



# Projektmanagement in der Softwareentwicklung

Vortrag vom 13.06.2005 in der Reihe

Projektmanagement Jour-Fixe des  
Kompetenzzentrums Projektführung und Projektmanagement,  
am GuT, Solingen

<http://www.project-city.de>

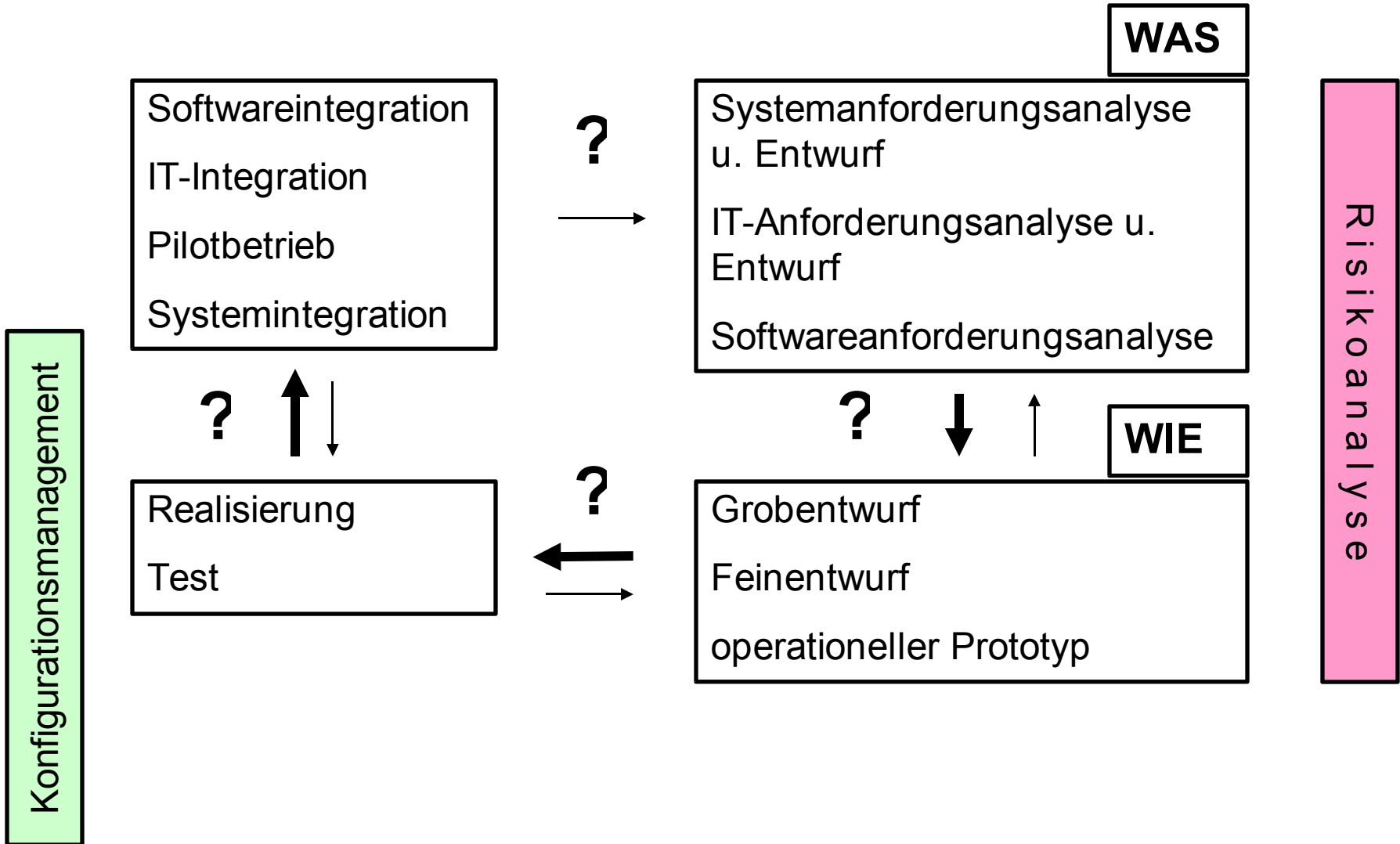
Ulfert Rieger

Infoforum GmbH  
Uellendahler Str. 586  
42111 Wuppertal

[ulfert.rieger@t-online.de](mailto:ulfert.rieger@t-online.de)

0202 / 7052318

# Klassische Vorgehensmodelle



# Wesentliche Merkmale eines Projektes

- sachlich: Einmaligkeit der Aufgabenstellung und Zielvorgabe
- zeitlich: Befristung mit genau festgelegten Start-Ende-Terminen
- budgetär: Kostenbegrenzung durch ein festgelegtes Budget
- personell: Bildung einer Projektgruppe nur für dieses Vorhaben

? Stehen Entwicklungsprojekte wirklich isoliert ?

? Ist die Aufgabenstellung einmalig, die Zielvorgabe klar ?

? Wie kommt es zur Bestellung des Kunden

→ I know it when I see it, but then it's too late ...

## Lösungsvorschlag: Das konventionelle Projekt wird geteilt in

1. Projektierung → Bereitstellung eines konkreten (!) Proiectums, damit der Kunde weiß, was er bestellt und es mit dem Endergebnis vergleichen kann

	Projektierung
Auftraggeber	Kunde, Thread-Manager
Ausführender	Projektor (Analyse-/Design-Service)
Input	freie Form
Output	Proiectum, Pilotsystem
Tests	funktional
Abnehmer	IT-Architekt
Outsourcing	möglich
Zeitraum	< 1 Monat
Parallelisierung	durchaus, persistenter Prototyp

## Lösungsvorschlag: Das konventionelle Projekt wird geteilt in

2. Architektierung → Projektübergreifende Bereitstellung der Architekturen.  
Architekten leiten aus dem Proiectum Bauplan und Modularisierung ab.

