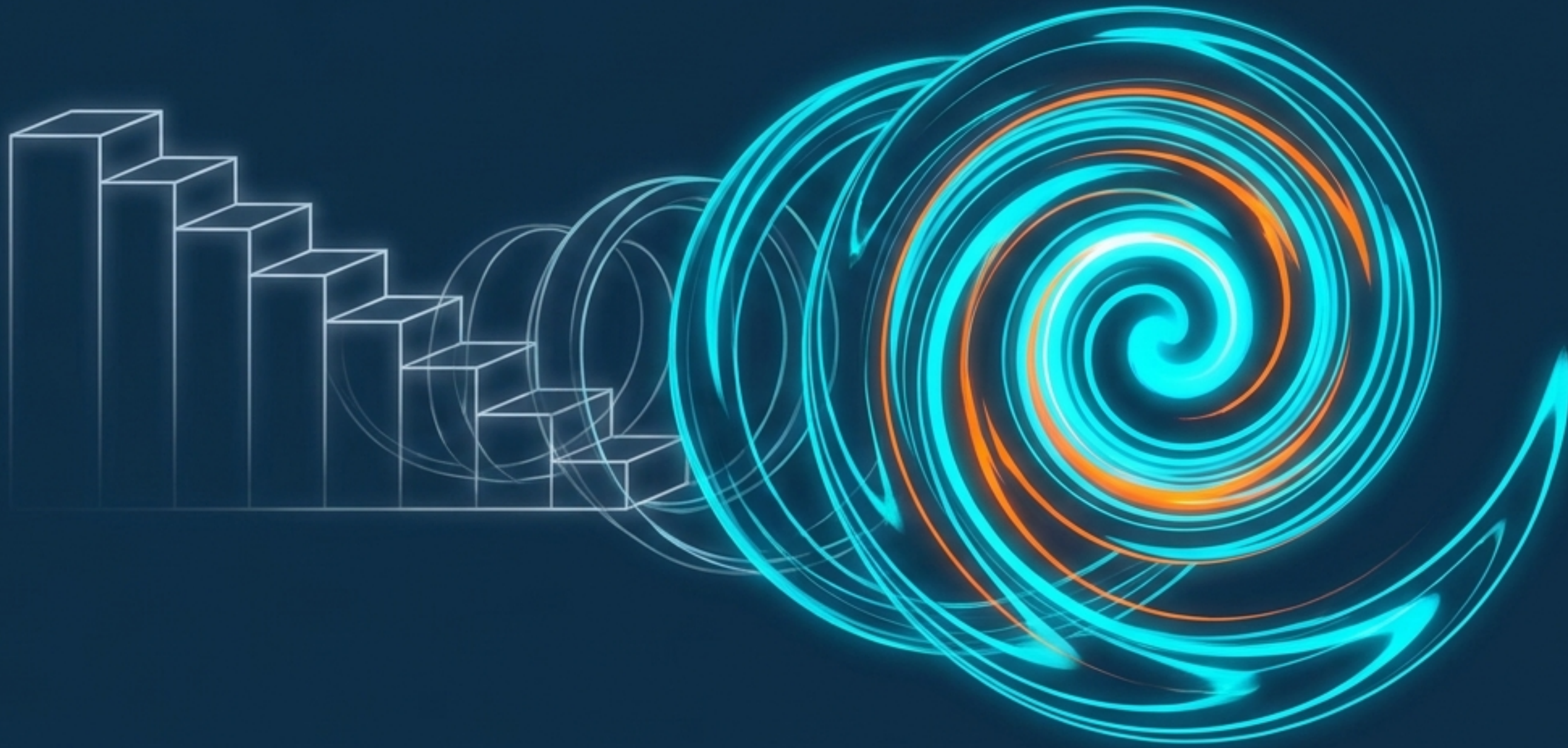


Vorgehensmodelle Tag 2

Spiralmodell, weitere iterative Ansätze und professionelle Modellauswahl in der Praxis



Der Weg in die Iteration: Grenzen der Planbarkeit

Klassische Modelle (Tag 1)

- ⚡ **Modelle:** Wasserfall, V-Modell XT
- ✓ **Charakteristik:** Plangetrieben, stark sequenziell.
- 🎯 **Voraussetzung:** Anforderungen müssen frühzeitig bekannt und über den Projektverlauf stabil sein.



Die Grenze: Bei technologischen Innovationen oder unklaren Anforderungen droht ein teures, spätes Scheitern.

Iteratives Denken (Tag 2)

- **Paradigma:** Risiko- und wertorientiertes Denken in Wiederholungsschleifen.
- **Fokus heute:**
 - ✓ Das Spiralmodell als konzeptionelle Brücke
 - ✓ Prototyping, RUP und XP
 - ✓ Methodische Abgrenzung zu Scrum
 - ✓ Faktoren der professionellen Modellauswahl

Das Spiralmodell (Boehm, 1988): Fokus auf das Risiko

Entstehung

Entwickelt von Barry W. Boehm als Antwort auf die Schwächen starrer Phasenmodelle bei komplexen Projekten.

Grundidee

Das Projekt durchläuft mehrere Schleifen (Spiralwindungen). Mit jeder Iteration steigt die Produktreife, während die Unsicherheit sinkt.

