
Vorlesung „Methoden des Software-Engineering“

Block F „Web-Engineering“
**Web-Software, Vorgehensmodelle und
Projektmanagement**

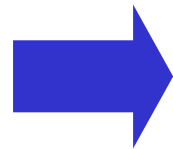
Nora Koch

Einheit F.1, 01.02.2007

WS 2006/2007

Gliederung Block F

- Web-Software, Vorgehensmodelle und Projektmanagement
- Methoden und Techniken zur Entwicklung von Web-Anwendungen



Web Engineering Vorlesung im SS07

Ziele der Vorlesung

- Überblick zu Web-Engineering (WE)
 - Produkt: Web-Software
 - Entwicklung: Software Engineering für Web-Software
- insbesondere
 - Unterschiede beim Entwicklungsprozess
 - Besonderheiten beim Projektmanagement

Historie

- **1989:** Tim Berners-Lee (CERN) schlägt neuen Netzdienst im Intranet vor: einheitliche Benutzeroberfläche und Querverweise zwischen Dokumenten
- **1990:** World Wide Web (WWW) wird der Öffentlichkeit vorgestellt
- **1992:** Erster Browser frei im Netz verfügbar
- **1993:** XMosaic für Unix und erste wissenschaftliche Arbeiten zum Thema Web-Engineering
- **1998:** Erster Workshop zu Web-Engineering
- **2001:** IEEE Multimedia Journal Special Ausgabe zu Web-Engineering
- 1st International Conference on Web Engineering (ICWE)
- **2002:** Journal on Web Engineering (Rinton Press)

World Wide Web (WWW / Web)

- Web ist ein über das Internet aufrufbares Hypermedia-System
 - Hypertext: Knoten (Dokumente/Dienste) und Verweise (Links)
 - Hypermedia: Hypertext + Multimedia-Elemente
 - Präsentation von Dokumenten/Diensten
 - Navigation durch diese Dokumente/Dienste
 - Inzwischen mit Abstand der populärste Dienst im Internet
- Web beeinflusst unser tägliches Leben
 - Lernen
 - Arbeit
 - Freizeit
- Allgegenwärtigkeit durch Natur des Web
 - globale und permanente Verfügbarkeit
 - komfortabler und einheitlicher Zugriff
 - verteilte Informationen in Form von Webseiten, von „jedem“ herstellbar

Web-Engineering: Definition

- (1) Web-Engineering ist die Anwendung systematischer und quantifizierbarer Ansätze (Konzepte, Methoden, Techniken und Werkzeuge) um Anforderungsbeschreibung, Entwurf, Implementierung, Test, Betrieb und Wartung qualitativ hochwertiger Web-Anwendungen kosteneffektiv durchführen zu können.*

- (2) Web-Engineering meint auch die wissenschaftliche Disziplin, die sich mit der Erforschung dieser Ansätze beschäftigt.*

Gerti Kappel, Birgit Pröll, Siegfried Reich & Werner Retschitzegger (2003)

Web-Engineering

Neue Disziplin zur Entwicklung von Software

- für ein spezielles Medium: das Web
 - Web-Server, Web-Clients, ...
- mit eigenem Paradigma: Hypertextstruktur
 - Knoten und Links
- unter Anwendung von Web-Technologien
 - CSS, AJAX, ...
- und Bedarf an speziellen Entwicklungsverfahren
 - für Trennung von Inhalt und Präsentation, ubiquitäre Anwendungen, ...

Web-Software

- **Web-Software (Web-basierte Systeme, Web-Systeme)**
 - Ressourcen werden über einen Webbrowser (Benutzerschnittstelle) verwendet
- **Website**
 - statisches Informationssystem
 - besteht aus Webseiten
- **Web-Anwendung (WA)**
 - beruht auf den Technologien und Standards des World Wide Web Consortiums (W3C)
 - stellt web-spezifische Ressourcen wie Inhalte und evtl. auch Dienste bereit
- **Web-Service**
 - unterstützt die direkte Interaktion mit anderen Software-Agenten unter Verwendung XML-basierter Nachrichten durch den Austausch über internetbasierte Protokolle
 - eindeutig identifizierbar anhand einer Uniform Resource Identifier (URI)

Web-basierte Systeme vs. Software Systeme

Web-basiert

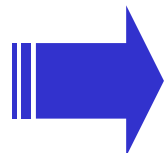
- stark Inhalts-/Informationsbasiert
- Relevanz der Präsentationsaspekte der **Benutzerschnittstelle** (look & feel)
- dynamische Anforderungen an Benutzerschnittstelle
- **kurze Entwicklungszyklen**
- unzählige **Nutzer**
- **Heterogenität** der Benutzer (unterschiedliche Fähigkeiten und Background)
- **Internationalität** (Sprache & Kultur)
- **selbsterklärend**

nicht-Web-basiert

- Funktionalität/Prozessorientierung
- funktionale Bedienbarkeit im Vordergrund
- eher stabile Anforderungen an Schnittstelle
- längere Entwicklungszyklen
- begrenzter Benutzerkreis
- überwiegend homogene Skills der Benutzer
- öfter für lokalen Bedarf
- Training, Benutzerhandbuch, Hilfe

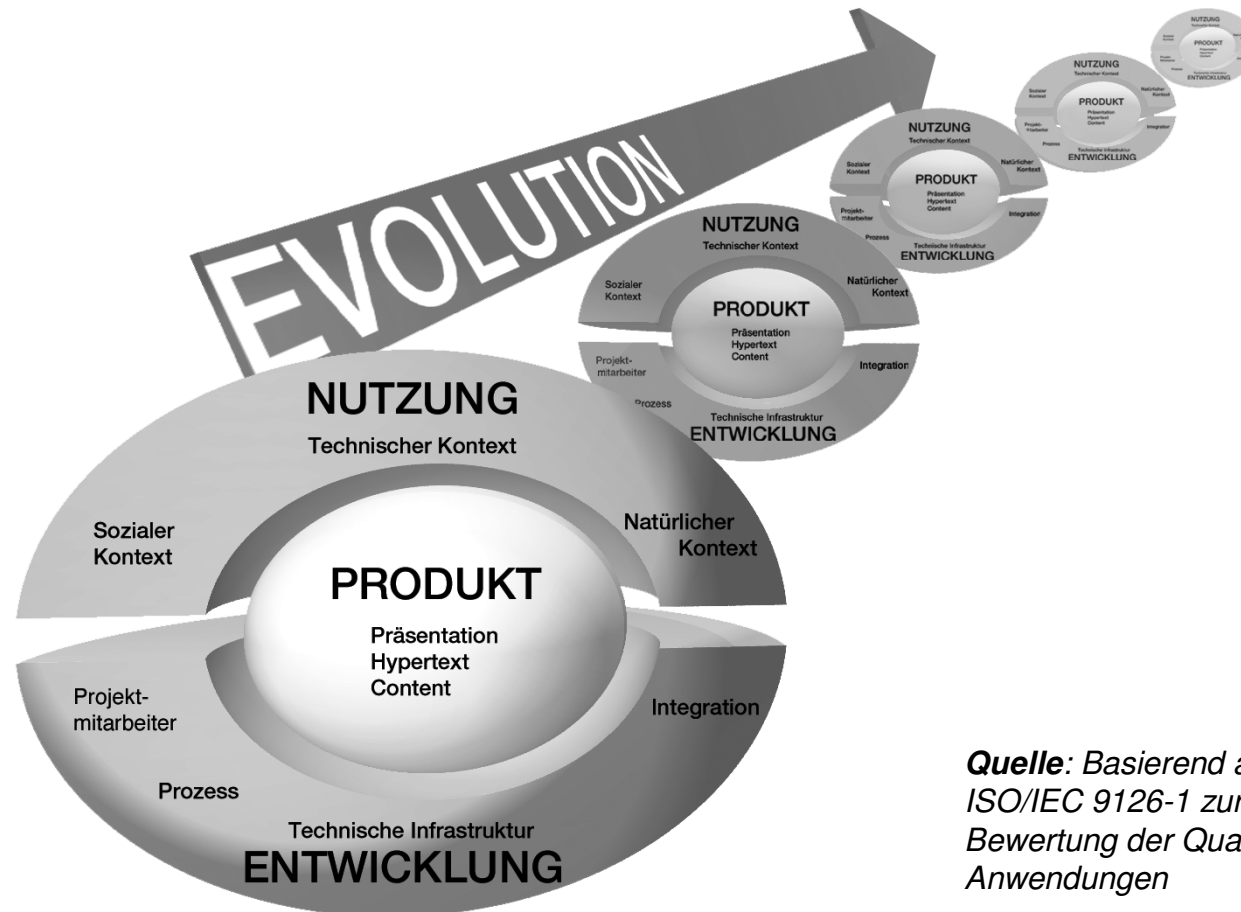
Web-Anwendungen vs. Anwendungen

- Unterschiede
 - **zusätzliche Charakteristika**, entweder in herkömmlichen Anwendungen nicht vorhanden (z.B. nichtlineare Navigation)
 - oder in Web-Anwendungen **besonders stark ausgeprägte Charakteristika** (z.B. Häufigkeit von Änderungen)
- Intensität
 - abhängig von der Art der Web-Anwendung
 - Beispiel: Aktualität und Konsistenz des Inhaltes bei transaktionalen WA wie E-Commerce-Systeme im Vergleich zu reinen Informationsanbieter



