

LMU
Ludwig-Maximilians-Universität München

**Projektmanagement:
Projektverlauf und Prozessmodelle**

Martin Wirsing
in Zusammenarbeit mit
Gefei Zhang

Institut für Informatik
Ludwig-Maximilians-Universität München

SS 2008

Ziele

- **Aufgabenfelder des Projektmanagement kennen lernen**
- **Tätigkeiten während des Projektverlaufs kennen lernen und im V-Modell wiederfinden**
- **Wichtige Prozessparadigmen und Vorgehensmodelle wiederholen und in Zusammenhang mit Projekten stellen:**
 - Sequentiell („Wasserfall“),
 - iterativ („spiralförmig“),
 - leichtgewichtig („agil“).

Projektmanagement M2 – Projektverlauf und Prozessmodelle 2

Aufgabenfelder Projektmanagement

- Laut Project Management Institute besteht PM aus den folgenden **9 Aufgabenfeldern:**
 - Integrierende Aufgaben (integration management)
 - Umfangsmanagement (scope management)
 - Zeitmanagement (time management)
 - Kostenmanagement (cost management)
 - Qualitätsmanagement (quality management)
 - Personalmanagement (human resource management)
 - Kommunikationsmgmt. (communication management)
 - Risikomanagement (risk management)
 - Beschaffungsmanagement.(procurement management)
- Es folgt eine Kurzübersicht über deren Themen

Projektmanagement M2 – Projektverlauf und Prozessmodelle 3

Vorbemerkung: Wann wie viel?

- Viele der nachfolgend beschriebenen Aufgaben sind nur für Großprojekte im vollen Umfang sinnvoll
- Bei mittelgroßen Projekten sollten viele davon vereinfacht werden
 - **aber welche und wie stark hängt vom Einzelfall ab**
- Bei kleinen Projekten sollten evtl. sogar manche ganz entfallen
 - **aber ob und welche hängt vom Einzelfall ab**
- Übertriebenes Projektmanagement ist schädlich!
- Zu wenig Projektmanagement ist auch schädlich!

➤ Augenmaß ist gefragt!
Das ist allerdings ohne Erfahrung ziemlich viel verlangt...

Projektmanagement M2 – Projektverlauf und Prozessmodelle 4

Integrierende Aufgaben (integration management)



Diese Tätigkeiten verbinden die Tätigkeiten der restlichen Felder miteinander:

- **Entwickeln des umfassenden Projektplans**
 - enthält Teilpläne aus den anderen Aufgabenfeldern
 - Anmerkung: Ein solcher Plan existiert immer (evtl. nicht detailliert, evtl. nicht einmal schriftlich)
- **Projektleitung**
 - konkrete Anweisungen geben, Entscheidungen fällen etc.
- **Projektüberwachung**
 - kontinuierlich den Ablauf gegenüber den Plänen verfolgen,
 - bei Abweichungen geeignete Maßnahmen ergreifen
- **Planänderungs-Überwachung und -Steuerung**
 - Änderungen am Plan auf Wirkungen abklopfen und entweder zurückweisen oder komplett ins Projekt einfädeln

Umfangsmanagement (scope mgmt)



- **Umfangsdefinition**
 - Was ist Aufgabe des Projekts? Was nicht?
 - Bei SW: Anforderungsbestimmung
- **Erarbeiten einer Produktzerlegung (WBS: Work Breakdown Structure)**
 - In welche handhabbaren Teile sollte man das Gesamtprodukt (genauer eigentlich: den konkreten Prozess) zerlegen?
 - Wie fügen sich diese Teile zum Ganzen zusammen?
 - Bei SW hauptsächlich: Entwurf + Prozessmodell
- **Umfangsverwaltung (bei SW: Anforderungsverwaltung)**
 - Änderungen am Umfang (durch externe Einflüsse, z.B. Kundenwunsch) werden nicht einfach zugelassen, sondern die Auswirkungen überprüft.
 - Akzeptierte Änderungen werden sorgfältig in die Umfangsbeschreibung eingearbeitet
 - und Beteiligte geeignet benachrichtigt

Zeitmanagement (time management)



