

VERTEILTE VERSIONSKONTROLLE MIT GIT

JOHANNES SONN
CT



GLIEDERUNG

- Problemstellung
- Was ist ein Versionskontrollsystem (VCS)?
 - Lokale Versionskontrollsysteme
 - Zentralisierte Versionskontrollsysteme (CVCS)
 - Verteilte Versionskontrollsysteme (DVCS)
- Was ist Git?
- Geschichte von Git
- Die drei Zustände
- Demo
- Vergleich mit Alternativen
- Fazit



PROBLEMSTELLUNG

- Für ein erfolgreiches Projekt sollte man:
 - Überblick über den Quellcode behalten
 - Änderungen des Quellcodes nachvollziehen können
 - Auf ältere Versionen zurückgreifen & zurücksetzen können
 - Möglicherweise mit anderen kollaborieren



PROBLEMSTELLUNG

- Lokale Kopien mit Versionsnummer oder Datum anfertigen
 - Man kann auf ältere Versionen zurückgreifen
 - Sehr fehleranfällig & unübersichtlich
 - Verbraucht viel Speicher
 - Nicht zum Kollaborieren geeignet



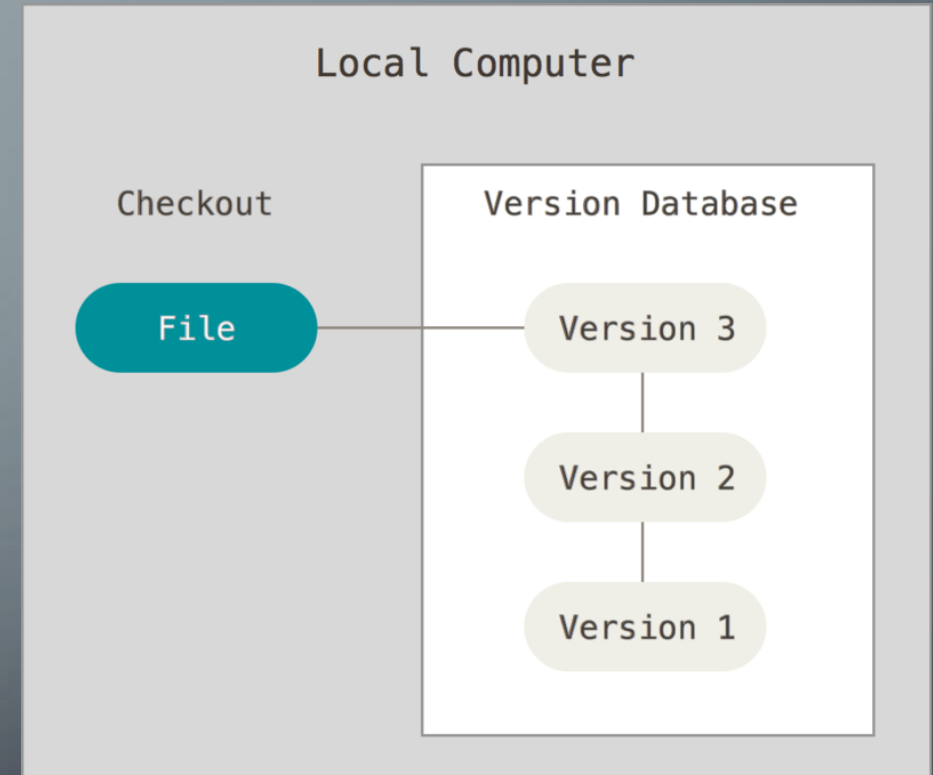
WAS IST EIN VERSIONSKONTROLLSYSTEM (VCS)?