Konzepte, Methoden, Techniken und Werkzeuge des traditionellen Software Engineering müssen **angepasst** werden

Charakteristika von Web-Anwendungen



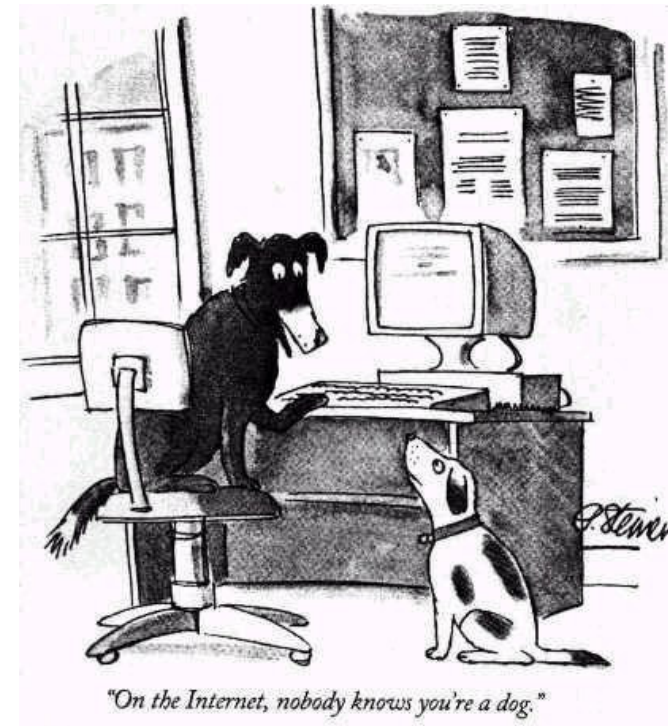
Quelle: Basierend auf ISO/IEC 9126-1 zur Bewertung der Qualität von Web-Anwendungen

Produktbezogene Charakteristika

- **Inhalt/Content**
 - Dokumentenzentrierter Charakter und Multimedialität
 - Qualitätsansprüche hinsichtlich Aktualität, Genauigkeit, Konsistenz, Verlässlichkeit und Umfang
- **Hypertextstruktur (Navigationsstruktur)**
 - Nicht-Linearität
 - Desorientierung und kognitive Belastung
- **Präsentation**
 - Ästhetik
 - Selbsterklärbarkeit

Nutzungsbezogene Charakteristika

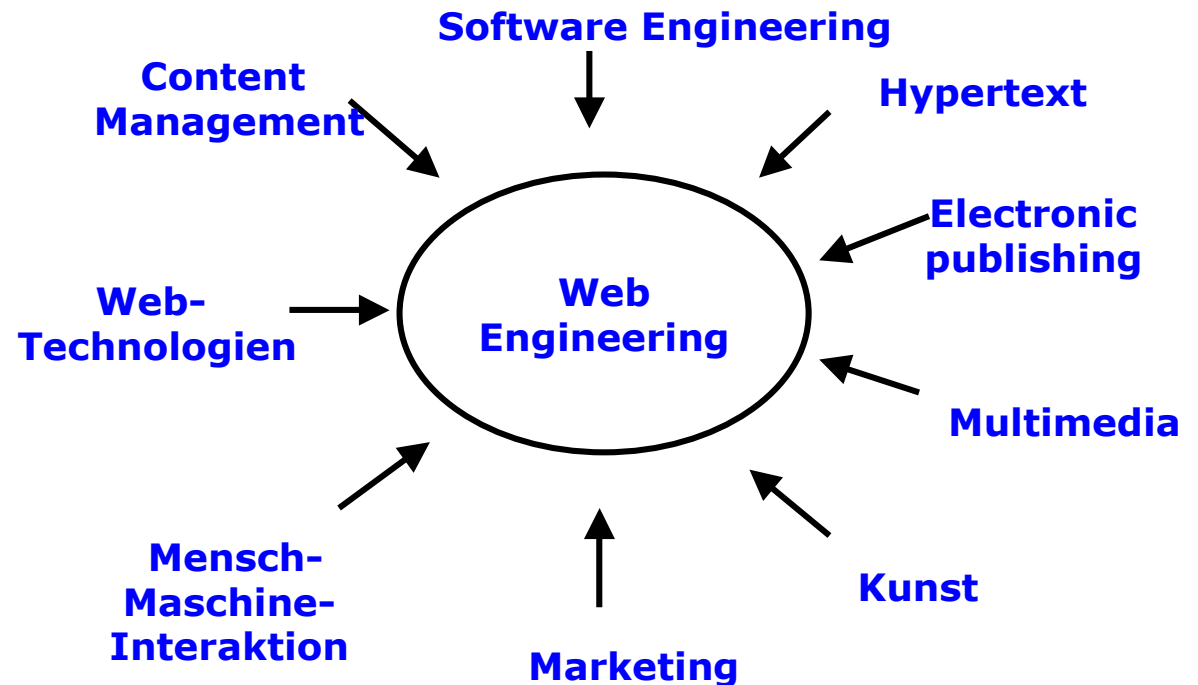
- Sozialer Kontext: Benutzer
 - Anonymität
 - Spontaneität
 - Multikulturalität
- Technischer Kontext: Netzwerke und Endgeräte
 - Qualität der Dienstleistung
 - Bandbreite, Sicherheit, ...
 - Hardware/Software
 - PDA, PC, Browsers,
- Natürlicher Kontext: Ort und Zeit
 - Globalität
 - Verfügbarkeit



(Steiner, 1993)

