

# Softwaretechnik C - Softwaremanagement



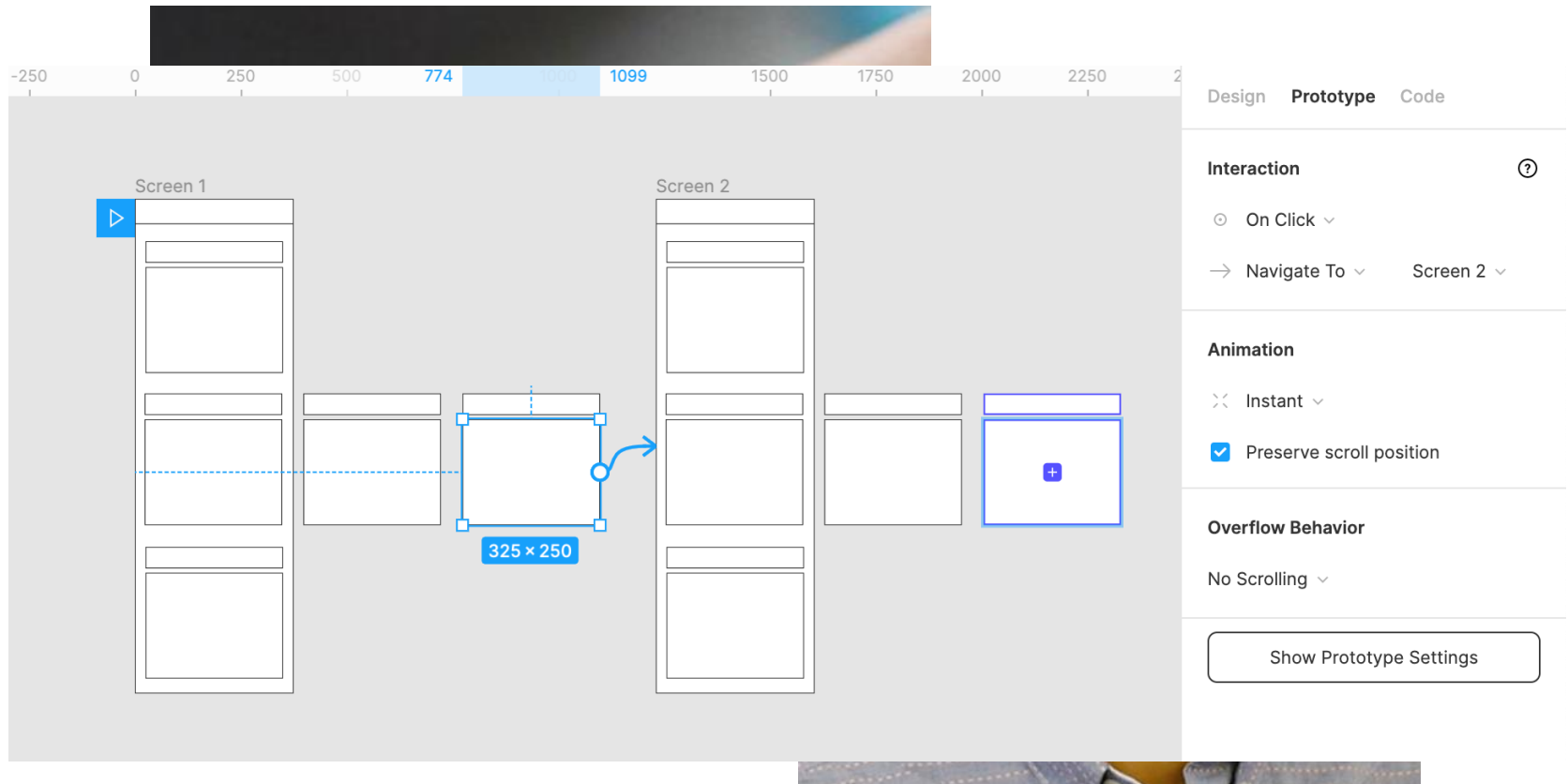
## LE 02: Vorgehens- und Prozessmodelle



# Organisatorisches

- Veranstaltungen SWT-C nächste Woche finden **nicht** statt (Vorlesungsfrei)
- Betroffen:
  - Vorlesung am **10.10.2022**
  - Praktikum am **11.10.2022**





Quelle: [iconstorm.com](https://iconstorm.com); [pinterest.com](https://pinterest.com); [figma.com](https://figma.com)

# Agenda

- Klassische Vorgehensmodelle
  - Wasserfallmodell
  - Nebenläufiges Modell
  - Spiralmodell
  - **Prototyping**
- Monumentale Vorgehensmodelle
  - **V-Modell XT**
  - **Rational Unified Process (RUP)**
- Agile Modelle
  - Scrum
  - Kanban (IT-Kanban)
  - Extreme Programming (XP)
  - Feature Driven Development
  - Adaptive Software Development
  - Crystal
  - Lean Software Development



- Klassische Vorgehensmodelle
  - Wasserfallmodell
  - Nebenläufiges Modell
  - Spiralmodell
  - **Prototyping**
- Monumentale Vorgehensmodelle
  - V-Modell XT
  - Rational Unified Process (RUP)
- Agile Modelle
  - Scrum
  - Kanban (IT-Kanban)
  - Extreme Programming (XP)
  - Feature Driven Development
  - Adaptive Software Development
  - Crystal
  - Lean Software Development



# Prototyping

## Grundidee und Einsatzgebiete (1/2)

- „Fail fast“
- **Prototyp-basierte** Vorgehensmodelle dienen der präzisen & vollständigen Erfassung der **Benutzeranforderungen** an ein Softwareprodukt
- Beim Prototyping werden **nur wesentliche** Systemmerkmale implementiert
- Durch **Erprobung und Testen** des Prototyps in Kooperation mit den Benutzern werden **neue, weitere Anforderungen** identifiziert
- Versuch, durch eine anschauliche und dem zu entwickelnden späteren Softwaresystem möglichst nah kommende vorläufige Version eine **frühzeitige und möglichst umfassende** Bestimmung der Anforderungen umzusetzen



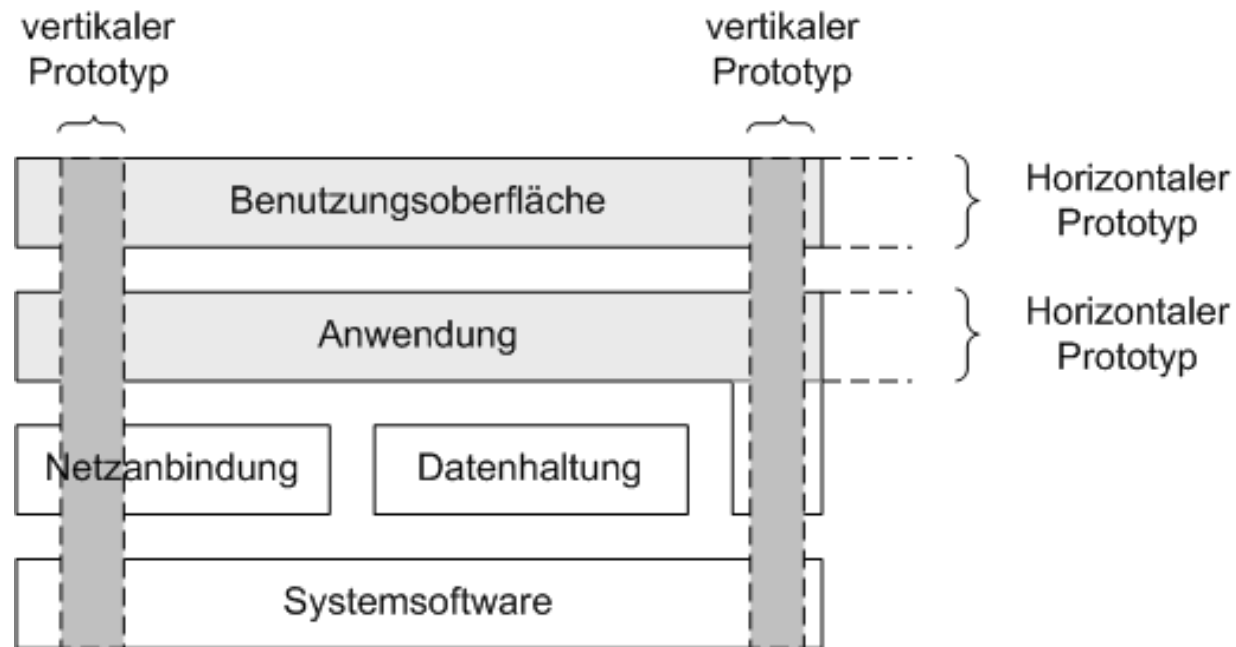
**Vertikales Prototyping**  
Durchstich-Prototyp realisiert  
Untermenge der Gesamtfunktionalität  
durch alle Ebenen

