

Vorlesung
Software Architektur-Modelle
Prozesse 2: moderne Vorgehensmodelle, Prozeßverbesserung

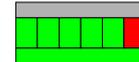
Dr. Harald Störrle

Ludwig-Maximilians-Universität München
Wintersemester 2001

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



1
13

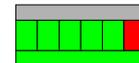
**1) Ein paar Nachträge vom
letzten Mal**

2) Leichtgewichtige Prozesse

22.1.2002
„Leichte“ Prozesse

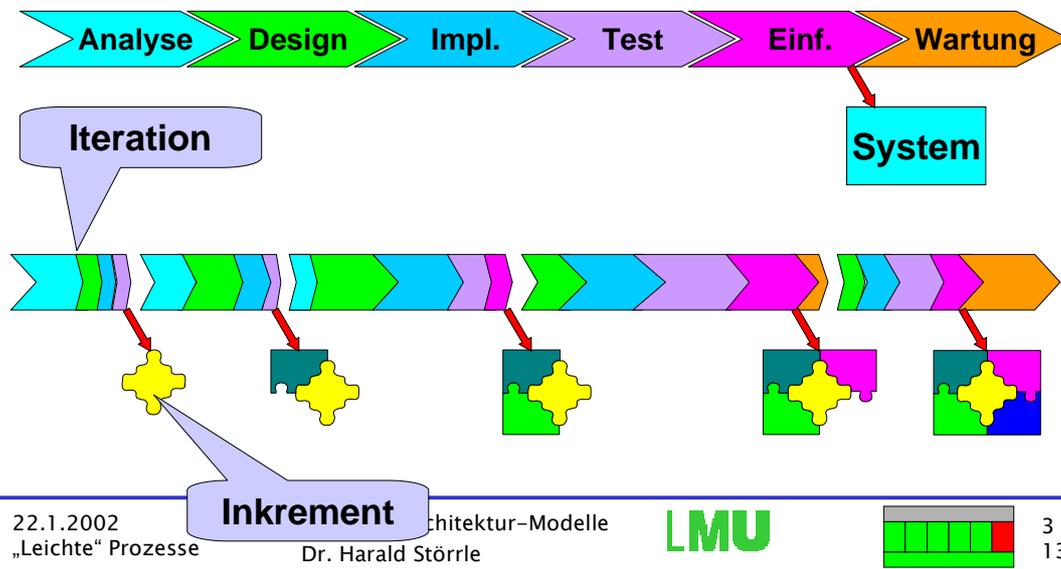
VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



2
13

Abwägung Wasserfall/Big Bang vs. Iterativ/Inkrementell



Browser für RUP und VM



22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



4
13

Wichtige, aber vernachlässigte Prozesse („Prozessweisen“)

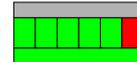
- Außenbeziehungen („Customer/Supplier“)
 - Aufwandsschätzungen
 - Angebotserstellung
 - Vertragsgestaltung (Seitenabsprachen, Schiedsgerichte, Klagewege, Mediation,...)
 - Auslieferung
 - Abnahme
 - Betrieb
 - Service-Level-Agreements
 - Fehler- & Anforderungsverwaltung
 - Architekturmigration
- Management („Project“)
 - Releaseplanung
 - Produktlinien Änderungsverwaltung
 - Zuliefer-Verwaltung
 - Architekturevaluierung

Hier wäre Prozessverbesserung besonders wichtig!

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



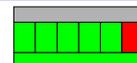
5
13

Beispiel 1: Die Methode „Inspektion“

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



6
13

Beispiel: Methode „Inspektion“ Ziel und Anwendungsbereich

- Ziel:
 - Auffinden, Klassifizieren, Beheben von Fehlern im Prüfgegenstand
 - Validation

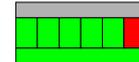
- Anwendbar auf:
 - im Prinzip auf alle Artefakte für die es ein Größenmaß gibt
 - (Programm: LoC, Entwurf: NoM, Textdokument: Seiten)
 - Kalibrierung/Bezugsgröße: Sprache, Stil- & Formatvorgaben
 - Inspektionsgruppe muß kompetent sein (doppelter Wortsinn)
 - mittlere und große Projekte, sehr frühe bis mittlere Phasen

- Einfache, aber wirkungsvolle Methode

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



7
13

Beispiel: Inspektion Durchführung

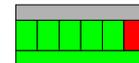
Schritt	Wer	Aktivitäten, Umfang, Dauer
Planung	M, A (, I)	Vorbedingungen prüfen Termine & Gegenstand festlegen
Überblick	M, A, I ₁ , ..., I _n	Autor stellt Gegenstände vor (ca. 1h) Gruppenabstimmung
Vorbereitung	M, I (, A)	Durchlesen (125NLoC/h·Person) Checklisten abarbeiten
Inspektion	M, A, I	L. trägt vor, I. en unterbrechen, A. erläutert (90-125 NLoC/h, max.2h) Protokoll durch Moderator
Korrektur	A	gemäß Protokoll (nur diese Fehler!)
Abnahme	M, A	ggf. weitere Sitzung

3-5h

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



8
13

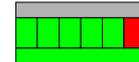
Beispiel: Inspektion Hinweise, Varianten

- Alle Rollen werden aus dem Projekt besetzt (wechselseitige Begutachtung verhindert Ausreißer).
- Kann bei sehr kleinen Projekten schwierig sein. Dafür gibt es („Walk Through“): Autor=Moderator=Leser, Inspektor stellt Fragen
- Re-Inspektion notwendig bei:
 - Korrekturen verändern Prüfgegenstand um mehr als 5% (z.B. NLoC/δNLoC)
 - #Fehler > 2•(φ frühere Inspektionen) ≈ 3–20 funktionale Fehler/1000 LoC
 - Killer-Fehler
- Beachte: Grundlagen für quantitative Erfassung werden gelegt!
 - Zeitreihen von Fehlerdichten
 - Raten/Aufwände von Reparaturen

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



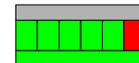
9
13

Beispiel 2: Die Methode ATAM

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



10
13

Beispiel: ATAM

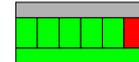
Architecture Tradeoff Analysis Method

- Eine Inspektion
 - der Entwurfsalternativen (nach Vision, vor Entwurf)
 - des Architektur-Entwurfs (nach Entwurf, vor Realisierung)
- Anpassungen erforderlich bei
 - Analysethemen und zu prüfende Standards (-> Design-Richtlinien)
 - Checklisten
 - Analysemethoden
 - Mengenmaße/-zahlen (-> Loc? NoM!)
 - Kosten/Nutzenmaße (offen!)
- Durchzuführen entweder von der Architekturgruppe, oder einer QS-Gruppe.