Entwicklungsbezogene Charakteristika

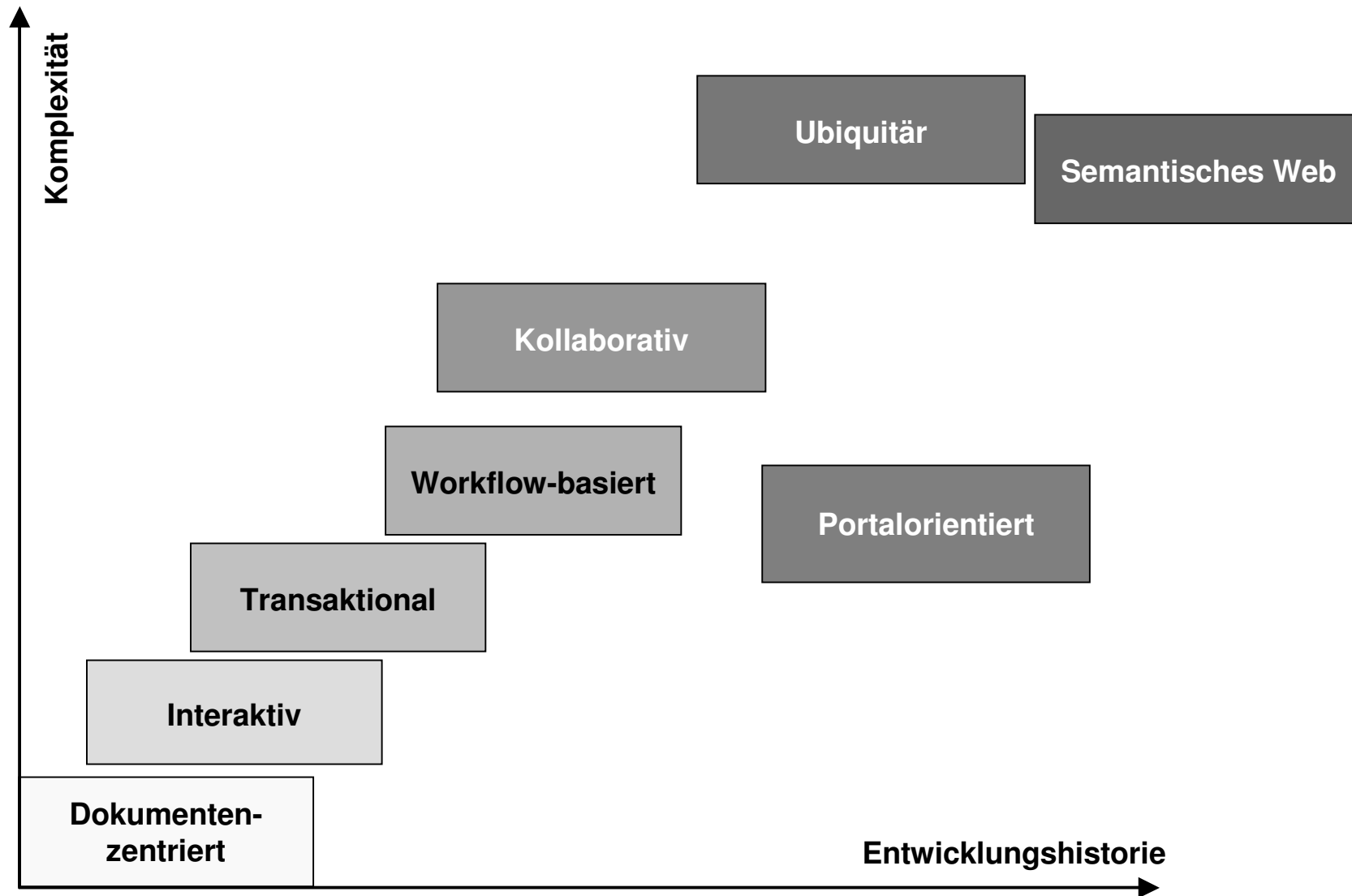
- Projektmitarbeiter
 - Interdisziplinarität
 - Community-Entwicklung (open source)
- Technische Infrastruktur
 - Inhomogenität (Webserver/Webclient)
 - Unreife
- Prozess
 - Flexibilität
 - Parallelität
- Integration
 - Interne Integration (Legacy-Systeme)
 - Externe Integration (Inhalt und Dienste externer Web-Anwendungen)



Evolutionenbezogene Charakteristika

- **Permanente Änderungen**
 - hohe Änderungsdynamik durch drei Dimensionen Produkt, Nutzung und Entwicklung
 - geänderte Anforderungen und Rahmenbedingungen
- **Konkurrenzdruck**
 - kürzere Produktlebenszyklen und sehr kurze Entwicklungszyklen
 - wenig Spielraum für systematischen Entwicklungsprozess
- **Schnellebigkeit**
 - hoher Termindruck durch raschen Wandel im Web
 - Berücksichtigung von permanente Wartung im Entwicklungszyklus

Kategorien von Web-Anwendungen

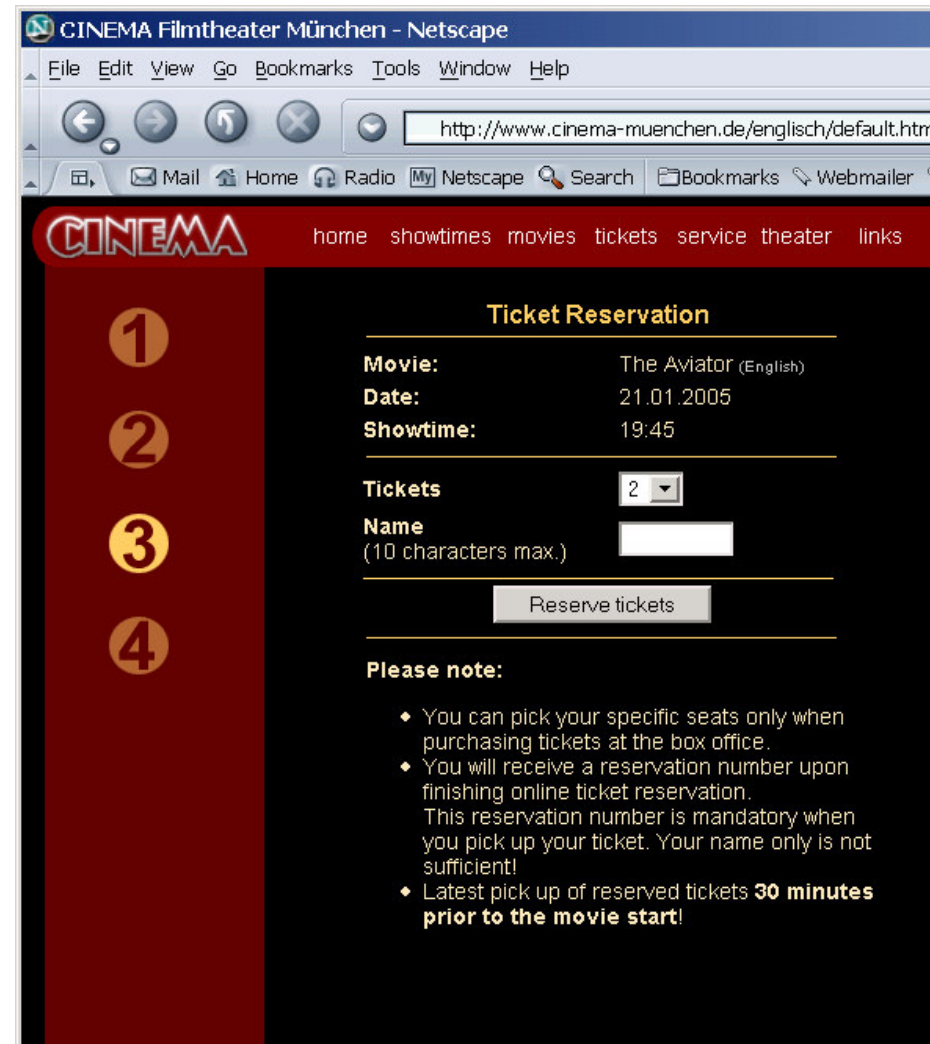


Dokumentenzentrierte Web-Anwendungen

- auch **Web-Informationssystem** (WIS) genannt
- Schwerpunkt: **Inhalt der Seiten**
- Webseiten werden als statischen HTML-Dateien auf Web-Server abgelegt und bei Anfrage an Web-Client geschickt
- Nachteile
 - Hoher Aufwand bei änderungsintensiven Webseiten
 - Gefahr von Inkonsistenzen
- Vorteile
 - Einfachheit und Stabilität des Systems
- Beispiele
 - Statische Homepages

Interaktive/Transaktionale Web-Anwendungen

- Benutzerinteraktionen
 - Eingabe/Selektieren
- Datenbanktransaktionen
- Vorteile
 - strukturierte Abfragen
 - effiziente und konsistente Datenverwaltung
- Beispiele
 - Suchmaschinen
 - Online-Banking
 - E-Shopping
 - Reservierungssysteme



Workflow-basierte Web-Anwendungen

- **Abwicklung von Geschäftsprozessen** ("workflows")
- **Nutzung von Web-Services**
 - gewährleistet Interoperabilität
- **Voraussetzung**
 - Strukturierbarkeit der Vorgänge
- **Beispiele**
 - Business-to-Business (B2B) im E-Commerce-Bereich
 - E-Government-Anwendungen im öffentlichen Bereich
 - Patienten-Workflows im Gesundheitsbereich

Kollaborative Web-Anwendungen

- Kooperationsplattformen (groupware) für
 - unstrukturierte Vorgänge
 - hohen Kommunikationsbedarf
- Beispiele
 - Arbeitsräume
 - Wiki
 - Gruppenterminplanung
 - E-Learning-Plattformen
 - Sitzungsunterstützung, z.B.
 - Chaträume



<http://de.wikipedia.org/wiki/>

Weitere Kategorien

- **Portalorientierte Web-Anwendungen**
 - **Single Point of Access** auf verteilte evt. heterogene Informationsquellen und Dienste
 - Beispiel
 - Unternehmens- & Open source Community-Portale
- **Ubiquitäre Web-Anwendungen**
 - allgegenwärtiger Zugriff auf (personalisierten) Diensten
 - Beispiele
 - Flughafeninformationen auf mobilen Endgeräten
- **Semantisches Web**
 - maschinenlesbares Informationen im Web
 - Wissensmanagement im Web
 - **Vernetzung und Wiederverwendung von Wissen** (Content Syndication)
 - **Finden von neuem relevanten Wissen** (Recommender Systeme)

