

Komponentenorientierte Vorgehensmodelle im Vergleich

Peter Fettke, Iulian Intorsureanu, Peter Loos

Juni 2002

Technische Universität Chemnitz
Professur Wirtschaftsinformatik II

peter.fettke@isym.tu-chemnitz.de, iintorsu@infoc.ase.ro, loos@isym.tu-chemnitz.de

4. Workshop Komponentenorientierte betriebliche Anwendungssysteme (WKBA 4)
GI-Arbeitskreis 5.10.3 "Komponentenorientierte betriebliche Anwendungssysteme"
11. Juni 2002, Universität Augsburg

Gliederung

- Einführung
- Überblick über die betrachteten Vorgehensmodelle
- Gegenüberstellung und Vergleich der Vorgehensmodelle
- Resümee und Ausblick

■ Ausgangssituation

- Bedeutung der komponentenbasierten Softwareentwicklung
- Ingenieurmäßige Entwicklung erfordert Vorgehensmodelle
- Zahlreiche Vorgehensmodelle bekannt

■ Problemstellung

- Erhebung von komponentenbasierten Vorgehensmodellen
- Beschreibung der Vorgehensmodelle
- Systematischer Vergleich

Betrachtete Vorgehensmodelle

Vorgehensmodell: mustergültige Beschreibung eines Entwicklungsprozesses

Auswahlkriterien:

- Explizite Komponentenorientierung
- Weites Aktivitätsspektrum, ausreichende Detaillierung
- Vorgehensmodell-Charakter

Bezeichnung	Autor(en)	Erscheinungs-jahr	Quelle(n)
Catalysis	D'Souza; Wills	1998	[DSWi1998], www.catalysis.org
Perspective	Allen; Frost	1998	[AIFr1998]
Rational Unified Process 2000	Rational Software	2001	[Rati2001]
V-Modell '97	Öffentliche Hand	1997	[o.V.1997a]

Überblick: Catalysis



■ Merkmale:

- iterative, inkrementelle, parallele und nicht-lineare Entwicklung
- Produktfamilien-Ansatz

■ Prozess wird aus Prozess-Muster gestaltet

- Unterschiedliche Granularität der Muster
- Beispiele: Short-Cycle Development, Factories

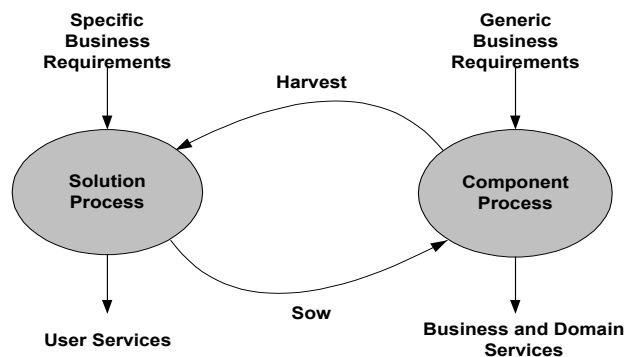
■ Wenige Vorschriften bezüglich formaler Spezifikation, jedoch Betonung der Qualität einer Spezifikation

Überblick: Perspective



■ Merkmale:

- Getrennte Prozesse für Komponenten- bzw. Systementwicklung
- Iterativ und inkrementell
- basiert auf „best practices“



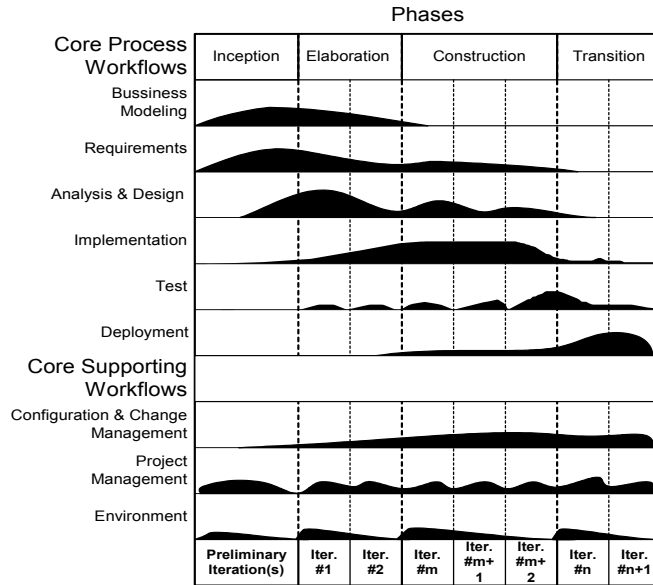
Überblick: Rational Unified Process



Merkmale:

