

---

# MASTERARBEIT

---

Herr Dipl.Ing.(FH), Mag.  
Andreas Gebert

**Maschinen- & Anlagenbau, neue  
Formen des Projektmanagements  
und der Projektbearbeitung,  
Nachhaltigkeitsaspekte**

Mittweida, 2022



---

## MASTERARBEIT

---

# Maschinen- & Anlagenbau, neue Formen des Projektmanagements und der Projektbearbeitung, Nachhaltigkeitsaspekte

Autor:  
Herr Dipl.Ing.(FH), Mag.  
**Andreas Gebert**

Studiengang:  
**Nachhaltigkeit in gesamtwirtschaftlichen Kreisläufen**

Seminargruppe:  
**NH19w1-M**

Erstprüfer:  
Frau Dipl.-Wirt.-Ing. Heike Diebler

Zweitprüfer:  
Herr Dipl. Ing. Klemens Unger

Einreichung:  
Mittweida, 23. Mai. 2022

Verteidigung/Bewertung:  
Mittweida, 2022





Institute for Knowledge Transfer and  
Digital Transformation

---

## MASTER THESIS

---

**Machinery and plant engineering;  
Introduction of agile methodologies  
into project management and  
order execution;  
Impact on sustainability aspects**

author:

Mr. Dipl.Ing.(FH), Mag.  
**Andreas Gebert**

course of studies:

**Sustainability in macroeconomic cycles**

seminar group:

**NH19w1-M**

first examiner:

**Ms. Dipl.-Wirt.-Ing. Heike Diebler**

second examiner:

**Mr. Dipl. Ing. Klemens Unger**

submission:

**Mittweida, 23<sup>rd</sup> May.2022**

defence/ evaluation:

**Mittweida, 2022**



## **Bibliografische Beschreibung:**

Gebert, Andreas:

Maschinen- & Anlagenbau, neue Formen des Projektmanagements und der Projektbearbeitung, Nachhaltigkeitsaspekte. - 2022. - I-VIII, 1-60, I-XXII S.

Mittweida, Hochschule Mittweida, IWD - Institut für Wissenstransfer und Digitale Transformation, Masterarbeit, 2022

## **Referat:**

An den kombinierten Maschinen- und Anlagenbau werden neue Anforderungen gestellt; Diese Anforderungen werden durch die Globalisierung der kompletten Wertschöpfungskette, allgemeine Steigerung des Komplexitätsgrades, durchgehenden Einsatz digitaler Werkzeuge und eine verbesserte Informationsbasis verursacht. Maschinen- und Anlagenbauer reagieren darauf mit dem Einsatz agiler Projektmanagement-Methoden insbesondere im Dienstleistungsbereich.



## Inhaltsverzeichnis

Inhaltsverzeichnis.....	I
Abbildungsverzeichnis.....	IV
Tabellenverzeichnis.....	VI
Abkürzungsverzeichnis.....	VII
1 Einleitung und Aufbau.....	1
1.1 Definitionen.....	1
1.1.1 Maschinenbau.....	1
1.1.2 Anlagenbau.....	1
1.1.3 Einordnung Maschinenbau/Anlagenbau entsprechend der Fertigungsprinzipien.....	2
1.1.4 Der gemischte Maschinen- und Anlagenbau.....	2
1.2 Aufbau der Arbeit.....	4
2 Aufgabenstellung (Problem Statement).....	5
3 Das klassische Modell, sequenzielle Vorgehensmodelle.....	7
3.1 Das Kaskadenmodell oder Wasserfallmethode.....	9
3.2 Die Grenzen des Wasserfall-Modells (Waterfall model).....	10
4 Das gewandelte Modell und neue Anforderungen.....	11
4.1 Anstieg des Komplexitätsgrades allgemein.....	11
4.1.1 Messung des Komplexitätsgrades – der Darnall-Preston Komplexitätsindex (DPCI).....	13
4.2 Spezielle Treiber des Komplexitätsgrades.....	13
4.2.1 3-dimensionales Projektmanagement.....	14
4.2.2 Neue Methoden des Supply Chain Managements.....	15
4.2.2.1 Traditionelle Beschaffungskonzepte.....	16
4.2.2.2 Modulare Beschaffung.....	16
4.2.2.3 In-Plants.....	16
4.2.2.4 Entmaterialisiertes Unternehmen.....	17
4.2.3 Die Verkomplizierung der physischen Transportbeziehungen.....	18
4.2.4 Das Koordinierungsproblem paralleler Aktivitäten bei gleichzeitiger geographischer Dislozierung der Akteure.....	19
5 Lösungsansätze.....	21
5.1 Voraussetzungen.....	21
5.1.1 Ordnungs- und Klassifizierungssystem.....	21

