

# PRÄSENTATION HERUNTERLADEN

- <https://github.com/marcoribi/git-training>



semabit

# THE GIT WAY

Marco Ribí

# AGENDA

- **Ziele**
- **Git – Die Lösung aller Probleme?**
- **Github (Hosting)**
- **Vorteile von Git**
- **Tägliche Arbeit in der Software-Entwicklung verbessern**

# ZIEL DES REFERATS

- Grundfunktionalität eines Version Control System (VCS)
- Was bietet Git (anhand von Beispielen)
- Was ist Github und mögliche Alternativen
- Verbesserung von Entwicklungs-Abläufen

# AGENDA

- Ziele
- **Git – Die Lösung aller Probleme?**
- Github (Hosting)
- Vorteile von Git
- Tägliche Arbeit in der Software-Entwicklung verbessern

# GIT – DIE LÖSUNG ALLER PROBLEME?

## Was ist ein Version Control System?

- Erfassung von Änderungen an Dokumenten (Dateien)
- Versionierung
- Vereinfacht zeitgleiches Arbeiten an einem Projekt
- Jede Änderung erhält Zeitstempel und Benutzerkennung
- Wiederherstellen von Änderungen

# GIT – DIE LÖSUNG ALLER PROBLEME?

## Geschichte

- Erfunden von Linus Torvalds
- git heisst umgangssprachlich „Blödmann“
- Vorhandene Systeme (BitKeeper) wurden plötzlich kostenpflichtig

# GIT – DIE LÖSUNG ALLER PROBLEME?

## Unterschiede zu anderen VCS:

- Nicht zentralisiert Ablage der Daten
- Einfache Bedienung
- Kein reines VCS (z.B. Deployment)



# GIT – DIE LÖSUNG ALLER PROBLEME?

## Repository

- Gefäß, in dem alle Daten gespeichert und Änderungen aufgezeichnet werden
- Besteht aus: Working Tree, Index und HEAD
- `git init <project_name> | git clone <url>`

# GIT – DIE LÖSUNG ALLER PROBLEME?

## Working Tree

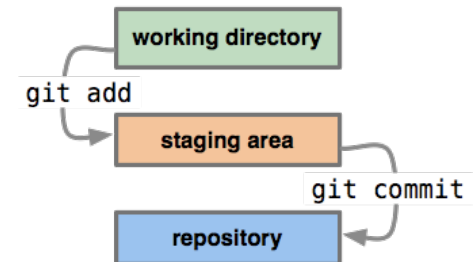
- Aktuelle Änderungen, die noch nicht „committed“ wurden
- `git status`

## Index (Staging Area)

- Hinzufügen der Änderungen in die Staging-Area
- Dient dazu, Änderungen an verschiedenen Dateien zu sammeln
- `git add <file|directory>`

## HEAD

- Referenz zum letzten Commit der aktuellen Umgebung
- Eindeutige Zusammenfassung von Änderungen, die im Repository hinterlegt (eingescheckt) sind
- `git commit -m"initial commit"`



# GIT – DIE LÖSUNG ALLER PROBLEME?

## Remotes

- Es wird unterschieden zwischen lokalen und remote Repo's
- Distributed => Verteiltes VCS
- Ermöglicht die Definition von entfernten Repositories
- Beispiel Github (Hosting von Git-Repositories)
- `git remote add <alias-remote> <url>`

## Auszug von definierten Remotes:

`origin git@github.com:semabit/tima.git`

`origin git@github.com:semabit/tima.git`

# GIT – DIE LÖSUNG ALLER PROBLEME?

## Branching

- Möglichkeit, eine Baumstruktur mit Branches (Äste) aufzubauen
- Funktionalitäten können in separaten Entwicklungszweigen isoliert voneinander entwickelt werden
- Bei einem Branch-Wechsel wird der Working Tree komplett ausgetauscht
- Standard-Branch heisst „Master“
- `git checkout -b <branch-name>`

```
db9dc3b175 ○ Merge branch 'master' of github.com:screenconcept/mobilezone-pricetag
cc4e815407 ○ extend manually import
3d63faddf7 ○ describe how resque-web works
1b6982bbf4 ○ describe offerpattern logic in readme
0e1882f501 ○ Fix: mysql server has gone away
85e64c28ea ○ Merge branch 'master' of github.com:screenconcept/mobilezone-pricetag
990639d1f0 ○ mount resque server as subpath
0b6c7e3d8f ○ add helping comment
201ed70dd9 ○ fix
75a27d141e ○ reactivate yallo single template
73b847dc2d ○ add 4 provider template, remove 5 provider and yallo
cdec148e31 ○ use rack 1.2.2
```

