

# **GIT**

Eine Einführung

Dominic Heun | 8. Mai 2019

AES ETTLINGEN - TGJ2/2

# Was ist GIT

Git als VCS

Einrichten einer Repository

Änderungen Speichern

Zusammenführen der Arbeit

Den Status der Repository anzeigen

Teilen von Arbeit

# Wie arbeitet man im Team mit eigenen Computern

- ▶ Synchronisation?

# Wie arbeitet man im Team mit eigenen Computern

- ▶ Synchronisation?
- ⇒ Fehleranfällig oder mehrere arbeiten gleichzeitig?

# Wie arbeitet man im Team mit eigenen Computern

- ▶ Synchronisation?
- ⇒ Fehleranfällig oder mehrere arbeiten gleichzeitig?
- ⇒ GIT löst Problem

# Versionskontrolle

- Verschiedene Versionen

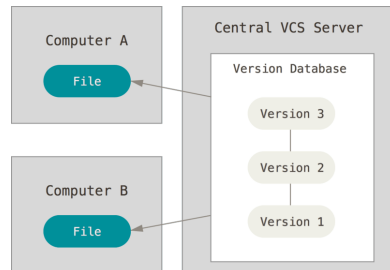


Abbildung: Versionskontrolle mit zentralem VCS Sever

[git-scm.com/book/en/v2/images/centralized.png](https://git-scm.com/book/en/v2/images/centralized.png);  
08.05.2018

# Versionskontrolle

- ▶ Verschiedene Versionen
- ▶ Jeder arbeitet an seinen *Versionen* der Dateien

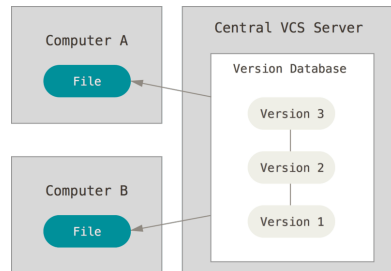


Abbildung: Versionskontrolle mit zentralem VCS Sever

[git-scm.com/book/en/v2/images/centralized.png](https://git-scm.com/book/en/v2/images/centralized.png)  
08.05.2018

# Versionskontrolle

- ▶ Verschiedene Versionen
- ▶ Jeder arbeitet an seinen *Versionen* der Dateien
- ▶ **Commit** gibt die Information weiter

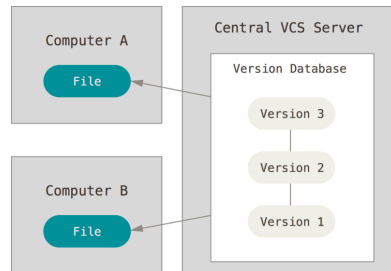


Abbildung: Versionskontrolle mit zentralem VCS Server

[git-scm.com/book/en/v2/images/centralized.png](https://git-scm.com/book/en/v2/images/centralized.png)  
08.05.2018



# Versionskontrolle

- ▶ Verschiedene Versionen
- ▶ Jeder arbeitet an seinen *Versionen* der Dateien
- ▶ **Commit** gibt die Information weiter
- ▶ **Merge** fügt die verschiedenen Versionen zusammen

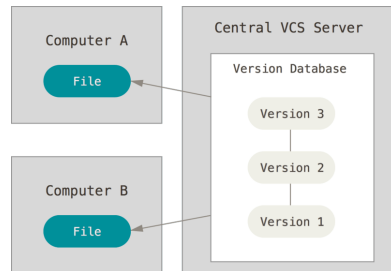


Abbildung: Versionskontrolle mit zentralem VCS Sever

[git-scm.com/book/en/v2/images/centralized.png](https://git-scm.com/book/en/v2/images/centralized.png);  
08.05.2018

Was ist GIT

**Git als VCS**

Einrichten einer Repository

Änderungen Speichern

Zusammenführen der Arbeit

Den Status der Repository anzeigen

Teilen von Arbeit

# Git

- Eine Möglichkeit unter vielen (*SVN, Git, Mercurial...*)



# Git

- ▶ Eine Möglichkeit unter vielen (*SVN, Git, Mercurial...*)
- ▶ Entwickelt von Linus Torwalds für den **Linux Kernel**



