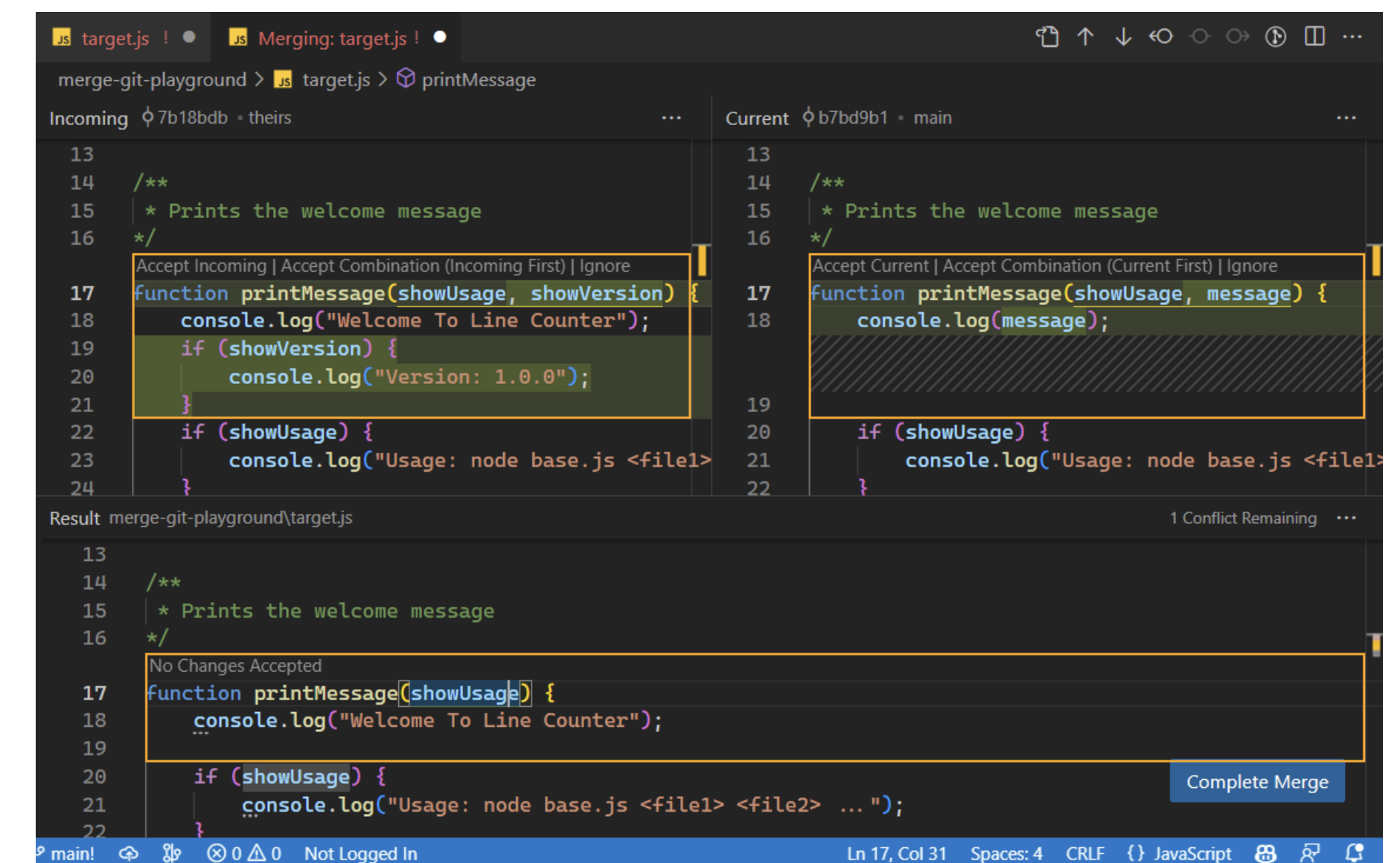
- VS Code hat einen eingebaute Lösungen.
- Inline Merge-Editor (oben)
  - Current Change: lokaler Stand
  - Incoming Change: einkommender Stand
  - Both: Kombination
- Three-Way Merge Editor (unten)
  - Links incoming, rechts current, unten Resultat.
- Optionen können per Click ausgewählt werden. Man kann auch direkt reinschreiben.



```

13
14 /**
15  * Prints the welcome message
16  */
17 <<<<<< HEAD (Current Change)
18 function printMessage(showUsage, message) {
19     console.log(message);
20 }
21
22 function printMessage(showUsage, showVersion) {
23     console.log("Welcome To Line Counter");
24     if (showVersion) {
25         console.log("Version: 1.0.0");
26     }
27 >>>>>> theirs (Incoming Change)
28 if (showUsage) {
29     console.log("Usage: node base.js <file1> <file2> ...");
30 }
31 }
32
33 /**

```



```

13
14 /**
15  * Prints the welcome message
16  */
17 function printMessage(showUsage, showVersion) {
18     console.log("Welcome To Line Counter");
19     if (showVersion) {
20         console.log("Version: 1.0.0");
21     }
22 }
23
24 if (showUsage) {
25     console.log("Usage: node base.js <file1> <file2> ...");
26 }

```

Quelle: [Microsoft](https://code.visualstudio.com/docs/sourcecontrol/merging)



# Zusätzliche Ressourcen

- Git Explained in 100 Seconds ([Youtube](#))
- Learn Git in 15 Minutes ([Youtube](#))
- A Grip On Git (<https://agripongit.vincenttunru.com/>)
  - kurze visuelle Einführung in Git
  - ~11 min read
- Learn Git Branching (<https://learngitbranching.js.org/>)
  - interaktives Tutorial für git im Webbrowser
  - ~30-60 min