## Inhaltsverzeichnis

5.1.2	Cooperation tools.....	21
5.1.3	Versionierungstools.....	22
5.1.4	CAD Systeme, die verteiltes Arbeiten erlauben.....	22
5.1.5	PPC/PPS (Produktionsplanung und Steuerung).....	23
5.2	Prozessorientierung.....	24
5.2.1	Swimlanes bei Maschinenbau-Komponenten.....	25
5.2.2	Swimlanes im Anlagenbau-Projektteil.....	26
5.2.3	Swimlanes von Sonderbauteilen im kritischen Pfad.....	27
5.3	Projektorganisation.....	27
5.3.1	Verdichtung und Colocation.....	29
5.3.2	Co-Manager in komplexen Projektteams.....	29
5.3.3	Die Integration der Kunden-Organisation.....	30
5.4	Agile Managementmethoden.....	31
5.4.1	Die APM Grundlagen.....	33
5.4.1.1	<i>Das agile Manifesto</i> .....	34
5.4.1.2	Die Erklärung der Wechselbeziehungen (DOI – Declaration of Interdependence).....	34
5.4.1.3	Die 12 Prinzipien von APM.....	35
5.4.2	Motivation agile Vorgehensmodelle im Maschinen- und Anlagenbau einzusetzen.....	35
5.4.2.1	Enge Kooperation erwünscht.....	35
5.4.2.2	Zeitersparnis durch früheres Abliefern von Arbeitsergebnissen.....	36
5.4.2.3	Zeitersparnis durch <i>„Ineinanderschieben“ sequenzieller Arbeiten</i> .....	37
5.4.2.4	<i>das Arbeitsergebnis ist nicht linear vorausplanbar</i> .....	38
5.4.2.4.1	Eine intuitive Sichtweise der Projektlandschaft.....	40
5.4.3	Übersicht und Entwicklungsgeschichte agiler Vorgehensmodelle.....	41
5.4.4	Scrum (Deutsch Gedränge).....	43
5.4.5	Die Integration von kaskadierten und agilen Methoden führt zum hybriden Projektmanagement.....	44
5.4.5.1	Die Zusammenführung der Swimlanes.....	44
5.4.5.2	Die grundsätzlichen Möglichkeiten entsprechend des PMLC (Project Management Life Cycle Model).....	45
5.4.5.2.1	Das lineare PMLC Modell.....	46
5.4.5.2.2	Das inkrementelle PMLC Modell.....	46
5.4.5.2.3	Das iterative PMLC Modell.....	46
5.4.5.2.4	Das adaptive PMLC Modell.....	46
5.4.5.2.5	Das extreme PMLC Modell.....	46
5.5	Koordinierungsmethoden.....	47
6	Nachhaltigkeitsaspekte.....	49

## Inhaltsverzeichnis

6.1 Ökonomische Nachhaltigkeit.....	49
6.2 Soziale Nachhaltigkeit.....	49
Index.....	53
Literaturverzeichnis.....	57
Anlagen.....	I
Anlage 1: Der Scrum Guide von Ken Schwaber & Jeff Sutherland.....	III
Anlage 2: Das ECPM Rahmenkonzept – Ablaufdiagramm von Robert K. Wysocki.....	XXXI