**Horizontales Prototyping**  
Nur eine Ebene bzw. Schicht des  
Softwaresystems wird entwickelt (oft GUI)

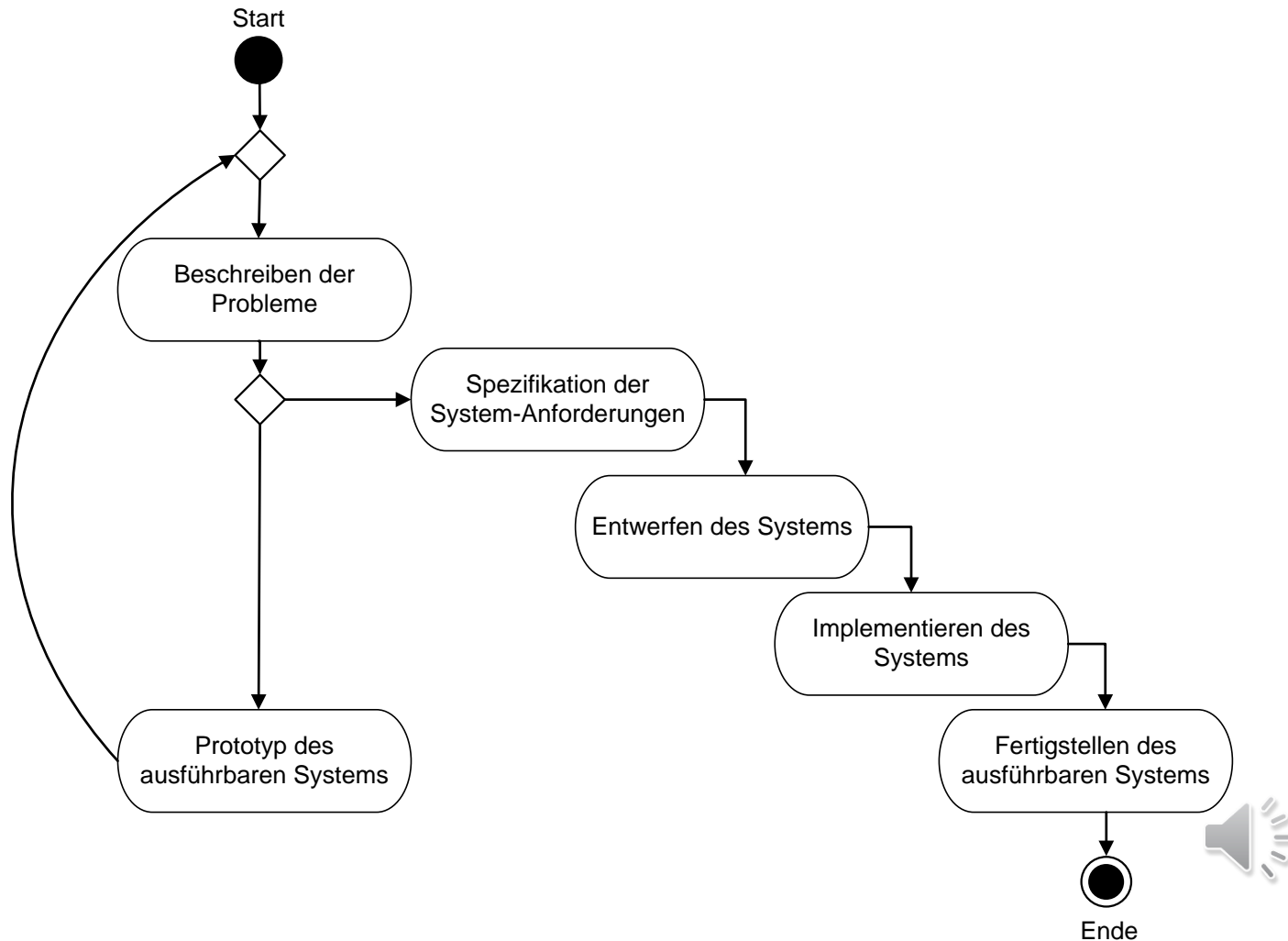


# Prototyping

## Horizontales vs. Vertikales Prototyping (2/2)



# Prototyping Überblick (1/2)



# Prototyping

## Überblick (2/2)

- Verschiedene Arten des Prototyping in Abhängigkeit zur projektspezifischen Zielsetzung
  1. Exploratives Prototyping
  2. Experimentelles Prototyping
  3. Evolutionäres Prototyping



- Prototyp dient als **Kommunikationsgrundlage** für die **Softwareentwickler** und die späteren **Benutzer** (oftmals GUI-Prototyp)
  1. Softwareentwickler verbessern so ihre Kenntnisse im Anwendungsbereich
  2. Benutzer können unpräzise Anforderungen an das zu erstellende Softwareprodukt konkretisieren
  3. Häufig genutzt für (iterative) Anforderungsentwicklung
- Es werden sogenannte **Wegwerf-Prototypen** (Throw-away-Prototypen) eingesetzt, die **nicht** zum späteren Softwareprodukt weiterentwickelt werden
- Dadurch sind beim explorativen Prototyping nicht von Beginn an die gleichen **Qualitätsanforderungen** zu realisieren, wie für das spätere Softwareprodukt
- **Ziel:** Nachweis über Tauglichkeit einer Idee / Spezifikation
- **Ergebnis:** belastbare **Anforderungsspezifikation**



# Prototyping

## 2. Experimentelles Prototyping

- Das experimentelle Prototyping wird während des Softwareentwicklungsprozesses eingesetzt, um **Lösungsalternativen** zu diskutieren und zu evaluieren
  - Jede Lösungsalternative wird **prototypisch implementiert**
  - Die **Evaluierung** (Erprobung und Bewertung) wird anhand von Tests an den Prototypen vorgenommen
- Prototypen können sowohl als Wegwerf-Prototypen realisiert als auch zum späteren Softwareprodukt weiterentwickelt werden
- Im letzteren Fall müssen von Beginn an die gleichen **Qualitätsanforderungen** wie an das spätere Softwareprodukt eingeplant und umgesetzt werden
- **Ziel:** Sammeln von Erfahrungen mit Prototyp
- **Ergebnis:** experimenteller Prototyp



# Prototyping

## 3. Evolutionäres Prototyping

- Das evolutionäre Prototyping wird in der Literatur häufig mit evolutionären bzw. **inkrementellen Vorgehensmodellen** gleichgesetzt
- Beim evolutionären Prototyping wird im Gegensatz zum explorativen Prototyping der Prototyp bzw. die vorläufige Version **zum späteren Softwareprodukt weiterentwickelt**
- Somit gelten bereits zu Beginn der Prototypentwicklung die gleichen **Qualitätsanforderungen** wie für das zu erstellende Softwareprodukt
- **Ziel:** Akzeptanz durch Benutzer überprüfen
- **Ergebnis:** Programm mit Grundfunktionalitäten



# Prototyping

## Bewertung: Vorteile

- Durch den frühzeitigen Einsatz von Prototypen kann das **Entwicklungsrisiko** deutlich reduziert werden
- GUI-Prototypen können durch geeignete WYSIWYG-Werkzeuge (What You See Is What You Get) **schnell und einfach** erstellt werden
- Prototyping verbessert durch die starke Rückkopplung mit den späteren Benutzerinnen und Benutzern:
  - die **Planbarkeit** des Softwareentwicklungsprozesses
  - die **Qualität** der erzeugten Ergebnisse
  - **reduziert Missverständnisse**
- Prototyping unterstützt und fokussiert die spezifischen Charakteristika eines Softwareentwicklungsprojekts (**frühe Integration der Stakeholder**)



# Prototyping

## Bewertung: Nachteile

- Tendenziell große Gefahr, dass ein Wegwerf-Prototyp aus Termingründen zum **endgültigen Softwareprodukt** weiterentwickelt wird
- Mithilfe von Prototypen wird oftmals die **fehlende bzw. unzureichende Dokumentation** im Rahmen eines Softwareentwicklungsprozesses substituiert

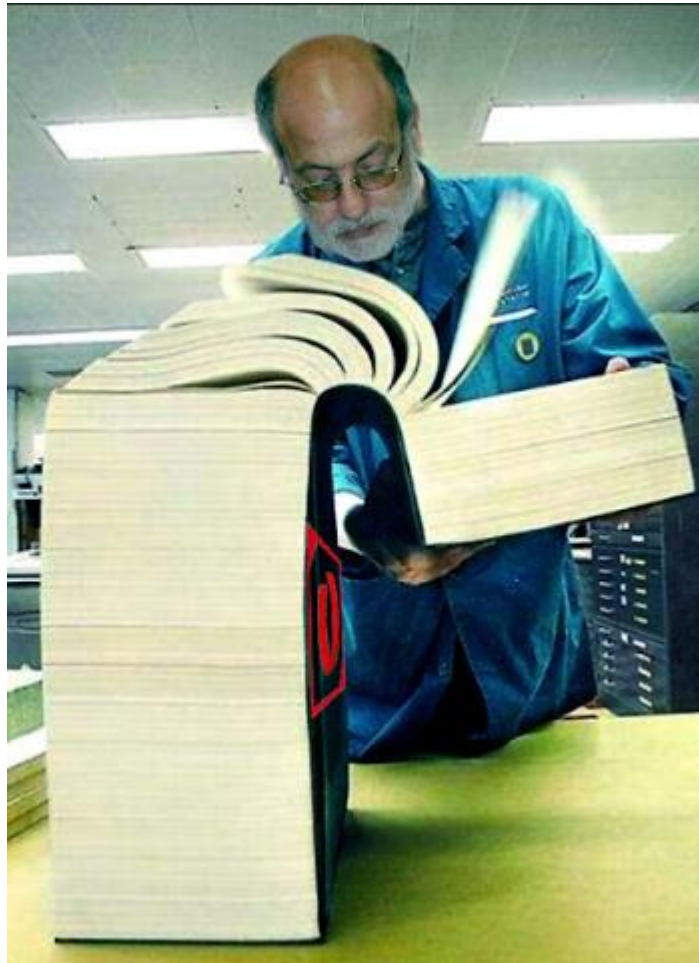


# Prototyping

## Klassifizierung und praktische Relevanz

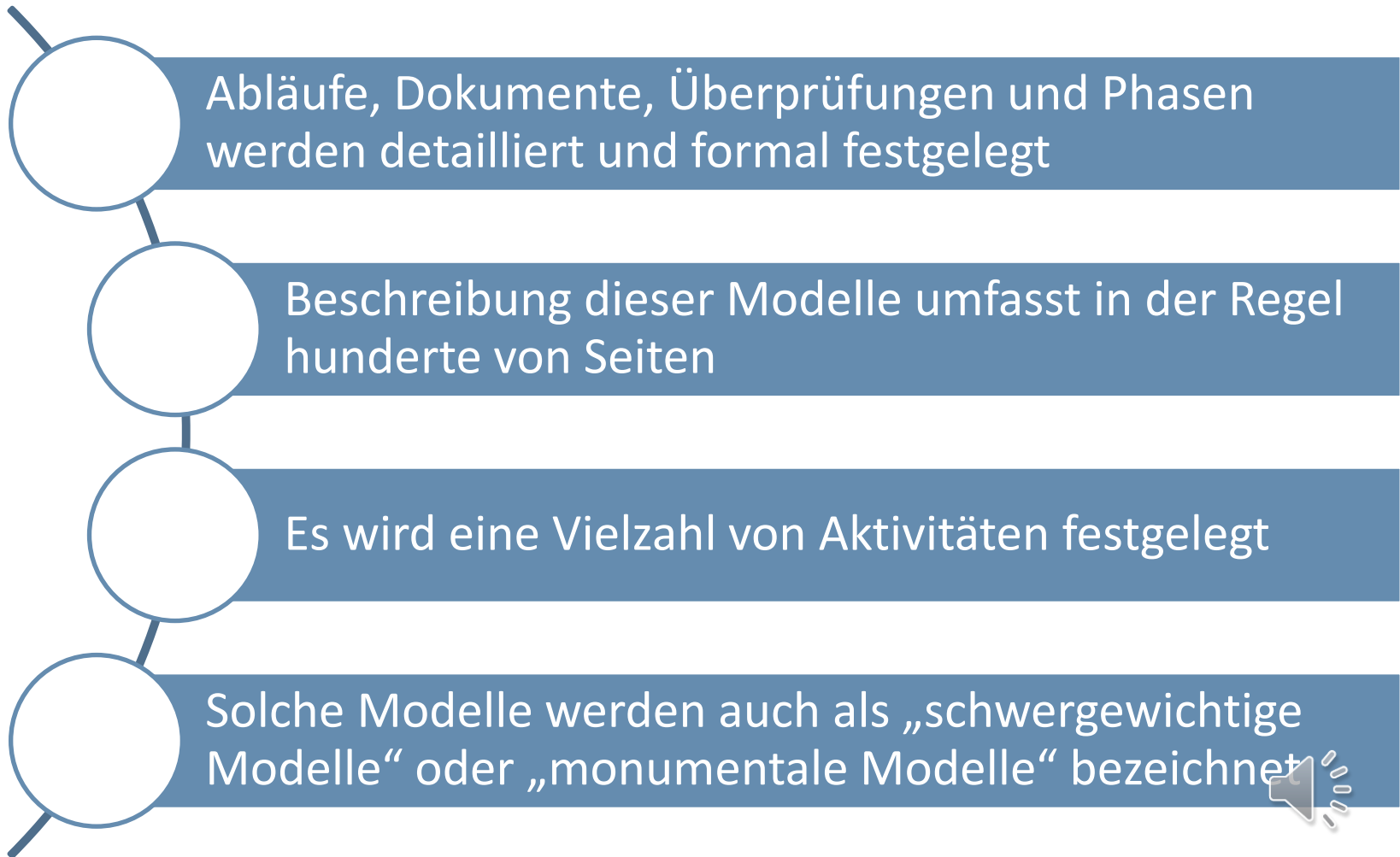
- Prototyping wird in der **Praxis** immer häufiger eingesetzt
- Oftmals gehört ein GUI-Prototyp zu den drei zentralen Ergebnissen der objektorientierten Analysephase (OOA):
  - Pflichtenheft (inkl. Anforderungsdokument)
  - OOA-Modell mit textueller Beschreibung der statischen und dynamischen Aspekte des zu entwickelnden Softwaresystems
  - Zugehörige UML-Abbildungen (UML als Spezifikationssprache für objektorientierte Softwaresysteme)
  - GUI-Prototyp und Benutzungshandbuch
- Wesentliche Methode des ***Design Thinking***