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU

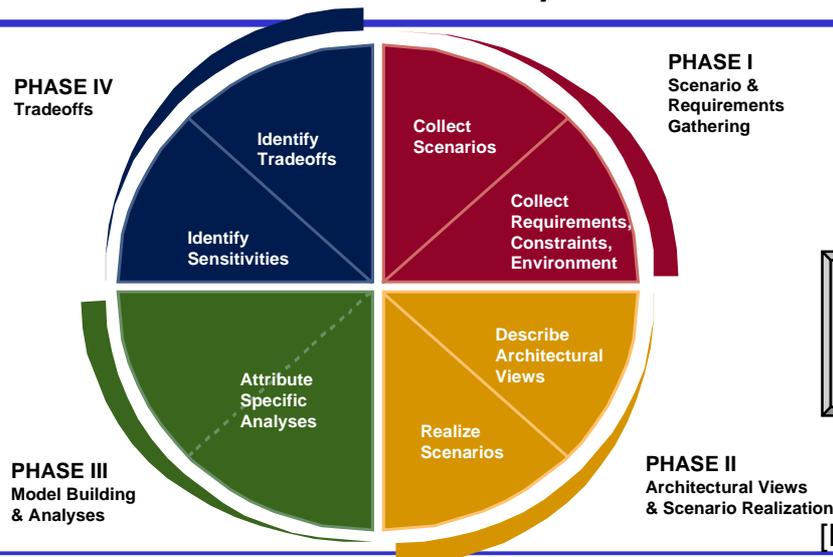


11
13

Beispiel: ATAM

Architecture Tradeoff Analysis Method

[D. Garlan]

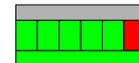


[D. Garlan]

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



12
13

Beispiel: ATAM Analysethemen

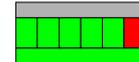
- **Simplicity of architecture**
 - uniformity, use of standards, shared infrastructure
- **Central role for error handling**
 - should be an arch driver, not an afterthought
- **Operations, Administration, and Management (OAM)**
 - often determines feasibility of architecture
- **Performance handled systematically, but at high level**
 - process boundaries made explicit in design
 - use back-of-envelope calculations to determine throughputs, bottlenecks, points of failure
 - allocate resource budgets to components
- **Monitoring capabilities**
 - to permit visibility into running system

[D. Garlan]

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



13
13

Availability Scenarios



- **The availability requirements and analysis are guided by two scenarios:**
 - A server suffers a software failure and is rebooted.
 - A server suffers a power supply failure and is replaced.
- **Availability requirements**
 - **AR1**: Server must not be unavailable for more than 60 minutes per year.

[D. Garlan]

22.1.2002
„Leichte“ Prozesse

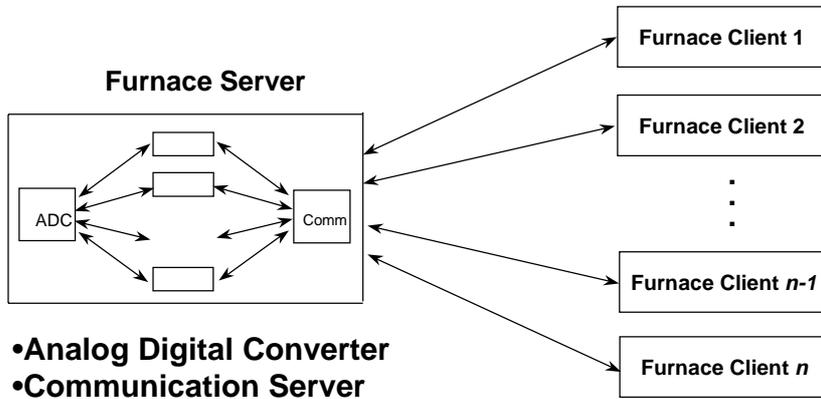
VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



14
13

Option 1: Client-Server

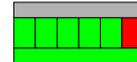


alle Kästchen sind *Prozesse*

[D. Garlan]

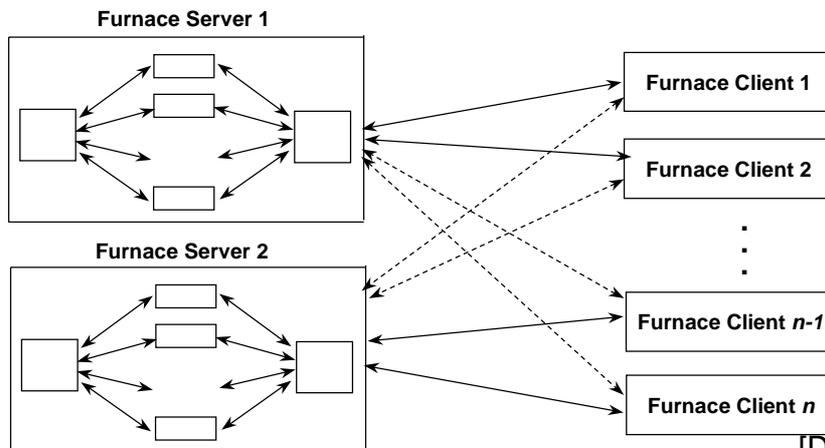
22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle



