

Martin Glinz Harald Gall

# Software Engineering

Wintersemester 2005/06

Kapitel 23

## Software-Konfigurationsverwaltung



Universität Zürich  
Institut für Informatik

# 23.1 Grundlagen

---

23.2 Identifikation und Verwaltung

23.3 Version, Konfiguration, Release

23.4 Änderungswesen

23.5 Problemmeldewesen

# Probleme

---

*Ändern Sie noch eben schnell...*

- Software ist **scheinbar leicht änderbar**
- Während der Entwicklung entstehen **viele Artefakte** in vielen **Versionen**
- Typische **Probleme**:
  - Codieren anhand der **falschen Version** des Entwurfs
  - Paralleles, **unkoordiniertes Ändern** durch mehrere Personen
  - **Undokumentierte Schnellreparaturen** an in Betrieb befindlicher Software
- ⇒ **Hohe Kosten**
- Probleme wachsen **überproportional** mit der Anzahl der Komponenten
- Das **Gegenmittel** heißt **Software-Konfigurationsverwaltung**

# Definitionen

---

**Software-Konfigurationsverwaltung (software configuration management)** – Die Gesamtheit aller Verfahren zur eindeutigen **Kennzeichnung** der Konfiguration eines Software-Systems mit dem Zweck, den **Aufbau** und alle **Änderungen** dieser Konfiguration systematisch zu **überwachen**, die **Konsistenz** des Software-Systems **sicherzustellen** und die Möglichkeit der **Rückverfolgung** anzubieten.

**Software-Konfiguration** – Eine Menge zusammenpassender Software-Einheiten.

**Software-Einheit (software configuration item)** – Der **kleinste**, im Rahmen der Konfigurationsverwaltung als **atomar** behandelte **Baustein** einer Konfiguration.

- Als **Ganzes** registriert, **freigegeben** oder **geändert**
- Zum Beispiel Programm-Module und Dokumente

23.1 Grundlagen

**23.2 Identifikation und Verwaltung**

---

23.3 Version, Konfiguration, Release


23.4 Änderungswesen

23.5 Problemmeldewesen

# Kennzeichnung von Software-Einheiten

---

- Software-Einheiten haben eine **eindeutige Kennzeichnung**
- Besteht aus einem **Namen** und einer **Versionsnummer**
- Kann weitere Informationen enthalten, zum Beispiel Name des Systems oder Teilsystems
- Die **Identität** einer Software-Einheit ist feststellbar, z.B. mit **Prüfsummen**



LOG 0027.03  
Stückliste  
Logistiksystem  
0372538-1

# Registrierung und Verwaltung

---

- Registrierung und Verwaltung der Software-Einheiten durch **Software-Bibliothekar**
- Typisch mit Hilfe einer **Datenbank** oder eines speziellen **Konfigurationsverwaltungssystems**
- Pro Einheit mehrere **Versionen** möglich

Nummer	Name	Typ	Ver	Prüfsumme	Status
...					
LOG 0021	Materialwesen	EntwDok	02	0873451-2	freigegeben
LOG 0027	Stückliste	Prog	03	0372538-1	freigegeben
LOG 0028	Verwendungs- nachweis	Prog	02	0576927-6	in Prüfung
...					

