

# Software Engineering

## Informatik II.

### 9. Software-Entwicklung – Dokumentation –

Dipl.-Inform. Hartmut Petters

# Vorwort – was ich noch zu sagen hätte ...

Basis dieser Vorlesung sind vor allem die folgenden Ausarbeitungen

- Vorlesungsskript „Software Engineering“  
von Prof. Dr. Martin Glinz Universität Zürich  
<http://www.ifi.unizh.ch/groups/req/courses/ses/>
- Vorlesungsskript „Informatik II – Software Engineering“  
von Frau Prof. Dr. Kühn FH Karlsruhe FB W  
<http://www.home.fh-karlsruhe.de/~kuin0001/inhalt.htm>
- Das Buch „Software Engineering“ 6. Auflage/2001  
von Prof. Ian Sommerville University of Lancaster (UK)  
Addison Wesley ISBN 3-8273-7001-9

Konkret entnommene Beiträge sind i.d.R. mit einem Quellen-Verweis gekennzeichnet – sollte dieser fehlen bitte ich um Nachsicht.

Den „**roten Faden**“ durch die Vorlesung habe ich dem Skript der Vorlesung von Prof. Dr. Martin Glinz entnommen und um eigene Beiträge erweitert bzw. aus den beiden anderen Quellen ergänzt.

Für die Möglichkeit der Verwendung der wesentlichen Inhalte möchte ich mich an dieser Stelle bei den Autoren herzlich bedanken.

# Grundlagen

## ■ Software

- Kein *materielles Produkt*
  - Manifestiert sich nur in
    - *Wirkungen*
    - **Dokumenten**
- ↳ **Dokumentation ist wichtig!**

## ■ Dokumentations-**Arten**

- *Produktdokumentation*  
dokumentiert das Produkt
  - Technisch
  - Benutzung
- *Projektdokumentation*  
dokumentiert die Entwicklung