- Protokolliert Änderungen von Dateien (Beispielsweise Quellcode) über die Zeit hinweg
- Bietet die Möglichkeit auf ältere Versionen zurückzugreifen
- Man kann nachvollziehen welche Änderung wann & durch welche Person vorgenommen wurde
 - Kann bei Fehlersuche helfen



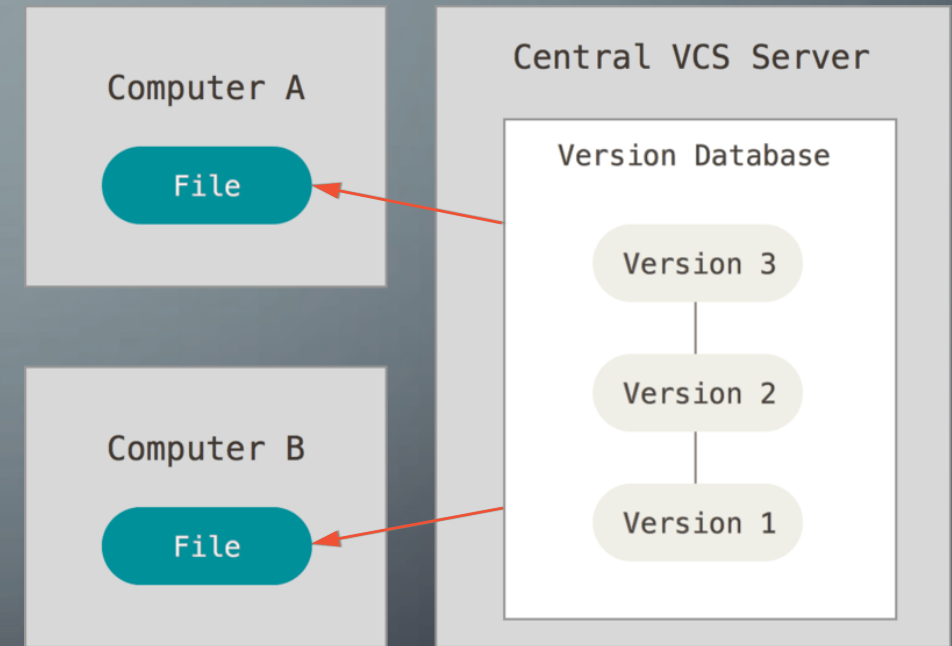
LOKALE VERSIONSKONTROLLSYSTEME

- Speichert den Änderungsverlauf in eine separate lokale Datei
- Eines der populärsten lokalen VCS war *RCS*
 - Es wird heutzutage immer noch mit vielen PCs ausgeliefert
- Probleme:
 - Umständlich für Zusammenarbeiten
 - Kann nicht zur Verwaltung ganzer Projekte genutzt werden
 - Ohne Backup besteht ein „Single Point of Failure“



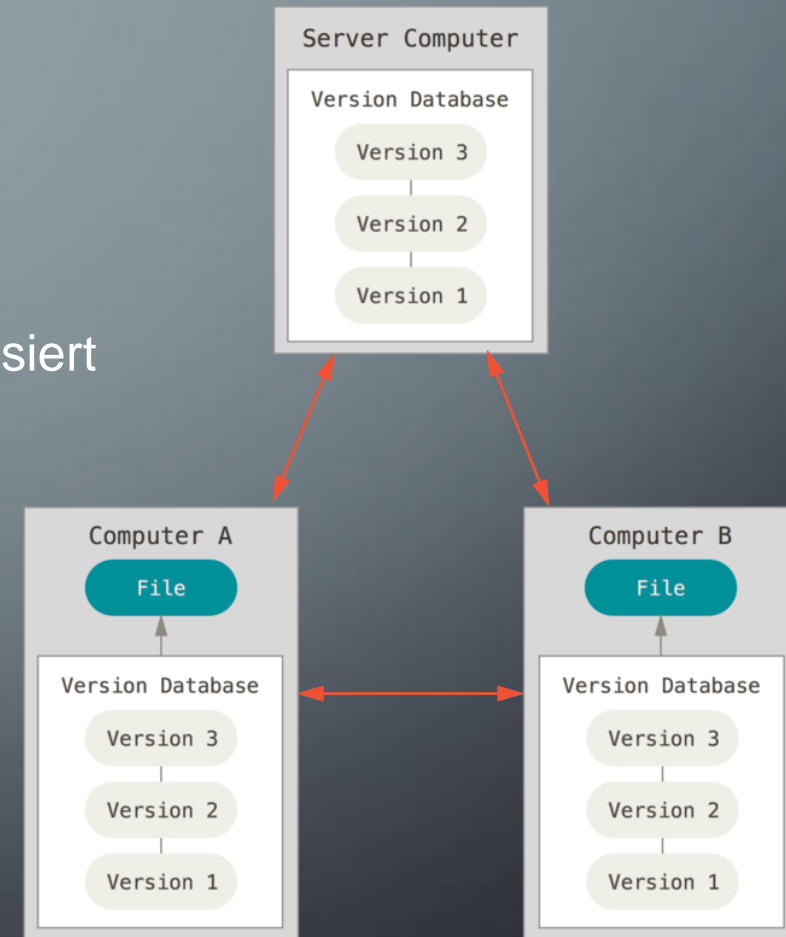
ZENTRALISIERTE VERSIONSKONTROLLSYSTEME (CVCS)