- zwei-dimensional strukturiert
- Iterativ und inkrementell

Prozessrahmenwerk

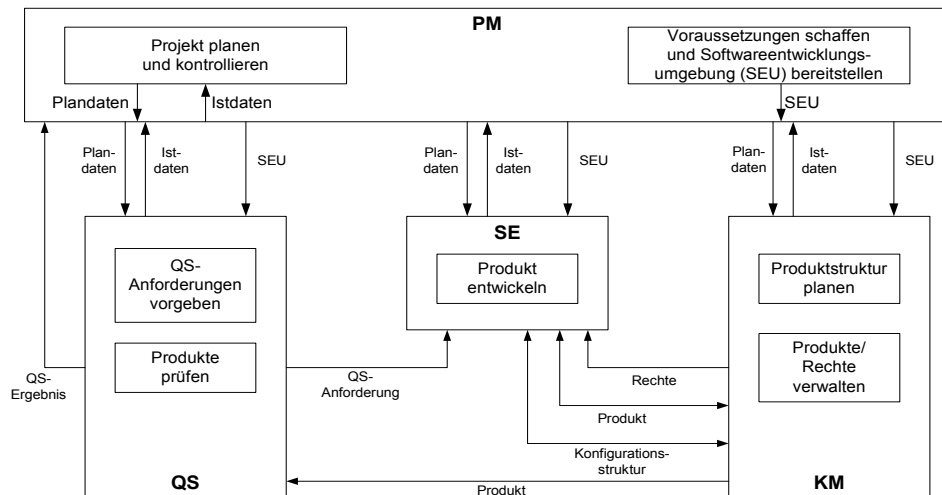


ISYM – Information Systems & Management

Überblick: V-Modell



Komponenten-Szenario: „Einsatz von Fertigprodukten“



ISYM – Information Systems & Management

Vergleichsrahmen



- **Klassifizierung**
- **Terminologie**
- **Komponentenbegriff**
- **Abdeckung des Lebenszyklus einer Komponente**
- **Abdeckung der Tätigkeitsbereiche**
- **Prozessarchitektur**
- **Prozesssteuerung**
- **Rollenabdeckung**
- **Adaption**

Vergleichsrahmen: Klassifizierung



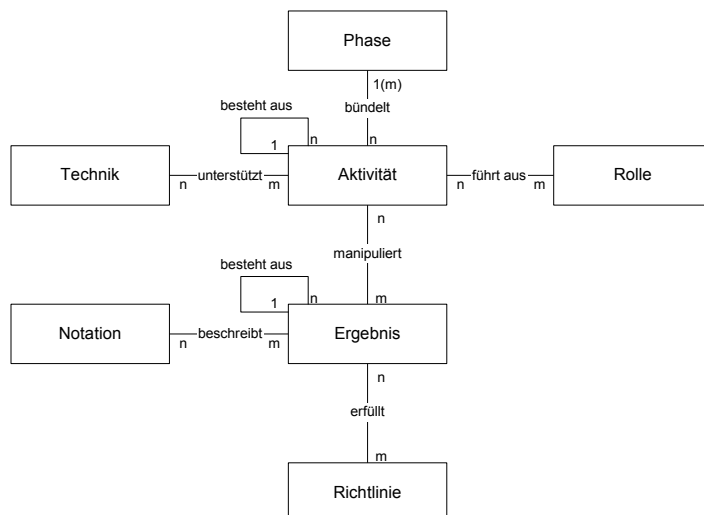
- **Art**
 - Evolutionär
 - Revolutionär
- **Komponentenbildung**
 - Top-Down
 - Bottom-Up
- **Systemgranularität**
 - Entwicklung einzelner Komponenten
 - Systementwicklung

Vergleich: Klassifizierung



	Catalysis	Perspective	RUP	V-Modell
Art	revolutionär	revolutionär	evolutionär	evolutionär
Komponentenbildung	Top-Down und Bottom-Up	Bottom-Up	Top-Down	Top-Down
Systemgranularität	Anwendung	Komponente und Anwendung	Anwendung	Anwendung

Vergleichsrahmen: Terminologie



Grundlegende Terminologie (Quelle: [NoSc1999, S. 169])

Vergleich: Terminologie



	Catalysis	Perspective	RUP	V-Modell
Phase	-	Stage	Phase	Phase
Aktivität	Activity	Task	Workflow, Activity, Step	Aktivität, Teilaktivität
Ergebnis	Deliverable	Deliverables, Inputs, Outputs	Artifact	Produkt, Teilprodukt
Rolle	-	Team Role	Worker	Rolle
Technik	Technique	Technique	-	Elementar-methode
Richtlinie	-	Practical Guidelines	Work Guidelines	Handbuchsammlung
Notation	Notation	Notation	Language	-

Vergleichsrahmen: Komponentenbegriff



- **Zusammensetzung aus Software-Artefakte**
- **Wiederverwendbarkeit**
- **Abgeschlossenheit**
- **Vermarktbarkeit**
- **Dienstangebote**
- **Wohldefinierte Schnittstelle**
- **Verbirgt die Realisierung**
- **Kombinierbarkeit**