Stärken

Frühe Risikotransparenz, schrittweise Konkretisierung, Validierung durch Zwischenergebnisse.

Schwächen

Hoher Dokumentationsaufwand, anspruchsvolle Risikoanalyse erfordert Experten, schwierige Aufwandsschätzung zu Beginn.



Alleinstellungsmerkmal: Risikofokus

Die systematische Identifikation und Behandlung größter Unsicherheiten steht im Zentrum. Ein geordneter, verlustarmer Projektabbruch ist hier keine Katastrophe, sondern eine explizite, valide Option der Projektsteuerung.

Die Architektur des Spiralmodells: Quadranten und Iterationen

1. Ziele & Alternativen bestimmen

Rahmenbedingungen (Budget, Zeit), Architektur- und Lieferantalternativen definieren.

2. Risiken bewerten & behandeln

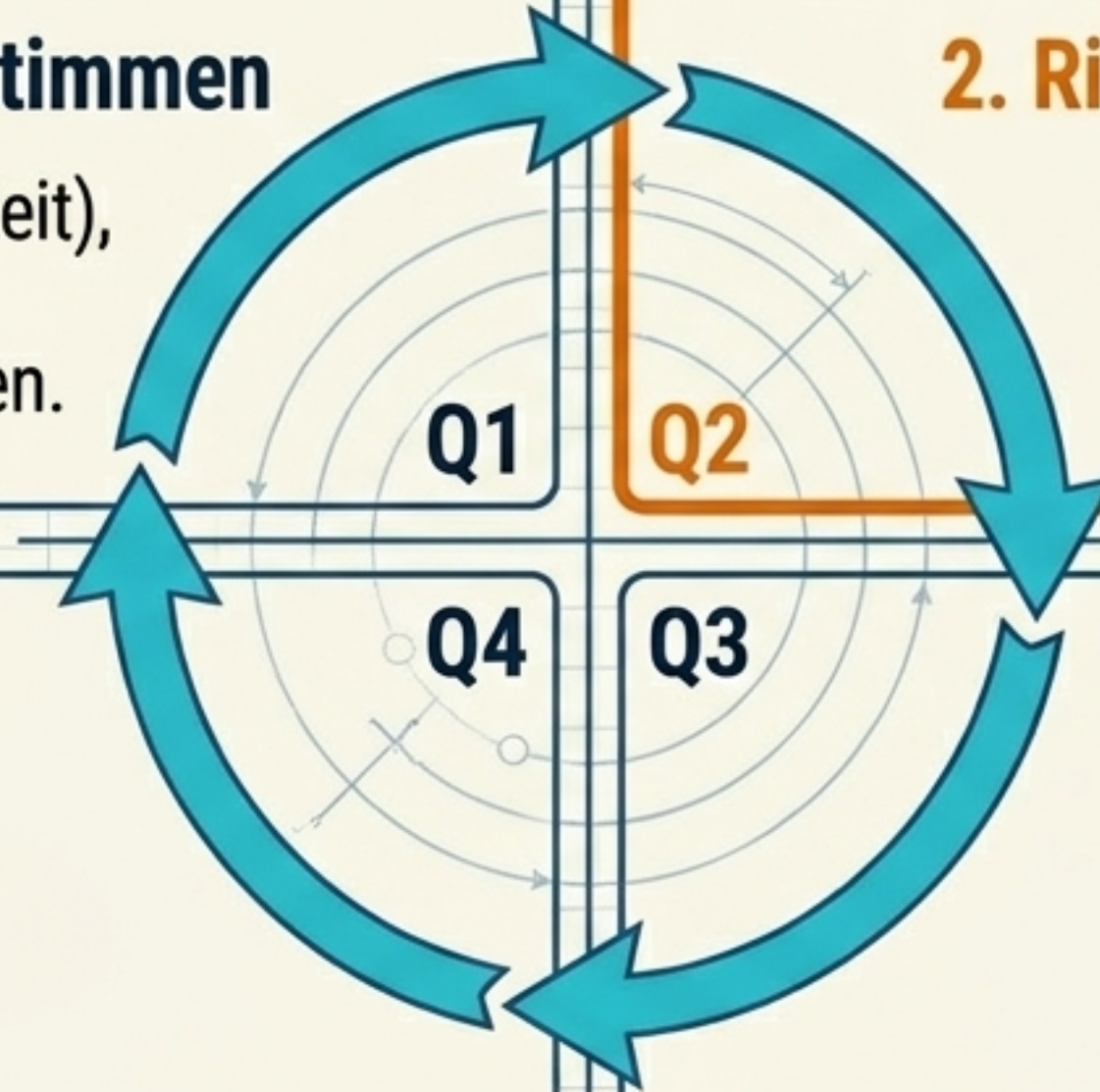
Größte Unsicherheiten analysieren (z.B. mittels Prototypen, Simulationen).

4. Planung der nächsten Iteration

Ergebnisbewertung und Go/No-Go-Entscheidung für die nächste Schleife.

3. Entwicklung & Verifikation

Das nächste Inkrement bauen und validieren.