Entwicklung von Web-Software

- Charakteristika: Dynamik der Entwicklung
 - neue oder überarbeitete Webseiten
 - neue Verlinkungen
 - kontinuierliche Verbesserung der Benutzerschnittstelle
 - Integration von neuen Web-Services
 - Anpassung an neue Standards und Technologien
- Vergleichbar mit den Softwareentwicklungspraktiken um 1970
 - viele spontane und einmalige Entwicklungen
 - basierend auf dem Wissen, der Erfahrung und dem Vorgehensstil individueller Entwickler
 - Wiederverwendung anhand des “Copy&Paste”-Paradigmas
 - keine geeignete Dokumentation der Entwurfsentscheidungen
 - inkrementelle Komplexität und Umfang der Web-Software

Anforderungen an Web-Engineering Ansätze

Web-Engineering ist die Anwendung systematischer und quantifizierbarer Ansätze (Konzepte, Methoden, Techniken und Werkzeuge)

- kurze Entwicklungszeiten
 - durchschnittlich 3 bis 6 Monate
- Koexistenz verschiedener Versionen
- Flexibilität für Wartung
 - Trennung von Inhalt, Navigationsstruktur und Präsentation
 - modellgetriebene automatische Generierung
- Integration
 - heterogene Ergebnisse von unterschiedlichen Mitglieder des Web-Teams
 - Verknüpfung von Web-Anwendungen
 - Wiederverwendung von Komponenten

Vorgehensmodell vs. Methode vs. Technik

- **Vorgehensmodell**
 - beschreibt ein Entwicklungsprozess aus dem organisatorischen Gesichtspunkt (im Großen)
- **(Entwicklungs)methode**
 - stellt das Vorgehen aus dem technischen Gesichtspunkt dar (im Kleinen)
- **(Entwicklungs)technik**
 - stellt Elemente und Regeln zu Verfügung

- **Vorgehensmodell verwendet Methoden**

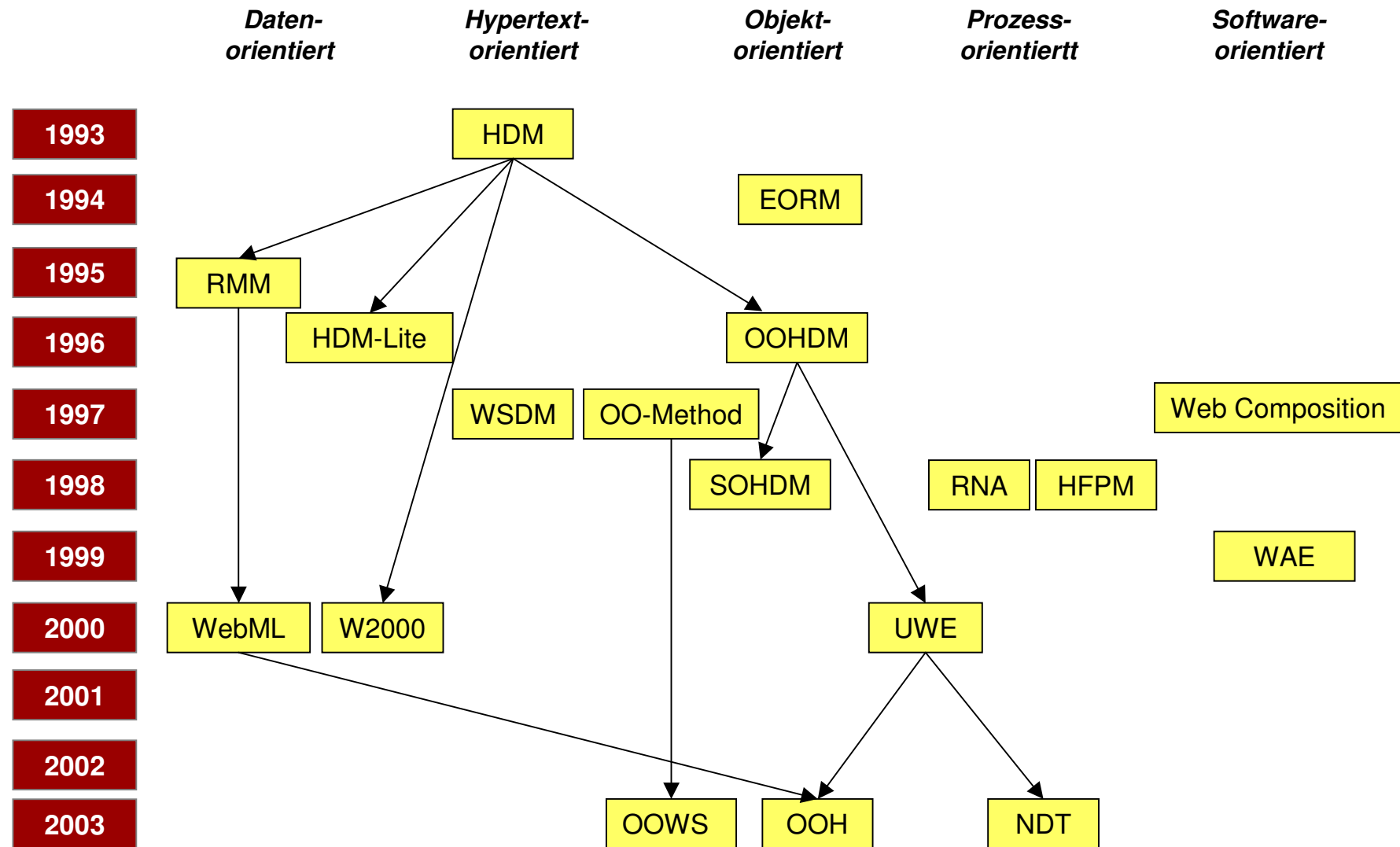
Selben Namen für Methode und Vorgehensmodell

 - Bsp: WSDM, UWE
 - Gegenbsp: HPFM (OOHDM), UX (WAE2)
- **Methode verwendet Techniken**

bewährte Techniken im Software-Engineering

 - Bsp: WebML verwendet ER, UWE verwendet UML
 - Gegenbsp: OOHDM verwendet eigene Diagrammtechniken

Vorgehensmodelle und Methoden



Vorgehensmodelle und Methoden (2)

EORM	Enhanced Object Relationship Model
HDM	Hypertext Design Model
HFPM	Hypertext Flexible Process Modeling
NDT	Navigational Development Technique
OO-H	Object-Oriented Hypermedia Method
OO-Method	Object-Oriented Method
OOHDM	Object-Oriented Hypermedia Design Method
OOWS	Object-Oriented Web Solution
RMM	Relationship Management Methodology
RNA	Relationship-Navigation Analysis
SOHDM	Scenario-based Hypermedia Design Method
UWE	UML-based Web Engineering
W2000	
WAE	Web Application Extension
Web Compostion	
WebML	Web Modeling Language
WSDM	Web Site Design Method

Vorgehensmodelle im Web-Bereich

- Wasserfallmodell-basierte
- Iterative (RUP-basierte)
- Modellgesteuerte
- Agile



„Tailoring“ von allgemeinen Vorgehensmodellen
für die Entwicklung von
Web-Anwendungen