Vergleich: Komponentenbegriff

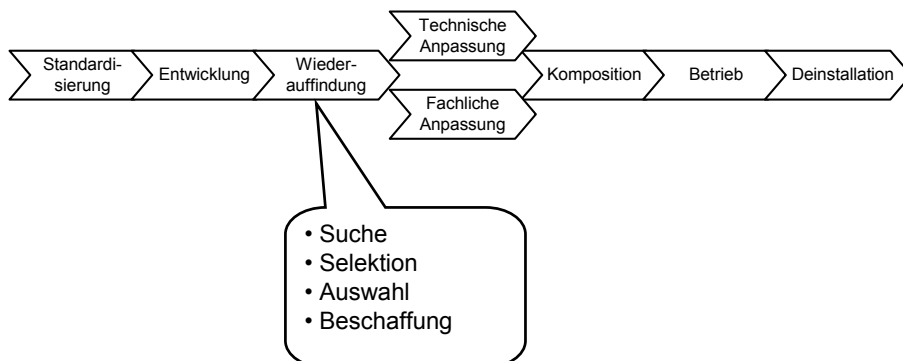


	Catalysis	Perspective	RUP	V-Modell
Bezeichnung	Component	Component	Component	Fertigprodukt
Software-Artefakte	(+)	(+)	(+)	(+)
Wiederverwendbarkeit	(+)	+	+	+
Abgeschlossenheit	+	+	(+)	(+)
Vermarktbarkeit	?	?	?	(+)
Dienstangebote	+	+	+	?
Wohldefinierte Schnittstelle	+	+	+	?
Verbirgt ihre Realisierung	(+)	+	?	?
Kombinierbarkeit	(+)	(+)	?	?

Legende: + Merkmal explizit angeführt, (+) Merkmal implizit unterstellt, ? keine Aussage möglich

n Systems & Management

Vergleichsrahmen: Abdeckung des Lebenszyklus



Lebenszyklus einer Komponente (in Anlehnung an [Turo1999, S. 13])

Vergleich: Abdeckung des Lebenszyklus



	Catalysis	Perspective	RUP	V-Modell
Standardisierung	How to Specify a Component	-	-	-
Entwicklung	How to Implement a Component	Component Process	Process Workflows	Systemerstellung
Wiederauffindung	-	(Search for possible areas of reuse)	(Project Management)	Realisierbarkeitsuntersuchung, Marktsichtung
Technische Anpassung	Reuse and Pluggable Design Frameworks in Code	-	-	(System-konfektion)
Fachliche Anpassung	-	-	-	(System-konfektion)
Komposition	Components and Connectors	(Roll Out)	Deployment	Integration
Betrieb	-	-	-	-
Deinstallation	-	-	-	-

Vergleichsrahmen: Abdeckung der Tätigkeitsbereiche



- **Projektmanagement**
- **Konfigurationsmanagement**
- **Qualitätsmanagement**
- **Systementwicklung**

Vergleich: Abdeckung der Tätigkeitsbereiche



	Catalysis	Perspective	RUP	V-Modell
Projektmanagement	-	-	Project Management	Projektmanagement
Qualitätsmanagement	-	-	-	Qualitätssicherung
Konfigurationsmanagement	-	-	Configuration & Change Mgmt	Konfigurationsmanagement
Entwicklung	Catalysis Process	Perspective Process	Core Process Workflows	Systemerstellung

Vergleichsrahmen



■ Prozessarchitektur

Anordnung und Ablauf der Phasen und Aktivitäten

■ Prozesssteuerung

- Aktivitätsorientiert
- Ergebnisorientiert
- Entscheidungsorientiert

■ Rollenabdeckung

Für die komponentenorientierte Entwicklung bedeutsame Rollen

■ Adaption

Anpassungsmöglichkeiten

Vergleich: Prozesssteuerung



	Catalysis	Perspective	RUP	V-Modell
Prozesssteuerung	entscheidungsorientiert	aktivitätsorientiert	aktivitätsorientiert	aktivitätsorientiert

Vergleich: Rollenabdeckung



- **Catalysis**
 - keine Rollen
- **Perspective**
 - Reuse Identifier, -Librarian, -Assessor, -Architect
- **RUP**
 - vermutlich: Architect, Designer, Implementer, Integrator
- **V-Modell**
 - spezifische Rollen für komponentenorientierte Entwicklung nicht definiert

Vergleich: Adaption



- **Catalysis**
 - Adaption per Prozess-Muster
- **Perspective**
 - Adaption empfohlen, aber nicht konkret beschrieben
- **RUP**
 - Environment-Arbeitsablauf: Anpassung zur Organisation
 - Roadmaps: Anpassung zum Entwicklungsumfeld
- **V-Modell**
 - Entwicklungsszenarien
 - Leitfaden

Resümee und Ausblick



- **Vorgehensmodelle haben jeweils spezifische Stärken und Schwächen**
- **Entscheidung über die Auswahl eines Vorgehensmodells benötigt weitere Untersuchungen**
- **spezifische Vorgehensmodelle je nach Handlungsoption**
 1. Vorgehensmodell für Komponentenentwickler
 2. Vorgehensmodell für Komponentenassemblierer
 3. Vorgehensmodell für Komponentengeneralist



- **Vorgehensmodelle haben jeweils spezifische Stärken und Schwächen**
- **Keine eindeutige Präferenz für ein Vorgehensmodell**
- **Spezifisches Vorgehensmodell je nach Handlungsoption**
 1. **Vorgehensmodell für Komponentenentwickler**
 2. **Vorgehensmodell für Komponentenassemblierer**
 3. **Vorgehensmodell für Komponentengeneralist**