- Entwickeln einer Aktivitätenliste
 - **Zu jedem Teilprodukt laut WBS gehören eine oder mehrere Aufgaben (Aktivitäten, also Prozesse)**
- (Zeit-) Aufwandsschätzung für die Aktivitäten
 - **Wie hoch ist der Aufwand (z.B. Personentage)?**
 - **Wie lange dauert die Erledigung (Kalendertage)?**
- Aufstellung eines Zeit- und Arbeitsplans (schedule)
 - **Wie hängen die Aufgaben von einander ab?**
 - **Wer erledigt von wann bis wann welche Aktivität?**
- Zeitplanüberwachung (schedule control)
 - **Wird der Plan eingehalten?**
 - **Kontinuierliche Fortschreibung des Zeitplans**

Kostenmanagement (cost management)



- Kostenschätzung
- Budgetaufstellung
- Kostenüberwachung
- Bei SW-Projekten sind außer den Personalkosten meistens nur ganz wenige Posten relevant (meist SW-Lizenzen, Reisekosten, ...)
- Dadurch hängen die Kosten so eng an den Zeitaufwänden, dass diese Aufgabe kaum separat gelöst werden muss

Qualitätsmanagement (quality management)



- Qualitätsplanung (quality planning)
 - **Aufstellen von Qualitätsanforderungen**
 - (bei SW: Anforderungsbestimmung)
- Planen, wie man sie erfüllt
 - (bei SW ungefähr: Qualitätsmanagement)
- Qualitätssicherung (quality assurance)
- Qualitätssteuerung (quality control)
 - **wenn die Qualität nicht stimmt, geeignete Maßnahmen ergreifen**
 - z.B. korrigieren, Teil wegwerfen und neu bauen, Aufgabe an andere Person übergeben, Entwicklungsprozess ändern, etc.

Personalmanagement (human resource management)



- Personalplanung
 - **Definition von Rollen, Verantwortlichkeiten, Weisungsbefugnisse**
 - (Prozessmodell & Organisationsplanung)
- Aufstellung eines Personalzeitplans
 - **Wie viele Leute mit welcher Qualifikation brauchen wir von wann bis wann?**
 - z.B. Analysten überwiegend am Anfang des Projekts,
 - Programmierer jedoch nicht von Anfang an
- Team einwerben, Teamformung
 - **Teilnahme an einem Projekt sollte freiwillig sein**
 - **Mitglieder müssen ein Gemeinschaftsgefühl entwickeln**
- Teamführung
 - **Leistung der Mitglieder verfolgen, Feedback geben, Konflikte auflösen, Arbeit bei Änderungen neu koordinieren**

Kommunikationsmanagement (communication management)



- Kommunikationsplanung
 - **Welche Beteiligten brauchen regelmäßig welche Information?**
- Informationsverteilung
 - **Den Betroffenen relevante Information stets zügig zukommen lassen**
- Fortschrittsberichte
 - **Statusberichte und Planfortschreibung regelmäßig erarbeiten und allen Betroffenen zuleiten**
- Interaktion mit Beteiligten
 - **Besprechungen und Schriftverkehr mit allen Beteiligten (z.B. Projektteam, Auftraggeber, Anwender, Interessierte), um deren Bedürfnisse zu erfüllen und Angelegenheiten aller Art zu klären**

Risikomanagement (risk management)



- Risiken identifizieren
 - **Welche Ereignisse könnten die plangerechte Durchführung bedrohen?**
- Risikoeinschätzung
 - **Risikogrößen abschätzen, Risikobehandlung priorisieren**
- Vorbeugung planen
 - **Was wird getan, um welchem Risiko vorzubeugen?**
- Gegenmaßnahmen planen
 - **Was wird getan, wenn welches Risiko eintritt?**
- Risikomanagement-Überwachung
 - **Kontinuierlich nach neuen oder veränderten Risiken Ausschau halten**
 - **Fortführung und Wirkung eingeleiteter Vorbeugungen und Gegenmaßnahmen überwachen. Ggf. korrigierend eingreifen**

Beschaffungsmanagement (procurement management)

- Planung und Durchführung der Beschaffung aller Dinge, die das Projekt braucht, aber nicht selbst herstellt
- Betrifft bei SW-Projekten oft SW-Lizenzen, gelegentlich Möbel u.ä. und manchmal Hardware
- Ist aber (außer bei Großprojekten) meist nicht sehr kompliziert

Projektmanagement M2 – Projektverlauf und Prozessmodelle 13

Projektverlauf

- Projektvorbereitung**
 - Festlegung der Projektziele
 - Zusammenstellung der groben Projektplanung ...in Koordination mit dem Auftraggeber!
 - Zusammenstellung und Nutzen von Know-How aus früheren Tätigkeiten und Projekten
 - Möglichst Durchführung eines Start-Workshops gemeinsam mit Auftraggeber und Projektteam
- Projektabschluss**
 - Zusammenstellung und Präsentation der Projektergebnisse
 - Gemeinsamer inhaltlicher und emotionaler(!) Abschluss des Projektes
 - Sicherung des Know-Hows und der "learned lessons" für nachfolgende Projekte
 - Prozessverbesserung