Typische Iterationsabfolge Machbarkeitsstudie → Erster Prototyp → Pilotversion → Produktivsystem

Das Prototypenmodell: Risikominimierung durch Anschauung

Grundidee: Greifbare Zwischenergebnisse ersetzen umfangreiche Papierspezifikationen zur frühen Validierung von Anforderungen.



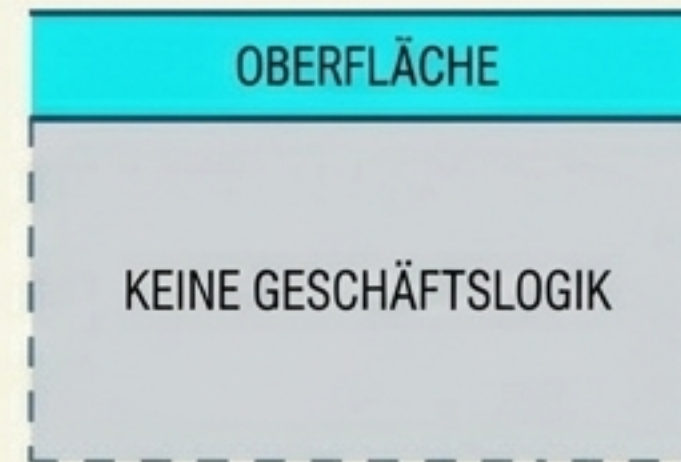
Wegwerf-Prototyp

Reines Vehikel zum Erkenntnisgewinn. Wird nach Anforderungsklä rung entsorgt.



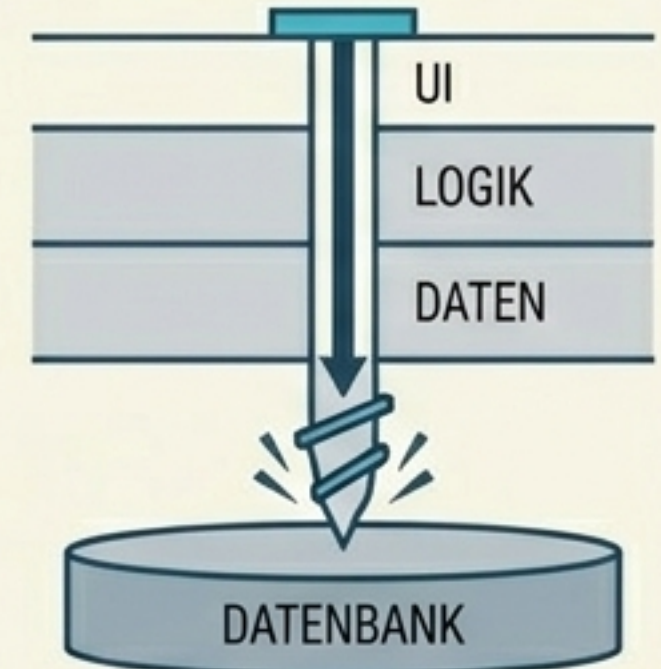
Evolutionärer Prototyp

Wird schrittweise zum finalen Endprodukt ausgebaut (erfordert saubere Basisarchitektur).



Horizontaler Prototyp

Zeigt die breite Oberfläche (z.B. UI-Mockup), besitzt aber keine tiefgehende Geschäftslogik.

