

# Besondere Fragen

Diese Fragen sind nicht so oft vorgekommen, sind aber relativ schwierig - daher bietet es sich an die Fragen zumindest genau durchzulesen (damit dann keine negative Überraschung droht).

## RUP Prinzip

### Allgemeine Erklärung

RUP = Rational Unified Process

Dabei handelt es sich um eine inkrementelle und iterative Vorgehensweise. Grundsätzlich gibt es dabei 4 Phasen, definierte Workflows und Disziplinen. Bei RUP kommt ein integriertes Anforderungsmanagement zum Einsatz, eine Komponenten-orientierte Architektur. Modelliert wird mit dem UML-Methodenframework und bei Meilensteine erfolgt jeweils die Verifikation von Produkten.

Vorteile:

- Real-world Szenarien
- Werkzeugunterstützung (zB Rational XDE, IBM)
- Vordefinierte Listen mit erforderlichen Artefakten

Nachteile:

- Hohe Komplexität
- Hoher Dokumentationsaufwand
- Anbieterabhängigkeit

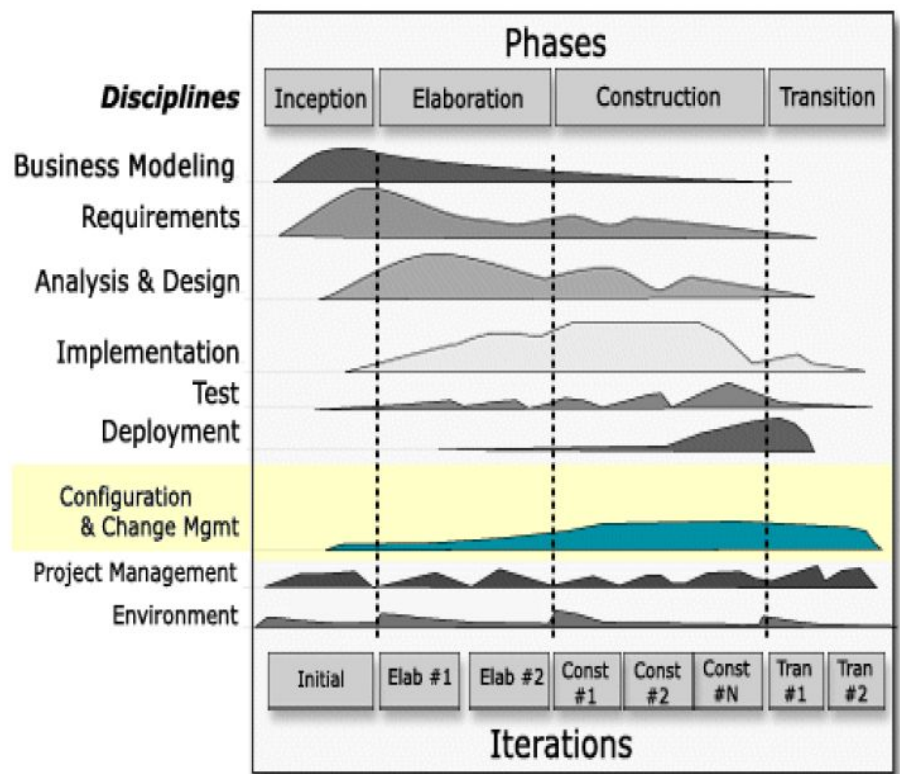
### Phasen

Inception (Beginn)

Elaboration (Concept & Design)

Construction

Transition (Auslieferung)