15
13

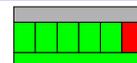
Option 2: Client-Server-Server



[D. Garlan]

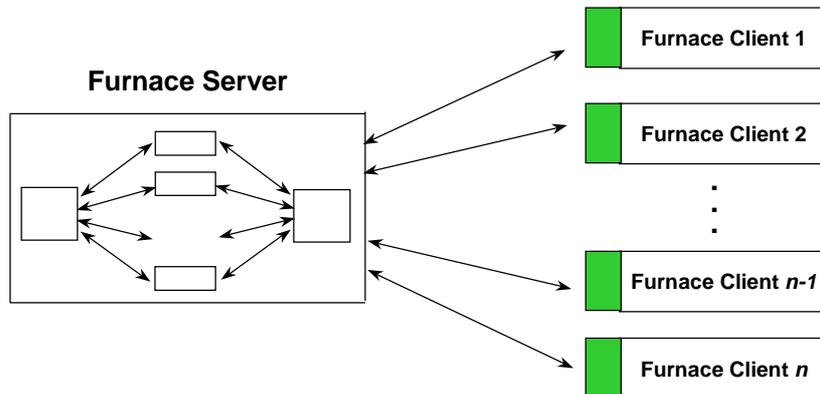
22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle



16
13

Option 3: Client-Cache-Server

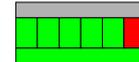


[D. Garlan]

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



17
13

Vergleich der Verfügbarkeit mit Markov-Modellen



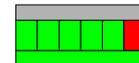
- Für jede Option werden Fehler- und Reparaturraten angenommen:
 - Server-Ausfallrate: $[0 \dots 24]$ p.a.
 - Reparatur-Rate: $[0, 1 \dots 12\text{h}]$ Warmstart...Servicemann
- damit werden Markov-Modelle aufgestellt
- Aber woher kommen die konkreten Zahlen?
 - a) Wissen
 - vergleichbare Systeme, z.B. der Vorläufer
 - Ziele
 - Vorgaben durch Hardware
 - b) Schätzen
 - c) Errechnen (siehe vorletzte Vorlesung)

[D. Garlan]

22.1.2002
„Leichte“ Prozesse

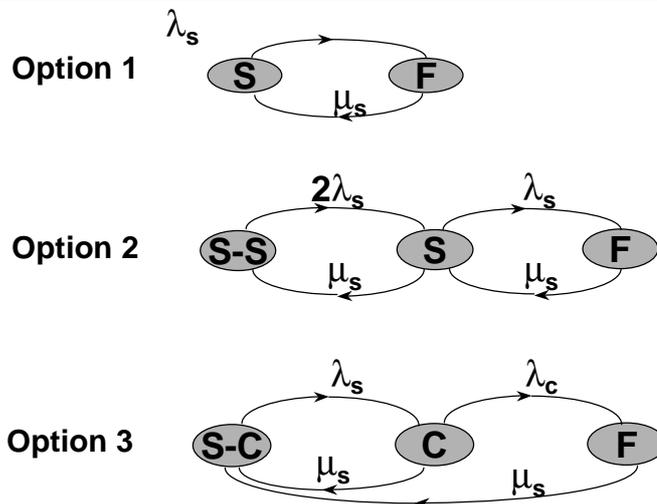
VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



18
13

Vergleich der Verfügbarkeit mit Markov-Modellen



Bedeutung, Analyse etc.:
siehe Vorlesung vom 15.1.
(oder Garlans Vortrag)

[D. Garlan]

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



19
13

Beispiele: Industrielle Prozesse (klassisch)

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



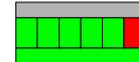
20
13

RUP und VM97 kamen schon jetzt noch ISO 12207 und 15504

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



21
13

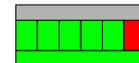
Industrielle Vorgehensmodelle 3: ISO 12207

- 4 Gruppen mit 18 Prozessen
- Prozesse haben Activities, diese haben Tasks
- Sehr detailliert gegliedert und beschrieben, sehr umfassend
- modern, und international standardisiert

22.1.2002
„Leichte“ Prozesse

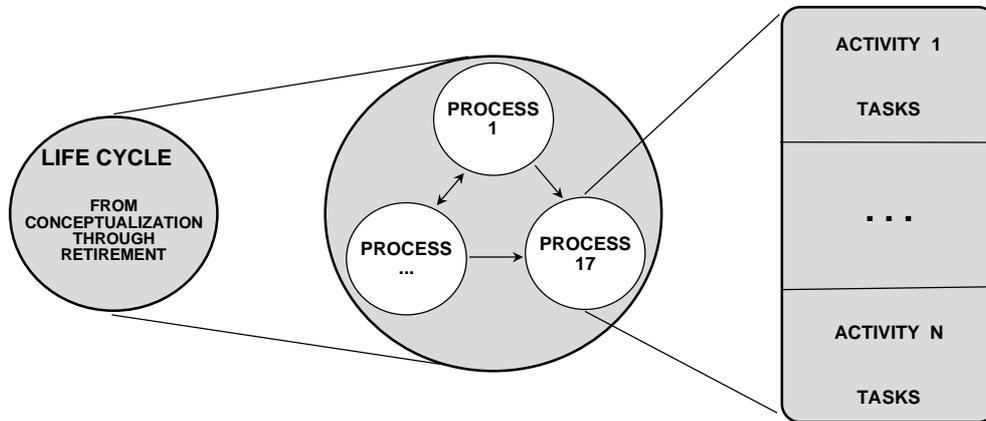
VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



22
13

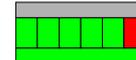
Industrielle Vorgehensmodelle 3: ISO 12207