# GIT – DIE LÖSUNG ALLER PROBLEME?

## Merging

- Isolierte Funktionalität zusammenfügen (mergen)
- Allfällige Konflikte lösen  
*System sagt, was nicht zusammengeführt werden konnte*
- `git merge <branch-name>`

# GIT – DIE LÖSUNG ALLER PROBLEME?

## Veröffentlichen von Änderungen

- Änderungen auf anderen Remotes verteilen (pushen)
- `git push <remote-name> <branch-name>`

## Aktualisieren des lokalen Repositories

- Änderungen von anderen Remotes holen (pullen)
- `git pull <remote-name> <branch-name>`

# AGENDA

- Ziele
- Git – Die Lösung aller Probleme?
- **Github (Hosting)**
- Vorteile von Git
- Tägliche Arbeit in der Software-Entwicklung verbessern

# GITHUB (HOSTING)

- Webbasierter Hosting-Dienst für Git-Repositories
- Einrichten von Remote Repo's (private / public for free)
- Hauptfokus auf Zusammenarbeit in einem Projekt (Forking, Pull Request, Wiki)
- Help-Center: [help.github.com](https://help.github.com) für alle Fragen
- Mögliche Alternative:  
GitLab  
Bitbucket



# AGENDA

- Ziele
- Git – Die Lösung aller Probleme?
- Github (Hosting)
- **Vorteile von Git**
- Tägliche Arbeit in der Software-Entwicklung verbessern

# VORTEILE VON GIT

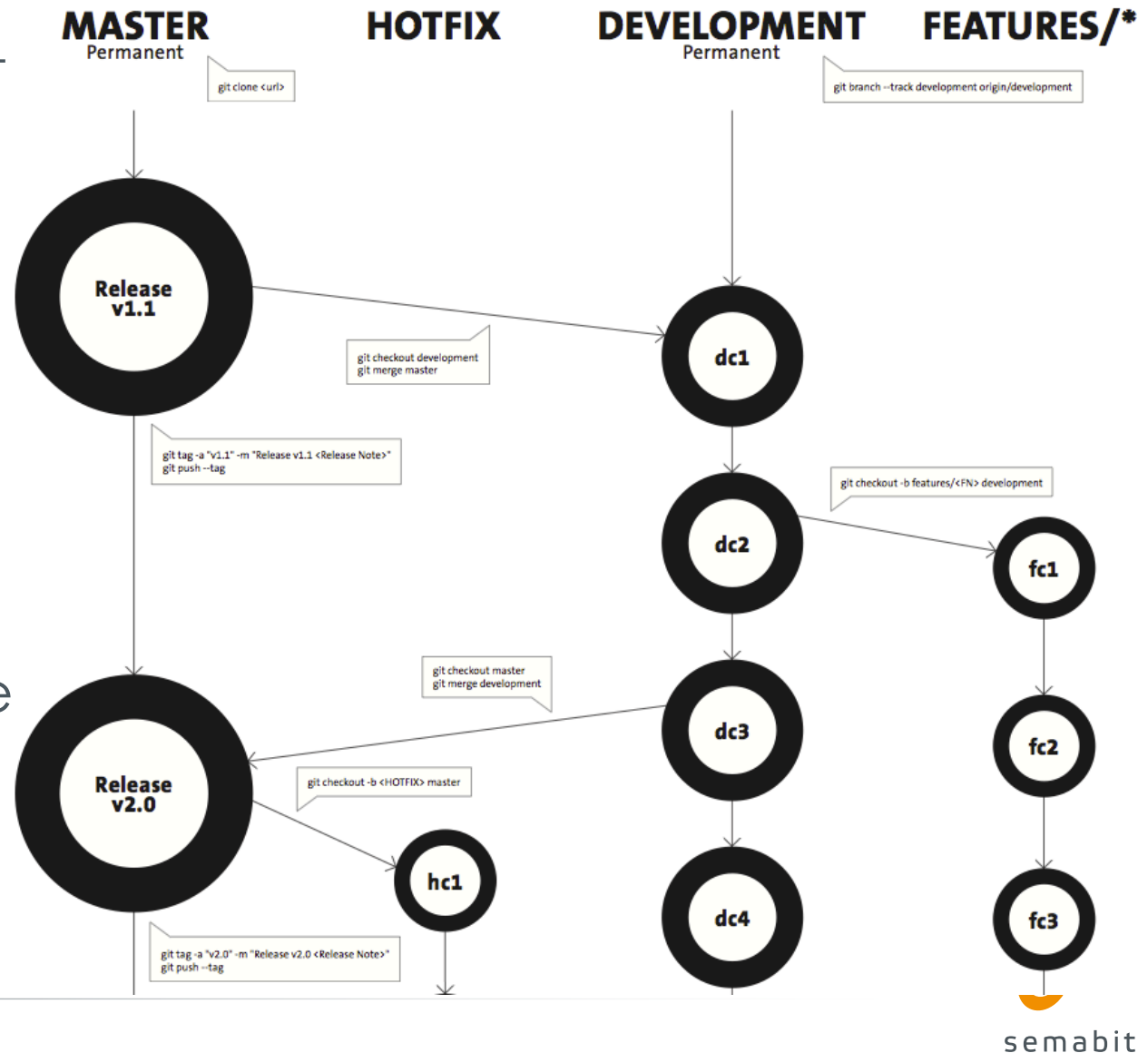
- Gleichzeitiges Arbeiten am gleichen Projekt
- Sicherung und Wiederherstellung von versionierten Dateien
- Gestaltung von Entwicklungsprozessen (Abläufe)
- Deployment über git