### Einsatzgebiet

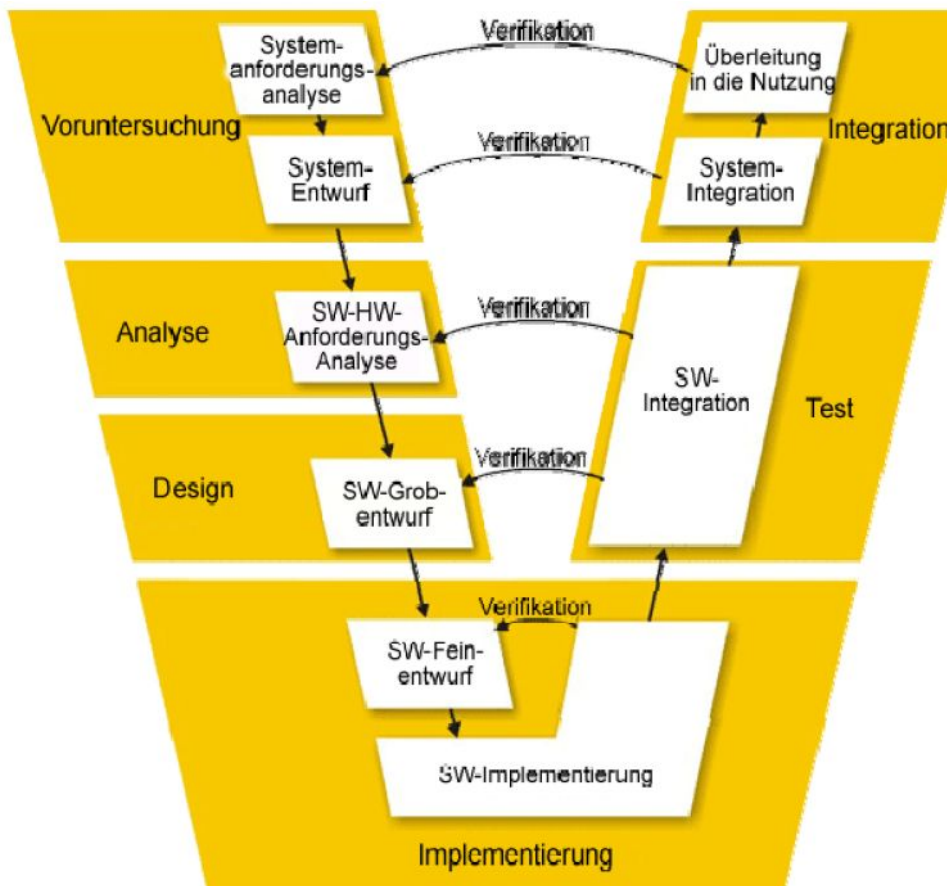
Grosse Projekte durch eine ganzheitliche Prozess-Sicht auf das gesamte Projekt (inkl. Deployment).

## V-Modell

Gesamtaufwand sinkt (Messung anhand des Lebenszyklus einer IT-Anwendung), somit auch insgesamt eine Kostenreduktion. Die Einarbeitungszeit für neue Mitarbeiter ist kürzer / schneller. Alle Beteiligten (Auftraggeber und Auftragnehmer) können den Projektfortschritt einfach kontrollieren. Insgesamt werden Projekte einfacher vergleichbar und es können genauere Aufwandsabschätzungen für zukünftige Projekte getroffen werden.

Innerhalb des V-Modells gibt es 4 Submodelle: Systemerstellung, Qualitätssicherung, Konfigurationsmanagement, Projektmanagement

Anwendung: für kleine Projekte zu detailliert, für große Projekte mit hohem Qualitätsanspruch geeignet (speziell Embedded Systems)