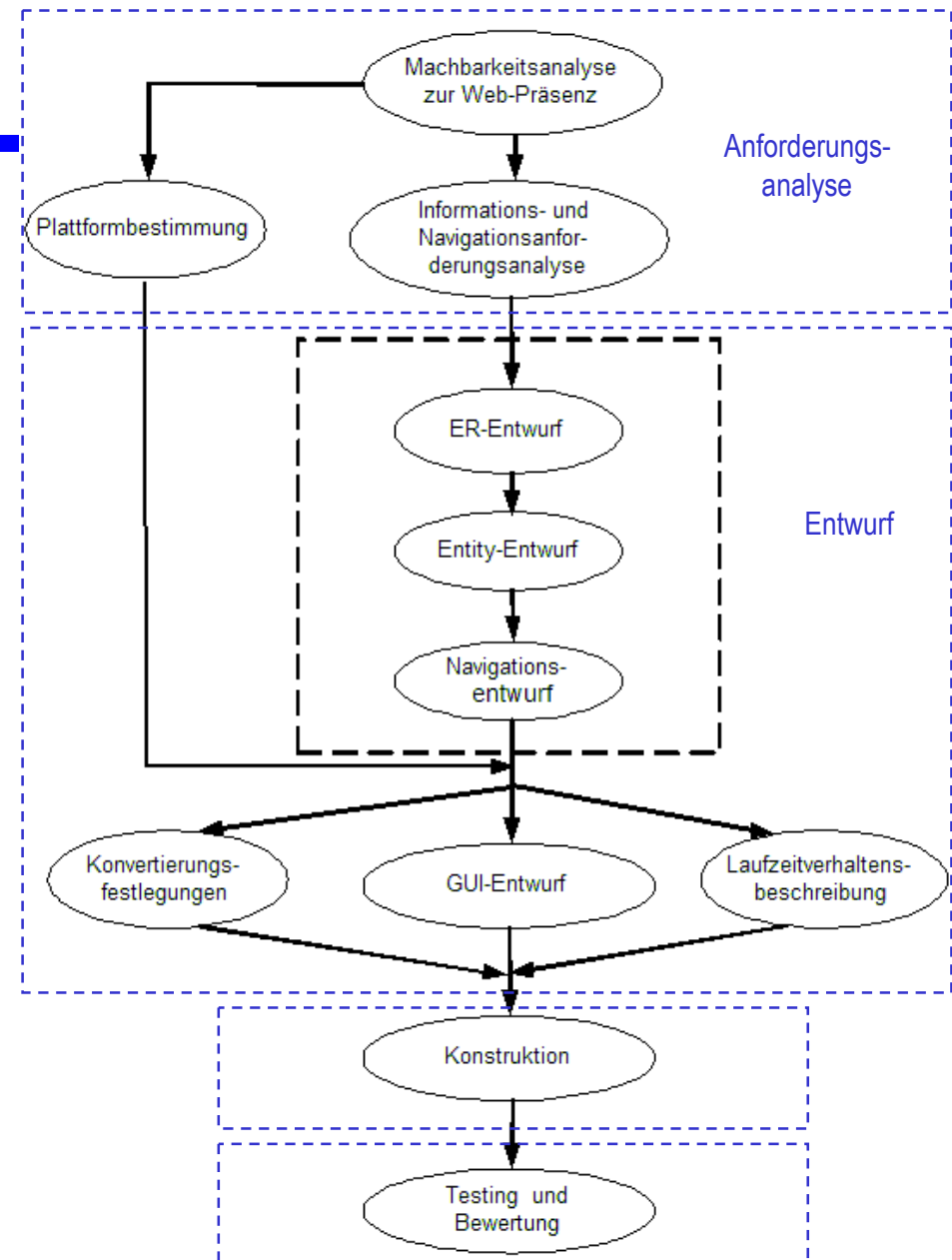
Wasserfallmodell-basierte Vorgehensmodelle

- geeignet für einfache, dokumentenzentrierte Web-Anwendungen
- visuelle Modellierung
- Fokus auf die Entwurfsphase
- Beispiele:
 - RMM: Relationship Management Methodology
 - WebML: Web Markup Language
 - WSDM: Web-System Design Method

RMM

- erster systematischer Ansatz für ein Web-Vorgehensmodell (1995)
- unterteilt in 4 Phasen
- plattform-unabhängiges Entity-Relationship (ER) Entwurfsmodell
- erweitert ER für die Modellierung von Web-Anwendungen
- verwendet Sketches und Storyboarding für den GUI-Entwurf

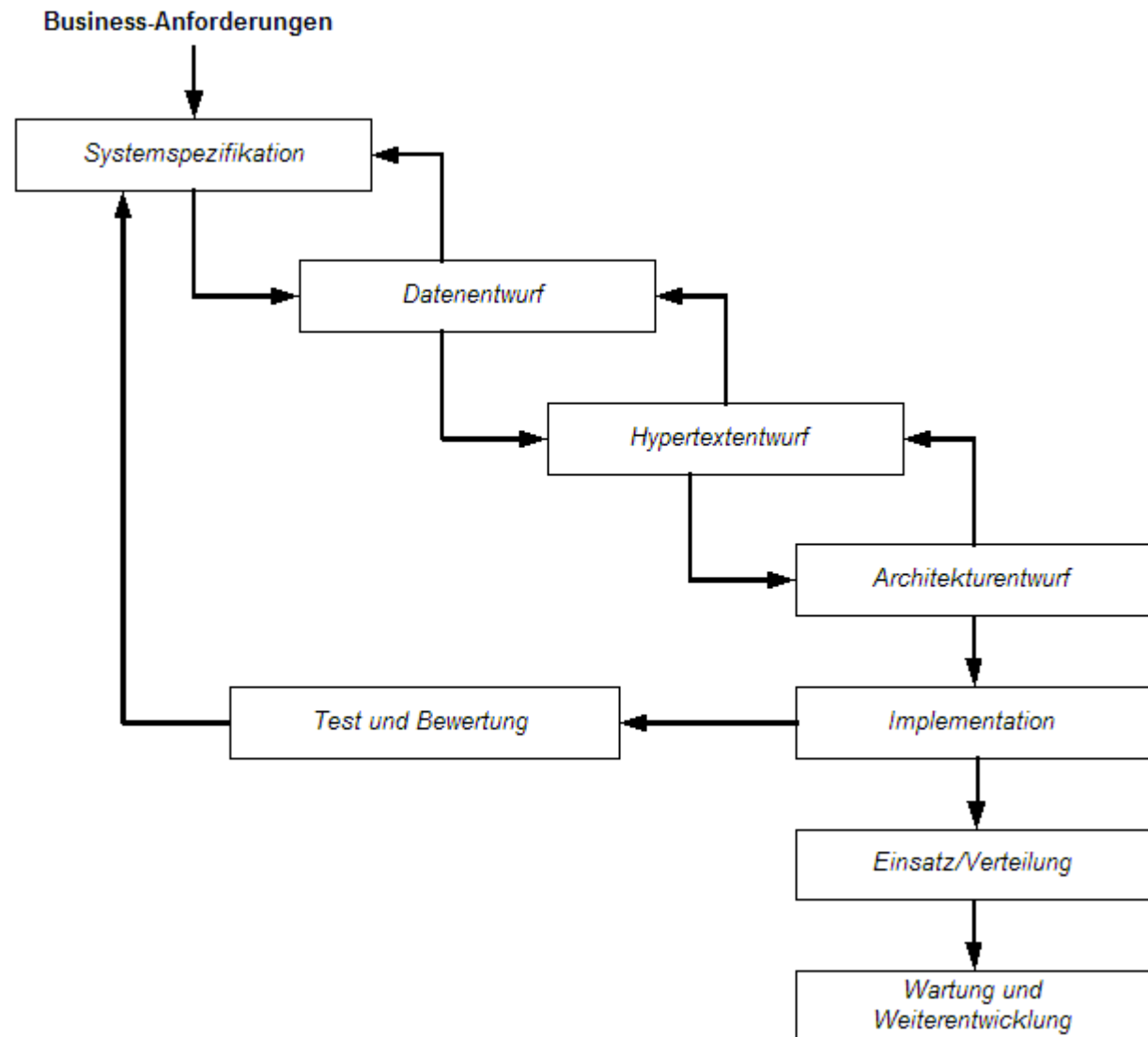
T. Isakowitz, M. Bieber, V. Balasubramanian (1995)



WebML

- visuelle Modellierung
- ER-basierte Notation
- nicht objektorientiert
- XML-basierte Spezifikation
- geeignet für daten-intensive Web-Anwendungen