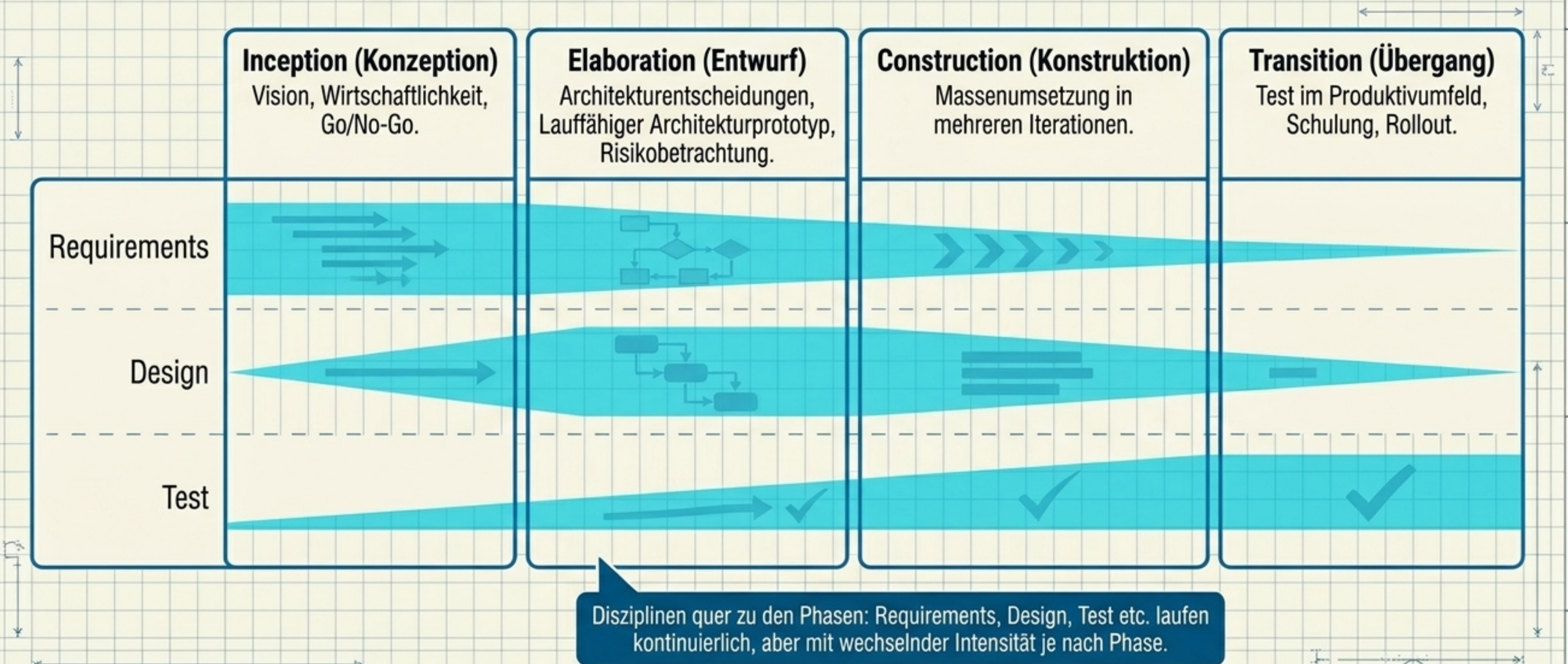


Vertikaler Prototyp

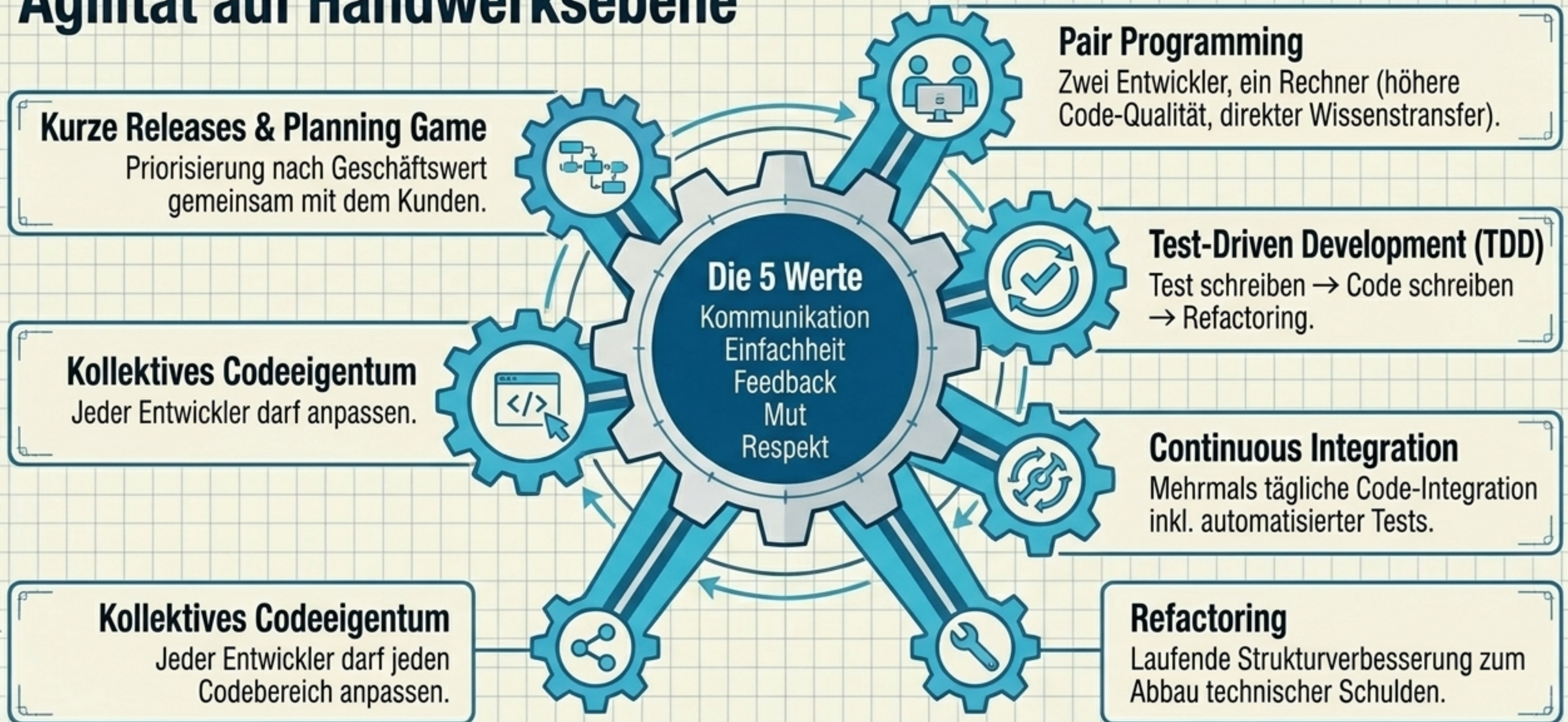
Ein tiefer Funktionsdurchstich von der UI bis zur Datenbank. Beweist die technologische Machbarkeit.