23.1 Grundlagen

23.2 Identifikation und Verwaltung

**23.3 Version, Konfiguration, Release**

---

23.4 Änderungswesen

23.5 Problemmeldewesen



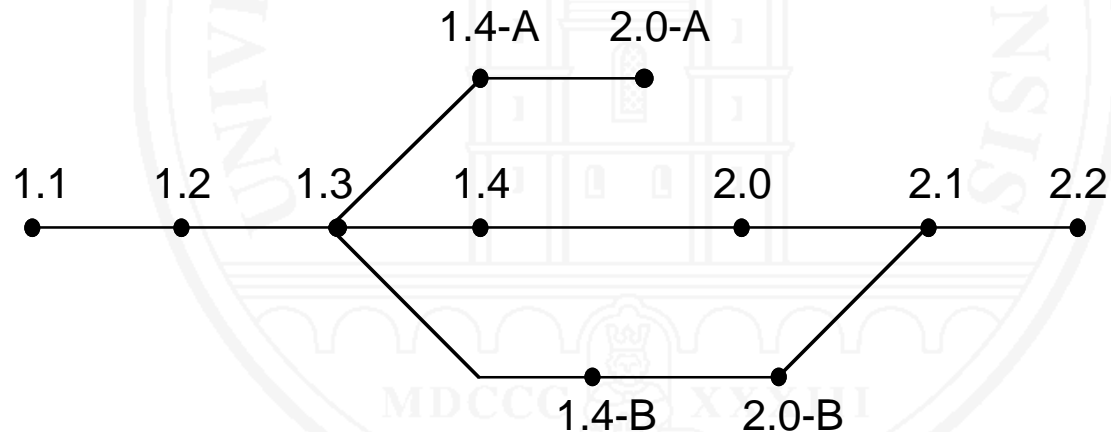
# Versionierung

---

- Einfachste Art der Versionierung: **aufsteigende Versionsnummern**



- Im allgemeinen Fall: **Revisionen** (aufsteigend) und **Varianten** (parallel)