- Klassische Vorgehensmodelle
  - Wasserfallmodell
  - Nebenläufiges Modell
  - Spiralmodell
  - **Prototyping**
- Monumentale Vorgehensmodelle
  - **V-Modell XT**
  - Rational Unified Process (RUP)
- Agile Modelle
  - Scrum
  - Kanban (IT-Kanban)
  - Extreme Programming (XP)
  - Feature Driven Development
  - Adaptive Software Development
  - Crystal
  - Lean Software Development





- Eines der **umfassendsten monumentalen Modelle** ist das V-Modell XT
- Stammt ursprünglich aus dem militärischen Bereich (V-Modell)
- Wurde erstmals **1992** im Bundesanzeiger veröffentlicht
- Ursprüngliche **Ziele**:
  - Entwicklung eines Vorgehensmodells für deutsche Bundeswehr, das anschließend für **Bundesbehörden** erweitert wurde
  - Sollte dazu beitragen, die **Softwareerstellung** und -beauftragung im Bereich der Bundesverwaltung zu **standardisieren**
- Ist heutzutage **für alle Bundesbehörden in Deutschland verbindlich**
- „XT“ steht für **Extreme Tailoring** und soll andeuten, dass das Modell für verschiedene Projekttypen zugeschnitten werden kann



# V-Modell XT

## Historie, Entstehung und Einsatzgebiete (4/4)

- Aktuell: Version 2.3
- Gesamtüberblick umfasst 439 Seiten
- Informationen und gesamte Dokumentation:
  - [http://www.cio.bund.de/Web/DE/Architekturen-und-Standards/V-Modell-XT/vmodell\\_xt\\_node.html](http://www.cio.bund.de/Web/DE/Architekturen-und-Standards/V-Modell-XT/vmodell_xt_node.html)