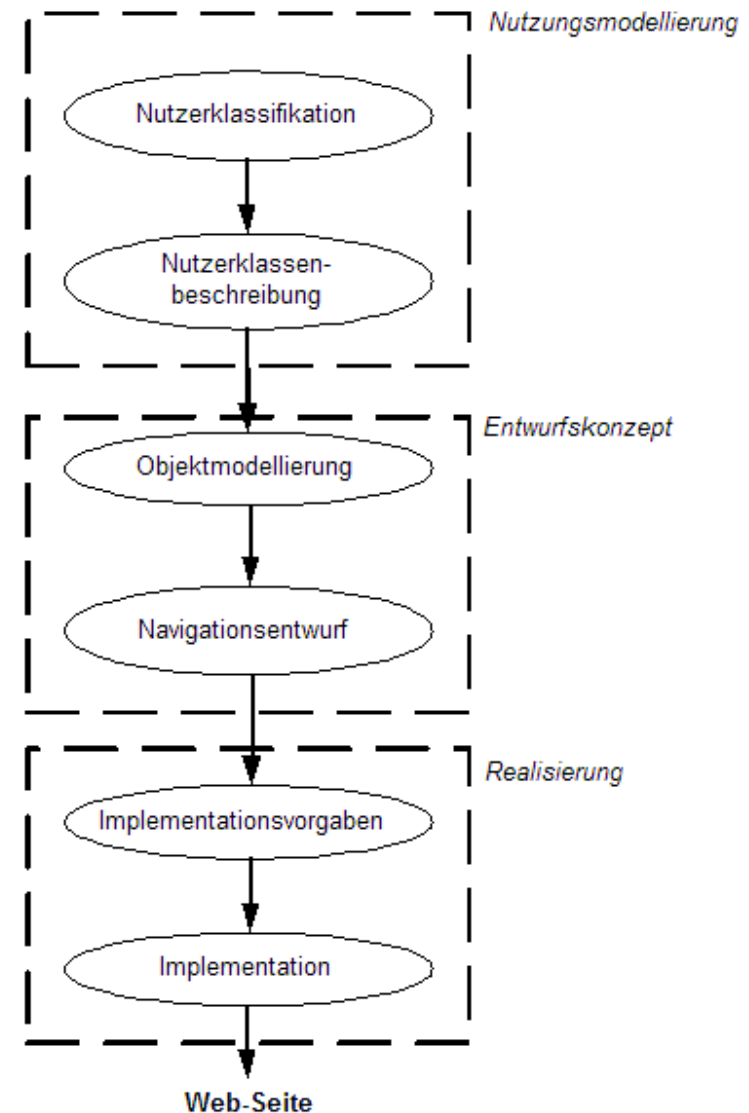
S. Ceri, P. Fraternali (2000-2006)



WSDM

- definiert einen minimalen wasserfallartigen Prozess
- nutzerorientierte Anforderungsanalyse
- zielt auf Entwicklung von Web-Informationssysteme
- objektorientierter Entwurf
 - Modellierung der Sichten
 - Context (Inhalt)
 - Navigation
- keine Präsentationsspezifikation

O. De Troyer (1998)



Iterative Vorgehensmodelle

- geeignet für komplexe Web-Anwendungen
- basieren auf Unified Process
- dokumentieren Entwicklungsprozesses
- Beispiele:
 - WAE2: Web Application Extension
 - UWE: UML-based Web Engineering (erste Version)

WAE2

- WAE2 basiert auf RUP* und ICONIX Unified Process**
- erweitert diese Prozesse um ein UX Modell
 - user experience (UX) sind die Aktivitäten des Teams, die sich um die Benutzerschnittstelle bemühen
 - Modellierung der Bildschirmseiten und dessen Inhalt
 - Storyboard Szenarien
 - Navigationsplan

Conallen, 2003

* Rational Unified Process

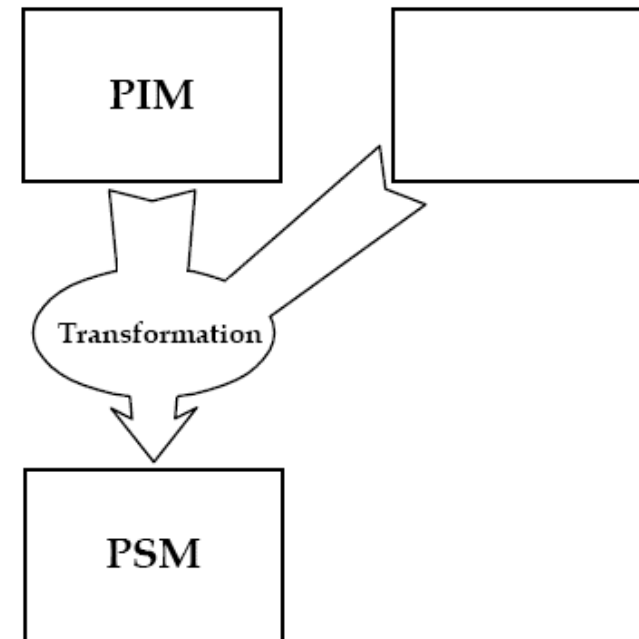
**Rosenberg & Kendall: Use-Case Driven Object Modeling with UML, Addison-Wesley, 2000

Modellgesteuerte Vorgehensmodelle

- basieren auf MDA Prinzipien (Model-Driven Architecture - OMG Standard)
- objektorientiertes Vorgehen
- Anwendungsfall- (Use Cases) gesteuerte Entwicklung
- architekturzentrierter Ansatz
- anwendbar für komplexere Web-Anwendungen (inhaltlich, struktur-technisch, funktional)
- Beispiele
 - UWE: UML-based Web Engineering
 - OOWS: Object-Oriented Web Solution

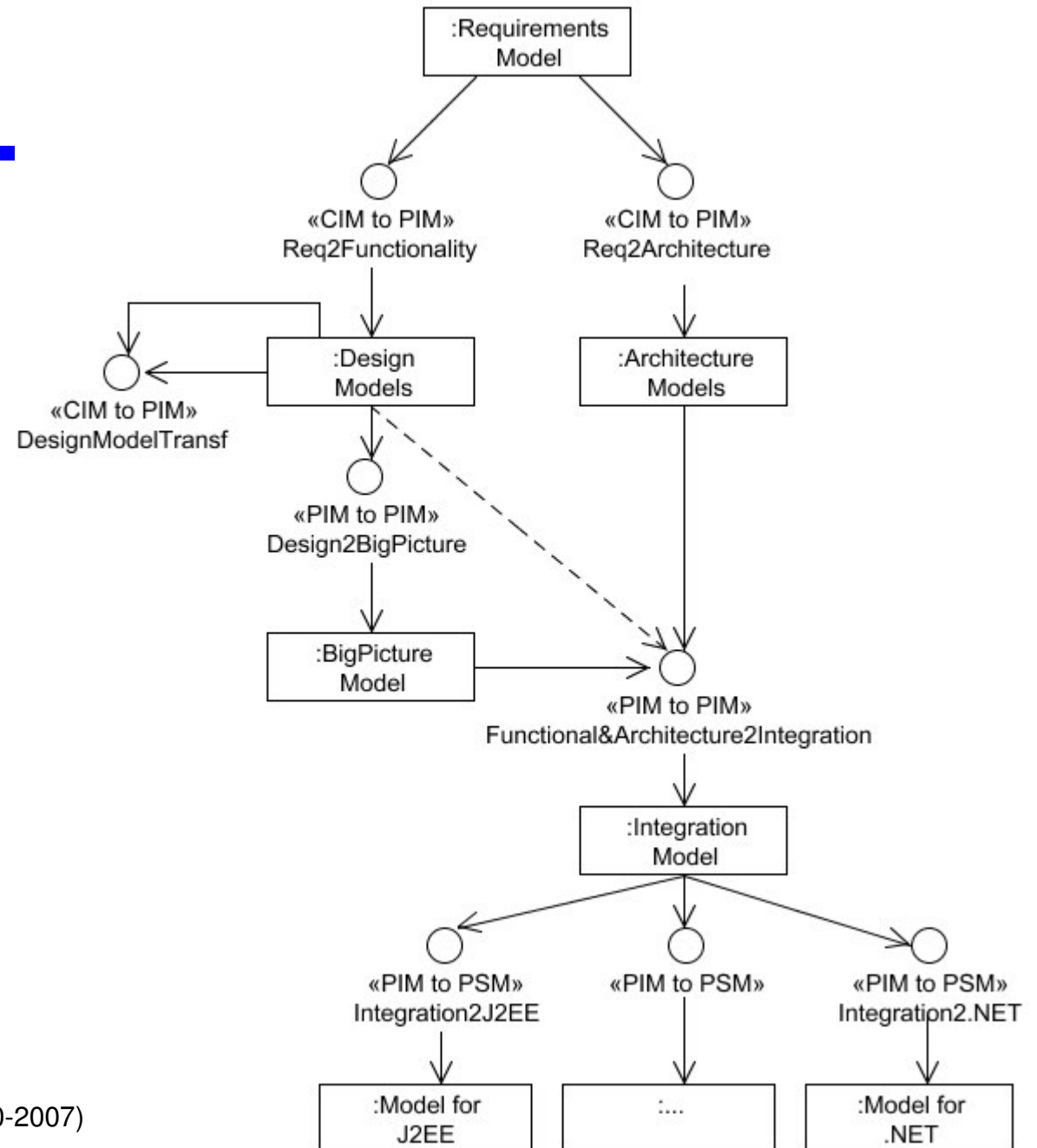
MDA: Model-Driven Architecture

- OMG Standard
- MDE : Model Driven Engineering
- MDD : Model Driven Development
- basiert auf Konstruktion von Modellen und Modelltransformation
- UML Modellierung
- Modellklassifizierung
 - CIM: Computation Independent Model
 - PIM : Platform Independent Model
 - PSM: Platform Specific Model
- Spezifikation von Modelltransformationen
 - Programmiersprachen (Java)
 - Sprachen zur Beschreibung von Graphtransformationen
 - Modelltransformationssprachen (ATLAS, QVT)