Projektmanagement M2 – Projektverlauf und Prozessmodelle 14

Projektverlauf

- Projektdurchführung**
 - Projektkontrolle**
 - Feststellung des Projektstatus
 - Feststellung von Planabweichungen
 - Techniken:
 - Fortschrittsanalyse
 - Risikoanalyse
 - Qualitätssicherungs-Maßnahmen
 - Projektsteuerung**
 - Durchführungsentscheidungen
 - Korrektivmaßnahmen
 - Techniken:
 - Risikomanagement
 - Qualitätsmanagement-Maßnahmen
 - Konfigurationsmanagement

Projektmanagement M2 – Projektverlauf und Prozessmodelle 15

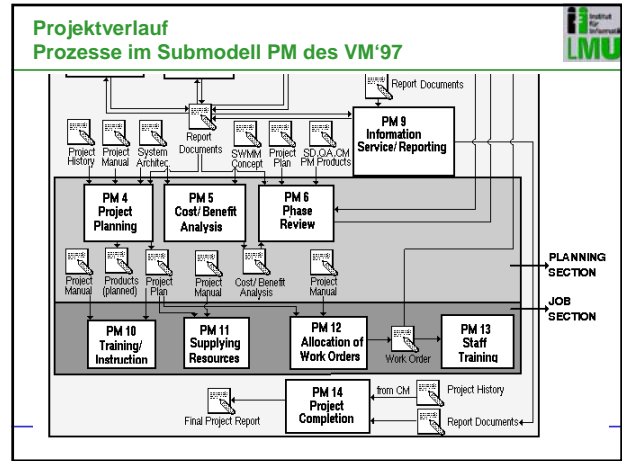
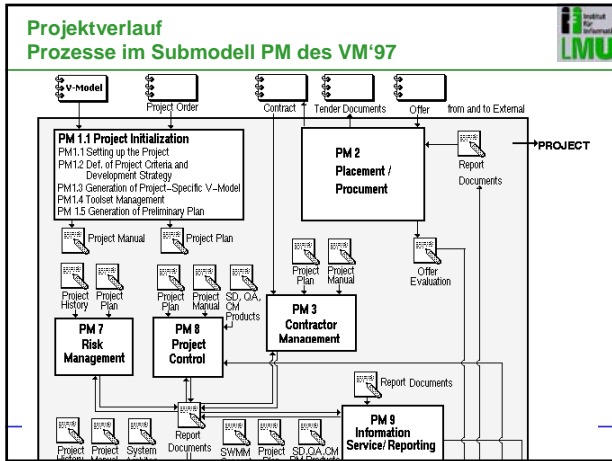
Projektverlauf

Prozesse im Submodell PM des VM'97

Prozesse nach VM'97 (Submodell PM)

PM1: Projektinitialisierung	PM8: Projektkontrolle und -steuerung
PM2: Vergabe/Beschaffung	PM9: Informationsdienst/Berichtswesen
PM3: Auftragnehmer-Management	PM10: Schulung/Einarbeitung
PM4: Feinplanung	PM11: Bereitstellung der Ressourcen
PM5: Kosten-/Nutzenanalyse	PM12: Vergabe von Arbeitsaufträgen
PM6: Durchführungentscheidung	PM13: Einweisung der Mitarbeiter
PM7: Risikomanagement	PM14: Projektabschluss

Projektmanagement M2 – Projektverlauf und Prozessmodelle 16



Tätigkeiten bei der Software-Entwicklung

- Anforderungsanalyse & Fachliche Konzeption
- Technische Konzeption (Design, Entwurf)
- Realisierung
- Integration & Test

Und deren Einordnung in **Vorgehensmodelle**

Projektmanagement M2 – Projektverlauf und Prozessmodelle 19

Das sequentielle Paradigma

- Das **sequentielle Paradigma** bezeichnet ein **sequentielles Vorgehen** mit **klar definierten Phasen und Ergebnissen**
- Bekannteste Vorgehensmodelle:**
 - Wasserfall-Modell
 - V-Modell

Projektmanagement M2 – Projektverlauf und Prozessmodelle 20

Wasserfallmodell (schematisch)