## Abbildungsverzeichnis

Illustration 1: Einordnung Maschinenbau/Anlagenbau entsprechend Fertigungsprinzipien nach J.C. Wortmann.....	2
Illustration 2: Darstellung der Arbeit mittels Ishikawa Diagramms.....	4
Illustration 3: Projektstrukturplan entsprechend Jörg Walter - Projektmanagement im Maschinenbau.....	7
Illustration 4: Primavera P6 - Projekt-Ablaufplan als Gant-Chart.....	8
Illustration 5: Arbeitsabschnitte bei der Erstellung von umfangreichen Computerprogrammen.....	9
Illustration 6: Schritte bei der Softwareentwicklung nach Royce.....	10
Illustration 7: klassische Projektorganisation im gemischten Maschinen- und Anlagenbau.....	14
Illustration 8: dreidimensionale Projektorganisation.....	14
Illustration 9: Projektleitung und dreidimensionale Projektmatrix.....	15
Illustration 10: Strategien des Supply Chain Managements nach Zentes et al.....	16
Illustration 11: Grad der Virtualität nach Eßig.....	17
Illustration 12: Warenströme in klassischen Lieferbeziehungen.....	18
Illustration 13: Warenströme bei komplexen Lieferbeziehungen.....	18
Illustration 14: Prinzip der Parallelisierung, Stärk in Anlehnung an B. Prasad.....	19
Illustration 15: Projektstrukturplan und Parallelaktivitäten.....	19
Illustration 16: Prinzipanalyse der Prozesse im kombinierten Maschinen- und Anlagenbau.....	24
Illustration 17: Swimlanes nach Wysocki (Vorrangdiagramm).....	25
Illustration 18: Maschinenbau Swimlane.....	25
Illustration 19: Anlagenbau Swimlane.....	26
Illustration 20: Sonderbauteile Swimlane.....	27
Illustration 21: beispielhafte Projektorganisation nach Domendos - Projectmanagement.guide.....	28
Illustration 22: zeitgemäße Projektorganisation.....	28
Illustration 23: komplexes Co-Manager Projektteam nach Robert K. Wysocki.....	30
Illustration 24: komplexes Projektteam mit integriertem Kundenteam.....	31
Illustration 25: Iteratives Project Management Life Cycle Modell.....	32
Illustration 26: APM - Entwicklungsgeschichte.....	32



## Abbildungsverzeichnis

Illustration 27: potentielle Zeitersparnis bei einzelnen Aktivitäten unter Anwendung agiler Methoden.....	36
Illustration 28: potentielle Zeitersparnis bei hintereinander folgenden Aktivitäten.....	37
Illustration 29: potentielle Zeitersparnis durch Ineinanderschieben sequenzieller Aktivitäten.....	38
Illustration 30: Die vier Quadranten der Projektlandschaft.....	40
Illustration 31: Auswirkungen von Komplexitätsanstieg und Unsicherheit.....	40
Illustration 32: Zeitlicher Ablauf der Entwicklung agiler Vorgehenstechniken.....	41
Illustration 33: Die Zusammenführung der wesentlichen Ablauftypen.....	44
Illustration 34: Das PMLC Modell wird von der Position in der Projektlandschaft bestimmt.....	45
Illustration 35: Modell zur Einschätzung der Auswirkungen von agilen Projektmanagementmethoden auf die soziale Nachhaltigkeit nach Francesco Albarosa und Rafael Valenzuela Musura.....	51

## Tabellenverzeichnis

Table 1: Übersicht der wichtigsten agilen Vorgehenstechniken (Herausgeber der ursprünglichen Literatur in Klammern).....	42
--	----

## Abkürzungsverzeichnis

3D CAD	3D-Computergrafik und Konstruktion
APM	Agile Project Management
BOM	Bill Of Material, Material-, Bauteileliste
CAD	Computer Aided Design
CIMOSA	Computer Integrated Manufacturing Open System Architecture
CODP	Customer Order Deployment Point
CPM	Complex Project Management
DOI	Declaration of Interdependence (Die Erklärung der Wechselbeziehungen)
DPCI	Darnall Preston Complexity Index
DSDM	Dynamic Systems Development Method
ECPM	Effective Complex Project Management
EPC	Engineering, Procurement, Construction
ERM	Entity Relationship Model
PERA	Purdue Enterprise Reference Architecture, Purdue – Unternehmensarchitektur
PERT	Program Evaluation and Review Technique
PID	Process & Instrumentation Diagram, Verfahrensfließbild
PMLC	Project Management Life Cycle, Projektmanagement – Lebenszykluskonzept
PPC	Production Planning and Control, Produktionsplanung und -steuerung
PPS	Produktionsplanung und Steuerung, Production Planning System
TPM	Traditional Project Management, Wasserfall Methode
VPN	Virtual Private Network

## Abkürzungsverzeichnis

<diese Seite bleibt aus redaktionellen Gründen frei>

# 1 Einleitung und Aufbau

Im ersten Kapitel wird die Motivation und Begriffsbestimmungen dieser Masterarbeit dargestellt. Darüber hinaus erfolgt ein kurzer Überblick zum Aufbau der Arbeit.

