LE 22
1

Software-Technik: Querschnitte und Ausblicke

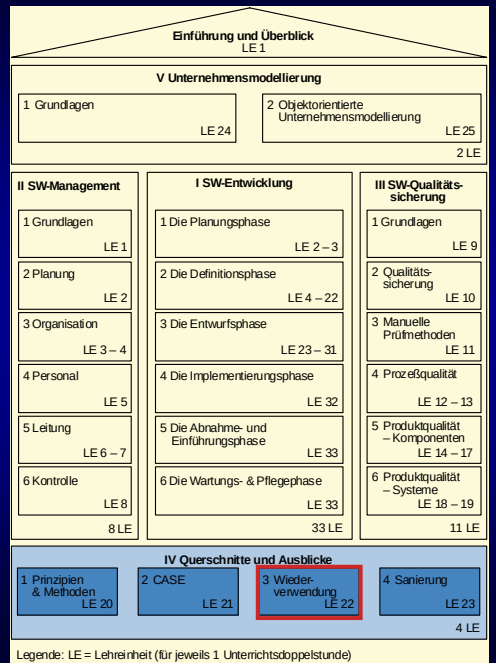
3 Wiederverwendung

[stark gekürzt]

Prof. Dr. Helmut Balzert
Lehrstuhl für Software-Technik
Ruhr-Universität Bochum

© Helmut Balzert 1998

LE 22
2



LE 22
3

Inhalt

3.1 Zur Problematik

3.7 Einführung der Wiederverwendung

3.4 Organisation

3.6 Kosten/Nutzen der Wiederverwendung

LE 22
4

3.1 Zur Problematik

- **»Diese Vorgehensweise, bei der darauf verzichtet wird, existierende Produkte als Komponenten zu verwenden und für jedes Produkt und jede zu entwickelnde Komponente die Marktchancen vor ihrer Entwicklung abzuschätzen, ist – so kann man es wohl sagen – industrielle Steinzeit.«**
- ***»Golden Rule of Reuseability:***
  - ***Before you can reuse something, you need to***
  - ***1 find it***
  - ***2 know what it does***
  - ***3 know how to reuse it.«***

LE 22 5

3.4 Organisation

- Reifegradstufen der Wiederverwendung

Stufe 1: Ad hoc-Wiederverwendung

- Wiederverwendung unsystematisch
  - Gelegentlich, in Abhängigkeit von der Arbeitsweise der Projektmitarbeiter, unkoordiniert und in der Regel undokumentiert
  - Regel: kopieren, anpassen und auseinanderlaufen lassen
  - Gemeinsame Versionsführung mit dem Original findet nicht statt
- Wartung und Fehlerbehebung mehrfach
- Großteil heutiger Wiederverwendung sicher in dieser Weise.

LE 22 6

3.4 Organisation

- Immer noch besser als überhaupt keine Wiederverwendung
- Einfach und nur mit kurzfristigem Nutzen
  - Nutzen wird wieder aufgehoben, wenn sich zu viele nicht dokumentierte Kopien unkontrolliert verbreiten und die Software unwartbar machen

Stufe 2: Wiederverwendung verfügbarer Software

- Durch Umsetzung geeigneter Maßnahmen systematisiert
- Grundlagen für Entwicklung wiederverwendbarer Komponenten werden geschaffen
- Verfügbare und wiederverwendbare Software muß gesammelt und strukturiert dokumentiert werden.

LE 22 7

3.4 Organisation

- Systemteile, die unterschiedlicher Änderungshäufigkeit unterliegen, werden lokalisiert
  - Varianten und Versionen entsprechender Systeme können effizienter entwickelt werden
- Verfahrensbasierter Zukauf von im Markt verfügbarer Software
  - Software-Entwicklung wird gezielt unterstützt
- Erstellung von modular aufgebauten Software-Systemen
  - Grundlage für die Wiederverwendung von Komponenten dieser Systeme
- Marktübliche und interne Standards werden eingehalten
  - portable Systeme.

LE 22 8

3.4 Organisation

Stufe 3: Entwicklung für Wiederverwendung

- Software-Entwicklung orientiert sich an Anwendungsdomänen
- Geeignete Infrastruktur für wiederverwendungs-orientierte Entwicklung wird aufgebaut
- Software-Komponenten werden erstellt
  - Nicht im Hinblick auf ihren Einsatz in einem Produkt
  - Sondern auf ihre künftigen Verwendungsmöglichkeiten
- Software-Komponenten mit Wiederverwendungspotential werden als inkrementell erweiterbare Halbfabrikate erstellt.