- **Eingeführt von Walker Royce 1970, Weiterentwicklung der „stufenweisen Entwicklung“ („stagewise development“) von Benington 1956.**
- **Ansatz:**
 - Top-down Stufenmodell mit eingeschränkter Rückkopplung
 - Phasensynchronisation durch (Zwischen-) Produkte
 - Geringer Managementaufwand

Charakteristika:

- Jede Aktivität muss beendet sein, bevor die nächste beginnt.
- Jede Aktivität ist in der richtigen Reihenfolge und in voller Breite vollständig durchzuführen.
- Am Ende jeder Aktivität steht ein fertiggestelltes Dokument.

Projektstart Ziel

Projektmanagement M2 – Projektverlauf und Prozessmodelle 21

Das sequentielle Paradigma: Vor- und Nachteile

- **Gut anwendbar bei klarer, relativ fixer Funktionalität:**
 - System- und Basissoftware wie OS, DB, Web-Server;
 - Branchen-Software wie SAP R/3.
- **Notwendig bei hohen Qualitäts- und Zuverlässigkeitsanforderungen:**
 - eingebettete Systeme (Automotive, Maschinenbau, Medizinische Geräte, Anlagensteuerung, Aerospace, Defense, ...);
 - Telekommunikations-Systeme;
 - rechtliche Anforderungen wie Revisionssicherheit oder Nachvollziehbarkeit.
- **Vorteile**
 - einfach durchzuführen
 - schränkt Freiheitsgrade stark ein, daher auch für sehr große Projekte anwendbar
 - sehr effizient bei bekannten und konstanten Anforderungen
 - ist gut zu vermessen (notwendig z.B. für Prozessverbesserung)
- **Nachteile**
 - Risiken gesammelt am Schluss („Big Bang“)
 - starr während des Ablaufs

Projektmanagement M2 – Projektverlauf und Prozessmodelle 22

Das iterative Paradigma

Iterativ heißt, dass der Entwicklungsprozess mehrfach iteriert wird: statt den „Wasserfall“ einmal zu durchlaufen, werden kleine Wasserfälle hintereinander gesetzt.

Projektmanagement M2 – Projektverlauf und Prozessmodelle 23

Das iterative Paradigma

- Das **iterative Paradigma** ist eine Weiterentwicklung des sequentiellen Paradigmas aus der Erkenntnis, dass Software länger lebt als erwartet und auch vom Funktionsumfang her gepflegt werden muss.
- Das iterative Paradigma unterstützt **inkrementelle Entwicklung**, d.h. dass das zu erstellende System nicht in einem Rutsch freigegeben wird, sondern in mehreren Stufen.
- **Bekannteste Modelle:**
 - **Spiral-Modell** (ein Meta-Vorgehensmodell)
 - **Unified Process (RUP)**
- Das agile Paradigma ist eine Form des iterativen Paradigmas in Verbindung mit frühem Prototyping, bei der nicht alle Tätigkeiten iteriert werden.

Projektmanagement M2 – Projektverlauf und Prozessmodelle 24

Rational Unified Process (RUP)

RUP ist eine Weiterentwicklung von Jacobsons Objectory

RUP hat UML als Sprache voreingestellt

RUP wird von Rational/IBM bezeichnet als

- inkrementell & iterativ,
- architektur-zentriert, und
- anwendungsfall-getrieben.

Phasen

Inception Elaboration Construction Transition

Business Modeling

Requirements

Analysis & Design

Implementation

Test

Deployment

Configuration Mgmt. Management Environment

Core Workflows

Iterationen

Preliminary Iteration(s) Iter. #1 Iter. #2 Iter. #n Iter. #n+1 Iter. #n+2 Iter. #m Iter. #m+1

Projektmanagement M2 – Projektverlauf und Prozessmodelle 25

Das iterative Paradigma: Vor- und Nachteile

- **Vorteile**
 - Risiken können früher erkannt werden
 - volatile Anforderungen können besser berücksichtigt werden
 - inkrementelle Auslieferung wird erleichtert
- **Nachteile**
 - Mehrarbeit
 - komplexeres Projektmanagement
 - schwerer messbar

Projektmanagement M2 – Projektverlauf und Prozessmodelle 26

Das adaptive Paradigma

- Unter einem **adaptiven SW-Prozess** versteht man eine Weiterentwicklung des iterativen Paradigmas, bei der
 - die Planung der Iterationen dynamisch erfolgt und
 - von Anfang an Prototypen erstellt werden
- **Charakteristikum:**
 - kontinuierliche Anpassung an Änderungen
- **Prinzipien:**
 - Individuum und Interaktion (geht) vor Prozess und Werkzeug
 - ausführbare SW vor vollständiger Dokumentation
 - Zusammenarbeit mit Kunden vor Vertragsverhandlung
 - Berücksichtigung von Änderungen vor Beharren auf Plan
- **Bekannte Vorgehensweisen:**
 - XP
 - SCRUM, ...