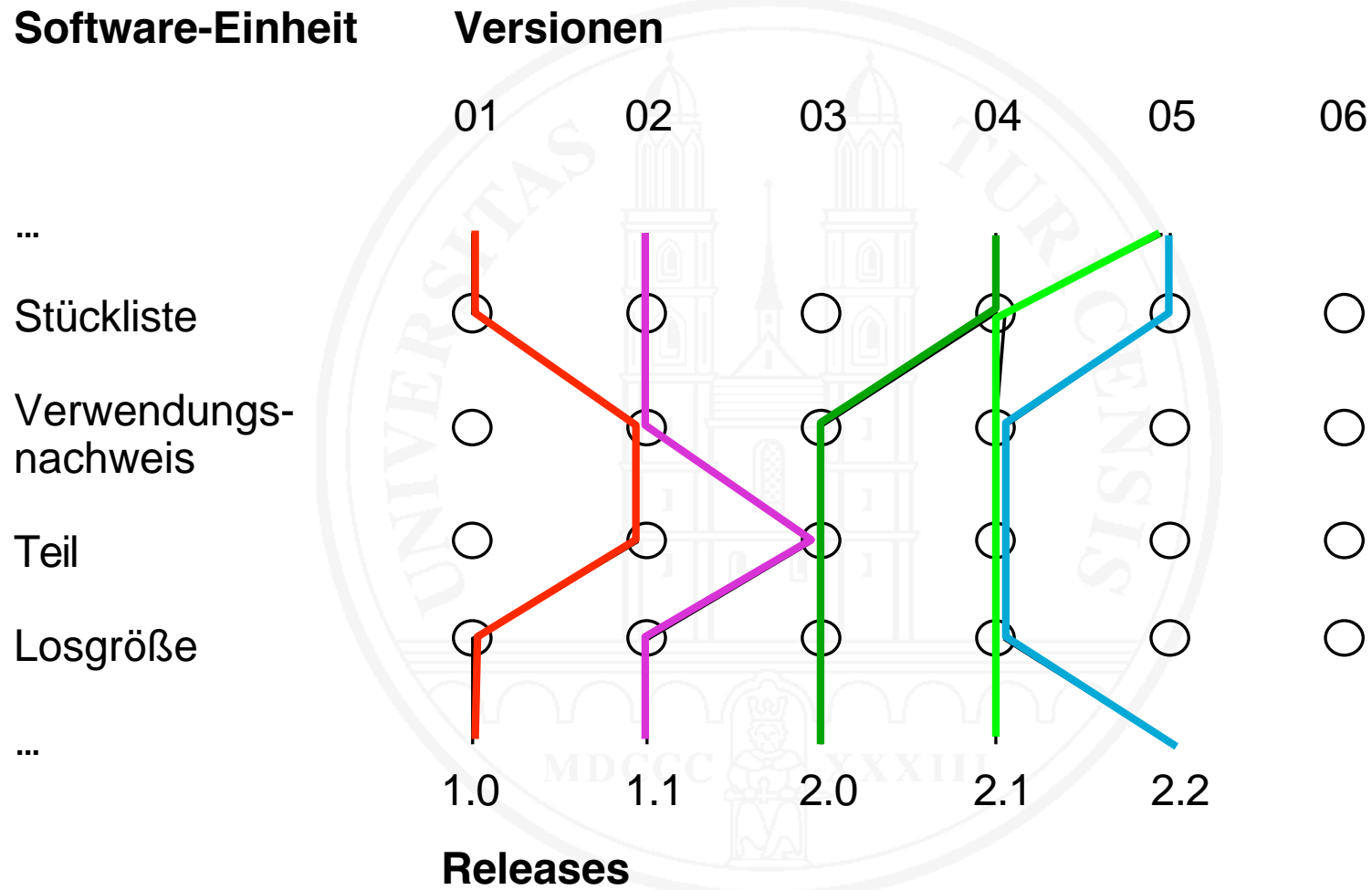
# Konfiguration und Release

---

**Release** – Eine **konsistente** Menge von Software-Einheiten, die **gemeinsam** zur Benutzung **freigegeben** werden.

- Dient vor allem
  - zur **Auslieferung** von Software-Produkten an **Kunden**
  - zur periodischen **Lieferungen** von **Nachträgen** und **Verbesserungen**
  
- Beantwortet u.a. folgende **Fragen**:
  - Welche Software-Einheiten gehören zu einer Konfiguration?
  - Wie hängen die Einheiten voneinander ab?
  - Wie wird ein auslieferbares System generiert?