22.1.2002
„Leichte“ Prozesse

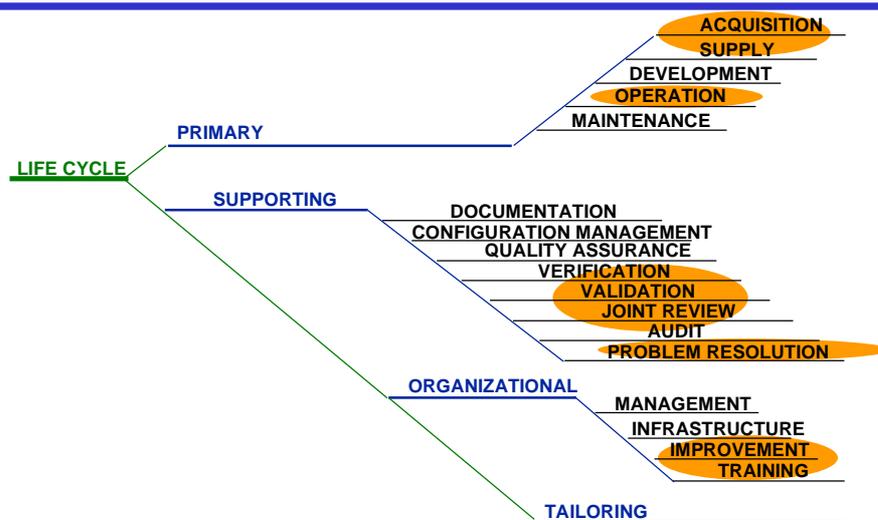
VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



23
13

Industrielle Vorgehensmodelle 3: ISO 12207



22.1.2002
„Leichte“ Prozesse

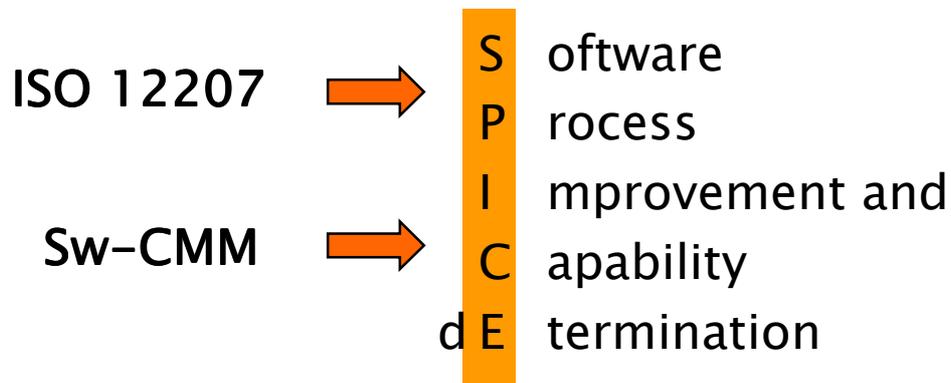
VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



24
13

Industrielle Vorgehensmodelle 4: ISO 15504 (SPICE)

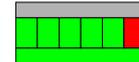


Moderner, terminologisch reifer, feinkörniger und flexibler als CMM

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



25
13

Industrielle Vorgehensmodelle 4: ISO-15504

5 Prozesskategorien

(P. category, process, base practice)

CUS : Customer-Supplier Relations

ENG : Engineering

PRO : Project

SUP : Support

ORG : Organization

6 Reifegradstufen für jeden einzelnen Prozess

0 – not performed

1 – performed informally

2 – planned and tracked

3 – well defined

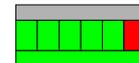
4 – quantitatively controlled

5 – continuously improving

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



26
13

Nächstes Thema: Prozessverbesserung

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



27
13

Prozessverbesserung: Das Capability Maturity Model (CMM)

- 1987, SEI, MITRE Corp.
- Anforderung aus dem DoD zur Zertifizierung von Zulieferern
- CMM v1.1 (v2.0 seit längerem in Vorbereitung)

- 5 Reifegradstufen

- Key Process Areas (KPAs), assoziiert mit Reifegradstufen

- Total Quality Management (TQM) auf Software angewandt

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



28
13

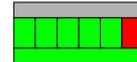
Prozessverbesserung: Die verschiedenen CMMs

- SPI
- verschiedene Arten von CMM
 - Sw
 - People
 - Sw Acquisition
 - Integrated Product Development
 - Systems Engineering
 - Test
 - ...

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



29
13

Prozessverbesserung: Die CMM-Reifegradstufen

Reifegradstufen für *Organisationen*

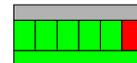
- 1 – Initial
- 2 – Repeatable
- 3 – Defined
- 4 – Managed
- 5 – Optimizing

Zu jeder Reifegradstufe sind „Key Process Areas“ (KPA) definiert, die erfüllt werden müssen, um eine Zertifizierung zu erreichen

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



30
13

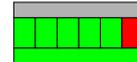
Prozessverbesserung: CMM Reifegradstufe 1 (Initial)

- Sw. Is characterised as ad hoc, and occasionally even chaotic. Few processes are defined, and success depends on individual effort and heroics
- 70% aller Organisationen in Deutschland (Umfrage des FhG IESE, Kaiserslautern, 2001)

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



31
13

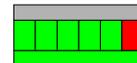
Prozessverbesserung: CMM Reifegradstufe 2 (Repeatable)

- Basic project management processes are established to track cost, schedule, and functionality. The necessary process discipline is in place to repeat earlier successes on projects with similar applications.
- KPAs
 - Requirements management
 - Sw. Project planning
 - Sw.P. tracking and oversight
 - Sw. Subcontract management
 - Sw. Config. management

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



32
13

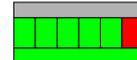
Prozessverbesserung: CMM Reifegradstufe 3 (Defined)

- The sw. process (swp) for both management and engineering activities is documented, standardized and integrated into a standard swp for the organization. All projects use an approved, tailored version of the organization`s standard swp for developing and maintaining sw.
 - KPA's
 - Organization process focus
 - Training program
 - Sw. Product engineering
 - Peer reviews
- Org. process definition,
Integrated sw. Management,
Intergroup coordination

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



33
13

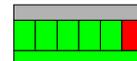
Prozessverbesserung: CMM Reifegradstufe 4 (Managed)