# Aufgaben der Dokumentation

## ■ *Wissenssicherung*

- Informationen aus den Köpfen holen
- Informationen Anderen zugänglich machen

## ■ *Kommunikation*

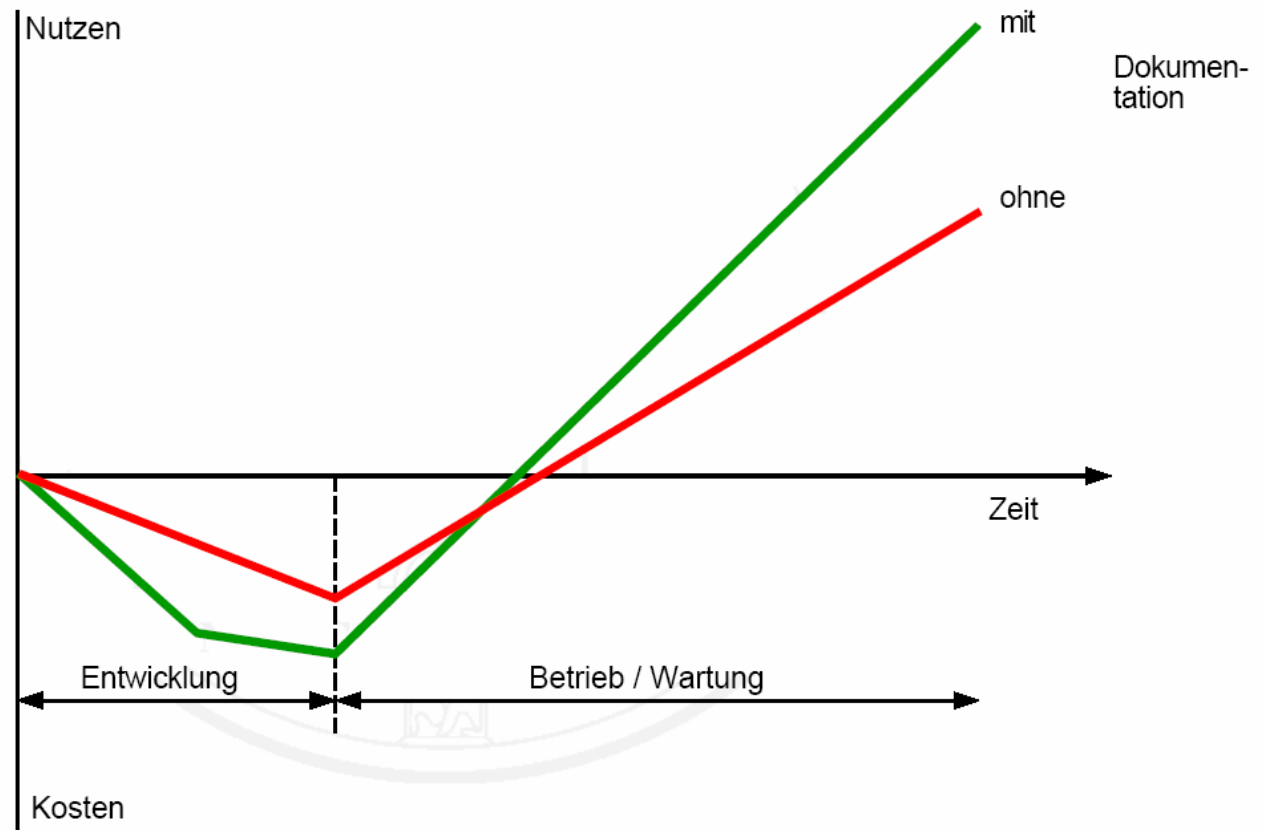
- Reden allein genügt nicht

## ■ *Sichtbar machen des Projektfortschritts*

- Dokumente sind die greifbaren Resultate des Entwicklungsprozesses

# Wirtschaftlichkeit der Dokumentation

- Nur soviel wie nötig
- Das Nötige aber sorgfältig + konsequent



# Produktdokumentation – 1te

- Die **Produktdokumente** dokumentieren ...
  - ... die *Anforderungen* an das System
  - ... das *Konzept* der *Lösung*
  - ... die *Einzelheiten / Details* der *Lösung*  
(Entwürfe + Realisierung)
  - ... die *Montage / Aufbau* der einzelnen Komponenten  
(Integration + Installation)
  - ... die *Planung* der *Tests* und der *Abnahme*
  - ... die Phasen der *Entwicklung* + *Auslieferung*
  - ... die *Handhabung* des Systems  
(Benutzerdokumentation)

# Produktdokumentation – 2te

## ■ Anforderungsspezifikation

Was von dem zu entwickelnden System *verlangt* wird

## ■ Lösungskonzept

Die *Architektur* der Lösung

- Gliederung der Lösung in Komponenten
- Kommunikation zwischen den Komponenten
- Ressourcen-Verteilung

## ■ Detailentwurf + Code

*Lösungsdetails* (Algorithmen + Datenstrukturen)

- Entwürfe sind
  - Entweder separat vom Programm-Code dokumentiert
  - Oder in Form von Kommentaren im Programmcode integriert

# Produktdokumentation – 3te

## ■ Testvorschriften

- Tests für die einzelnen Komponenten
- Tests für die einzelnen Integrationsschritte
- Systemtest (nach letztem Integrationsschritt)

## ■ Abnahmevorschriften

- Abnahme = formaler Abschluss einer Entwicklung
- Prüfung, ob das System die im Anforderungskatalog festgelegten Anforderungen erfüllt



# Produktdokumentation – 4te

## ■ Integrationsplan

- Wie die einzeln fertig gestellten Komponenten zu einem, in einer Testumgebung lauffähigen Gesamtsystem integriert werden

## ■ Installationsanleitung

- Wie ein, auf der Ziel-Hardware lauffähiges System *konfiguriert*...
- ... und auf dieser Ziel-Hardware *installiert* wird
- Welche *zusätzliche Fremd-Software* erforderlich ist (Middleware, Treiber, ...)

# Produktdokumentation – 5te

## ■ Benutzerhandbuch

Die *Bedienungsanleitung* für das System

- Welche Funktionen stellt das System bereit?
- Wie startet man das System?
- Wie bedient man das System / einzelne Funktionen?