Was ist GIT

Git als VCS

Einrichten einer Repository

Änderungen Speichern

Zusammenführen der Arbeit

Den Status der Repository anzeigen

Teilen von Arbeit

# Clone

- Repository laden
- Verschiedene Anbieter Möglich
  - GitLab
  - GitHub
  - Bitbucket

## Command

```
git clone <url>
```

# Clone

- Repository laden
- Verschiedene Anbieter Möglich
  - GitLab
  - GitHub
  - Bitbucket

## Eine Repository Clonen

`https://github.com/TheDome/githubPresentation2019.git`

# Init

- Repository erstellen

## Command

```
git init
```



Was ist GIT

Git als VCS

Einrichten einer Repository

**Änderungen Speichern**

Zusammenführen der Arbeit

Den Status der Repository anzeigen

Teilen von Arbeit

# Add

- Dateien *Stagen*

## Command

```
git add files..
```

# Add

- ▶ Dateien *Stagen*
- Auf commit vorbereiten

## Command

```
git add files..
```

# Add

- ▶ Dateien *Stagen*
- Auf commit vorbereiten
- ▶ Mehrere Dateien nacheinander für einen Commit vorbereiten

## Command

```
git add files..
```

# Commit

- ▶ Hinzufügen von Änderungen
- ▶ Lädt Änderung direkt in den GIT Baum

## Command

```
git commit -m «Message» files..
```

# Remove

- ▶ Dateien mit GIT löschen
- ▶ Eigentlich auch durch löschen und commit möglich
- ▶ Zeichnet direkt in dem Versionsbaum auf

## Command

```
git rm [-r] [-] «file»...
```

# Remove

- Dateien mit GIT löschen
  - Eigentlich auch durch löschen und commit möglich
  - Zeichnet direkt in dem Versionsbaum auf
- r Nimmt hierbei **alle** Dateien in einem Verzeichnis

## Command

```
git rm [-r] [-] «file»...
```

# Differences

- Änderungen gar nicht mehr bewusst

## Command

```
git diff «file»
```



# Differences

- Änderungen gar nicht mehr bewusst
- Durchsuchen

## Command

```
git diff «file»
```

# Differences

- Änderungen gar nicht mehr bewusst
- Durchsuchen
- Wissen, wann Änderung gemacht wurde

## Command

```
git diff «file»
```

# Differences

- ▶ Änderungen gar nicht mehr bewusst
- ▶ Durchsuchen
- ▶ Wissen, wann Änderung gemacht wurde

## Command

```
git diff «file»
```

## Output

```
diff -git a/presentation.pdf
b/presentation.pdf
index ab4f937..8df41aa 100644
Binary files a/presentation.pdf and
b/presentation.pdf differ
diff -git a/presentation.tex
b/presentation.tex
index c48b0c4..709be6b 100644
-- a/presentation.tex
+++ b/presentation.tex
@@ -222,7 +222,7 @@
```

Was ist GIT

Git als VCS

Einrichten einer Repository

Änderungen Speichern

**Zusammenführen der Arbeit**

Den Status der Repository anzeigen

Teilen von Arbeit

# Branches

- *Zweige*

Bilder von [git-scm.com/book/en/v2/Git-Branching-Branches-in-a-Nutshell](https://git-scm.com/book/en/v2/Git-Branching-Branches-in-a-Nutshell)

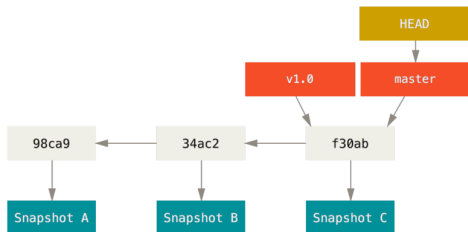
# Branches

- *Zweige*
- Möglichkeit, verschiedene **Versionen** zu haben

Bilder von [git-scm.com/book/en/v2/Git-Branching-Branches-in-a-Nutshell](https://git-scm.com/book/en/v2/Git-Branching-Branches-in-a-Nutshell)