# AGENDA

- Ziele
- Git – Die Lösung aller Probleme?
- Github (Hosting)
- Vorteile von Git
- **Abläufe in der Software-Entwicklung verbessern**

# ABLÄUFE MIT GITHUB

- Zusammenarbeit im Team kann schwierig sein
- Mangelnde Code-Qualität
- Hilfe bei Code-Reviews
- Immer lauffähige Version verfügbar



# GUIDES / INSTALLATION

- <http://rogerdudler.github.com/git-guide>
- <http://gitimmersion.com>
- Help auf Github benutzen

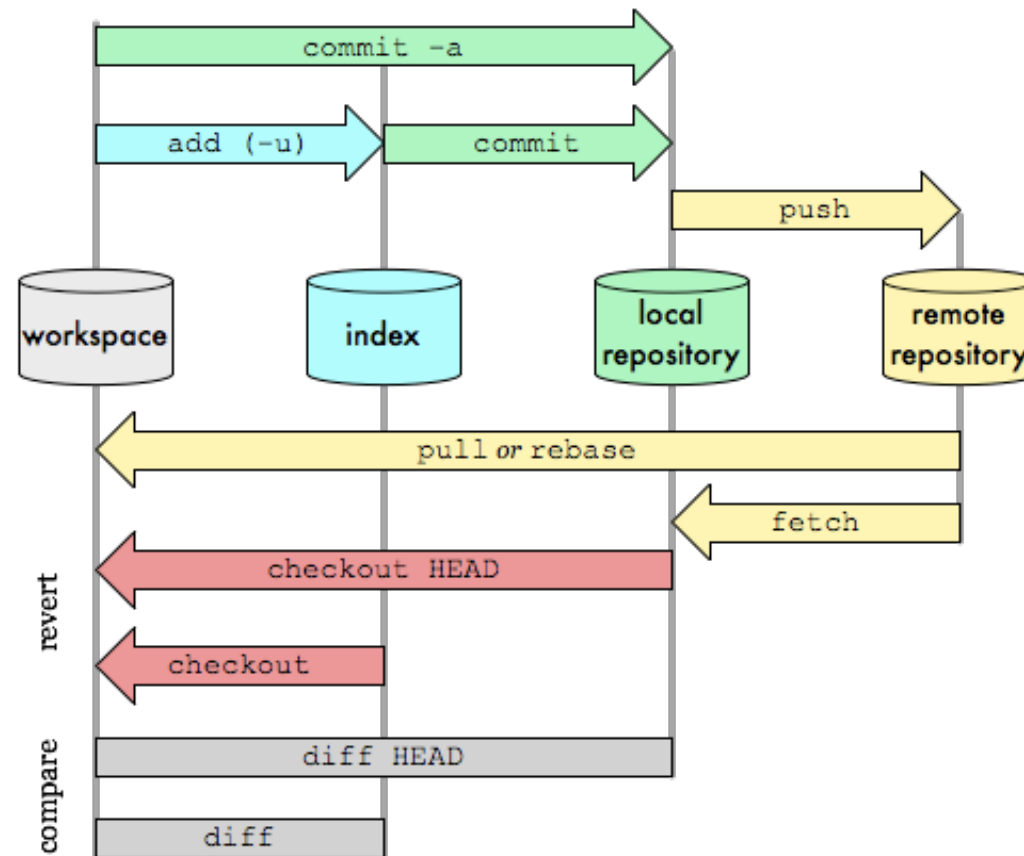
# TOOLS

- <https://desktop.github.com>
- <https://www.sourcetreeapp.com>
- <http://gitx.frim.nl>
- <https://www.git-tower.com/mac> (kostenpflichtig)

# SUMMARY

## Git Data Transport Commands

<http://osteele.com>



# HERZLICHEN DANK!

**Git** – Ist es wirklich die Lösung aller Probleme?

**NEIN, aber ...**