# Eine Folge von Releases



23.1 Grundlagen

23.2 Identifikation und Verwaltung

23.3 Version, Konfiguration, Release

**23.4 Änderungswesen**

---

23.5 Problemmeldewesen

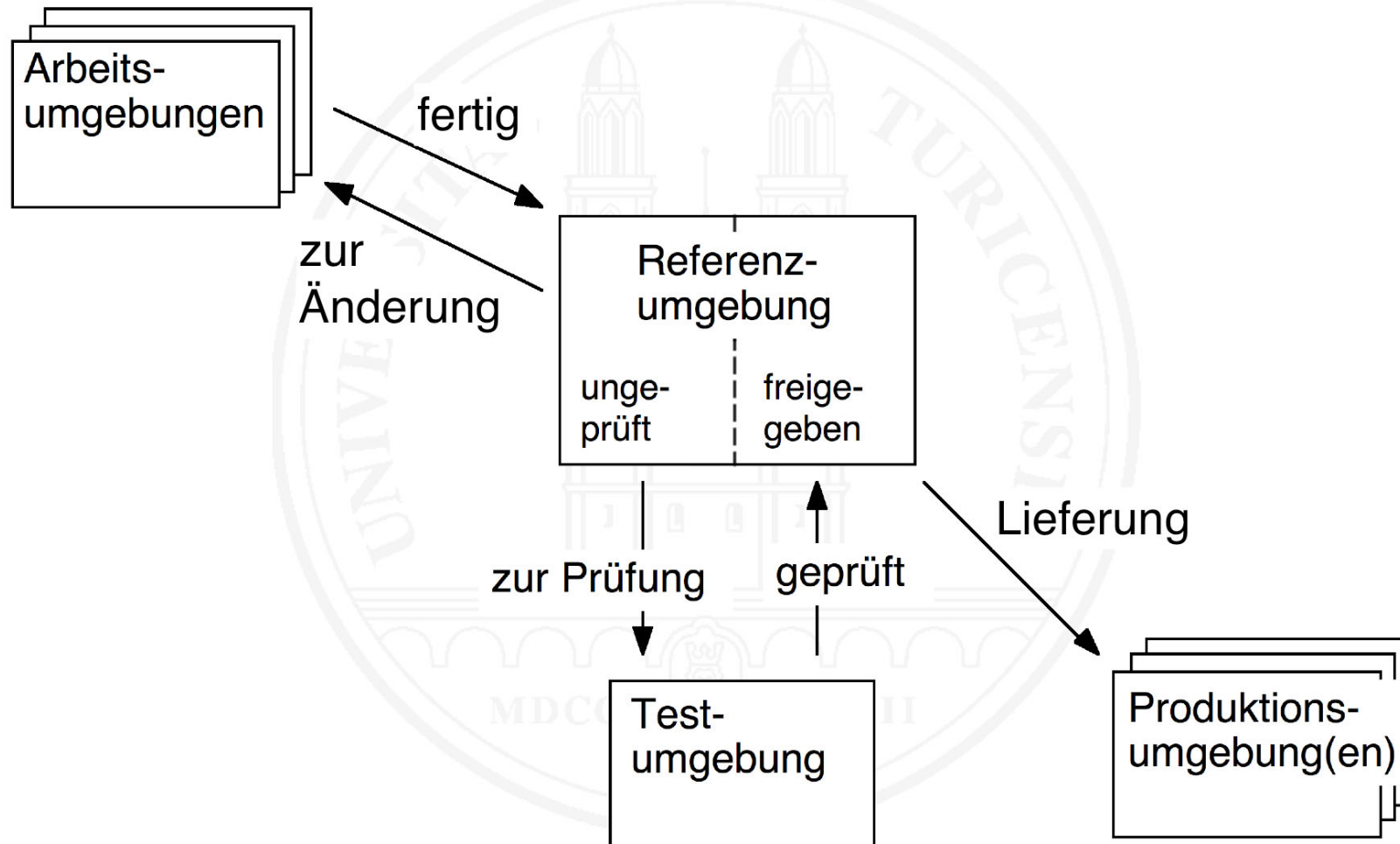
# Änderungswesen

---

- **Getrennte Umgebungen** für
  - Entwicklung (**Arbeitsumgebung**)
  - Verwaltung (**Referenzumgebung**)
  - Test (**Testumgebung**)
  - Operativen Einsatz (**Produktionsumgebung**)
- **Freie** Änderungen nur in **Arbeitsumgebungen**
- **Strikt reglementiertes** Änderungsprozedere für Software-Einheiten in der **Referenzumgebung**
- Änderungen in der **Produktionsumgebung** sind **verboten**
- **Änderungsstand** einer Software-Einheit jederzeit **nachweisbar**
- Definierter **Prozess** für den Ablauf von Änderungen

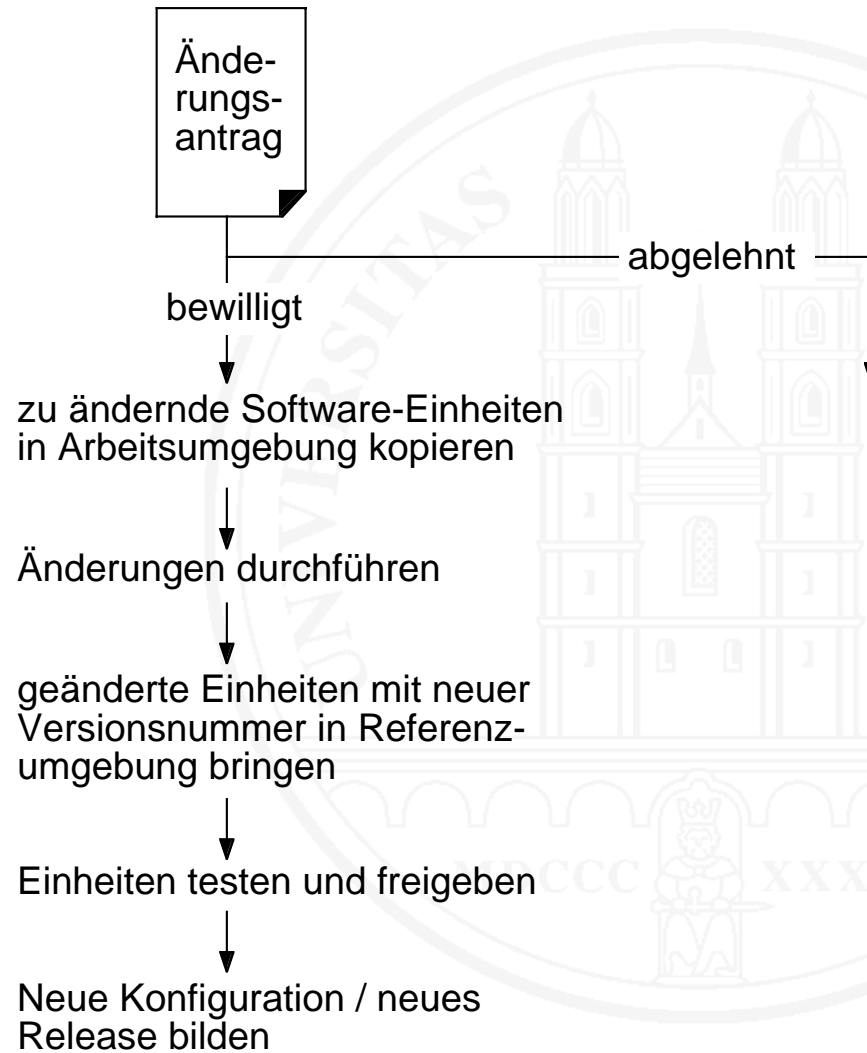
# Umgebungen in der Konfigurationsverwaltung

