

LANDESBERUFSSCHULE 4 SALZBURG

# Informatik

---

## Versionierung

LBS 4

Dieses Skript dient als zusätzliche Lernunterlage für Informatik

## Inhalt

Versionierung .....	3
Systeme: .....	3
Zentral oder dezentral .....	3
Begrifflichkeiten: .....	3
CVS: .....	3
Funktionsweise: .....	3
SVN: .....	4
Funktionsweise: .....	4
Wichtig: .....	4
Wichtige Begriffe SVN: .....	4

## Versionierung

Versionsverwaltungstools dienen zum Teilen und Verwalten von Quellcode oder anderen Dateien. Damit können mehrere Personen an einem Projekt arbeiten, wobei sichergestellt wird, dass alle Änderungen übernommen werden.

Das System behält den Überblick, welche Datei gerade aktuell ist, und sorgt zugleich dafür, dass die Teilnehmer des Projekts ausschließlich mit den aktuellsten Versionen arbeiten.

### Systeme:

Es gibt einige Systeme in der Open-Source-Community. CVS, SVN und GIT gehören zu den bekanntesten Systemen.

Wir werden nur diese drei Systeme betrachten und an einem Beispiel testen. Vorweg kann man GIT und SVN nicht miteinander vergleichen, weil diese andere Konzepte verfolgen.

### Zentral oder dezentral

CVS und SVN arbeiten zentral. Git und Mercurial arbeiten dezentral. Die Vor- und Nachteile jedes Systems sind für den jeweiligen Einsatzzweck zu berücksichtigen.

### Begrifflichkeiten:

- **Repository:** Container der die aktuellen Daten verwaltet
- **Revision:** Momentaufnahme der Daten im Projekt
- **Tag:** Name für eine Momentaufnahme; Lesezeichen oder Schlagworte (z.B.: version-1.3.3)
- **Arbeitskopie:** Dateibaum der lokal abgespeichert ist und bearbeitet wird
- **commit:** Daten an das Repository senden (Autor, Datum und Grund angeben)
- **update:** Daten zwischen Repository und Arbeitskopie aktualisieren
- **Merge:** zusammenführen von mehreren unabhängigen Veränderungen an einer Datei
- **Trunk:** Hauptentwicklungszweig
- **Branches:** Neben-Entwicklungszweig
- 

### CVS:

CVS (Concurrent Versions System) wurde zur Verwaltung von Quellcode entwickelt wurde. CVS ist vom Ursprung ein Kommandozeilentool das für gängige Betriebssysteme zur Verfügung steht.

Darauf basierend wurden grafische Oberflächen entwickelt, die das Handling erleichtern.

Unter CVS wird der Quellcode an einer zentralen Stelle (Repository) verwaltet. Im Repository können alle Dateien verändert werden, wobei die älteren Versionen erhalten bleiben. Diese können eingesehen werden und sind bei Bedarf wiederherstellbar.

### Funktionsweise:

Als erstes wird der aktuelle Stand des Projektes aus dem Repository geholt (check out). Dabei werden die Metadaten angelegt, welche die Änderungen erkennbar machen. Nach dem verändern der Daten werden die neuen Versionen im Repository

aktualisiert. (check in). Die Metadaten werden aktualisiert und die Änderungen werden serverseitig berechnet.

Viele kommerzielle Management-Werkzeuge basieren auf CVS.

### **SVN:**

SVN (Apache Subversion) ist ebenfalls eine freie Software zum Verwalten von Dateien. SVN ist eine Weiterentwicklung von CVS, es wurden die Konzepte modernisiert und weiterentwickelt. Vor allem wurde ein anderes Datenbankkonzept eingeführt. Die Versionierung erfolgt ebenfalls an einer zentralen Stelle, dem Repository. Dieses kann lokal oder im Netzwerk zur Verfügung gestellt werden. Einzelne Daten sind dort als Projekt zusammengefasst.

### **Funktionsweise:**

SVN verwaltet auch lokal eine Kopie (Arbeitskopie) des Projektes. Dadurch erhöht sich der Speicheraufwand. Die Änderungen werden atomar,-- vollständig – oder gar nicht geschrieben. SVN schreibt vorerst die veränderten Dateien in einen Cache. Die Daten im Repository bleiben unberührt.

Beim Aktualisieren werden die neuen Daten mit dem *commit* Befehl zum Repository gesendet. Falls kein Lock auf die Datei ausgeführt wurde, muss überprüft werden, ob eine neuere Version vorhanden ist. Wenn ja, dann müssen Sie die Dokumente zusammenführen. (merge)

Dadurch erhält man eine unabhängige Verwaltung der Dateien in einem Projekt.

Der Zugriff auf das Repository kann mittels Rechteverwaltung (pfadbasierend) administriert werden. (lesen, schreiben, beides)

SVN-Clients gibt es für die Kommandozeile, als grafisches Tool und als Plug-In für IDE's. Für Eclipse gibt es das Plug-In Subversive.

<https://eclipse.org/subversive/downloads.php>

### **Wichtig:**

- nur kompilierbaren Code einchecken
  - Experimente in einem Branch ablegen
- nur notwendige Ressourcen einchecken
- aussagekräftige Kommentare
- keine binär-Dateien

### **Wichtige Begriffe SVN:**

- File-Lock → sperrt eine Datei vor Veränderung
- Delta/Diff → es werden nur die Veränderungen gespeichert
- check-out → neue Version vom Repository holen
- check-in → Dateien an das Repository senden