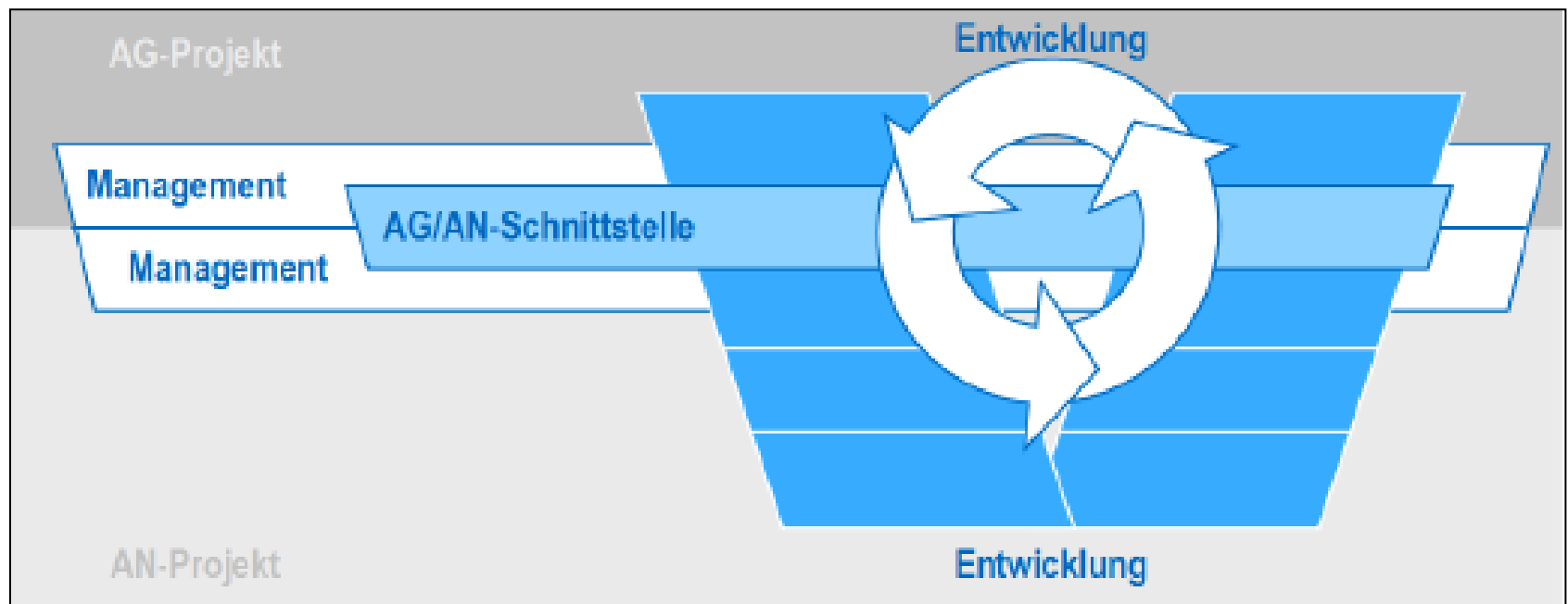


## Exkurs: Das V-Modell



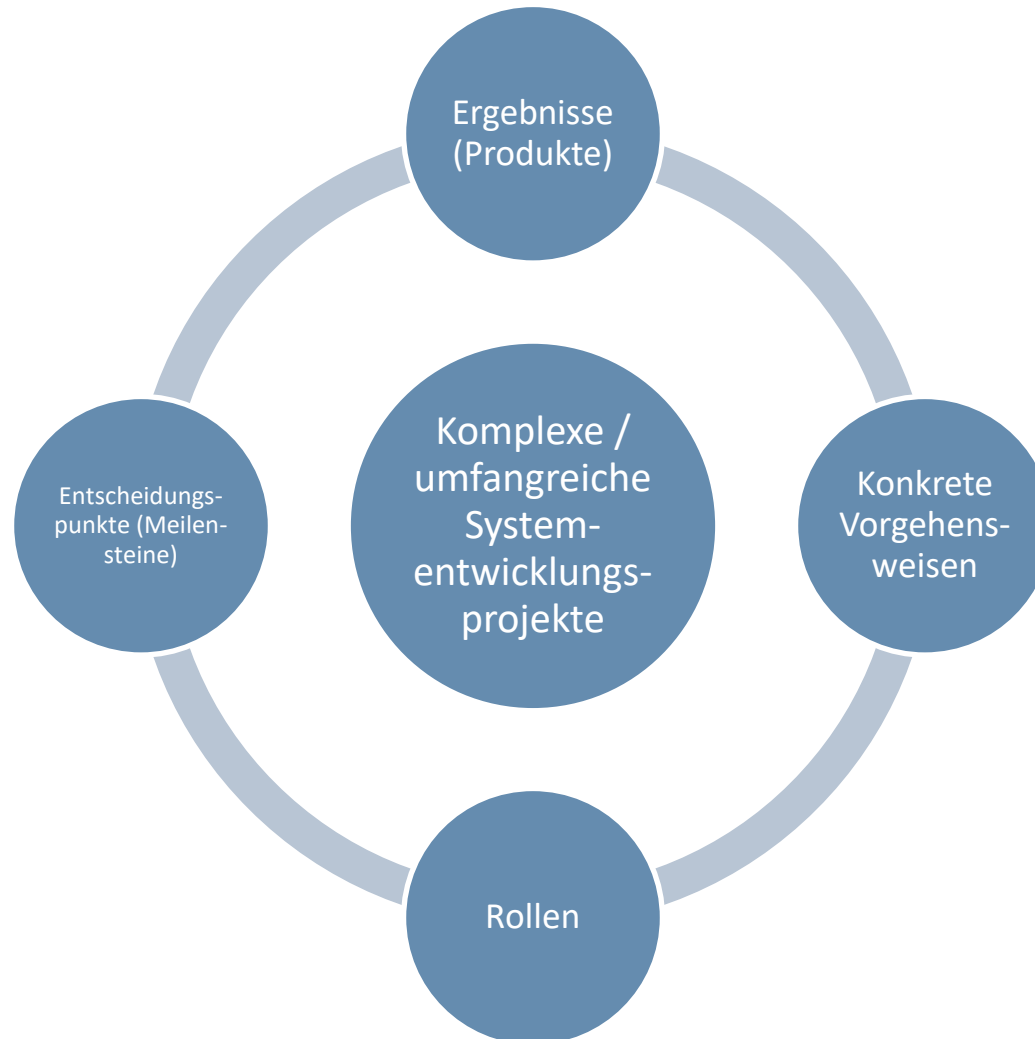
# V-Modell XT

## Überblick



# V-Modell XT

## Aufgaben



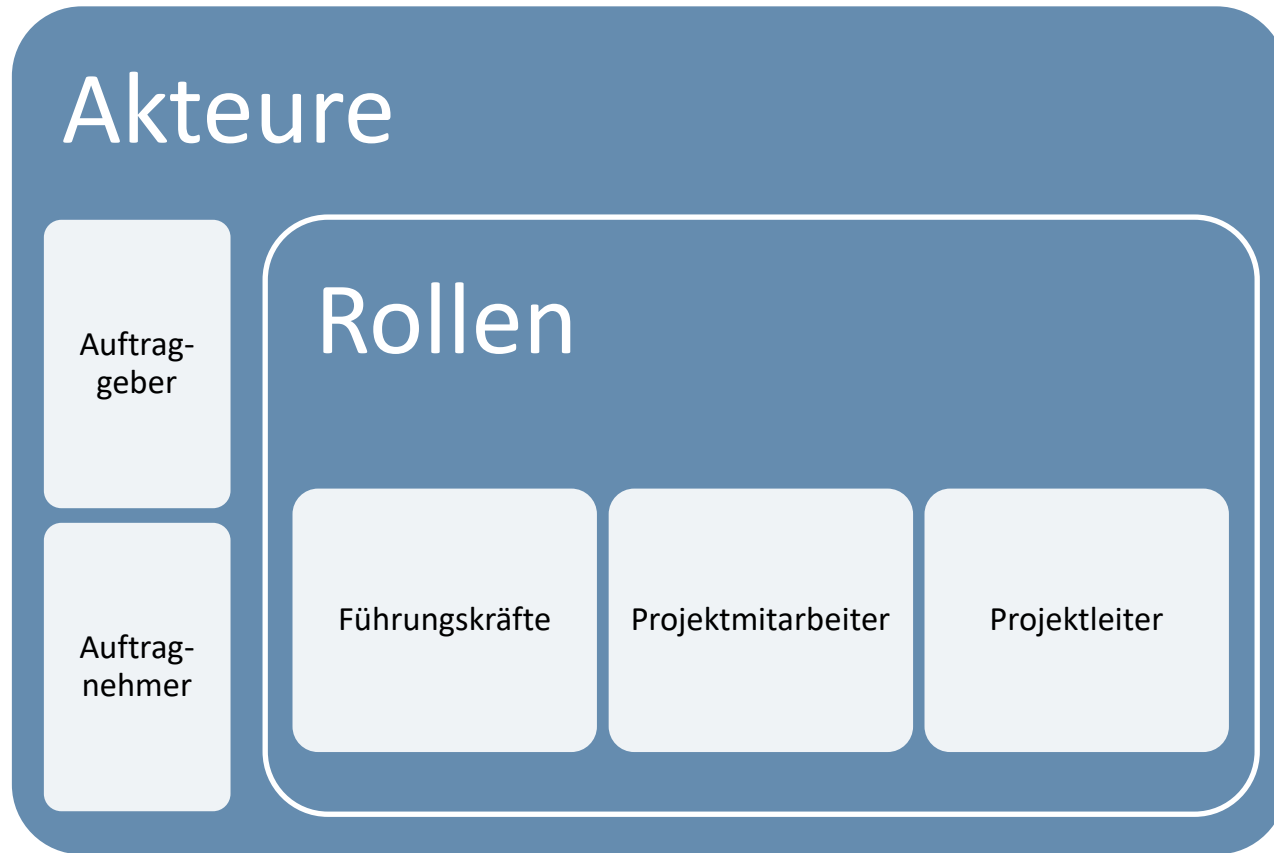
# V-Modell XT

## Ziele und Zielgruppen (1)



# V-Modell XT

## Ziele und Zielgruppen (2)

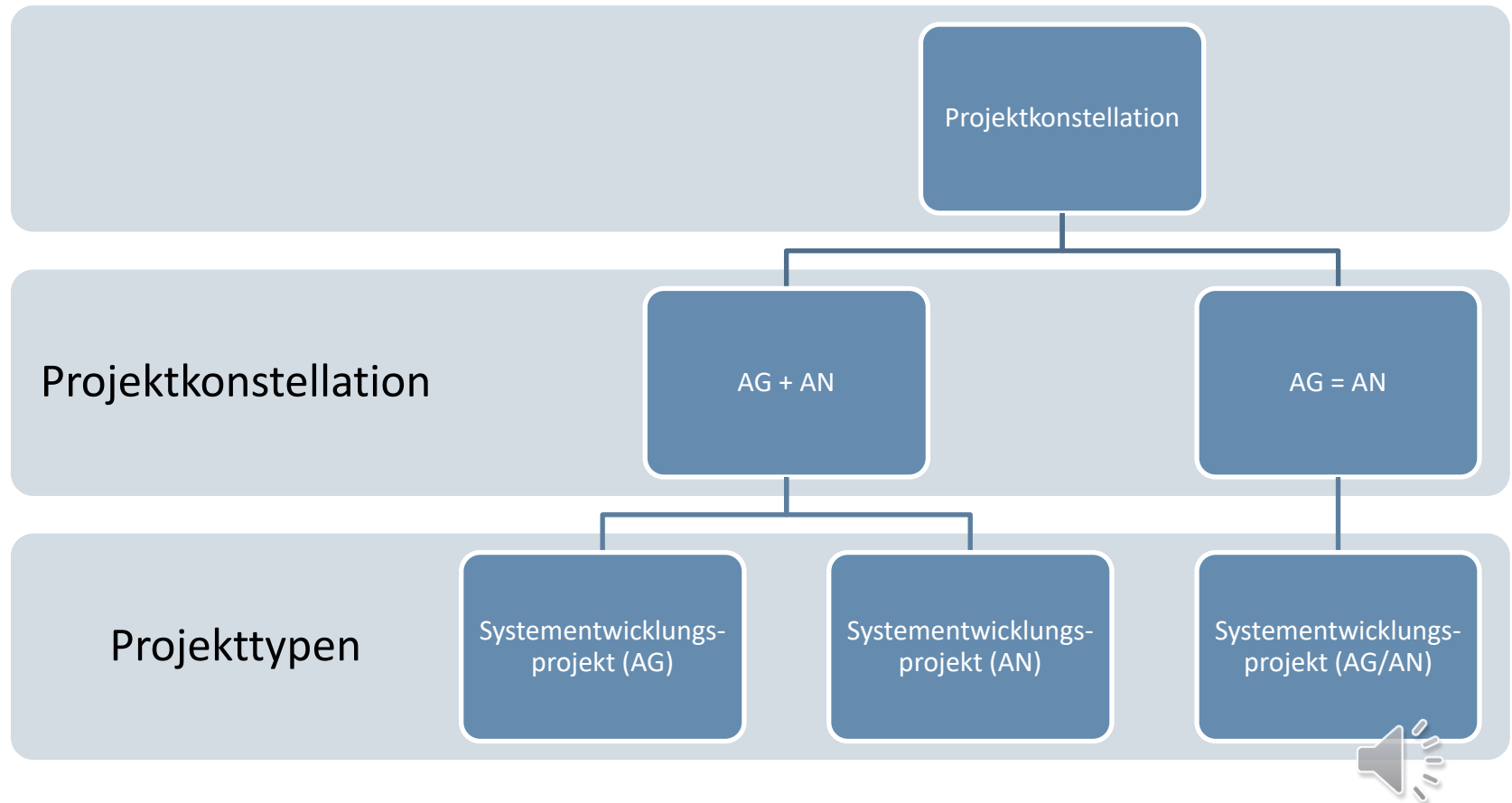


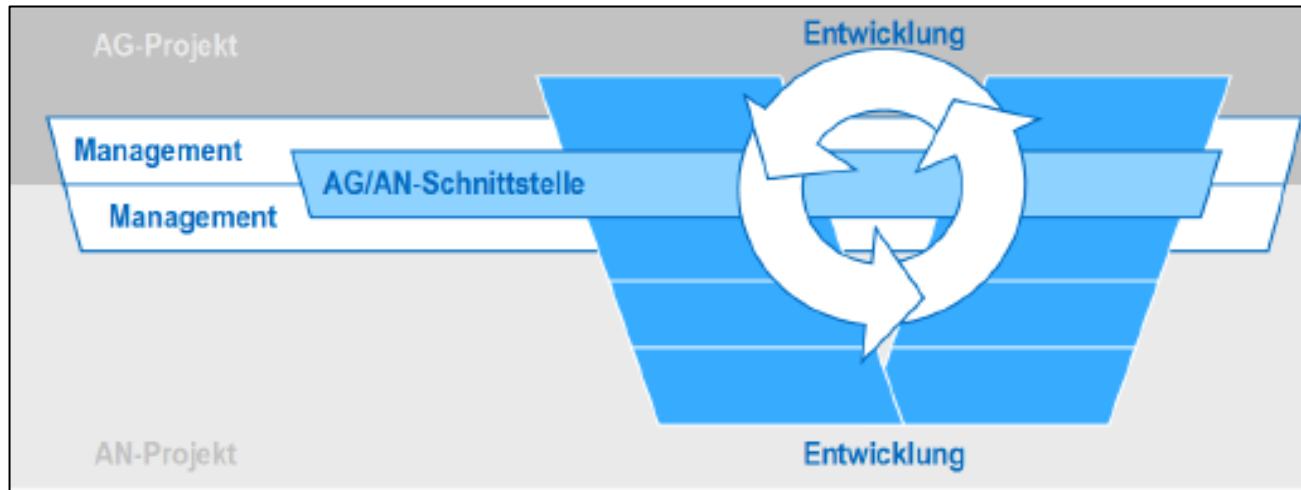
- **Projekttypen und Projektdurchführungsstrategien** bilden die Grundstruktur des V-Modells XT
- **Vorgehensbausteine** beschreiben die Bearbeitung von Aufgaben
- V-Modell XT lässt sich an spezifische **Projektkonstellationen** anpassen
- Die Aufgabenverteilung an der **Schnittstelle zwischen Auftraggeber und Auftragnehmer** ist klar und nachvollziehbar beschrieben