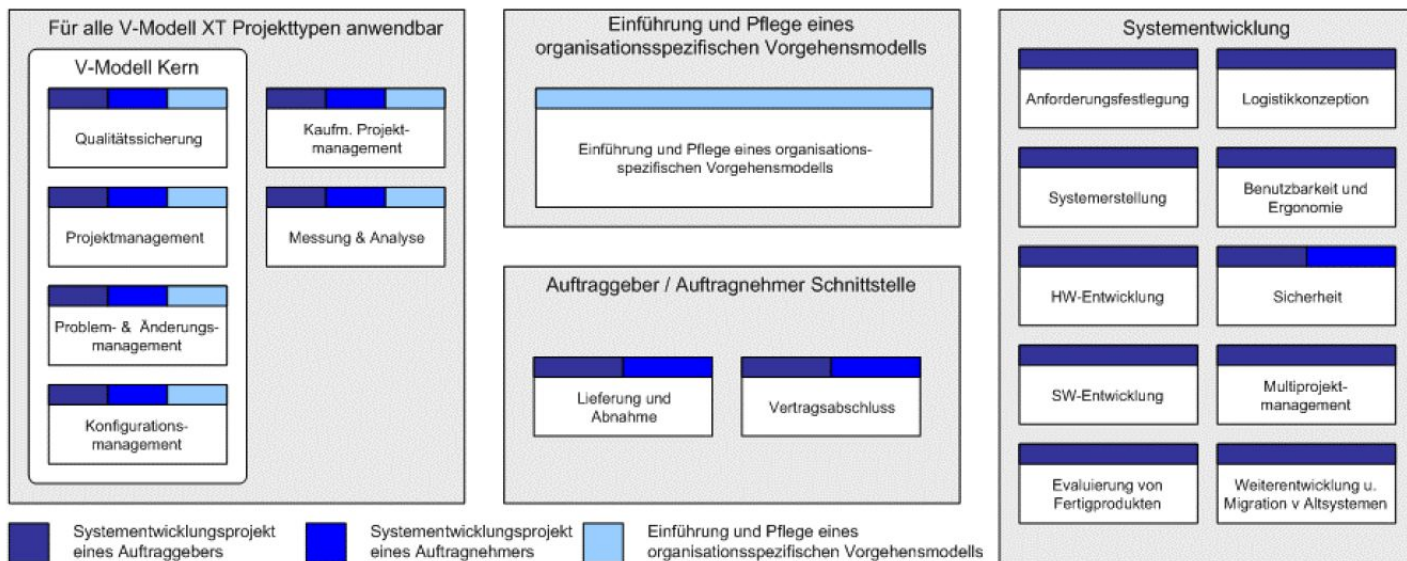
*Auch Skizze anschauen.*

## V-Modell XT

Weiterentwicklung des V-Modells von 1997 und wurde 2005 veröffentlicht. Ist ein verpflichtendes Vorgehensmodell für IT-Projekte im öffentlichen Bereich in Deutschland.

Zielsetzung:

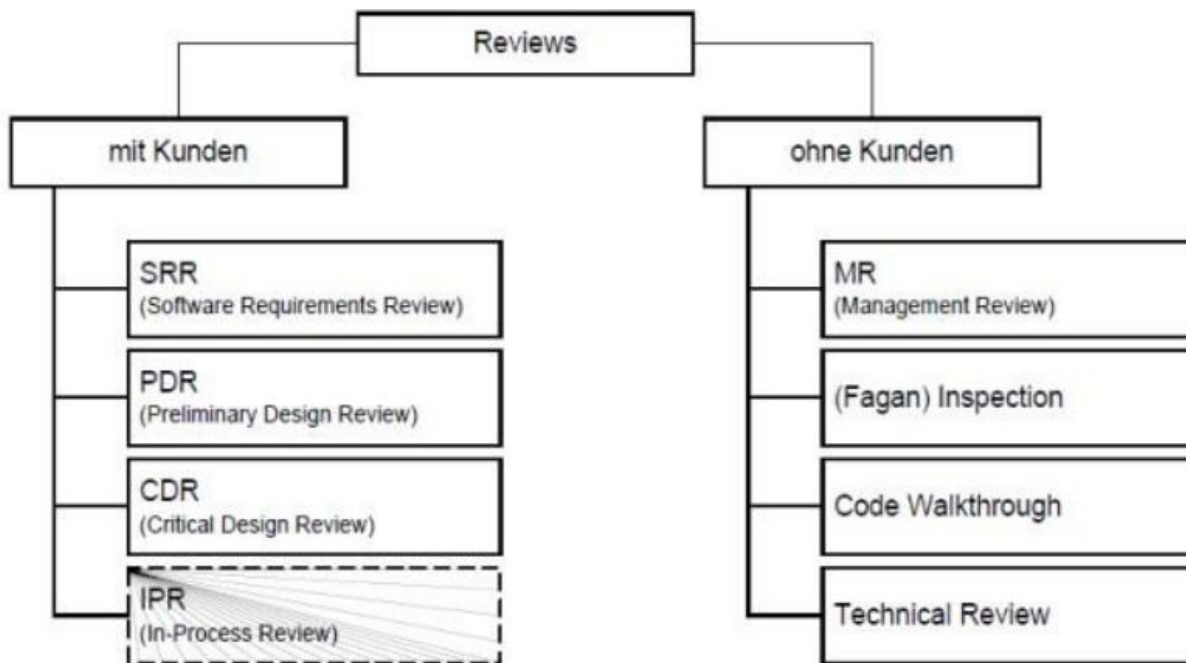
- Verbesserung der Unterstützung von Anpassbarkeit, Anwendbarkeit, Skalierbarkeit und Änder- und Erweiterbarkeit des V-Modells.
- Berücksichtigung des neuesten Stand der Technik (Best-Practice).
- Kompatibilität zu formalen Richtlinien und Standards
- Erweiterung des Anwendungsbereiches auf die Betrachtung des Systemlebenszyklus; Integration des Auftraggebers in das Projekt.
- Integration eines Prozessmodells zur "Einführung und Pflege eines organisationsspezifischen Vorgehensmodells"