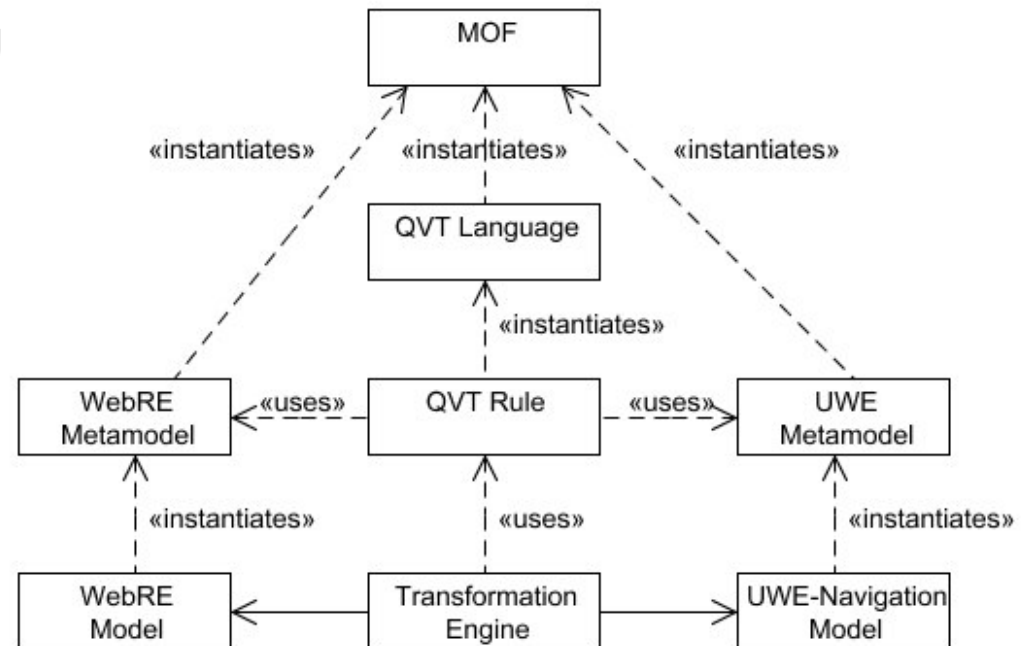
UWE: Modelle

- Konstruktion von Web-spezifischen Modellen
 - Anforderungsspezifikation
 - Design Modelle
 - Content
 - Navigation
 - Präsentation
 - Geschäftslogik
 - Adaptivität
 - Integration (auch PIM)
 - „Big Picture“ (Model checking)
 - „Integration“
 - plattformspezifische Modelle



UWE: Modelltransformationen

- Automatische Generierung
 - Modelle
 - Code
- Unterschiedliche Techniken
 - Java im CASE Tool ArgoUWE
 - Graphtransformationen für Big Picture
 - QVT & ATLAS zur Applikationscode Generierung (JavaBeans/JSP)

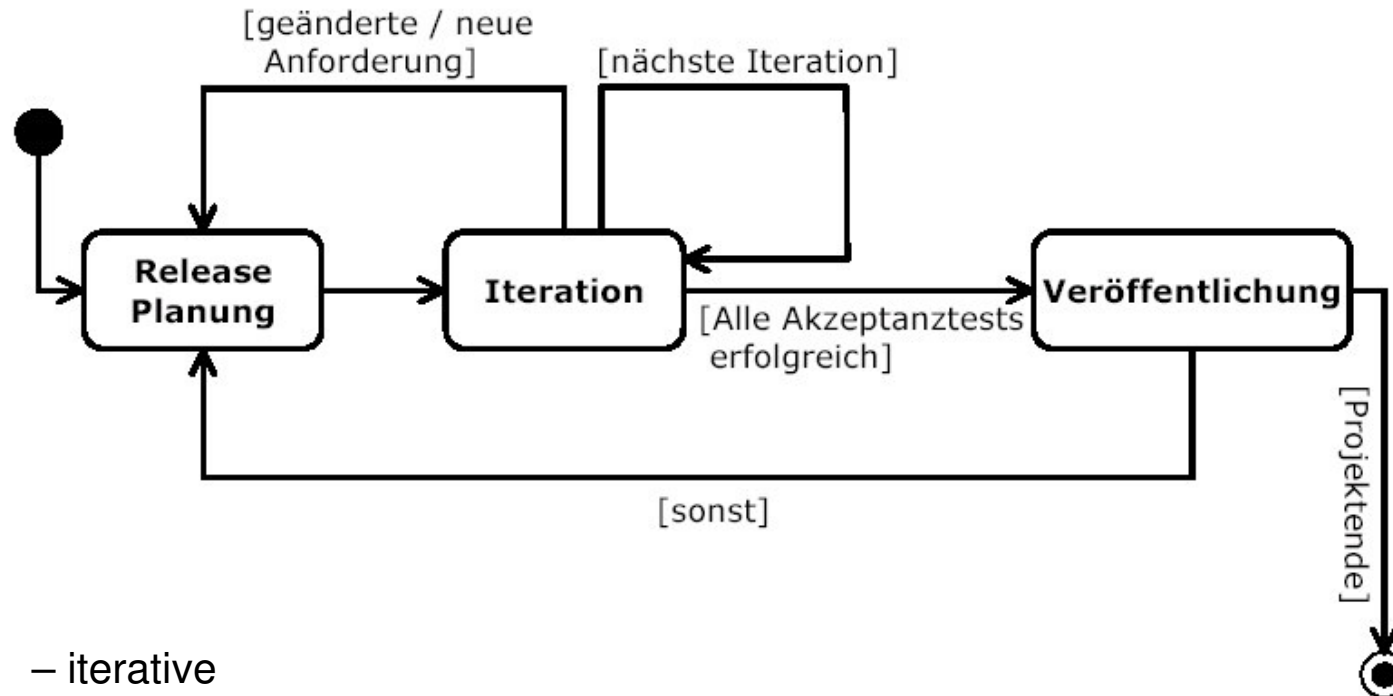


Koch, Knapp, Zhang, .. (2000-2007)

Agiles Vorgehen

- Kommunikation
 - intensiver persönlicher Austausch zwischen den Projektbeteiligten (Programmierung in Paaren)
- Einfachheit
 - wiederholt nach der einfachsten Lösung suchen
- Feedback
 - Grad der Zufriedenheit des Kunden bestimmen
- Mut
 - Verantwortung den Projektbeteiligten übergeben
- Beispiel
 - XP- Extreme Programming