- Ermöglicht Zusammenarbeit
- Zentraler Server wird benötigt
 - Verwaltet alle versionierten Dateien
 - Viele Clients können die Dateien, zum Bearbeiten, vom Server *abholen* (Checkout)
 - Nach Bearbeitung werden die Änderungen wieder beim Server *abgeliefert* (Checkin)
- Leicht zu administrieren
- Beispiele: Subversion, CVS, SCCS
- Probleme
 - Wenn Server nicht erreichbar ist, kann niemand mit anderen arbeiten oder neue Versionen speichern
 - „Single Point of Failure“



VERTEILTE VERSIONSKONTROLLSYSTEME (DVCS)

- Jeder erhält eine vollständige Kopie des Repositories (Projekt / Versionsdatenbank)
 - Redundanz, durch mehrere Anwenderrechner
- Jedes Repository kann mit jedem anderem synchronisiert werden
- Dateien können von mehreren Leuten gleichzeitig bearbeitet werden
- Arbeiten auch ohne Verbindung zum Server möglich
- Höhere Geschwindigkeit, da alle Dateien lokal vorhanden sind
- Beispiele: Git, BitKeeper, Bazaar



WAS IST GIT?

- Verteiltes Versionskontrollsystem
 - Kein Server notwendig
- Kostenlos & Open Source (github.com/git/git)
- Wurde von Linus Torvalds (Erfinder des Linux-Kernels) initiiert
 - Aus Notwendigkeit, zur Entwicklung des Linux-Kernels
- Ermöglicht Kollaborieren bei nicht-linearem Workflow