Rational Unified Process (RUP): Strukturierte Iterationen

Ein iterativ-inkrementelles, stark dokumentationsgetriebenes Modell aus den 1990ern (Rational Software).
Prägte viele moderne objektorientierte Vorgehensweisen.



Extreme Programming (XP): Agilität auf Handwerksebene



Methodische Abgrenzung: Spiralmodell ↔ Scrum

Gemeinsamkeiten: Beide arbeiten iterativ-inkrementell, reagieren auf Veränderungen und validieren Zwischenergebnisse.

Spiralmodell

Treiber

Wird von Risiko gesteuert

Iterationslänge

Variabel, je nach Ziel der Schleife

Rollen

Keine festen Rollenvorgaben

Planungsroutinen

Neuplanung nach Quadranten-Logik

Endpunkt

Inkludiert die explizite Abbruchentscheidung

Dokumentation

Umfangreiche formale Artefakte

Scrum

Wird von Geschäftswert gesteuert

Feste Zeitfenster (1-4 Wochen Sprints)

Klar normiert (PO, Scrum Master, Dev-Team)

Etablierte Events (Sprint Planning, Refinement)

Endet bei Erreichung des Produkt-Ziels

So schlank wie möglich ("gerade ausreichend")

Die drei großen Modellfamilien im Überblick

Klassisch (Plangetrieben)

Beispiele:
Wasserfall, V-Modell XT

Sequenzielle Phasen,
umfangreiche Vorab-
Spezifikation, späte Lieferung.
Änderungen sind teuer und
streng formalisiert.

Iterativ (Schleifenbasiert)

Beispiele:
Spiralmodell, RUP

Mehrere Analyse-Bau-Test-
Zyklen.
Konkrete Risikoadressierung,
schrittweise Annäherung an
die optimale Lösung.

Agil (Anpassungsfähig)

Beispiele:
Scrum, Kanban, XP

Hohe Team-Selbstorganisation,
extrem kurze Lieferzyklen,
minimale Dokumentation.
Wandel gilt als
Wettbewerbsvorteil.

Systematischer Paradigmen-Vergleich

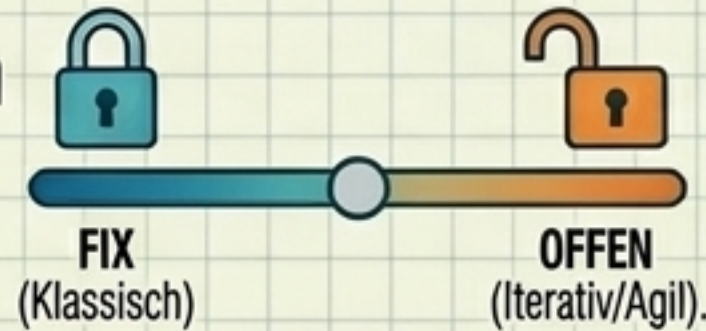
Kriterium	Klassisch	Iterativ	Agil
Anforderungen	Stabil, früh vollständig fixiert	Werden pro Schleife konkretisiert	Veränderlich, laufend verfeinert
Planung	Umfassend, vollständig vorab	Detailliert pro Iteration	Just-in-time (Sprints)
Lieferung	Einmalig (Big Bang) am Projektende	Inkrement nach jeder Iteration	Regelmäßig in sehr kurzen Zyklen
Risikoumgang	Wird oft erst spät im Test sichtbar	Iterativ und explizit adressiert	Kontinuierlich sichtbar und bearbeitet
Kundenkontakt	Nur an definierten Phasenübergängen	Am Ende jeder großen Iteration	Laufend, kontinuierliche Integration
Änderungen	Sehr teuer, erfordern formelle Prozesse	Möglich und gut einplanbar	Erwartet, fester Teil des Systems
Steuerungsaufwand	Sehr hoch am Anfang (Vorab-Planung)	Mittel	Mittel (verteilt über laufende Events)

Diagnose-Kriterien: Modellauswahl in der Praxis

Die Modellauswahl ist keine Geschmackssache, sondern eine fachliche Ingenieursentscheidung basierend auf Rahmenbedingungen.

1. Stabilität der Anforderungen

Fix (Klassisch) vs. Offen (Iterativ/Agil).



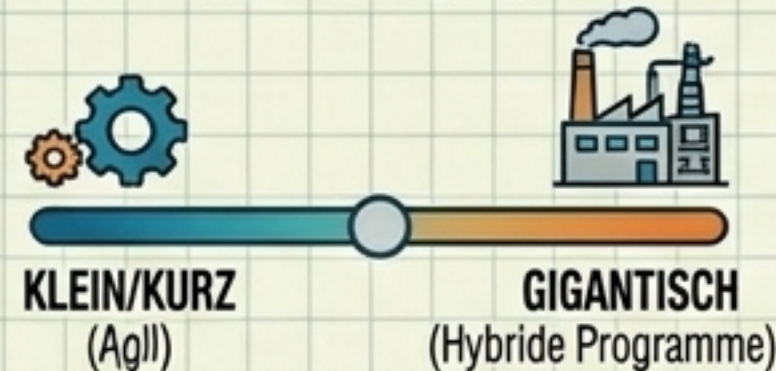
2. Technologisches Risiko

Hohe Unsicherheit zwingt zu Spirale oder Prototypen.