Das Projektmanagementmodell zur Bewältigung gemischter Maschinen- und Anlagenbauvorhaben hat sich die letzten Jahre verändert. Der wesentliche Treiber für diese Veränderungen erscheint der Komplexitätszuwachs bei der Abarbeitung entsprechender Projekte. Dazu zuallererst Definitionen um das Phänomen Maschinen- und Anlagenbau einordnen zu können:

## 1.1 Definitionen

### 1.1.1 Maschinenbau

Im Maschinenbau werden Produkte entwickelt/konstruiert<sup>1</sup>, die dann in Produktion/Fertigung gebracht werden. Maschinenbauprojekte sind klassische Produktentwicklungsprojekte.

Das Entscheidende daran ist, dass für die Fertigung konstruiert wird (im Gegensatz zum Anlagenbau, dort wird für die Montage vor Ort konstruiert). Eine Besonderheit der von mir beschriebenen Aktivitäten ist, dass praktisch für jedes Projekt Unikate konstruiert werden, auch wenn im Projekt selbst Maschinen (die Hauptaggregate) multipliziert (oder gespiegelt) als multiple Betriebsgrößenvariation<sup>2</sup> vorkommen. Die Hauptaggregate setzen sich wiederum aus Teil- oder Submaschinen zusammen, die von Projekt zu Projekt in Anzahl, Zusammensetzung und Typ variiert werden, des Weiteren werden diese Teil- oder Submaschinen auch projektspezifischen Modifikationen unterzogen.

Beispiele für derartige Maschinen sind Stranggießmaschinen, Walzwerke, Papiermaschinen.

### 1.1.2 Anlagenbau

Im Gegensatz zum Maschinenbau kombiniert man im Anlagenbau technische Komponenten<sup>3</sup> und/oder eben Maschinen zu einem Gesamtsystem; es geht daher um das Konzipieren von Prozessen und Abläufen und dabei um die Kombination anderweitig konstruierter Komponenten (Standardkomponenten wie z.B. Pumpen, gesondert konstruierte Maschinen).

Dabei konstruiert der Anlagenbau für die Montage, das Ergebnis der Planungsarbeiten wird erst am Einbauort sichtbar. Das bedeutet dass auch Fehler in Auslegung und Konstruktion

---

1 PMMB071: Wie unterscheiden sich eigentlich Projekte im Maschinenbau vom Anlagenbau, Seite 1, <https://bibliothek.projektmanagement-maschinenbau.de/portfolio-item/transkription-pmmb071-wie-unterscheiden-sich-eigentlich-projekte-im-maschinenbau-vom-anlagenbau/>, abgerufen am 12. Juni 2021, Pforzheim [1]

2 multiple Betriebsgrößenvariation, <https://wirtschaftslexikon.gabler.de/definition/multiple-betriebsgroessenvariation-38975/version-262395>, Prof. Dr. Marion Steven, Wiesbaden 2018, [2]