LE 22
9

3.4 Organisation

- Funktionalität dieser Software kann ohne Eingriff in den Quellcode ausgebaut werden
- Inkrementell erweiterbare Software erleichtert Erstellung von Versionen und Varianten
- Erstellung von Halbfabrikaten kann projektun-abhängig oder abgetrennt von anwendungs-spezifischen Projektarbeiten geschehen
- Halbfabrikate auf der Grundlage einer Analyse der Anwendungsdomäne
  - Ziel: Domänenmodell, das eine generelle und damit wiederverwendbare Architektur für die Produkte der untersuchten Anwendungsdomäne darstellt.

LE 22
10

3.4 Organisation

- Weitere notwendige Aktivitäten
  - Einrichtung eines Wiederverwendbarkeits-Archivs
  - Durchführung einer wiederverwendungs-spezifischen Aus- und Weiterbildung
  - Schaffung einer entsprechenden innerbetrieblichen Kommunikationsstruktur *(Bulletin Boards* usw.)
  - Bereitstellung von Richtlinien für standardisierte Auswahl, Bewertung und Anpassung von Komponenten.

LE 22
11

3.4 Organisation

Stufe 4: Verwendung von Domänenmodellen und statistische Steuerung des Prozesses

- Entwicklung von Anwendungen mit Domänenmodellen
- Steuerung der wiederverwendungsorientierten Software-Entwicklung auf statistischer Basis
- Für jede wohldefinierte Anwendungsdomäne wird ein Modell bereitgestellt
  - Das Modell soll eine generelle und damit wiederverwendbare Architektur für die Produkte der untersuchten Anwendungsdomäne beschreiben
- Analyse der jeweiligen Domäne erlaubt eine Integration in den Software-Entwicklungsprozeß.

LE 22
12

3.4 Organisation

- Jedes Domänenmodell besteht aus
  - Architektur
  - Menge integrierter Komponenten, den Halbfabrikaten, aus eigenem und kommerziell verfügbarem Bestand
- Schwerpunkt der Software-Entwicklung
  - Modellierung von Anwendungen
  - Nicht Programmierung
- Anwendungen entstehen durch das Montieren von Halbfabrikaten
- Anwendungsentwickler entwickelt nur die anwendungsspezifischen Teile.

LE 22
13

3.4 Organisation

- Folgende Aktivitäten außerdem:
  - Komponentenkreislauf für Entwicklung, Zertifizierung und Benutzung von Komponenten wird installiert
  - Wiederverwendbarkeitsorientierte CASE-Umgebungen werden eingerichtet
  - Projektverlauf wird durch Kennzahlen und Kosten/Nutzen-Modelle statistisch gesteuert
    - Kennzahlen werden aus Messungen ermittelt.

LE 22
14

3.4 Organisation

Stufe 5: Organisationsweite Ausrichtung auf Wiederverwendung

- Aktivitäten aller Unternehmensbereiche vollständig auf Wiederverwendung ausgerichtet
- Anwendungen werden wie in Stufe 4 durch Montage von bereits vorhandenen, weitgehend standardisierten Halbfabrikaten erstellt
- Neu: auch nicht softwareentwickelnde Unternehmensbereiche handeln wiederverwendungsorientiert
- Alle Verkaufsaktivitäten werden unter Berücksichtigung verfügbarer Komponenten ausgeübt.

LE 22
15

3.4 Organisation

- Jede Software-Komponente wird als Vermögenswert des Unternehmens betrachtet
- Höhe der Vermögenswerte und Lebenszykluslänge dieser Komponenten finden Eingang in die...
  - Entscheidungen der Software-Erstellung
  - strategischen Entscheidungen des Unternehmens.

LE 22
16

3.4 Organisation

- Wiederverwendungsstufen

Stufe	Wiederverwendung in Prozent	Anforderungen
5 Bereichsbezogene Wiederverwendung	80 – 100%	Bereichsanalysen und -architekturen
4 Konsequente Wiederverwendung	50 – 70%	Bibliotheken, Prozesse, Metriken, Training
3 Planbare Wiederverwendung	30 – 40%	Zustimmung und Unterstützung vom Management, Motivations-programme, Bibliotheken
2 Ausschlachten von Altanwendungen	10 – 50%	Glück und Wartungsprobleme
1 Keine oder ad hoc–Wiederverwendung	< 20%	Keine organisatorischen Änderungen

LE 22
17

3.7 Einführung der Wiederverwendung

- Hindernisse bei der Einführung der Wiederverwendung
  - Ökonomisch
    - Fehlendes *Commitment*
    - Unklare Geschäftsstrategie
    - Investitionshöhe
    - Aufwandsgeschäft
    - Fehlende Nutzungs- und Verwertungsrechte
  - Organisatorisch
    - Im Prozeß nicht vorgesehen
    - Verantwortung nicht zugewiesen
    - Fehlender Katalysator
    - Fehlende Infrastruktur.