- Detailed measures of the software process and product quality are collected. Both the software process and products are quantitatively understood and controlled.
- KPA's
 - Quantitative process management
 - Software quality management

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



34
13

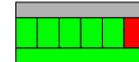
Prozessverbesserung: CMM Reifegradstufe 5 (Optimizing)

- Continuous process improvement is enabled by quantitative feedback from the process and from piloting innovative ideas and technologies.
- KPAs
 - Defect Prevention
 - Technology change management
 - Process change management

22.1.2002
„Leichte“ Prozesse

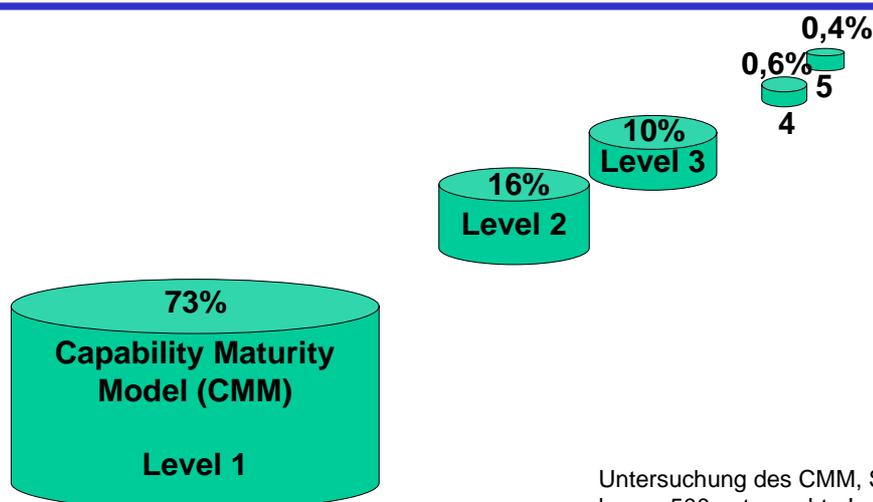
VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



35
13

Prozessverbesserung: Was bringt's?

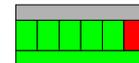


Untersuchung des CMM, Stand 3.3.2000
knapp 500 untersuchte Institutionen

22.1.2002
„Leichte“ Prozesse

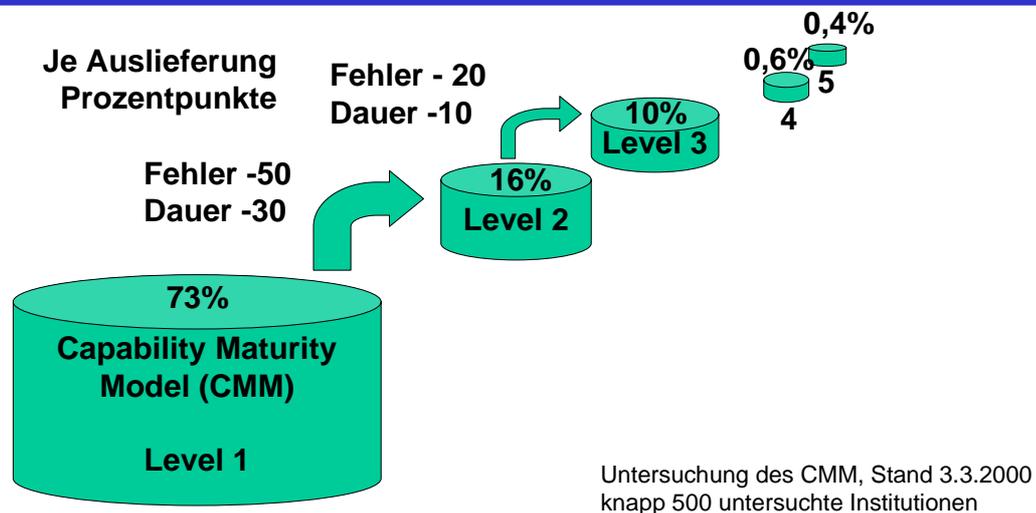
VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



36
13

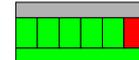
Prozessverbesserung: Was bringt's?



22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



37
13

Prozessverbesserung: Was bringt's?

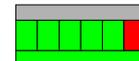
CMM Level	Monate	Personen-Monate	gefundene Fehler	ausgelieferte Fehler	Gesamtkosten (1000 US\$)
1	29,8	593,5	1348	61	5.440
2	18,5	143	328	12	1.311
3	15,2	79,5	182	7	728
4	12,5	42,8	96	5	392
5	9	16	37	1	146

Daten für ein 200 KLoC-Projekt (Schätzung der SemaTech)

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

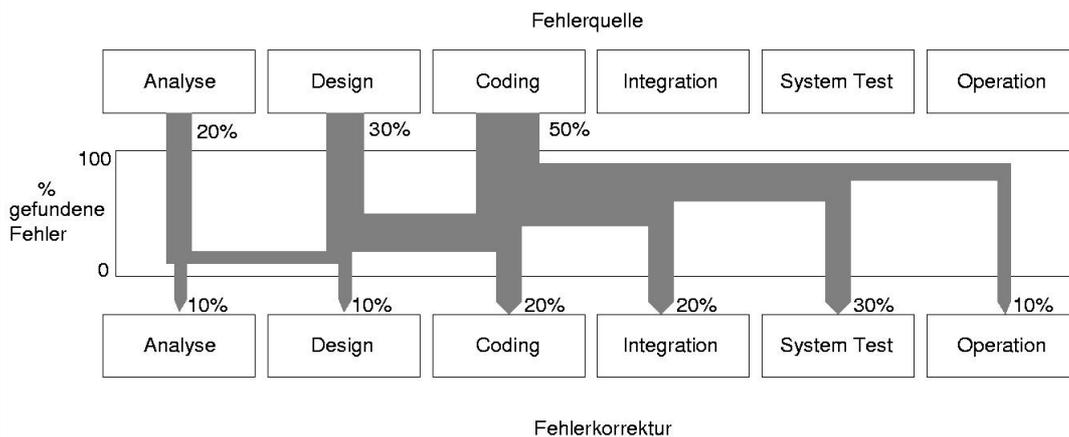
LMU



38
13

Wie können gleichzeitig Kosten und Dauer sinken, bei verbesserter Qualität?