	Architektierung
Auftraggeber	Thread-Manager
Ausführender	IT-Architekt
Input	Proiectum
Output	System-Blaupause
Tests	Prototyping
Abnehmer	Fabrikant und Integrator
Outsourcing	nein
Zeitraum	< 2 Monate
Parallelisierung	nein

# Lösungsvorschlag: Das konventionelle Projekt wird geteilt in

3. Fabrikation → Bereitstellung der Module.

	Fabrikation
Auftraggeber	Thread-Manager
Ausführender	Fabrikant
Input	System-Blaupause & Proiectum
Output	Bausteine
Tests	umfassend (Haftung)
Abnehmer	Integrator
Outsourcing	möglich!
Zeitraum	< 4 Monate
Parallelisierung	notwendig

## Lösungsvorschlag: Das konventionelle Projekt wird geteilt in

4. Integration → Aggregation der Module. Unbedingt sorgfältiges Konfigurationsmanagement (Changemanagement-Prozesse) einhalten.

	Integration
Auftraggeber	Thread-Manager
Ausführender	Integrator
Input	System-Blaupause & Proiectum
Output	System (Softwareprodukt)
Tests	umfassend (Haftung)
Abnehmer	Betreiber
Outsourcing	nein
Zeitraum	< 5 Monate
Parallelisierung	

## Lösungsvorschlag: Das konventionelle Projekt wird geteilt in

5. Abnahme → Abnahme durch Betreiber + Kunde. Vergleich von Bestellung mit Ergebnis. Zwischenzeitliche Änderungen laufen über Changemanagement.

	Abnahme
Auftraggeber	Thread-Manager
Ausführender	Betreiber & Pilotanwender
Input	Pilotbetrieb
Output	Abnahmebericht
Tests	umfassend
Abnehmer	Kunde
Outsourcing	nein
Zeitraum	
Parallelisierung	nein

# Lösungsvorschlag: Charakteristik

- Statt Top→down, Grob→Detail, Abstrakt→Konkret: Permanente Adaption!
- Einfache Prozesse, ingenieurmäßiges Vorgehen
- Thread-Manager organisiert bereichsübergreifend, ergebnisorientiert  
Threads = steuerbare Einheiten des Bestellnetzes → alles unter Changemanagement
- Intensives Testen, QS, Regressionstests
- Threads dauern kurz, müssen i.A. nicht gewartet werden, basieren auf dem Proiectum, Ownership liegt beim Jobnehmer
- Permanente, visualisierte Übersicht ← Techniken des Projektmanagements

## Erfolgsfaktoren für die Praxis:

• Nutzeranbindung	19	
• Klare Anforderungsspezifikation	15	
• Realistische Vorstellungen	10	
• Klare Visionen und Zielsetzungen	3	47
• Gutes PM ~ gute Planung	11	
• Kürzere Projektphasen	9	
• Klare Ownership	6	
• Hart arbeitendes, konzentriertes Team	3	29
• Managementunterstützung	16	
• Kompetentes Team	8	24

Quelle: Chaos Report, Standish Group

## Umgehung für die Praxis:

- Technische Kompetenz des PL ist notwendig aber nicht hinreichend
- Kooperation mit Zulieferern + Abnehmern (Betrieb!) selbst sichern  
→ Beziehungsmanagement, kleiner Dienstweg, persönliche Bindung
- Kaufen! Lösungen oder Kompetenz. Sonst: Schwierigkeitsgrade nicht unterschätzen.
- Reduktionen nicht diskutieren, machen. Erfolg wird immer relativiert...

# Agiles Projektmanagement

- Die handelnden Personen und deren Zusammenarbeit bei der Zielerreichung sind für den Erfolg wesentlicher als Tools und Methoden
- Funktionierende Lösungen sind wichtiger als eine lückenlose Dokumentation
- Das einvernehmliche Ändern überholter Anforderungen hat Priorität gegenüber bloßer Vertragserfüllung
- Die Reaktion auf veränderte Anforderungen geht vor Erfüllung eines starren Projektplanes oder eines starren Vertragsrahmens
- Die vorgenommenen Änderungen werden kurzfristig ausgeliefert und vom Kunden hinsichtlich der Benutzbarkeit bewertet

# Agiles Projektmanagement: wie?

- Tägliche, enge Zusammenarbeit und direkte Kommunikation zwischen Kunden, Projektleitern und Entwicklern
- Dynamik im Projekt ist alltäglich, nicht störend
- Grundfunktionalität mit Rapid Prototyping entwickeln, dann verfeinern
- Kick-off Workshop legt Anforderungen fest, Schlüssigkeit und Widerspruchsfreiheit prüfen
- Dann Anforderungsliste priorisieren und ständig überwachen. Einfache und auswechselbare Lösungen haben Vorrang
- Extreme Programming, wo immer möglich, kurze Schritte, geringes Risiko, ständig lauffähiges System → Ist Kunde (dessen PL) kompetent?
- Ggf. Kundensicht intern substituieren → Doppelspitze
- Paarweises Arbeiten hebt Motivation, verringert Fehler

# Fazit:

- Lösungsvorschlag und Umgehung stehen im Kontrast
- Ist agiles Projektmanagement Pfuscherei? Jein. Nutzt ja Erkenntnisse und erprobte Methoden, ist aber flexibler für die Umgebungsbedingungen
- Zweigleisigkeit ist Erfolgsfaktor
- Mikro-Machtverhältnisse bei Auftragnehmer *und* Auftraggeber beachten !!
- Erfolg zählt, das ist Funktionierender Betrieb und zufriedene Kunden