# V-Modell XT

## Projektkonstellationen und Projekttypen (1/4)

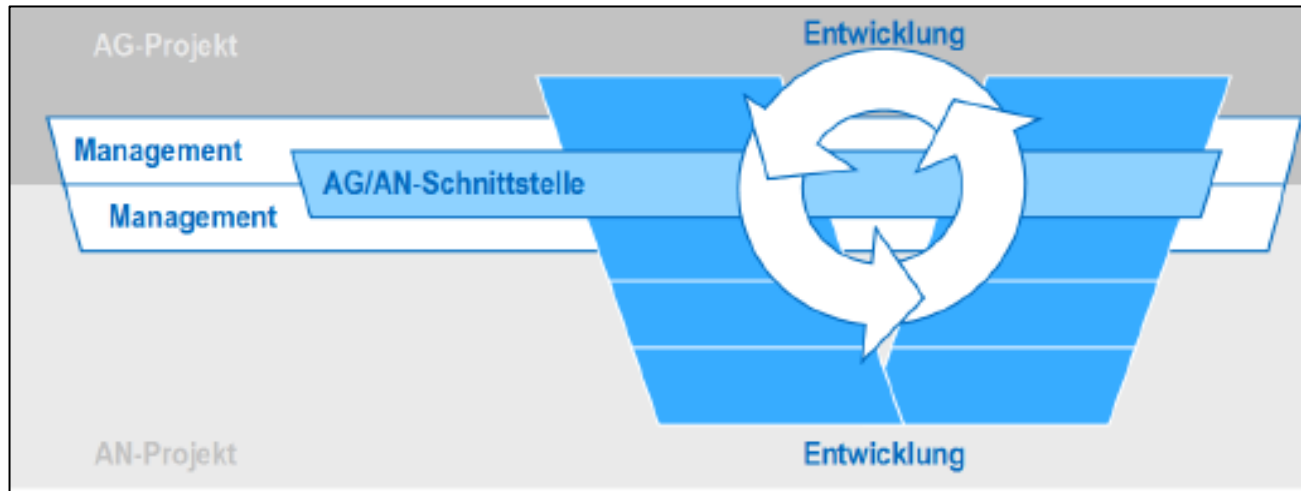




### ■ Projektkonstellation: **AG + AN**

- Jeder der beiden Projekttypen enthält die gleichen Management-mechanismen (weiß), die jedoch auf Auftraggeber- und Auftragnehmerseite jeweils projektspezifisch ausgestaltet und durchgeführt werden
- Über eine AG/AN-Schnittstelle (hellblau) tauschen AG und AN relevante Produkte aus, wie zum Beispiel den Vertrag oder die Lieferung





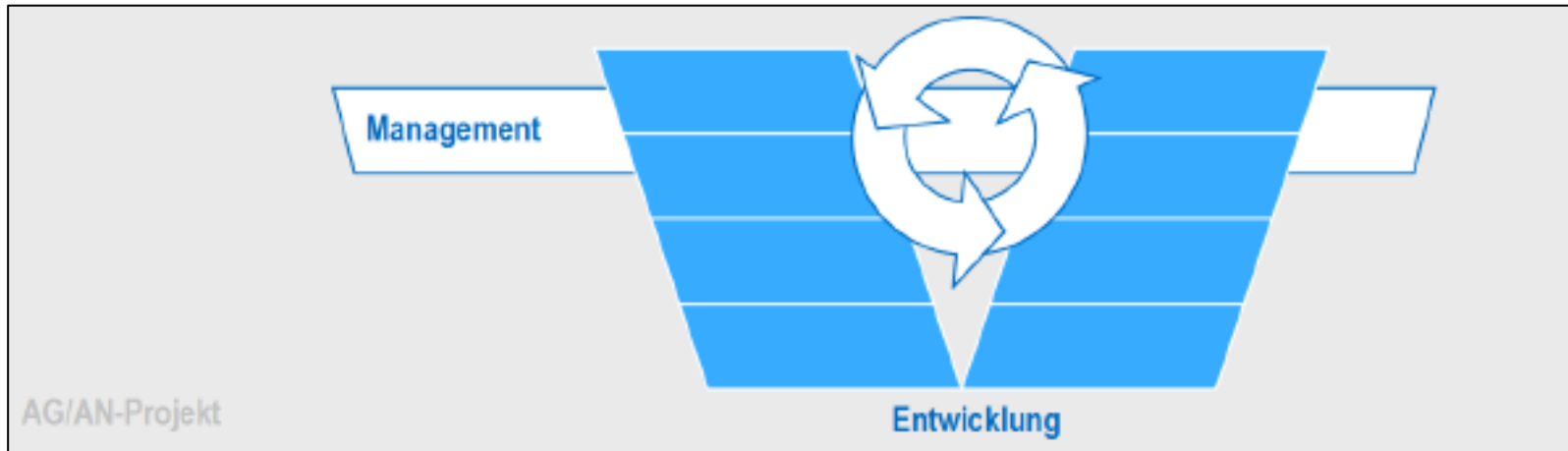
### ■ Projektkonstellation: **AG + AN**

- Der AN führt letztlich nach den Vorgaben des AG (Lastenheft mit Anforderungen) die Systementwicklung (dunkelblau) durch
- Zu jedem AG-Projekt wird damit durch den beauftragten Auftragnehmer ein AN-Projekt initiiert
- AG- und AN-Projekt laufen somit parallel ab



# V-Modell XT

## Projektkonstellationen und Projekttypen (4/4)




### ■ Projektkonstellation: **AG = AN**

- Ein dritter Projekttyp heißt Systementwicklungsprojekt (AG/AN)
- Anforderungsfestlegung, Projektabwicklung und Entwicklung finden innerhalb einer Organisation statt
- kein außenstehender AG vorhanden => AG/AN-Schnittstelle kann entfallen
- Bereiche Management und Entwicklung sind in diesem Projekttyp relevant
- Rollen des AG und AN werden organisationsintern besetzt, z. B. durch verschiedene Abteilungen.



- Grobe Unterscheidung in Projekttypen stellt erste grundlegende Anpassung der relevanten Teile des V-Modell XT dar
- Konkretes Projekt kann durch **Projekttypvarianten** und davon abhängig durch **Projektmerkmale** weiter charakterisiert werden
- Das V-Modell XT schlägt für **jeden Projekttyp zwei Projekttypvarianten** vor:

Projekttyp	Systementwicklungsprojekt (AG)	Systementwicklungsprojekt (AN)	Systementwicklungsprojekt (AG/AN)
Projekttypvariante	AG-Projekt mit einem Auftragnehmer	AN-Projekt mit Entwicklung, Weiterentwicklung oder Migration	AG/AN-Projekt mit Entwicklung, Weiterentwicklung oder Migration
	AG-Projekt mit mehreren Auftragnehmern	AN-Projekt mit Wartung und Pflege	AG/AN-Projekt mit Wartung und Pflege 

Projekttyp	Systementwicklungsprojekt (AG)	Systementwicklungsprojekt (AN)	Systementwicklungsprojekt (AG/AN)
Projekttypvariant	AG-Projekt mit einem Auftragnehmer	AN-Projekt mit Entwicklung, Weiterentwicklung oder Migration	AG/AN-Projekt mit Entwicklung, Weiterentwicklung oder Migration
	AG-Projekt mit mehreren Auftragnehmern	AN-Projekt mit Wartung und Pflege	AG/AN-Projekt mit Wartung und Pflege

- **AG-Projekte** können 1. mit einem AN oder 2. mit mehreren ANs durchgeführt werden
  - Im 2. Fall wird Systementwicklung auf verschiedene ANs und Teilprojekte aufgeteilt
  - Projektfortschritt wird auf Teilprojektebene UND auf Gesamtprojektebene überprüft
- Projekttypvarianten für **AN-Projekte bzw. AG/AN-Projekte** beschäftigen sich
  - mit der Weiterentwicklung oder Migration von Systemen oder
  - definieren ein Vorgehen für die Wartung und Pflege von Systemen



# V-Modell XT

## Vorgehensbausteine (1/3)

- V-Modell XT fasst ähnlich gelagerte Tätigkeiten zu einem so genannten **Vorgehensbaustein** zusammen
- Einige Vorgehensbausteine finden bei allen Projekten Anwendung und werden daher als **V-Modell-Kern** bezeichnet.
- Dazu gehören:
  - Projektmanagement (PM)
  - Qualitätssicherung (QS)
  - Konfigurationsmanagement (KM)
  - Problem- und Änderungsmanagement (PA)



# V-Modell XT

## Vorgehensbausteine (2/3)

### ■ Weitere Vorgehensbausteine im V-Modell XT

- Anforderungsfestlegung
- Lieferung und Abnahme (AG)
- Lieferung und Abnahme (AN)
- Vertragsschluss (AG)
- Vertragsschluss (AN)
- Kaufmännisches Projektmanagement
- Messung und Analyse
- Evaluierung von Fertigprodukten
- HW-Entwicklung
- SW-Entwicklung
- Benutzbarkeit und Ergonomie
- Systemerstellung
- Sicherheit, Sicherheit (AN)
- Logistikkonzeption
- Weiterentwicklung und Migration von Altsystemen



# V-Modell XT

## Vorgehensbausteine (3/3)

- Je nach Projekttyp gibt es **verpflichtende und optionale Vorgehensbausteine**
- **Beispielsweise** gibt es für ein Systementwicklungsprojekt (AG) die folgende Konstellation:

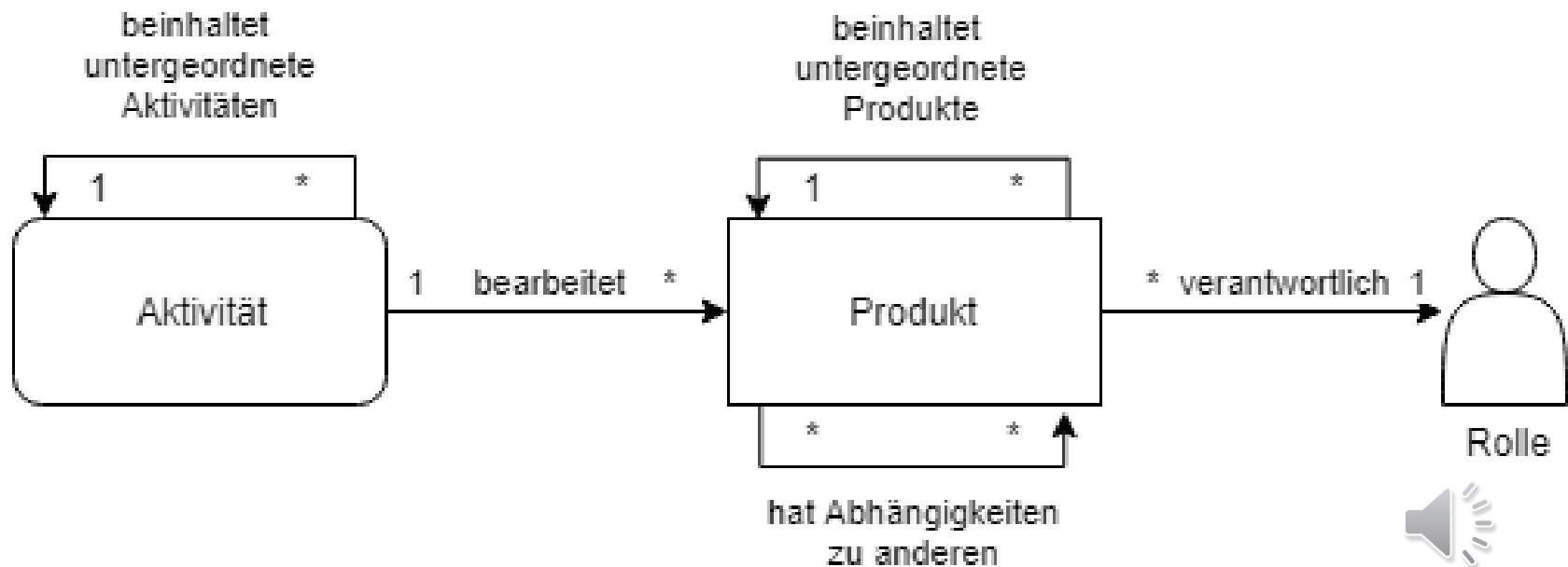
Verpflichtende Vorgehensbausteine	Optionale Vorgehensbausteine
<ul style="list-style-type: none"><li>• Projektmanagement</li><li>• Qualitätssicherung</li><li>• Konfigurationsmanagement</li><li>• Problem- und Änderungsmanagement</li><li>• Anforderungsfestlegung</li><li>• Vertragsschluss (AG)</li><li>• Lieferung und Abnahme (AG)</li></ul>	<ul style="list-style-type: none"><li>• Kaufmännisches Projektmanagement</li><li>• Messung und Analyse</li><li>• Systemsicherheit</li><li>• Evaluierung von Fertigprodukten</li><li>• Multi-Projektmanagement</li></ul>