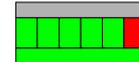
=> Vermeidung sinnloser Arbeit



22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



39

13

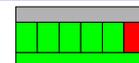
Prozessverbesserung: Fazit

- Man sollte sich um den Software-Prozess kümmern- es lohnt sich.
- Das wichtigste ist, sich überhaupt zu kümmern. *Wie* das geschieht, ist eher zweitrangig.
- Dies sind empirisch belegte Fakten, keine Meinungen oder Moden.

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



40

13

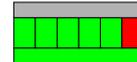
Prozessverbesserung: Probleme

- Die Durchsetzung von Prozessverbesserungen nach CMM erfordern
 - Sachverstand,
 - Entschlossenheit und
 - Durchhaltevermögenseitens des Managements erforderlich:
Pro Reifegradstufe rechnet man 2–3 Jahre!
- Das CMM (und SPICE/ISO15504) sind sehr stark auf Wasserfallmodell ausgerichtet.

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



41
13

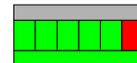
Nächstes Thema: Leichtgewichtige Prozesse

auch: Agile Processes

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



42
13

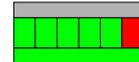
Einige Beobachtungen

- Individuelle Produktivität ist stark unterschiedlich (Faktor 10)
- Es gibt Gruppendynamik
 - Hawthorne-Effekt
 - Punktabfrage
 - WED-Technik
- Brooks´ Gesetz

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



43
13

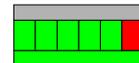
Brooks´ Gesetz (1)

- Der Zeitbedarf für jede Tätigkeit in Teams besteht immer aus zwei Elementen:
 - 1 – Die eigentliche Arbeit
 - 2 – Der Aufwand für Verwaltung und Abstimmung
- Ohne (2) würde die notwendige Dauer d eines Projektes mit der Zahl n der Mitarbeiter abnehmen, also $d \approx 1/n$.

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



44
13

Brooks´ Gesetz (2)

- Aber jeder Mitarbeiter muß mit jedem anderen kommunizieren (Aufwand jeweils k). Also ist die tatsächliche Dauer

$$d \approx 1/n + k \binom{n}{2} \approx 1/n + k \frac{1}{2} n^2$$

- Dadurch kommt es zu Brooks´ Gesetz: „*Adding manpower to a late project makes it later*“
- Mit anderen Worten: Schlechte Planung am Anfang führt unweigerlich zur Überziehung von Terminen (und Budget).

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



45
13

Organisationsformen

- Wie kann man trotzdem die Herstellung großer Software Systeme organisieren?
 - Hierarchisch (Baumartig)
 - Team
 - Fraktal (rekursives Netzwerk)

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



46
13

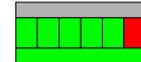
Leichtgewichtige Prozesse: Motivation

- Akzeptanzprobleme bei klassischen Modellen
 - Bürokratie
 - Anspruchsvoll
 - Durchgängigkeit erforderlich
- Outsourcing / contracting out: Zuliefer-, Ausliefer-, und Beschaffungsprozesse
- Integration statt Herstellung: über verschiedene Firmen hinweg – firmenweite Standardisierung reichte aus

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



47
13

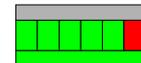
Leichtgewichtige Prozesse: Übersicht

- Prozessmuster
- Chefprogrammiererteam
- XP
- Crystal Series
- SCRUM

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



48
13

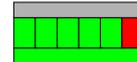
Leichtgewichtige Prozesse 1: Prozessmuster

- Analog zu Struktur–Mustern, gleiche Darstellung, gleiche Anwendung
- Vorteile
 - zum Planen oder Dokumentieren
 - zum Strukturierten, Problemgetriebenen Aufbau eines Prozesses
 - abgestufte/angepaßte Formalität
 - Akzeptanz
- Beispiele
 - Ambler–Bücher, Catalysis, Vilbig et al.
 - Störrle (Diss, Profes, EWSPT)

22.1.2002
„Leichte“ Prozesse

VL Software Architektur–Modelle
Dr. Harald Störrle

LMU



49
13

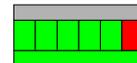
Leichtgewichtige Prozesse 2: Chefprogrammiererteam

- Baker (IBM) schlägt 1972 das „Chefprogrammierer–Team“ vor,
- Das Chefprogrammierer–Team (CPT) besteht aus
 - Chefprogrammierer: verantwortlich für Analyse, Entwurf, und Realisierung des Systems
 - Projektassistent: vertritt den CP und ist ihm Diskussionspartner
 - Projektsekretär: übernimmt alle Verwaltungsaufgaben
 - Spezialisten: je nach Bedarf für Sprache, Programmierung, Test, Werkzeuge, Aspekte, Komponenten, ...

22.1.2002
„Leichte“ Prozesse

VL Software Architektur–Modelle
Dr. Harald Störrle

LMU



50
13

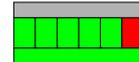
Leichtgewichtige Prozesse 2: Chefprogrammiererteam

- Vorteile:
 - Brooks´ Gesetz greift nicht
 - sehr effizient
- Nachteile:
 - funktioniert nur für kleine Gruppen
 - personell sehr schwierig zu besetzen (Assistent ist eine undankbare Rolle)

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



51
13

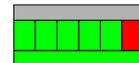
Leichtgewichtige Prozesse 2: Exkurs Gruppendynamik

- Manchmal gibt es eine sehr starke wechselseitige Befruchtung und Verstärkung innerhalb einer Gruppe.
- Manchmal auch das genaue Gegenteil.
- Man spricht in beiden Fällen von Gruppendynamik.
- Bsp.: Hawthorne-Effekt