Projektmanagement M2 – Projektverlauf und Prozessmodelle 27

XP (Extreme Programming)

- **Merkmale**
 - XP arbeitet mit **kleinen Releases**, unterteilt in Iterationen und Arbeitspakete.
 - **Anforderungsanalyse**
 - Aufgaben und Anforderungen in Form von **Stories**
 - Beschränkung auf **wenige Anforderungen pro Iteration**
 - **Pair Programming**
 - Zwei Personen vor einem Rechner, einer programmiert, der andere ist Sparringspartner.
 - Enthält Elemente des **Prototyping**, allerdings **ohne Wegwerfen**.
 - **Auftraggeber** wird mit besser **eingebunden**, kann konkrete Ergebnisse sehen.
- **Prozess:**
 - **Test-first:** Zuerst Tests (Spezifikation) schreiben, dann programmieren.
 - **Kontinuierliches Refactoring** zur Verbesserung der Code-Qualität.

Projektmanagement M2 – Projektverlauf und Prozessmodelle 28

Das adaptive Paradigma: Vor- und Nachteile



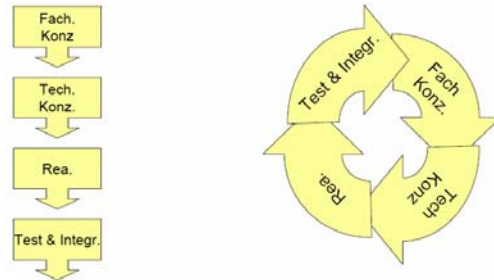
Vorteile

- Gut einsetzbar bei unklaren Zielen und sich ändernden Anforderungen/Umgebung
- Verspricht besseres Kosten/Nutzen-Verhältnis
- Vermutlich durchschnittliche Code-Qualität besser

Nachteile

- **Ergebnis ist nicht vorhersagbar**
 - Qualitätseigenschaften können nicht garantiert werden
 - Oft nicht nachvollziehbar, wie eine Funktion zustande kommt
- **80-90% aller Software läuft auf eingebetteten Systemen.**
 - Das bezieht sich z.B. auf Maschinen und Anlagen jeder Art und Größe, Verkehrsmittel, Militärische Anwendungen, Medizinische Geräte.
 - Hier sind Fehler fatal - ein „agiler“ Prozess kommt nicht (oft) in Frage.
- **Refactoring geht nicht bei Nicht-oo-Sprachen.**
 - Aber ein Großteil (ca. 90%) der bestehenden Applikationen weltweit ist in Cobol, PL1, Assembler und C, und OO-Sprachen sind nicht immer optimal.

Wasserfallmodell versus Inkrementelle Software-Entwicklung



- Was ist besser?

Wasserfallmodell versus Inkrementelle Software-Entwicklung



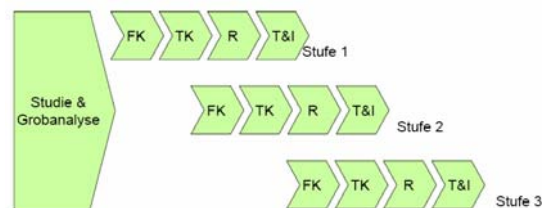
Wasserfall

- Hohe Sicherheit für Software-Anbieter
- Gesamtblick (aber: zu viele Details)
- Unüberschaubare Konzeptpapiere
- Geringere Flexibilität, aber Change-Request-Verfahren
- Nutzen erst bei Einführung
- „Deckel drauf bekommen“
- Einfache Struktur, Qualitätssicherung zwischen Phasen
- Entspricht Denkweise: Geld für definierte Leistung

Iteratives Vorgehen

- Früher Nutzen für Kunden
- Besseres, qualifizierteres Feedback
- Kosten für Übergangslösungen
- Schwieriger zu managen
- Geringeres Einführungsrisiko