# V-Modell XT

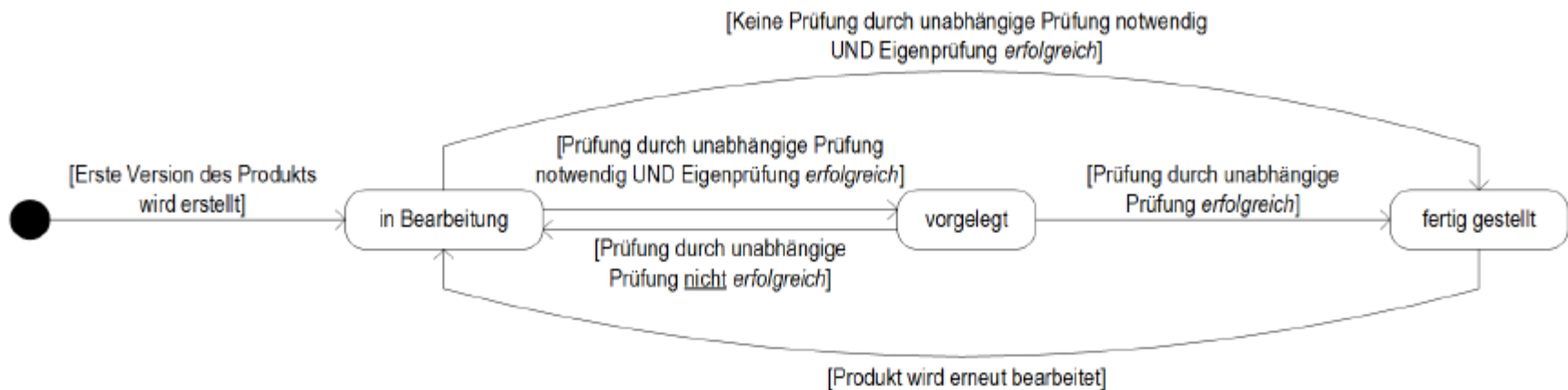
## Vorgehensbausteine und Entscheidungspunkte (1/3)

- Modulare, aufeinander aufbauende Vorgehensbausteine
- Vorgehensbaustein enthält alle Bestandteile zur Bearbeitung einer konkreten Aufgabenstellung
- Kapselt Aktivitäten, Produkte, Rollen



# V-Modell XT

## Produktzustandsmodell



## Vorgehensbausteine und Entscheidungspunkte (2/3)

- V-Modell XT besteht aus **21 modularen Vorgehensbausteinen**, die aufeinander aufbauen
- Ein Vorgehensbaustein enthält alle Bestandteile, die zur Bearbeitung einer konkreten Aufgabe notwendig sind
  - Produkt (zu erzeugendes Ergebnis), Aktivität, verantwortliche Rolle
- Jedes Produkt besitzt einen **Bearbeitungszustand**
  - in Bearbeitung
  - vorgelegt
  - fertig gestellt
- Jedes Produkt wird von genau einer Aktivität fertiggestellt, die in Teilaktivitäten gegliedert werden können
  - Teilaktivität ist eine Arbeitsanleitung, die in sich geschlossen und ohne Zeitverzug durchzuführen ist
- Am Ende der Aktivität muss das erzeugte Produkt **geprüft** werden



## Vorgehensbausteine und Entscheidungspunkte (3/3)

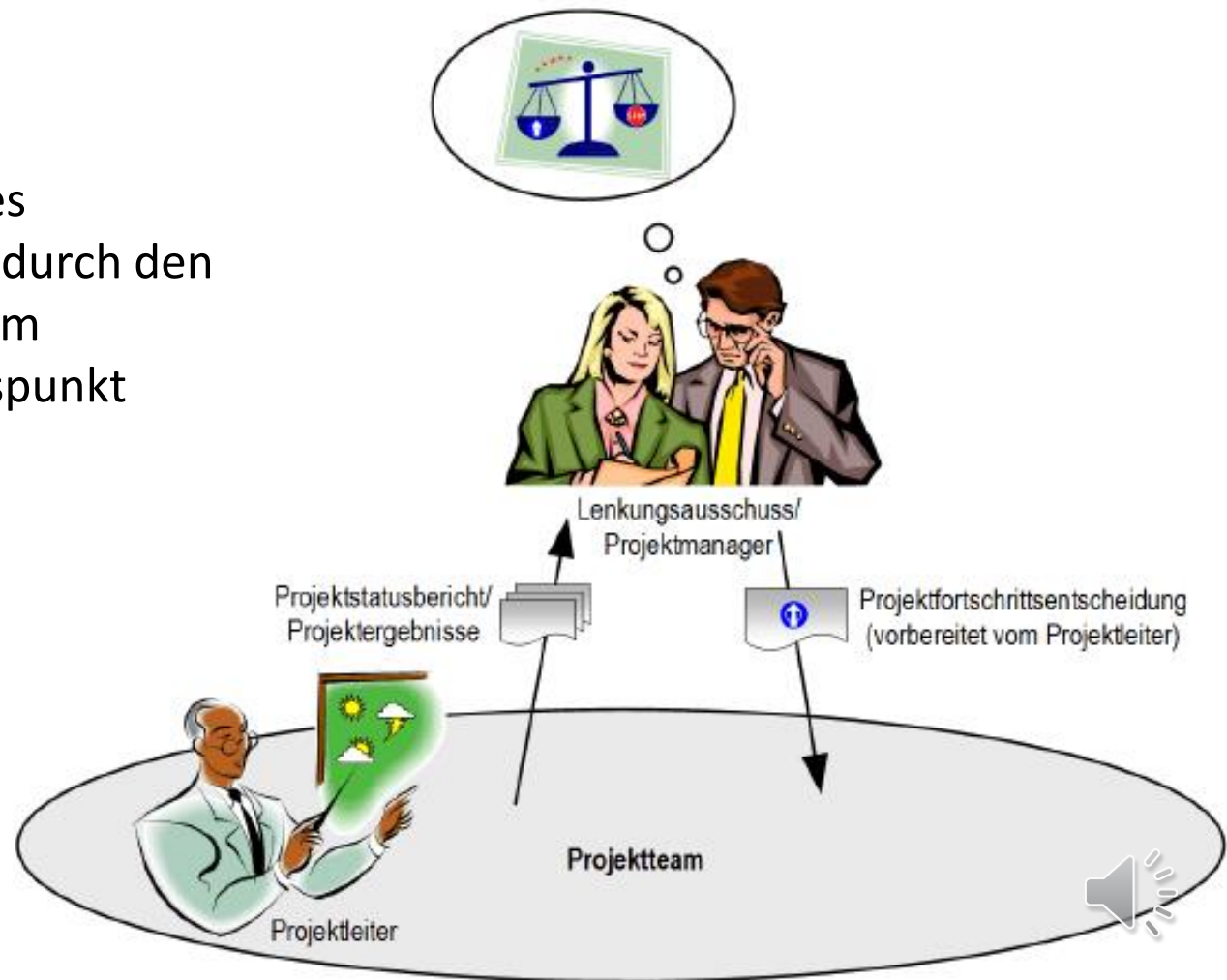
- Zu jeder **Projektdurchführungsstrategie** gibt es **Projektfortschrittsstufen**, die in einer festgelegten Reihenfolge erreicht werden müssen
- Das Erreichen einer Projektfortschrittsstufe wird durch einen sog. **Entscheidungspunkt** markiert
- Ein Entscheidungspunkt stellt somit einen **Meilenstein** im Projekt dar, an dem der aktuelle Stand des Projekts evaluiert wird



# V-Modell XT

## Entscheidungspunkte (1)

- Vorstellung des Projektstands durch den Projektleiter am Entscheidungspunkt



# V-Modell XT

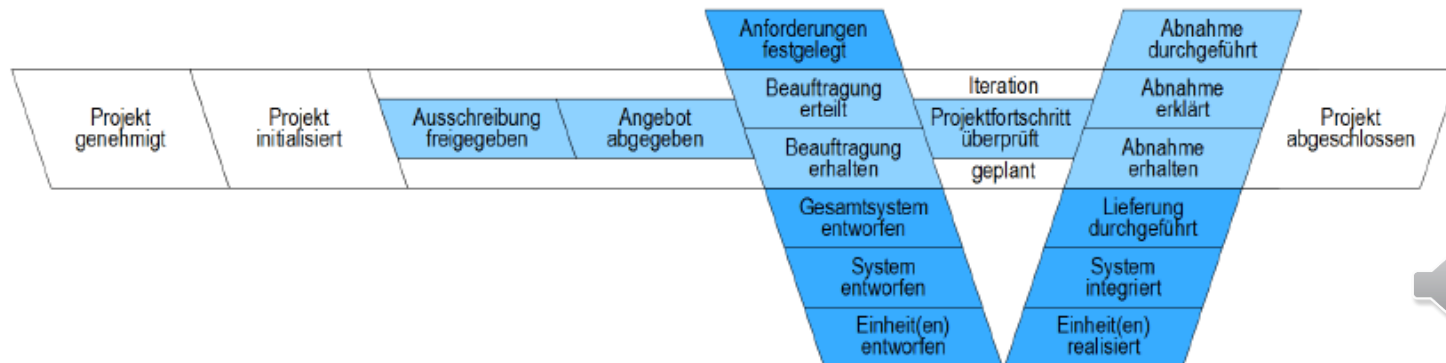
## Entscheidungspunkte (2)