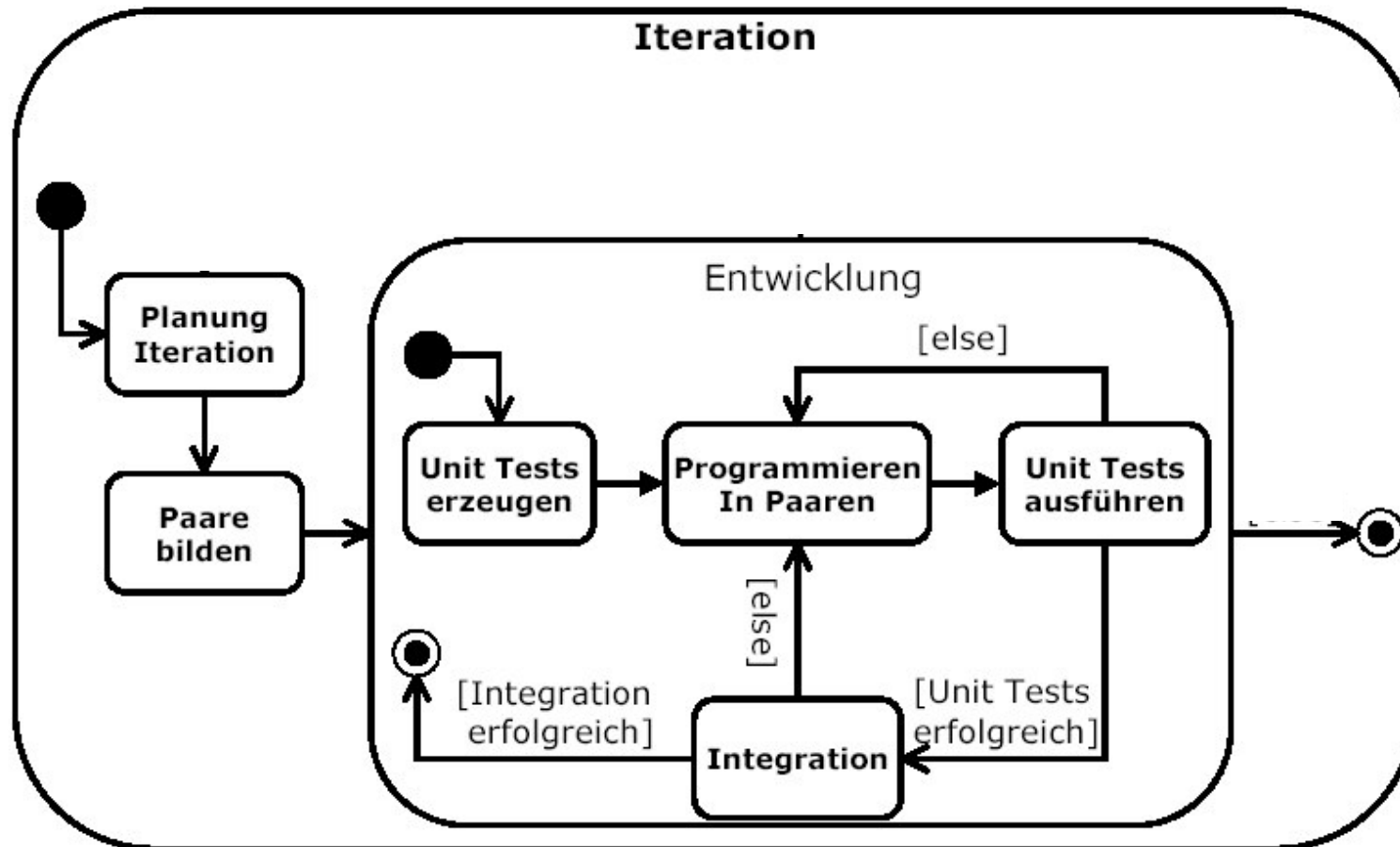
Ablauf eines XP-Projektes



- iterative
- basierend auf user stories
- test first

Engels, Lohmann & Wagner (2003)

Ablauf einer Iteration in XP-Projekten



Engels, Lohmann & Wagner (2003)

Agiles Vorgehen für den Web-Bereich

- geeignet
 - können mit häufigen Änderungen umgehen
 - fördern die Kommunikation im Web-Team
 - machen kurze Entwicklungszeiten möglich
 - gehen iterativ vor
 - Testen steht im Vordergrund
- ungeeignet
 - für komplexere WA und große Web-Teams
 - bei Web-Anwendungen mit hohen Sicherheitsanforderungen
 - bei räumlich verteilter Entwicklung

Projektmanagement im Web-Bereich

- Typische Probleme bei größeren Web-Projekten
 - gesetzten Geschäftsziele werden nicht erreicht (84%)
 - Verzögerungen bei der Projektplanung (79%)
 - Projekt-Budget überschritten (63%)
 - Funktionalität nicht den Anforderungen entsprechend (53%)
 - geringe Qualität der Lieferungen (52%)

Cutter Consortium (November 2000)
www.cutter.com/research/2000/crb001107.html

Risikofaktoren in Web-Projekten

- kurze Entwicklungszyklen
 - rasche technologische Änderungen
 - parallele Entwicklung
 - enge Terminplanung wegen Konkurrenzdruck
 - multidisziplinäres Web-Team
 - Kommunikationsprobleme
 - Akzeptanz der Web-Anwendung
 - Anonymität des Benutzers
 - Ästhetik
 - unbekannte kulturelle Unterschiede
- Risiko der Projektabwicklung*
- soziales Risiko*
- technisches Risiko*
- Risiko des Produktes*
-

Management des Web-Entwicklungsteam

- **Projektleiter-Skills**

- Führungsvermögen eines multidisziplinären Teams
- Schaffung einer Kommunikationsbasis zur Zusammenarbeit
- Fähigkeit zur dauernden Motivation und Konfliktlösung

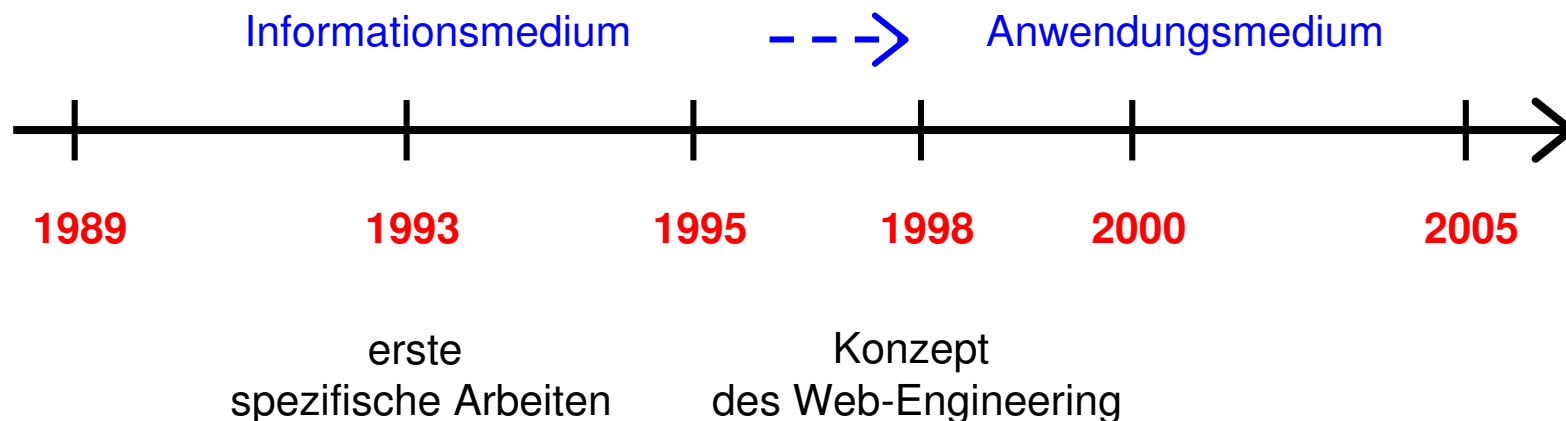


Zusammenfassung

- **Web-Engineering**
 - Definition, Ziele und Entwicklung
 - Bedarf einer eigenen Disziplin
- **Web-Software**
 - Charakteristika
 - Kategorien von Web-Anwendungen
- **Vorgehensmodelle in der Web-Domäne**
 - Modellgesteuerter Prozess
 - Agiles Vorgehen
- **Projektmanagement**
 - Entwicklungsteam
 - Risikofaktoren

Entwicklung des Web-Engineering

- Web-Engineering ist eine junge Disziplin
 - Methoden werden ständig überarbeitet
 - Techniken werden verbessert
 - Werkzeuge haben Prototyp-Charakter
 - zu wenig gesammelte Erfahrungen
 - sehr aktuelles Forschungsgebiet



Ausblick: Block F – Einheit 2

- Entwicklungsprozess im Web-Engineering
- Methoden zur Entwicklung von Web-Anwendungen
 - Systematisches Vorgehen
 - Techniken zur Analyse, Spezifikation, Verifikation und Implementierung
 - Werkzeuge (CAWE: Computer Aided Web Engineering)
 - Praxisbezogene Erfahrungen (best practice)

Literatur

- **Web Engineering – Systematische Entwicklung von Web-Anwendungen**
Gerti Kappel, Birgit Pröll, Siegfried Reich & Werner Retschitzegger
dpunkt.verlag, 2003
überarbeitete Englische Version John Wiley, 2006
- **Web Engineering**
Reiner Dumke, Matthias Lothar, Cornelius Wille & Fritz Zbrog
Pearson, 2003
- **Hypermedia & the Web – An Engineering Approach**
David Lowe & Wendy Hall
John Wiley & Sohn, 1999