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



52
13

• Gelingt es, eine kleine Gruppe

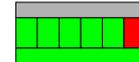
Leichtgewichtige Prozesse 2: XP

- XP ist ein Prozess, kein Vorgehensmodell
- XP deckt ab:
 - funktionale Anforderungen, teilweise Ergonomie und Last
 - Programmierung
 - Test
- Aber nicht
 - Legacies, Reengineering, große Systeme, Prozessverbesserung, Messen, non-OO-Entwicklung

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

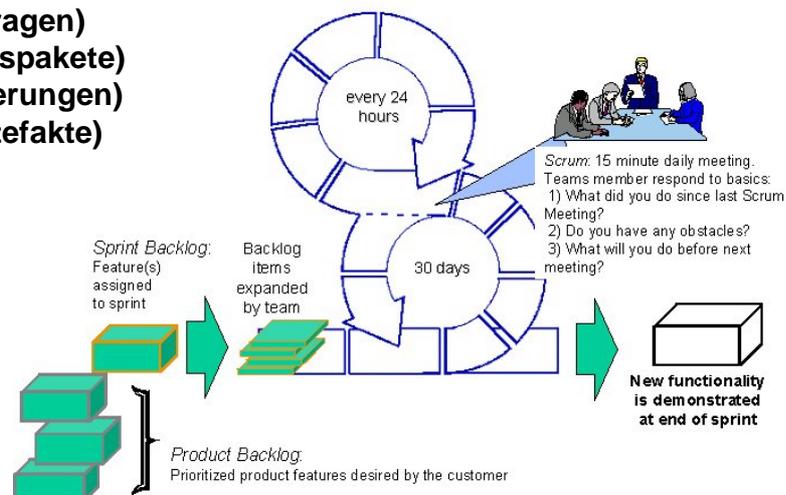
LMU



53
13

Leichtgewichtige Prozesse 3: SCRUM

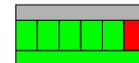
Issues (Offene Fragen)
Problems (Arbeitspakete)
Backlog (Anforderungen)
Components (Artefakte)



22.1.2002
„Leichte“ Prozesse

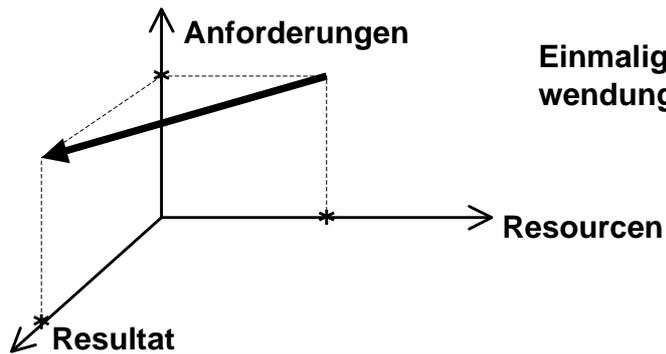
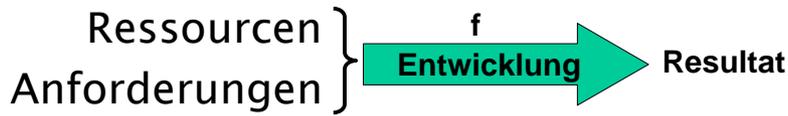
VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



54
13

Einschub Kontrolltheorie: deterministisches Chaos



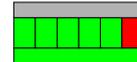
Einmalige Anwendung von f

3D ist eine sehr grobe Vereinfachung!

22.1.2002
„Leichte“ Prozesse

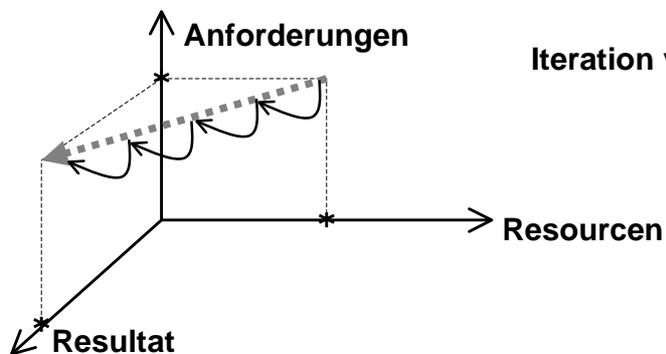
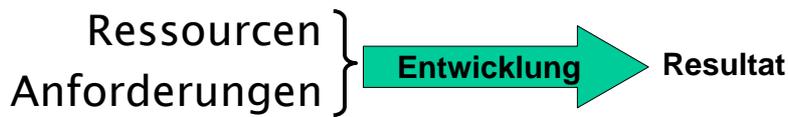
VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



55
13

Einschub Kontrolltheorie: deterministisches Chaos



Iteration von f

22.1.2002
„Leichte“ Prozesse

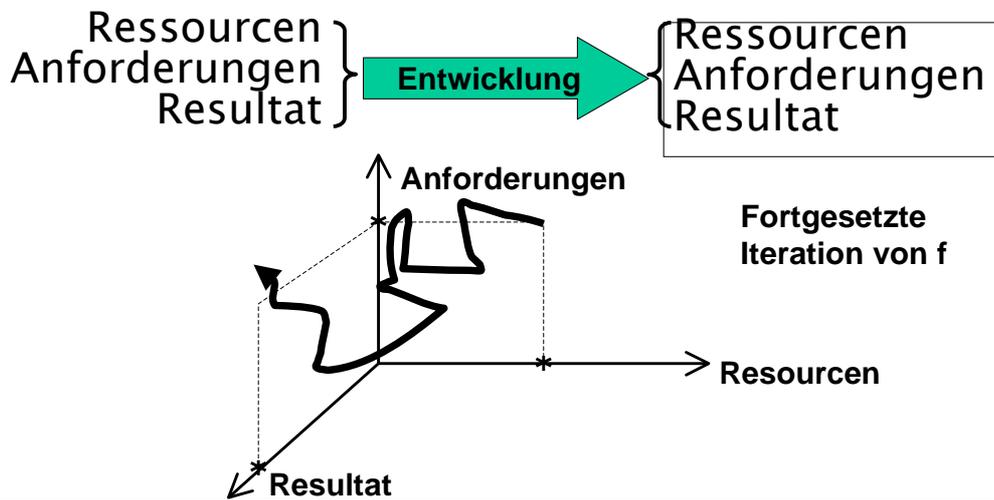
VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