# Branches

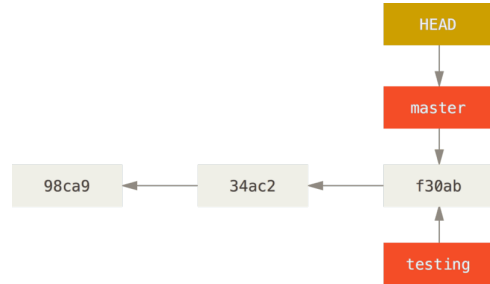
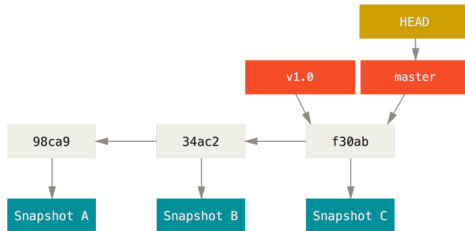
- *Zweige*
- Möglichkeit, verschiedene **Versionen** zu haben



Bilder von [git-scm.com/book/en/v2/Git-Branching-Branches-in-a-Nutshell](https://git-scm.com/book/en/v2/Git-Branching-Branches-in-a-Nutshell)

# Branches

- *Zweige*
- Möglichkeit, verschiedene **Versionen** zu haben



Bilder von [git-scm.com/book/en/v2/Git-Branching-Branches-in-a-Nutshell](https://git-scm.com/book/en/v2/Git-Branching-Branches-in-a-Nutshell)



# Branches anzeigen

- Anzeigen aller Branches

# Branches anzeigen

- Anzeigen aller Branches

## Command

```
git branch
```

# Branches anzeigen

- Anzeigen aller Branches

## Command

```
git branch
```

## Ausgabe

```
$ git branch
```

# Branches anzeigen

- Anzeigen aller Branches

## Command

```
git branch
```

## Ausgabe

```
$ git branch  
iss53  
* master  
testing
```

# Branches Zusammenführen

- Verschiedene Versionen

# Branches Zusammenführen

- Verschiedene Versionen
- Man will auf eine Version

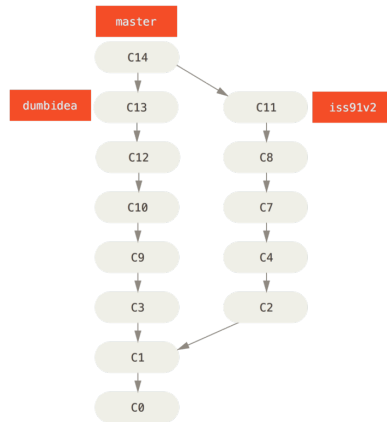


Abbildung: Verschiedene Branches

[git-scm.com/book/en/v2/Git-Branching-Branching-Workflows](https://git-scm.com/book/en/v2/Git-Branching-Branching-Workflows),  
08.05.2019

# Merge

- Branches Zusammenführen

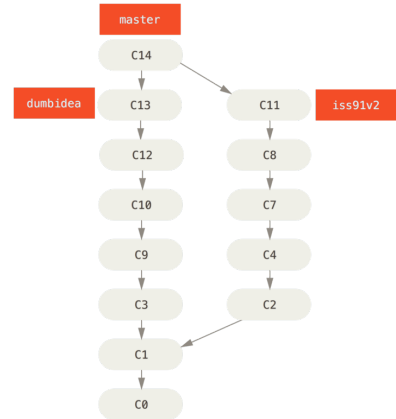


Abbildung: Verschiedene Branches

[git-scm.com/book/en/v2/](https://git-scm.com/book/en/v2/)

Git-Branching-Branching-Workflows, 08.05.2019

# Merge

- Branches Zusammenführen
- Manche haben Konflikte

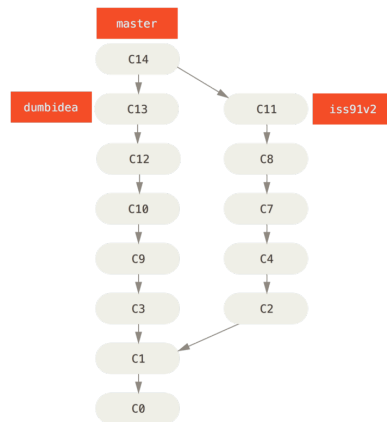


Abbildung: Verschiedene Branches

[git-scm.com/book/en/v2/](https://git-scm.com/book/en/v2/)