In der Anwendung:

- Flexible Anwendung durch Anpassung an Gegebenheiten (Projekttyp, Projektmerkmale)
- Werkzeugunterstützung (Open Source): V-Modell XT Projektassistent, V-Modell XT Editor
- Qualitätsstandards damit möglich

## Arten von Reviews



Ziemlich viel zum aufzählen und auswendig lernen (wenigstens Abkürzungen, Text kann man sich gut ableiten daraus)

- SRR Software Requirements Review
  - verfolgt das Ziel, nach der Definition der Anforderungen und vor dem Entwurf ein Review durchzuführen, um alle Unklarheiten sowohl seitens des Kunden als auch seitens des Entwicklerteams auszuräumen
  - Kunden sollen teilnehmen
  - Nach SRR müssen Ziele & Anforderungen allen Beteiligten bekannt sein
  - Im Anschluss Start Entwurfs und Designphase
- PDR Preliminary Design = Vorentwurfsplanung
  - In Entwurfsphase wird Architektur und Grobstruktur überprüft
  - Zum Sicherstellen, dass alle Anforderungen umgesetzt werden können
- CDR Critical Design Review
  - Bei besonders kritischen Komponenten sinnvoll um gründlich Entwurf und Design zu analysieren
  - In Praxis: PDR und CDR meist in einem Review Zyklus zusammengefasst
- IPR
  - Für sehr große und komplexe Projekte kann Review auch während dem Prozess erforderlich sein um Kunden Fortschritt vorzulegen
  - Entspricht im wesentlichen Reviews agiler Vorgehensweisen
- MR Management Reviews
  - Zur formellen Bewertung des Projekts / Projektstaterhebung.

- Bei Abweichungen → Gegenmaßnahmen
  - Meist im Projektplan mit eingeplant aber auch bei außergewöhnlichen Ereignissen zusätzlich möglich
- TR Technical Review
  - konkreten Teil der Software überprüfen und bewerten
  - Fehlersuche in bestehender Software kann auch TR sein
  - Entweder eingeplant oder integrale Bestandteile des Dev-Prozesses
- Code-Walk-Through & Inspektion
  - Eigentlich selbsterklärend
  - Code-Walk-Through: Qualität der Implementierung beurteilen
  - Inspektion: formales Review in frühen Phasen um schwere Fehler zu finden
- IRP integrierter Review-Prozess
  - Sourcecode-Managementsystem zb kann dazu eingesetzt werden (Sonar-Lint)
  - Qualität sicherstellen
- ICW