Verbesserung: Gestuftes Vorgehen



- **Wasserfall als Vorgehen stößt an Grenzen**
 - Große Projekte werden handhabbar
- **Verringern des Risikos**
 - Fachliches Risiko
 - Technisches Risiko
- **Die elegante Art, „nein“ zu sagen: Stufen bedeuten, dass Anforderungen nicht oder erst später erfüllt werden**
 - Widerstände müssen überwunden werden: Management-Aufgabe

Gestuftes Vorgehen



- Bei Aufsetzen des Projekts
 - Immer den Umfang des Projekts im Auge behalten
 - Bei größerem Umfang Stufung versuchen
 - Umfang niemals unterschätzen!
- Nach Fachkonzeption oder Studie
 - Zusätzlicher Schnittpunkt für Stufenbildung (fachlich)
- Indizien für zu großen Projektumfang
 - Umfang der Konzeptpapiere: größer als 200 Seiten?
 - Feedback der Reviewer/externen Beteiligten: wirken sie noch mit, lesen sie die Papiere?

Beispiele für gestuftes Vorgehen im Projekt



	Motivation / Einsatz	Besonderheiten
<p>Verzahntes Wasserfallmodell</p>	<ul style="list-style-type: none"> ▪ Kleines Projekt ▪ Klarer, überschaubarer Funktionsumfang ▪ Frühe Gesamtspezifikation erforderlich 	<ul style="list-style-type: none"> ▪ Ist Spezialfall einer Stufe mit nur einem Inkrement ▪ Funktionsumfang früh definiert und weitgehend fix
<p>Inkrementelles Vorgehen</p>	<ul style="list-style-type: none"> ▪ Schnelle Ergebnisse & schnelles Lernen – auch bei komplexer Funktionalität ▪ Risiko reduzieren durch „Wichtigstes zuerst“ 	<ul style="list-style-type: none"> ▪ Frühes Feedback durch schnell lauffähiges Teilsystem ▪ Schrittweises Verfeinern
<p>Inkrementell mit Vorspezifikation</p>	<ul style="list-style-type: none"> ▪ Gesamtspezifikation zu Beginn der Stufe ▪ Gesamtaufwand leichter planbar als bei inkrementellem Vorgehen 	<ul style="list-style-type: none"> ▪ Mischform der beiden obigen Modelle ▪ Schwerpunkt ab zweitem Inkrement auf Realisierung (weniger Konstruktion)

Zusammenfassung (1)



- Laut **Project Management Institute** besteht PM aus den folgenden **9 Aufgabenfeldern**:
 - **Integrierende Aufgaben** (integration management)
 - **Umfangsmanagement** (scope management)
 - **Zeitmanagement** (time management)
 - **Kostenmanagement** (cost management)
 - **Qualitätsmanagement** (quality management)
 - **Personalmanagement** (human resource management)
 - **Kommunikationsgmt.** (communication management)
 - **Risikomanagement** (risk management)
 - **Beschaffungsmanagement** (procurement management)
- Ein **Projektverlauf** besteht aus
 - **Projektvorbereitung**
 - **Projektdurchführung** mit Projektkontrolle und -Steuerung
 - **Projektabschluss**

Zusammenfassung (2)



- **Das sequentielle Prozessparadigma (Wasserfall, klassisches V-Modell) ist**
 - einfach durchzuführen
 - auch für sehr große Projekte anwendbar
 - sehr effizient bei bekannten und konstanten Anforderungen

ABER

 - Risiken gesammelt am Schluss („Big Bang“) und starr während des Ablaufs
- **Das iterative Paradigma (RUP, Spiralmodell) erleichtert**
 - frühe Erkennung von Risiken
 - Berücksichtigung sich ändernder Anforderungen
 - inkrementelle Auslieferung

ABER

 - es erfordert Mehrarbeit, komplexeres Projektmanagement und ist schwerer messbar

Zusammenfassung (3)



- **Das agile Prozessparadigma (XP) ist**
 - Gut einsetzbar bei unklaren Zielen und sich ändernden Anforderungen/Umgebung
 - Verspricht besseres Kosten/Nutzen-Verhältnis
 - Vermutlich durchschnittliche Code-Qualität besser
- ABER
 - das Ergebnis ist nicht vorhersagbar
 - Qualitätseigenschaften können nicht garantiert werden
- Ein **gestuftes Vorgehen** verbindet die Vorteile von sequentiellem und iterativem Vorgehen und vermeidet Nachteile wie „Big-Bang“ und hohem Mehraufwand.