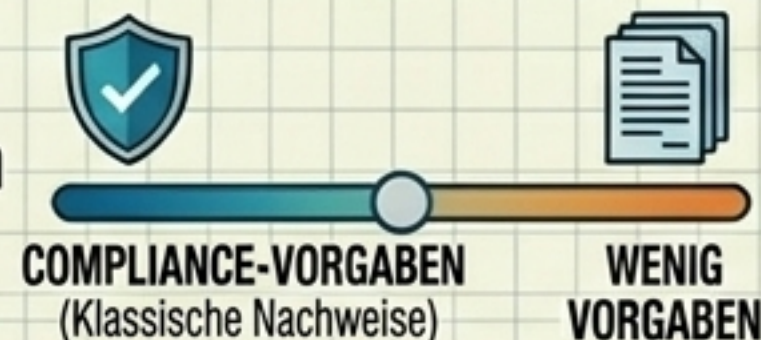
3. Projektgröße & -dauer

Klein/Kurz (Agil) vs. Gigantisch (Hybride Programme).



4. Regulatorisches Umfeld

Compliance-Vorgaben verlangen oft klassische Nachweise.



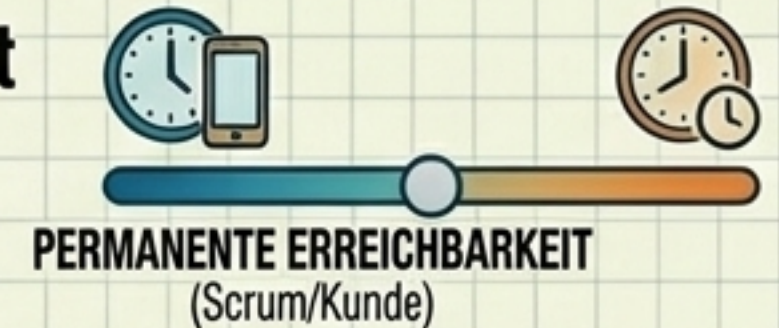
5. Reife der Organisation

Agilität benötigt Eigenverantwortung und kulturelle Reife.



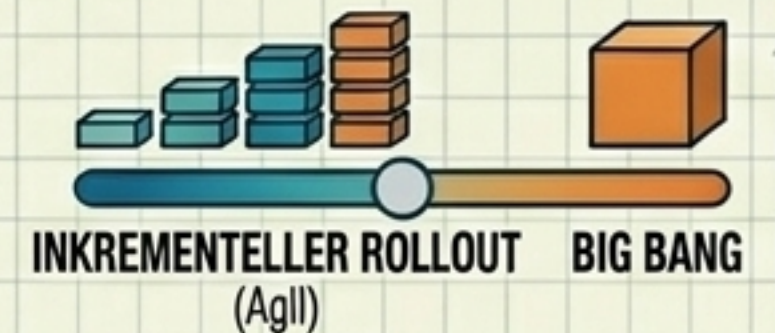
6. Stakeholder-Verfügbarkeit

Scrum erfordert permanente Erreichbarkeit des Kunden.



7. Wirtschaftliche Lieferform

Inkrementeller Rollout ist prädestiniert für agil.



8. Schnittstellen

Altsystem-Abhängigkeiten erzwingen oft sequenzielle Meilensteine.



Typische Projektzuordnungen

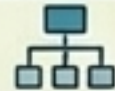
Behördliches IT-Vorhaben

(Regulatorik, definierter Umfang)



Infrastruktur-Migration im Mittelstand

(Standardisiert, wenig Unklarheiten)



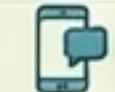
Forschungsprojekt für neue KI-Diagnostik

(Hohes technisches/fachliches Risiko)



Entwicklung einer neuen Mobile-App

(Unklare Marktreaktion, Time-to-Market wichtig)



Wartung & Dauer-Weiterentwicklung

(Laufender Fluss an kleinen Aufgaben)



Sicherheitskritische Embedded Software

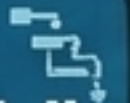
(Medizingerät, Normen)



V-Modell XT



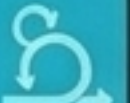
Wasserfall /
schlankes V-Modell



Spiralmodell



Scrum



Kanban



V-Modell XT
(mit Risikomanagement)



Die Unternehmensrealität: Hybride Modelle

In der Praxis werden Elemente verschiedener Familien kombiniert, um Stabilität mit Reaktionsfähigkeit zu vereinen.

Typische Hybrid-Muster:



1. Wasserfall außen, Scrum innen

Budgetierung, Phasengliederung und Berichtslinie erfolgen klassisch. Die Umsetzung der Arbeitspakete erfolgt agil in Sprints.