---



# Prozess für den Ablauf einer Änderung

---



23.1 Grundlagen

23.2 Identifikation und Verwaltung

23.3 Version, Konfiguration, Release

23.4 Änderungswesen

**23.5 Problemmeldewesen**

---



# Das Problemmeldungswesen

---

- Systematische Behandlung von **Kundenproblemen**
- Grundlage: organisiertes **Problemmeldungswesen**
- **Problemmeldungsformular**
- Geordneter Bearbeitungsablauf (**Problemmeldeprozess**)
  - **Registrierung** eingegangener Problemmeldungen
  - **Analyse** der Meldung
  - **Vorläufige Antwort**
  - **Problembehebung**
  - **Abschließende Antwort**
  - **Abschluss** und Ablage der Problemmeldung
  - **Auslieferung** von neuem Release

# Problemmeldung – 1

<b>Problemmeldung</b>		<b>Nr.</b>
<b>Verfasser</b>		
Name _____		Datum _____
Firma _____	Telefon / Fax / E-mail _____	
Adresse _____		
<b>Betrifft</b>		<b>Problem ist</b>
<input type="checkbox"/> Produkt _____		reproduzierbar <input type="checkbox"/> ja <input type="checkbox"/> nein
<input type="checkbox"/> Leistung _____		umgehbar <input type="checkbox"/> <input type="checkbox"/>
<input type="checkbox"/> anderes _____		
Verwendete Hardware _____		<b>Problem betrifft</b>
Betriebssystem _____		<input type="checkbox"/> Programme
		<input type="checkbox"/> Unterlagen
		<input type="checkbox"/> Leistungen
		<b>Antwort erwartet bis</b>
		_____
<b>Problembeschreibung</b>		<input type="checkbox"/> Problembeschreibung in Beilage

# Problemmeldung – 2

<b>Problembeschreibung</b> <input type="checkbox"/> Problembeschreibung in Beilage		
<b>Zu treffende Maßnahmen</b>		<b>Klassifizierung der Maßnahmen</b> Fehlerbehebung <input type="checkbox"/> Anpassung <input type="checkbox"/> Erweiterung <input type="checkbox"/> Beratung/Info <input type="checkbox"/> Schulung <input type="checkbox"/>
<b>Verantwortlicher Sachbearbeiter</b>		
_____	_____	_____
Name	Datum	Visum
<b>Zwischenbescheid an Kunde</b> (erforderlich, wenn Meldung nicht bis zum vom Kunden erwarteten Termin erledigt werden kann)		
_____	_____	_____
Datum	Datum	Visum
<b>Problem erledigt und Kunde informiert</b>		
_____	_____	_____
Name	Datum	Visum

# Literatur

---

Siehe Literaturverweise im Kapitel 11 des Skripts.

Im Skript [M. Glinz (2005). *Software Engineering*. Vorlesungsskript, Universität Zürich] lesen Sie Kapitel 11.

Im Begleittext zur Vorlesung [S.L. Pfleeger, J. Atlee (2006). *Software Engineering: Theory and Practice*, 3rd edition. Upper Saddle River, N.J.: Pearson Education International] lesen Sie in Kapitel 9.1 die Seiten 423-428 sowie im Kapitel 11.5 die Seiten 524-526.