<http://bit.ly/1LXEzKC>
<http://bit.ly/2ugC6qt>

Verteilte Versionsverwaltung mit Git
Johannes Sonn



GESCHICHTE VON GIT

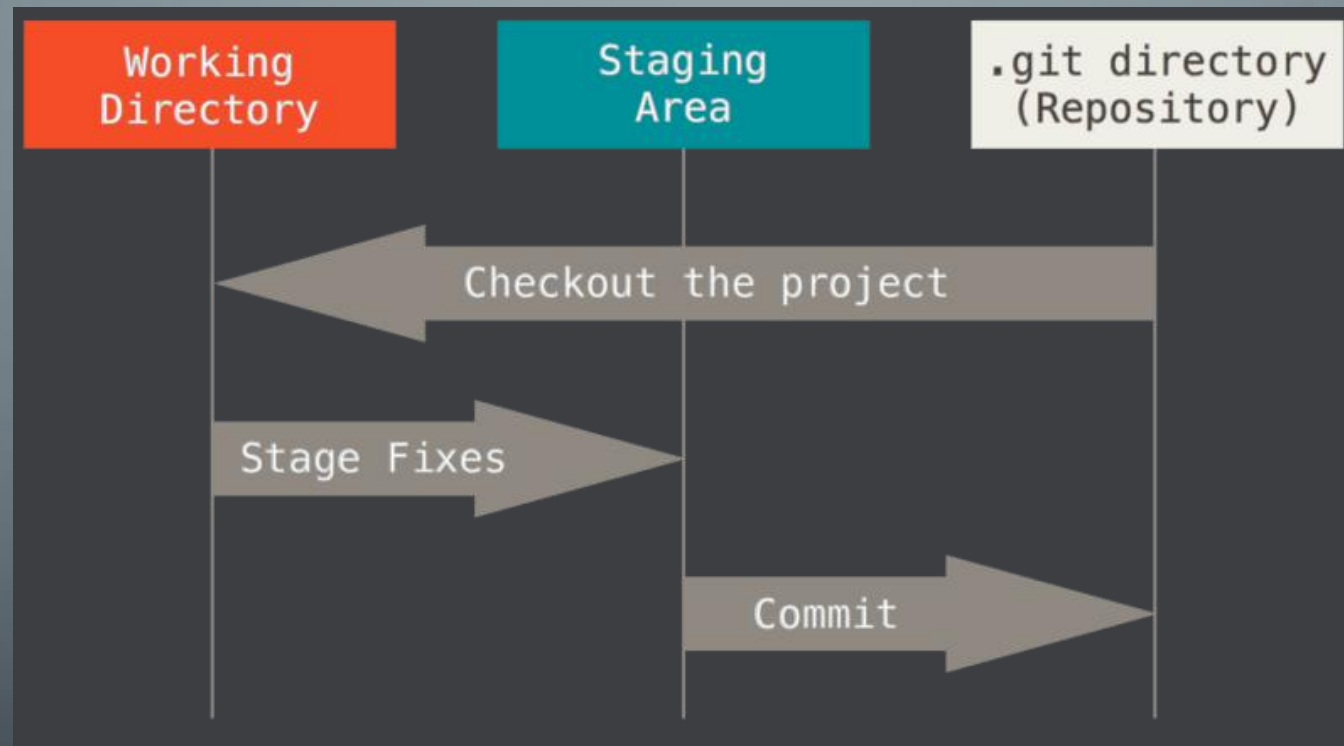
- Zur Entwicklung des Linux-Kernels wurde BitKeeper verwendet
 - Lizenzänderung von BitKeeper → nicht mehr kostenlos
 - Andere Systeme waren nicht effizient genug
- Ziele:
 - Hohe Geschwindigkeit → Alle Dateien lokal vorhanden
 - Einfaches Design
 - Vollständig verteilt
 - Fähigkeit große Projekte zu verwalten (beispielsweise den Linux-Kernel)
 - Sicherheit gegen Verfälschung (Checksummen: SHA-1) <https://shattered.io/>
 - BitKeeper-ähnliche Abläufe → Leichter Umstieg

<http://bit.ly/2gRvxH1>
<http://bit.ly/2ugC6qt>

Verteilte Versionsverwaltung mit Git
Johannes Sonn



DIE DREI ZUSTÄNDE



DEMO

Verteilte Versionsverwaltung mit Git
Johannes Sonn



VERGLEICH MIT ALTERNATIVEN

VCS / Kriterium	VCS-Typen	OS	Server nötig	Kollaboration möglich	Open-Source	Check-summe
RCS	Lokales VCS	?	✗	✗	✓	✗
Subversion	CVCS	Win/Mac/Linux/...	✓	✓	✓	✓
Perforce	CVCS	Win/Mac/Linux/...	?	✓	✗	✓ (MD5)
Git	DVCS	Win/Mac/Linux	✗	✓	✓	✓ (SHA-1) ₁
Bazaar	DVCS	Win/Mac/Linux/...	✗	✓	✓	✓

1: <https://shattered.io/>



FAZIT

- Git ist weitverbreitet
 - Große Community
 - Viele Firmen nutzen es
 - Viele Antworten im Internet zu finden
- Ich nutze Git und GitHub seit Anfang 2016
- Kann es auf jeden Fall empfehlen



Noch Fragen?

Verteilte Versionsverwaltung mit Git
Johannes Sonn



QUELLEN & LINKS



<https://git-scm.com/>

Verteilte Versionsverwaltung mit Git
Johannes Sonn