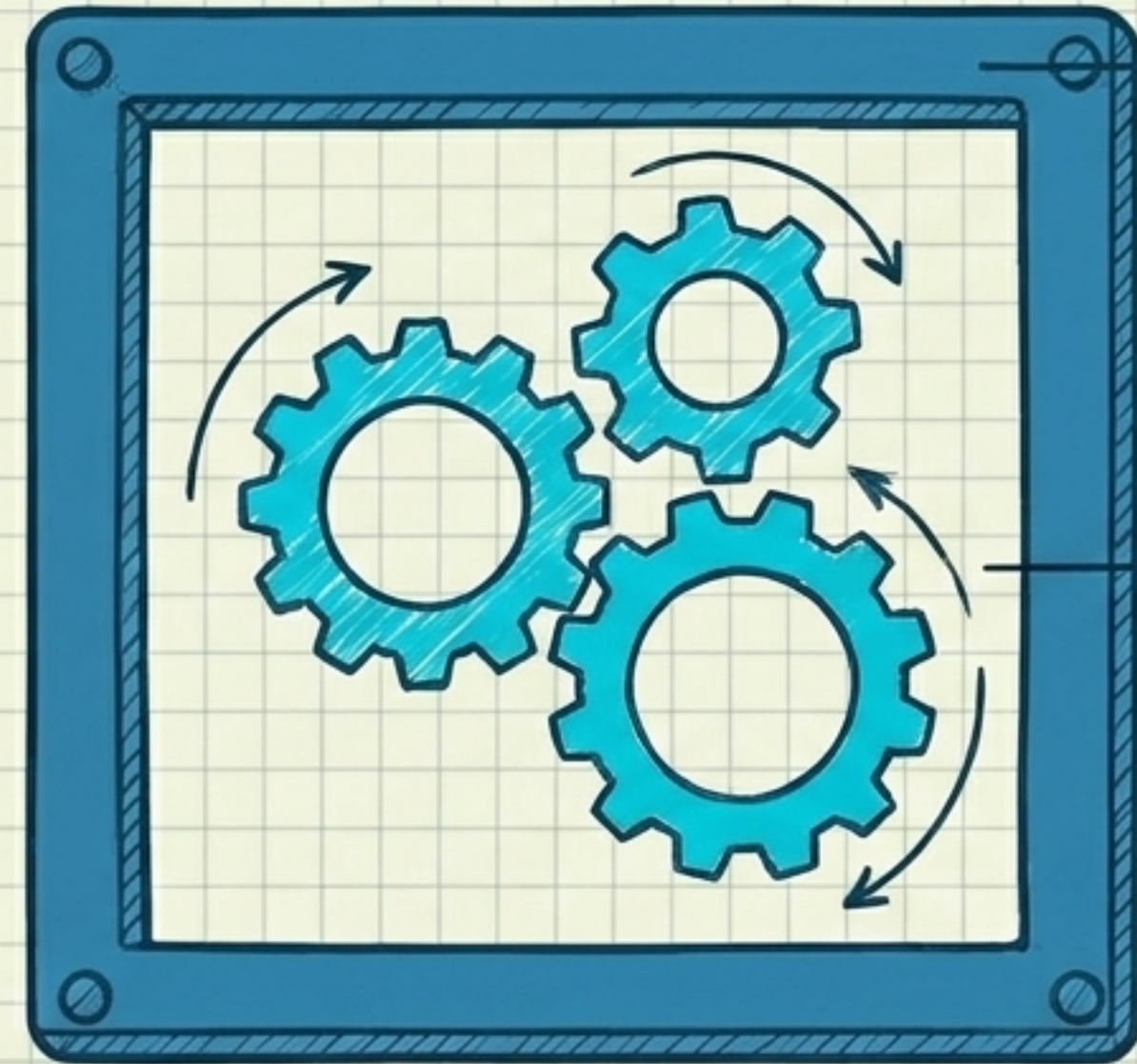
2. Spirale mit agilen Iterationen

Die strategische Risikosteuerung folgt der Spirale, die Quadranten der Umsetzung werden als Scrum-Zyklen gefahren.



3. Scaled Agile (SAFe, LeSS)

Skalieren agile Techniken auf Unternehmensebene inklusive klassischer Programmplanung.