56
13

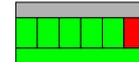
Einschub Kontrolltheorie: deterministisches Chaos



22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

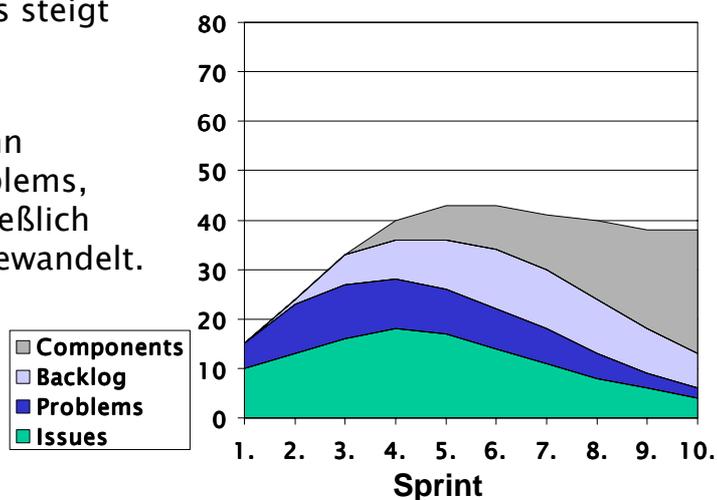
LMU



57
13

Leichtgewichtige Prozesse 3: Idealtypischer Ablauf von SCRUM

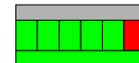
- Die Zahl der Issues steigt schnell an.
- Issues werden dann sukzessive in Problems, Backlog, und schließlich Components umgewandelt.



22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU

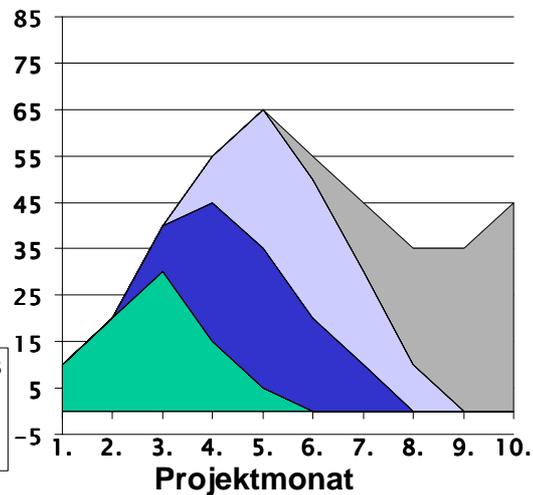
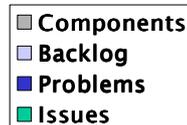
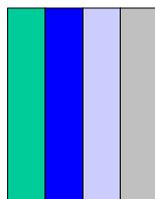


58
13

Leichtgewichtige Prozesse 3: Idealtypischer Ablauf mit Wasserfallmodell



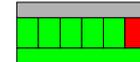
vs.



22.1.2002
„Leichte“ Prozesse

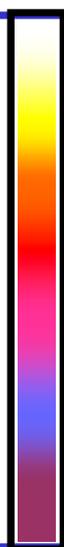
VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



59
13

Leichtgewichtige Prozesse 4: Crystal Series



Clear	3-8
Yellow	10-20
Orange	25-50
Red	50-100
Maroon	100-200
Blue	200-500
Violet	800+

„Software Development
as a cooperative game“

Erfahrungen?
Beschreibungen?

Alistair Cockburn: „Surviving OO Projects“
www.crystallmethodologies.org

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



60
13

Literatur Empfehlenswert

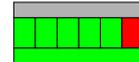
- Bücher von Watts Humphrey:
 - Managing the Software Process (DAS CMM–Buch)
 - The Personal Software Process
- Paulk, Weber, Curtis, Chrissis:
 - The Capability Maturity Model

alle AW, SEI–Series in Software Engineering
richtig teuer, richtig gut.

22.1.2002
„Leichte“ Prozesse

VL Software Architektur–Modelle
Dr. Harald Störrle

LMU



61
13

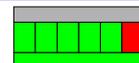
Literatur Weniger empfehlenswert

- Bücher von Scott Ambler

22.1.2002
„Leichte“ Prozesse

VL Software Architektur–Modelle
Dr. Harald Störrle

LMU



62
13

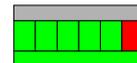
Literatur Industrielle Prozesse

- Dröschel/Wiemers: *Das Vorgehensmodell des Bundes Oldenbourg*, 1997
- Jacobson, Booch, Rumbaugh: *The Unified Software Development Process* Addison-Wesley, 2000
- D`Souza & Wills: *Catalysis* Addison-Wesley, 1999
- Beck: *Extreme Programming Explained* Addison-Wesley, 2001

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



63
13

Literatur Code-Inspektion

- Wheeler, Brykczynski, Meeson: *Software Inspection* IEEE Press, 1996
- Darin ausführliche Literaturhinweise

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



64
13

Literatur

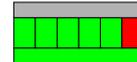
Organisation und Management

- Womack, Roos, Jones: *Die zweite Revolution in der Automobilindustrie* Campus, 1991
- Krause: *Die Kunst des Kriegs für Führungskräfte* Ueberreuther, 1995
- Malik: *Führen, Leisten, Leben DVA, 2001*
- DeMarco, Lister: *Wien wartet auf Dich!* Hanser 1991 (Neuaufgabe 1999)

22.1.2002
„Leichte“ Prozesse

VL Software Architektur-Modelle
Dr. Harald Störrle

LMU



65
13