Git-Branching-Branching-Workflows, 08.05.2019



# Merge

- Branches Zusammenführen
  - Manche haben Konflikte
- ⇒ Selbst lösen

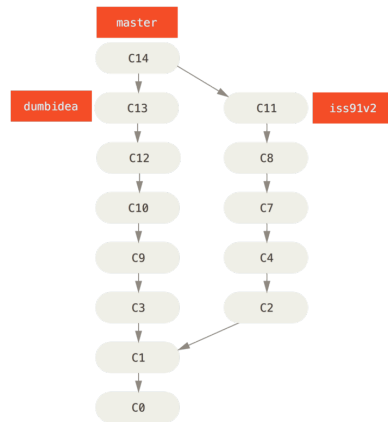


Abbildung: Verschiedene Branches

[git-scm.com/book/en/v2/](https://git-scm.com/book/en/v2/)

Git-Branching-Branching-Workflows, 08.05.2019

# Merge

- Branches Zusammenführen
  - Manche haben Konflikte
- ⇒ Selbst lösen

## Command

```
git merge [<commit>...]
```

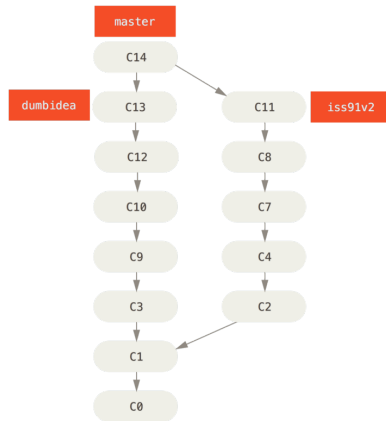


Abbildung: Verschiedene Branches

[git-scm.com/book/en/v2/](https://git-scm.com/book/en/v2/)

Git-Branching-Branching-Workflows, 08.05.2019

Was ist GIT

Git als VCS

Einrichten einer Repository

Änderungen Speichern

Zusammenführen der Arbeit

**Den Status der Repository anzeigen**

Teilen von Arbeit

# Status

- ▶ Keinen überblick über Dateien, die schon committed sind

# Status

- ▶ Keinen überblick über Dateien, die schon committed sind
- ▶ git weiß es

# Status

- ▶ Keinen Überblick über Dateien, die schon committed sind
- ▶ git weiß es

## Command

```
git status
```

# Log - Die Historie

- ▶ Keinen überblick über Dateien, die schon committed sind

# Log - Die Historie

- ▶ Keinen überblick über Dateien, die schon committed sind
- ▶ git weiß es



# Log - Die Historie

- Keinen Überblick über Dateien, die schon committed sind
- git weiß es

## Command

```
git log
```

# Log - Die Historie

- Keinen Überblick über Dateien, die schon committed sind
- git weiß es

## Command

```
git log
```

## Command

```
commit c8765b34740f5a9bc0bc0be531474d32ce3548b0 (HEAD -> master)
Author: TheDome <TheDome@users.noreply.github.com>
Date: Wed May 8 23:04:40 2019 +0200
```

Add branches

Was ist GIT

Git als VCS

Einrichten einer Repository

Änderungen Speichern

Zusammenführen der Arbeit

Den Status der Repository anzeigen

Teilen von Arbeit

# push

- ▶ Lokal gespeichert im Baum

# push

- ▶ Lokal gespeichert im Baum
- ▶ Mit mehreren Teilen
- ▶ Auf einen Server hosten

# push

- ▶ Lokal gespeichert im Baum
- ▶ Mit mehreren Teilen
- ▶ Auf einen Server hosten
- ▶ *GitHub, GitLab, Bitbucket*

# push

- ▶ Lokal gespeichert im Baum
- ▶ Mit mehreren Teilen
- ▶ Auf einen Server hosten
- ▶ *GitHub, GitLab, Bitbucket*

## Command

```
git push
```

# push

- ▶ Lokal gespeichert im Baum
- ▶ Mit mehreren Teilen
- ▶ Auf einen Server hosten
- ▶ *GitHub, GitLab, Bitbucket*

Abbildung: Zentral gehostete git  
Repository

## Command

```
git push
```