LE 22
18

3.7 Einführung der Wiederverwendung

- Soziologisch
  - *Not-invented-here*-Syndrom
  - Widerstand gegen Veränderungen
  - Existenzängste »*Re-Use* ist ein Job-Killer«
  - Selbstverständnis des Entwicklers / geändertes Rollenbild
- Technisch
  - Fehlende Erfahrung mit praktischen Anwendungen
  - Mangelndes *Know-how*
  - Schwächen im *Software-Engineering*-Prozeß
  - Fehlende Werkzeuge.

LE 22
19

3.6 Kosten/Nutzen der Wiederverwendung

- Kosten/Nutzen der Wiederverwendung
- Qualitative Aussagen
  - Faustregel: »Formel 3«
    - Software muß *dreimal entwickelt* werden, bevor sie wirklich wiederverwendbar entwickelt werden kann
    - Bevor die »Früchte der Wiederverwendung geerntet« werden können, muß Software *dreimal wiederverwendet* werden
  - *break-even*-Punkt bei ca. 3 Wiederverwendungen einer gezielt für die Wiederverwendung entwickelten Komponente
  - Danach können die um 30% bis 50 % höheren Entwicklungskosten wieder hereingespielt werden.

LE 22
20

3.6 Kosten/Nutzen der Wiederverwendung

- 60% höhere Erstellungskosten:
  - 25% für zusätzliche Verallgemeinerung
    - z.B. Parametrisierung, Entwurf, *review*
  - 15% für zusätzliche Dokumentation
  - 10% für zusätzliches Testen
  - 5% für Archivablage und Wartung
- Wiederverwendung und Gesamtproduktivität

	0%	25%	50%
Gesamtzeit in MM	81,5	45	32
Anzahl der Mitarbeiter	8	6	5
Kosten je Zeile	40,74	22,5	16
Zeilen pro MM	165	263	370
Ersparnis	0%	45%	61%

  - Es wurden 3 reale Anwendungen mit 10.000 Zeilen COBOL-Code untersucht.

LE 22 21

3.6 Kosten/Nutzen der Wiederverwendung

- Wiederverwendung und Gesamtproduktivität

	0%	10%	30%	50%	80%
Gesamtzeit in MM	200	176	131	89	33
Ersparnis	0%	12%	34%	56%	84%

  - Es wurden Objective-C-Programme mit 20.000 - 30.000 Zeilen untersucht

- Kosten / Nutzen-Verhältnis bei Klassenbibliotheken
  - Bibliotheksbestand wirkt sich gravierend auf die Kosten / Nutzen-Relation aus
  - Sind am Anfang keine Bibliotheksklassen verfügbar, dann wird die Kosten / Nutzen-Relation erst für das 3. Projekt < 1.

LE 22 22

3.6 Kosten/Nutzen der Wiederverwendung

- Japanische Erfahrungen
  - Nicht primär eine Frage der Technik
  - Seit Ende der 80er Jahre hat Wiederverwendung eine hohe Priorität
  - Ergebnisse durch straffe Organisation und strikte Einhaltung von Methoden
  - NEC
    - 6,7fache Produktivitätssteigerung
    - 2,8fache Qualitätsverbesserung im Bereich betriebswirtschaftlicher Anwendungen
    - Erreicht durch
      - Identifikation und Standardisierung von 32 logischen Schablonen und 130 gemeinsamen Algorithmen
      - Strikte Anwendung ihrer CASE-Umgebung.

LE 22 23

3.6 Kosten/Nutzen der Wiederverwendung

- Fujitsu
  - Nach Aufbau eines Wiederverwendbarkeitsarchivs Termintreue 70%
  - Vorher nur 20%
- Amerikanische Erfahrungen
  - Raytheon
    - Analyse und überarbeiteter Entwurf von 5000 im Einsatz befindlichen COBOL-Programmen
    - 50%-ige Produktionssteigerung in 6 Jahren
    - In neuen Systemen durchschnittlich 60% Code wiederverwendet
  - Unabhängige amerikanische Studie
    - > 70% kürzere Lieferzeiten durch Wiederverwendung .

LE 22 24

3.6 Kosten/Nutzen der Wiederverwendung

- Hewlett-Packard
  - Projekt 1
    - Fehlerreduktion: 51%
    - Produktivitätssteigerung: 57%
  - Projekt 2
    - Fehlerreduktion: 24%
    - Produktivitätssteigerung: 40%
    - Reduktion der Entwicklungszeit: 42%
- Quantitative Aussagen
  - Jede wiederverwendbare Software-Komponente muß als betrieblicher Vermögenswert *(asset)* angesehen werden
  - Dann kann Wiederverwendung wie jede andere finanzielle Investition kalkuliert werden.