- In allen Projekten gibt es folgende Entscheidungspunkte:
  - Projekt genehmigt, Projekt geplant, Iteration geplant, Projekt abgeschlossen
- Eine Projektdurchführungsstrategie legt mit den Entscheidungspunkten fest, wann welche Produkte fertiggestellt sein müssen

**Weiß:** Managementaspekte, sind in jeder Projektdurchführungsstrategie vorhanden

**Hellblau:** beschreiben Entscheidungspunkte an der Schnittstelle AG/AN

**Dunkelblau:** beziehen sich auf die eigentliche Systemerstellung

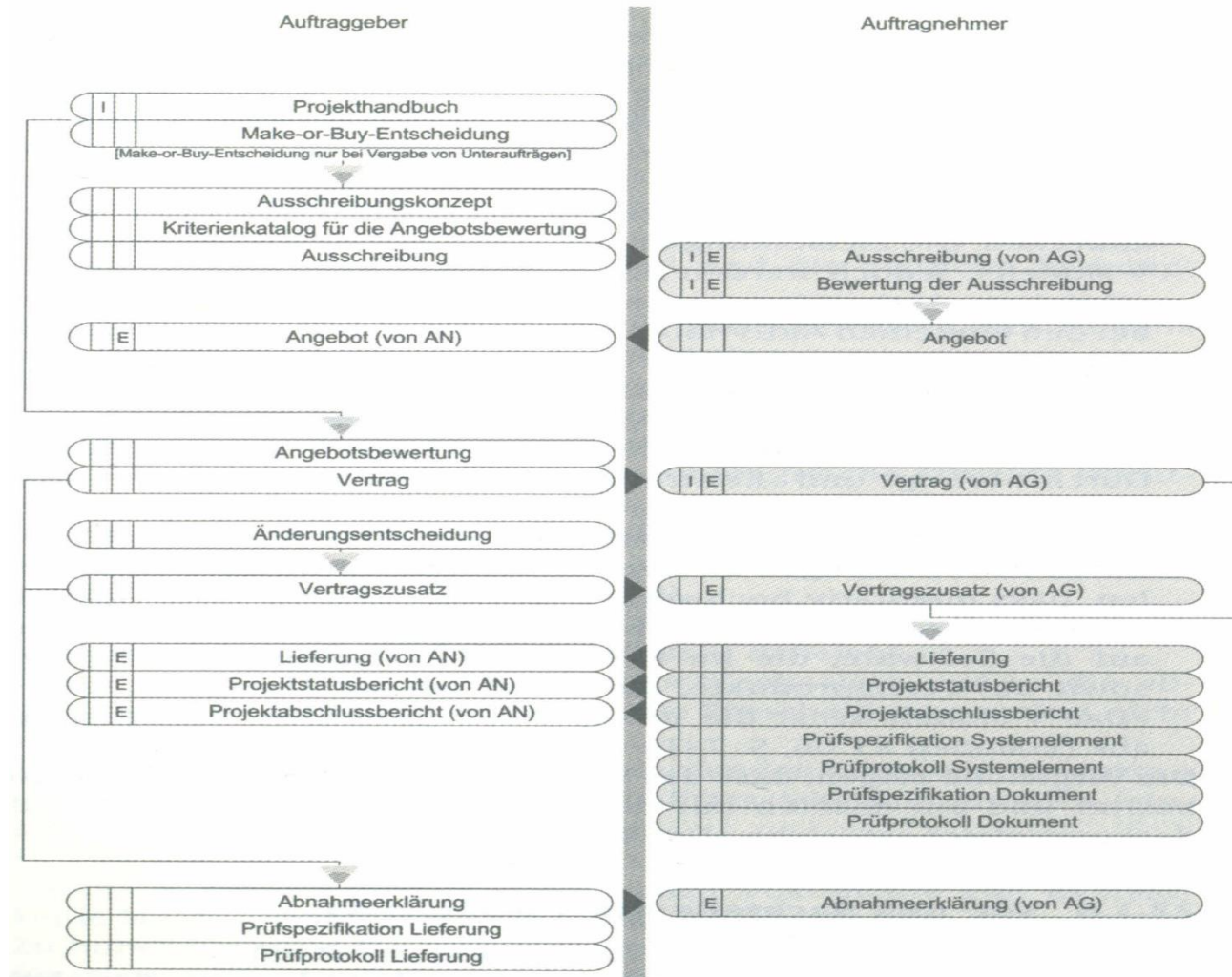


- Wird vom V-Modell XT unterstützt und **Werkzeug-überwacht**
  - Kostenfreie Lizenz unter „[www.v-modell.iabg.de](http://www.v-modell.iabg.de)“ downloadbar
- Ist die **projektspezifische Anpassung** erfolgt, dann liegt damit auch die **Projektdurchführungsstrategie** fest
- Jede Durchführungsstrategie enthält **Entscheidungspunkte**, die den Projektfortschritt kennzeichnen
- Der Projektleiter nimmt anhand der Projektdurchführungsstrategie eine projektspezifische Ausgestaltung der Entscheidungspunkte im Rahmen der Projektplanung vor
- Nach der Projektinitialisierung – also bereits während des laufenden Projekts – kann der Projektleiter Anpassungen vornehmen; sog. dynamisches Tailoring



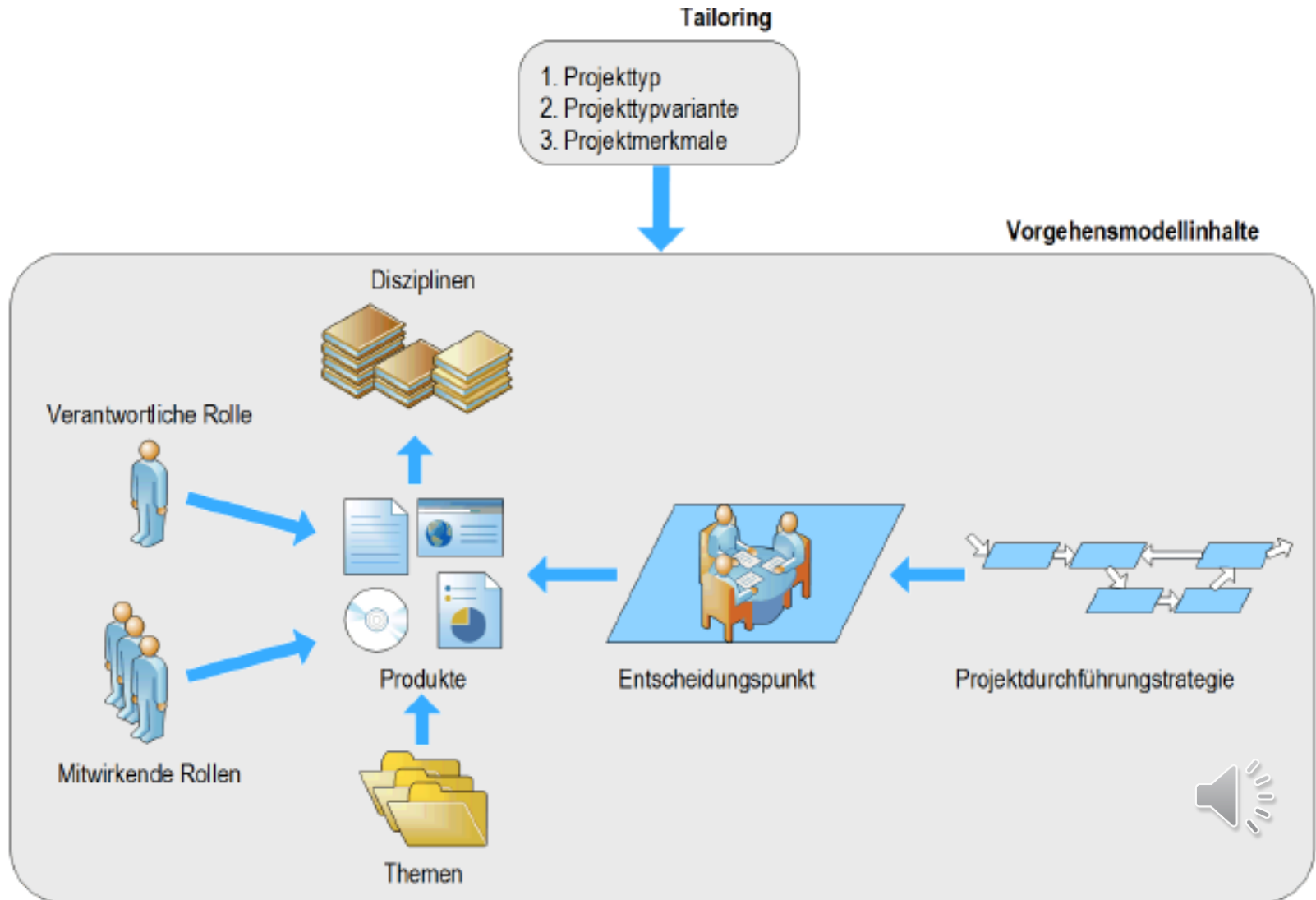
# V-Modell XT

## Schnittstelle Auftraggeber/Auftragnehmer



# V-Modell XT

## Zusammenfassung



# V-Modell XT

## Bewertung: Vorteile

- Nachvollziehbares und konsistentes Modell
- Umfangreich und durchaus anwendbar dokumentiert und beschrieben
- Vorgehensmodell das verschiedene:
  - Projektgegenstände,
  - Projektrollen,
  - Projekttypen und weitere Projektmerkmale unterscheidet
  - und dafür verschiedene Projektdurchführungsstrategien anbietet
- Trennung und Zusammenspiel zwischen Auftraggeber- und Auftragnehmersicht sind wohl definiert und gut durchdacht
- Mithilfe geeigneter Werkzeuge ist ein statisches und/oder dynamisches Tailoring des Modells und eine Projektplanung sehr gut durchführbar



# V-Modell XT

## Bewertung: Nachteile

- Hoher organisatorischer Aufwand zur Anwendung und Umsetzung
  - V-Modell XT ist für sehr große Projekte geeignet (ab ca. 500 PT)
- Extrem dokumentenzentriert: ohne die abgenommenen bzw. freigegebenen Dokumente können Entscheidungspunkte nicht verlassen werden
- Sehr hoher Projektmanagementaufwand alleine für die Umsetzung und Einhaltung des V-Modells, der dann nicht für das eigentliche Projekt zur Verfügung steht
- Das ist aber bei prinzipiell allen Modellen der Fall