Hinweis:

Eingebettete Systeme haben kein Benutzerhandbuch, ihre Benutzung wird im übergeordneten System dokumentiert.

## ■ Glossar

Erklärt die verwendeten *Begriffe + Abkürzungen*

- Hilfreich für Entwickler + Benutzer
- Sollte in jedem größeren Entwicklungsprojekt (projektbegleitend) erstellt werden.

# Projektdokumente

## ■ **Projektplan**

- Dokumentiert den geplanten *Projektablauf*
- Stellt *SOLL* und *IST* gegenüber
- Dokumentiert den *Projektablauf* + *Projektabschluss*

## ■ **Qualitätsplan**

- Projektspezifische *Vorgaben* für die *Qualität*
- *Durchführungsbestimmungen* für den Ablauf

## ■ **Projekt-Protokoll**

- *Sammlung* aller *Schriftstücke* + *Berichte* aus dem Projekt, wie
  - Planungsunterlagen
  - Sitzungsprotokolle
  - Review- und Abnahme-Protokolle
  - Änderungsdokumente

# Dokumenten-Erstellung

- Parallel zur Entwicklung
  - schritthaltend
  - Permanent
- Keine „Hinterher“-Dokumentation
- Eventuell *Überarbeitung + Aktualisierung* der Dokumente bei *Projektabschluss* (Historie)
- Dokumente *genehmigen + unterzeichnen*
  - Ersteller
  - Projekt-Manager
  - Prüfer (Abnahme)

# Dokumentenverwaltung

- Dokumente unterliegen dem **Konfigurations-Management** (↪ Kapitel 10)
- **Klassifikation**
  - *Leichtes Finden* durch *geordnete Ablage*
  - + sinnvolle *Klassifikation*
- **Freigabewesen**
  - Nur *freigegebene Dokumente* sind *gültig*
  - *Änderungshistorie* + *Änderungsdokumente* mitführen
- **Änderungswesen**
  - Nur *aktuelle Dokumente* sind hilfreich
  - Aktuelle Dokumente sind *Voraussetzung für Wartbarkeit*

# Zusammenfassung

- **Dokumentation ist wichtig / unabdingbar**
  - Fortschrittskontrolle
  - Maßeinheit für Erreichtes
  - Basis für sinnvolle Kommunikation
  - Wissens-Absicherung
  - Unterliegt dem Konfigurations-Management
- **Dokumentation umfasst**
  - Anforderungsspezifikation
  - Lösungskonzept, Detailentwurf + Code
  - Test- und Abnahme-Vorschriften
  - Integrationskonzept + Installationsanleitung
  - Benutzerhandbuch + Glossar
  - Projektplan / Qualitätsplan / Projekt-Protokoll

# Literatur – Software Engineering

- Skript Informatik II Prof. Dr. Kühn / Fb W FH Karlsruhe  
<http://www.home.fh-karlsruhe.de/~kuin0001/inhalt.htm>
- Skript Software Engineering Prof. Dr. Martin Glinz Universität Zürich  
[http://www.ifi.unizh.ch/groups/req/courses/se\\_I/](http://www.ifi.unizh.ch/groups/req/courses/se_I/)
- Skript Software Engineering II Bernd Kahlbrandt FH Hamburg  
<http://www.informatik.fh-hamburg.de/~khh/st4se2/>
- Software Engineering  
Ian Sommerville (ISBN3-82737-001-9)
- Software Engineering  
- Grundkurs für Praktiker –  
Roger S. Pressman (ISBN 3-89028-163-X)
- Software Entwurf  
- Methoden und Werkzeuge –  
A. Schulz (ISBN 3-486-21608-2)
- Software Engineering und Prototyping  
Thorsten Spitta (ISBN 3-540-17542-3)
- CASE  
Helmut Balzert (ISBN 3-411-03224-3)
- Software-Qualitätssicherung  
Ernest Wallmüller (ISBN 3-446-15846-4)

# Software Engineering

## Informatik II.

### 10. Software-Entwicklung – Konfigurations-Management –