Klassisch
(Waterfall)
Rahmen

Agil (Scrum)
Sprints

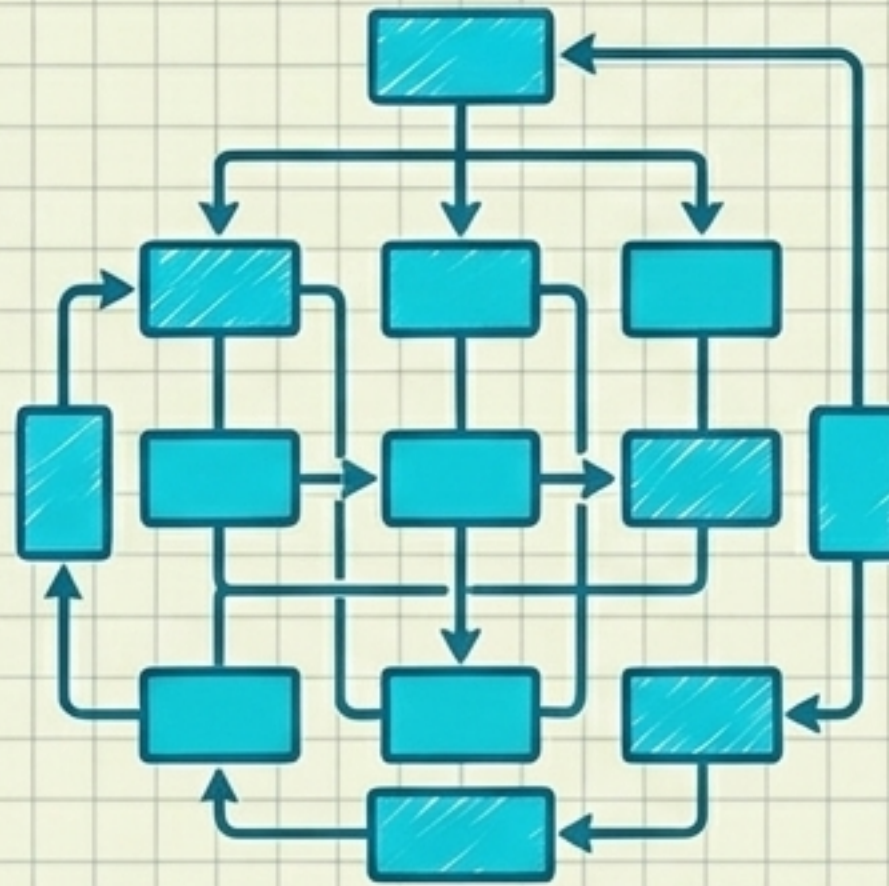
Grundvoraussetzung: Widerspruchsfreiheit. Ein klassisches wöchentliches Reporting-Raster funktioniert nicht mit einem autonomen 3-Wochen-Sprint-Team. Die Mechanik muss harmonisiert werden.

Die Architektur des Vorgehens: Zusammenfassung



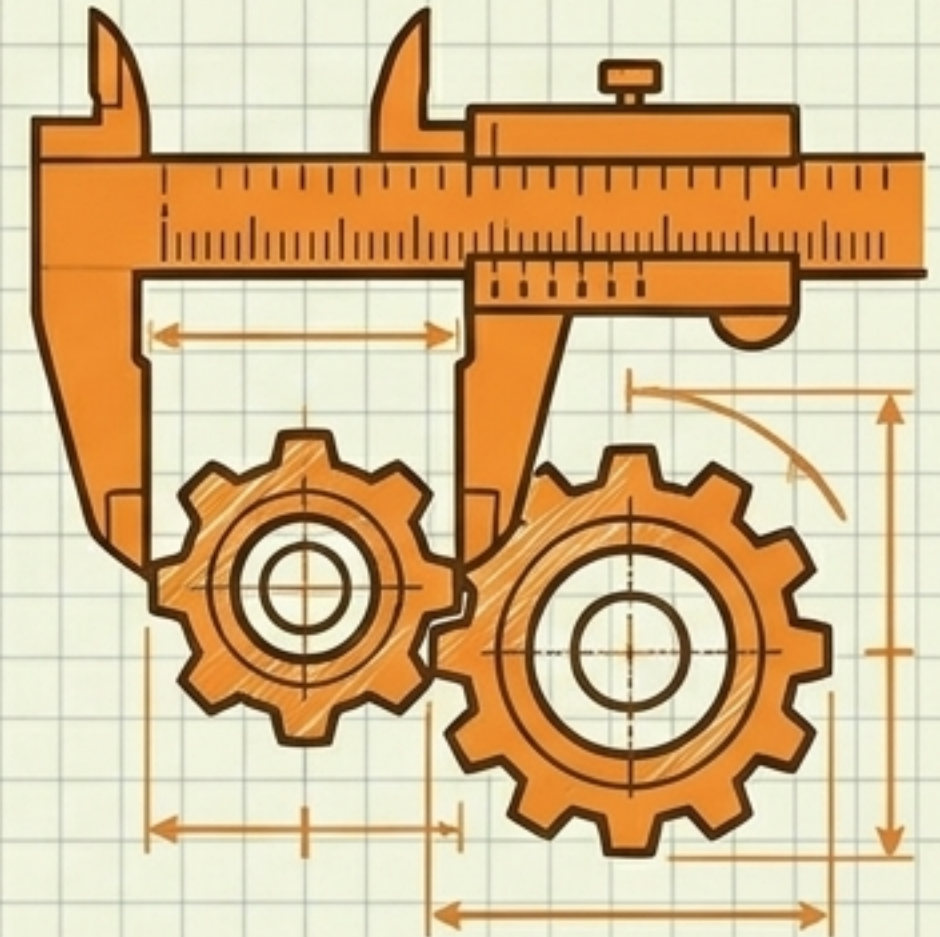
Die Brückenfunktion

Das Spiralmodell brachte Iteration und aktives Risikomanagement in die Projektwelt. Methoden wie Prototyping, RUP und XP lieferten Werkzeuge, die heute in modernen Frameworks leben.



Das methodische Raster

Die Einteilung in Klassisch, Iterativ und Agil dient als gedankliches Fundament zur Einordnung von Projektmethoden und -bedürfnissen.



Modellauswahl als Ingenieursaufgabe

Die Wahl des Vorgehens ist niemals Geschmackssache. Sie resultiert aus einer harten Analyse von Technologie-Risiko, Compliance und Unternehmenskultur. Nur wer die reinen Modelle versteht, kann funktionsfähige Hybride bauen.