# V-Modell XT

## Klassifizierung

- Prozess- und Qualitätsmodell
- Monumentales Modell mit Anpassbarkeit an verschiedene Projekttypen und Projektdurchführungsstrategien
- Modell mittlerer Granularität
- Systemorientiertes Modell mit Software- und/oder Hardwareanteilen
- Modell für die Entwicklung in Teams
- Gut geeignetes Vorgehensmodell
  - für die Erstellung von Individualsoftware in großen Projekten
  - für den Kauf von Softwareprodukten in großen Projekten



- Klassische Vorgehensmodelle
  - Wasserfallmodell
  - Nebenläufiges Modell
  - Spiralmodell
  - **Prototyping**
- Monumentale Vorgehensmodelle
  - V-Modell XT
  - **Rational Unified Process (RUP)**
- Agile Modelle
  - Scrum
  - Kanban (IT-Kanban)
  - Extreme Programming (XP)
  - Feature Driven Development
  - Adaptive Software Development
  - Crystal
  - Lean Software Development



# Rational Unified Process - Historie, Entstehung und Einsatzgebiete

- Stellt einen generischen Prozessrahmen für die objektorientierte Softwareentwicklung zur Verfügung
- Entwicklung ist eng verbunden mit der Entstehung der Unified Modeling Language (**UML**)
  - Booch, Jacobson und Rumbaugh haben UML entwickelt und standen ebenfalls Pate bei der Entwicklung des RUP
  - Alle drei haben bei der Firma Rational Software gearbeitet (seit 2003 Teil des IBM-Konzerns)
- RUP beschreibt im Prinzip, wie die UML optimal einzusetzen ist
- In den RUP sind die jahrelangen Entwicklungserfahrungen von Rational Software eingeflossen



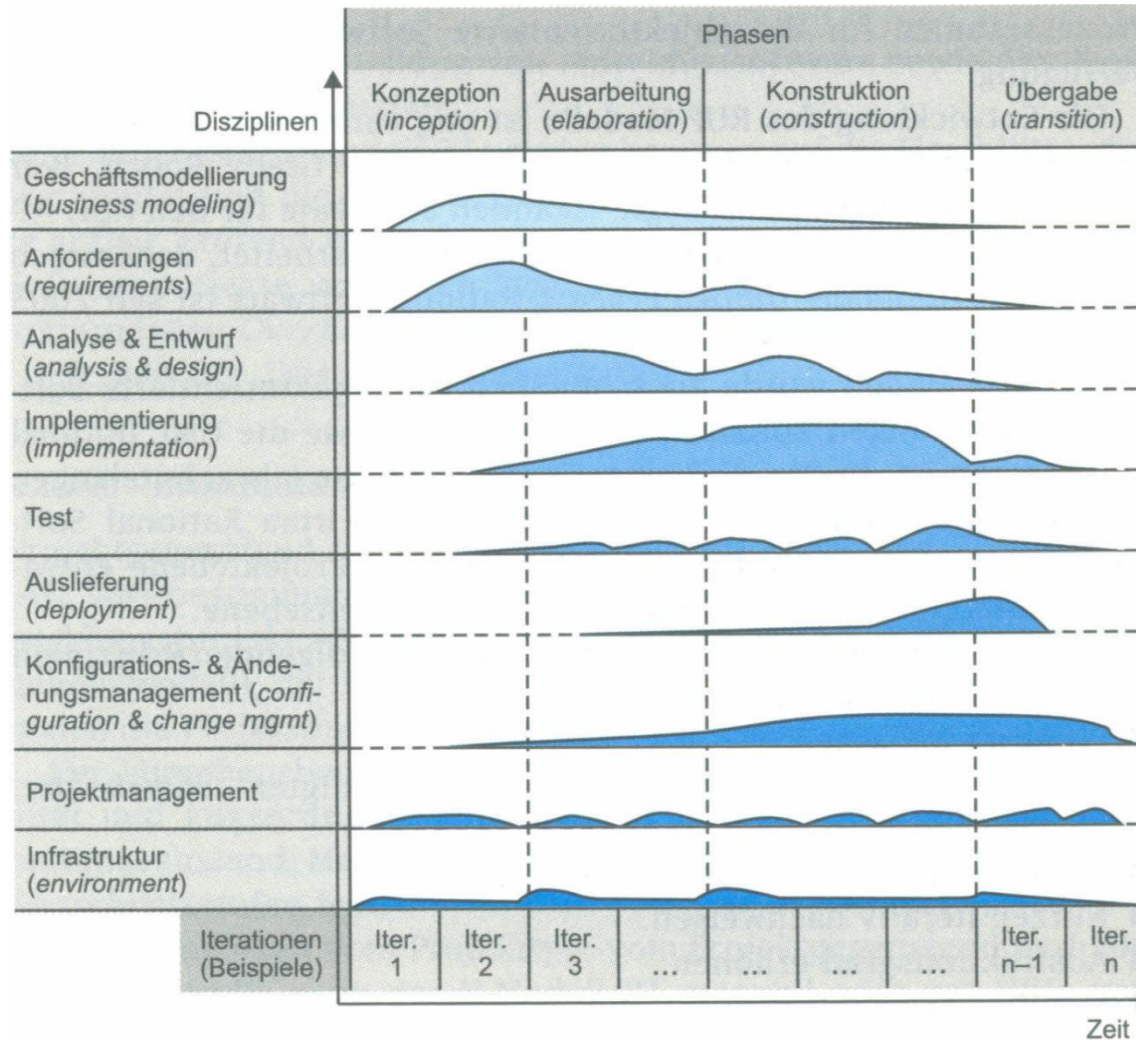
# Rational Unified Process - Grundprinzipien und Struktur (1)

- Das RUP-Modell orientiert sich an folgenden Prinzipien für eine geschäftsorientierte Entwicklung:
  - Prozesse anpassen
  - Konkurrierende Prioritäten von Projektbeteiligten (Stakeholder) ausgleichen
  - Teamübergreifend zusammenarbeiten
  - Nutzen iterativ nachweisen
  - Sukzessive Abstraktionsgrad erhöhen
  - Ständige Konzentration auf die Qualität
- RUP sieht vor, diese Prinzipien und Praktiken durch geeignete Werkzeuge und zusätzliche Dienstleistungen zu unterstützen



# Rational Unified Process

## Grundstruktur (3)



# Rational Unified Process - Grundprinzipien und Struktur (2)

- RUP ist ein Softwareprodukt der Firma IBM
  - Besteht aus verlinkter Wissensbasis mit Mustervorlagen und ausführlichen Beschreibungen für die verschiedenen Aktivitätstypen
- Das RUP-Prozessmodell besteht aus zwei Dimensionen:
  - Die horizontale Achse repräsentiert die Zeit und zeigt die Lebenszyklus-Aspekte des Prozesses
  - Es werden hierbei vier Phasen unterschieden, die jeweils mit einem Meilenstein abgeschlossen werden:
    1. Konzeption (engl. Inception)
    2. Ausarbeitung (engl. Elaboration)
    3. Konstruktion (engl. Construction)
    4. Übergabe (engl. Transition)



# Rational Unified Process - Grundprinzipien und Struktur (4)

- Alle vier Phasen der horizontalen Achse lassen sich – je nach Planungsstrategie – in mehrere Iterationen gliedern.
- Die vertikale Achse repräsentiert neun Disziplinen (engl. Disciplines), die Aktivitäten logisch gruppieren und phasenübergreifend sind.
- Dies verhält sich somit anders als beim Wasserfallmodell.
- Zu den neun Disziplinen zählen:
  1. Geschäftsmodellierung
  2. Anforderungen
  3. Analyse und Entwurf
  4. Implementierung
  5. Test
  6. Auslieferung
  7. Konfigurations- und Änderungsmanagement
  8. Projektmanagement
  9. Infrastruktur



# Rational Unified Process - Inhalt und Aufbau (1)

- Ein wichtiges Ziel des RUP ist es, in der Frühphase des Projekts (Konzeptions- und Ausführungsphase),
  - a) eine **funktionsfähige Architektur zu konzipieren** und
  - b) einen **funktionsfähigen Prototypen** zu realisieren.
- Ein weiteres Ziel ist die **frühe Minimierung der Projektrisiken**
- In jeder Phase können mehrere Iterationen durchlaufen werden
- Sind alle Phasen durchlaufen, dann ist ein Entwicklungszyklus abgeschlossen
- Das Ergebnis ist eine **Software-Generation**
- Neues Produkt durchläuft beim ersten Mal initialen Entwicklungszyklus
- Weiterentwicklungen werden **Evaluationszyklen** genannt; es entsteht die nächste Generation, wobei die Schwerpunkte anders verteilt werden



# Rational Unified Process - Vorteile des RUP

- gut geeignet für die objektorientierte und modellbasierte Neu- und Weiterentwicklung von Individualsoftware
- Gute Flexibilität und Anpassbarkeit
- Umfangreiche Erweiterungen verfügbar
- Klare Handlungsanweisungen
- Gute Schablonen, Vorlagen und Werkzeuge verfügbar
- Große internationale Verbreitung und Akzeptanz des RUP
- Deutsche Version des RUP ist verfügbar
- Weniger Dokumentation als im V-Modell XT notwendig



# Nachteile des RUP

- Sehr hoher Verwaltungs- und Einarbeitungsaufwand bei kleinen und mittleren Projekten
- => Standardmodell ist nur für sehr umfangreiche und große Softwareentwicklungsprojekte geeignet
- Um den RUP projektspezifisch anpassen zu können, ist umfangreiches und sehr detailliertes Wissen über den gesamten RUP erforderlich
- Erheblicher Verwaltungs-Overhead auch für die Dokumentation
- Keine Trennung von Auftraggeber- und Auftragnehmersicht



# Klassifizierung

- Monumentales Modell mit inkrementell-iterativer Vorgehensweise
- Prozess- und Qualitätsmodell; allerdings kein reines, denn:
  - Integriert die objektorientierte Entwicklung in der Analyse- und Entwurfsphase
  - RUP kann auch bei klassischer prozeduraler Softwareentwicklung eingesetzt werden
- Mittel- bis feingranulares Modell
- Modell unterstützt die OOA und OOD
- Modell für die Neuerstellung/Weiterentwicklung von Individualsoftware
- Auch für geografisch verteilte Entwicklungsteams geeignet
- Nimmt eher eine auftragnehmerorientierte Perspektive ein



# Herzlichen Dank für Ihre Aufmerksamkeit !