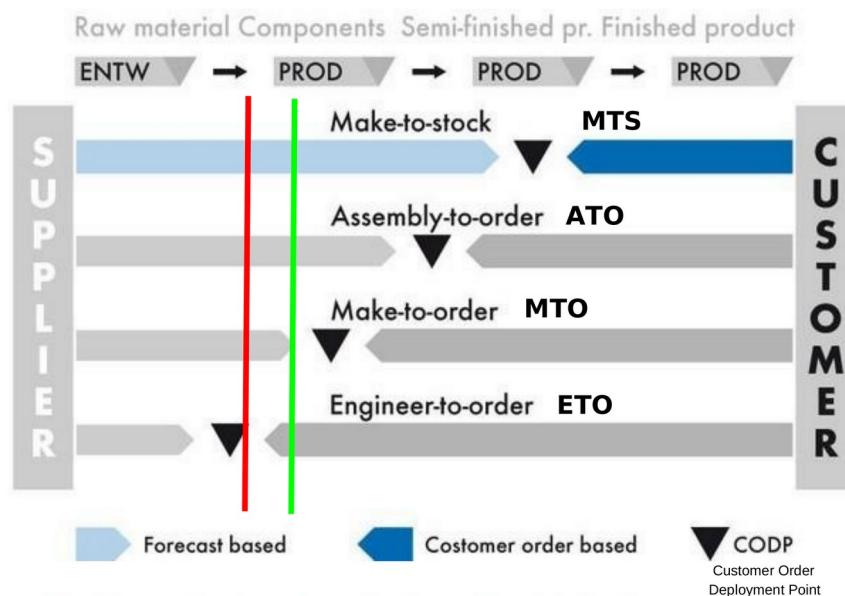
3 PMMB071: Wie unterscheiden sich eigentlich Projekte im Maschinenbau vom Anlagenbau, Seite 2, <https://bibliothek.projektmanagement-maschinenbau.de/portfolio-item/transkription-pmmb071-wie-unterscheiden-sich-eigentlich-projekte-im-maschinenbau-vom-anlagenbau/>, abgerufen am 12. Juni 2021, Pforzheim [1]

erst am Einbauort zu Tage treten. Das klassische Beispiel dafür sind Anlagen der Petrochemie.

### 1.1.3 Einordnung Maschinenbau/Anlagenbau entsprechend der Fertigungsprinzipien

Entsprechend der Klassifizierung der (Push/Pull) Fertigungsprinzipien nach Wortmann<sup>4</sup> lässt sich Maschinenbau in Relation zu Anlagenbau entsprechend „Customer Order Deployment Points“ (CODP) in der Wertschöpfungskette einordnen.

Maschinenbau / Anlagenbau entsprechend CODP:



**Machinery Engineering: Designed for fabrication**

**Plant Engineering: Designed for installation**

Illustration 1: Einordnung Maschinenbau/Anlagenbau entsprechend Fertigungsprinzipien nach J.C. Wortmann

Die Planungs- und Konstruktionsaktivitäten sind demnach beim Maschinenbau typischerweise weiter fortgeschritten als beim Anlagenbau obwohl auch im Maschinenbau projektbezogen praktisch Unikate konstruiert werden. Allerdings ist der Anteil der projektspezifischen Änderungen oder Neukonstruktion gemessen am gesamten Konstruktionsvolumen signifikant niedriger als beim Anlagenbau.

### 1.1.4 Der gemischte Maschinen- und Anlagenbau

Vereint Maschinen- und Anlagenbau jeweils im Sinne der vorangegangenen Beschreibungen: Es werden Maschinen konstruiert und entsprechend zur Fertigung gebracht, parallel dazu werden diese Maschinen in eine Anlagenstruktur eingebettet, die erst während der Montage in allen Ausprägungen zu Tage tritt. Während der Planungsphase treten alle erdenklich möglichen Koordinierungsprobleme im Maschinenbau selbst, zwischen Maschinenbau und Anlagenbau und zwischen den jeweils agierenden Disziplinen Prozessplanung,

<sup>4</sup> Nature of Master Production Scheduling, Kapitel: "A Classification Scheme for Master Production Scheduling" von J.C. Wortmann, Seite 103 ff, Efficiency of Manufacturing Systems, von C. Berg, D. French and B. Wilson, New York, 1983 [3]

## 1 Einleitung und Aufbau

Verrohrungsplanung, Maschinenkonstruktion und den entsprechenden Hilfsdisziplinen wie z.B. Hydraulik, Pneumatik, Elektrik auf.

Die vorliegende Arbeit bezieht sich ausschließlich auf gemischten Maschinen- und Anlagenbau.

<der Rest der Seite bleibt aus redaktionellen Gründen frei>

## 1.2 Aufbau der Arbeit

Nachfolgend der Versuch die logischen Zusammenhänge der Arbeit mittels eines auf den Kopf gestellten Ishikawa Diagramms darzustellen. Grau ist die Beschreibung des Istzustandes gekennzeichnet, grün sind die Lösungsmöglichkeiten markiert.

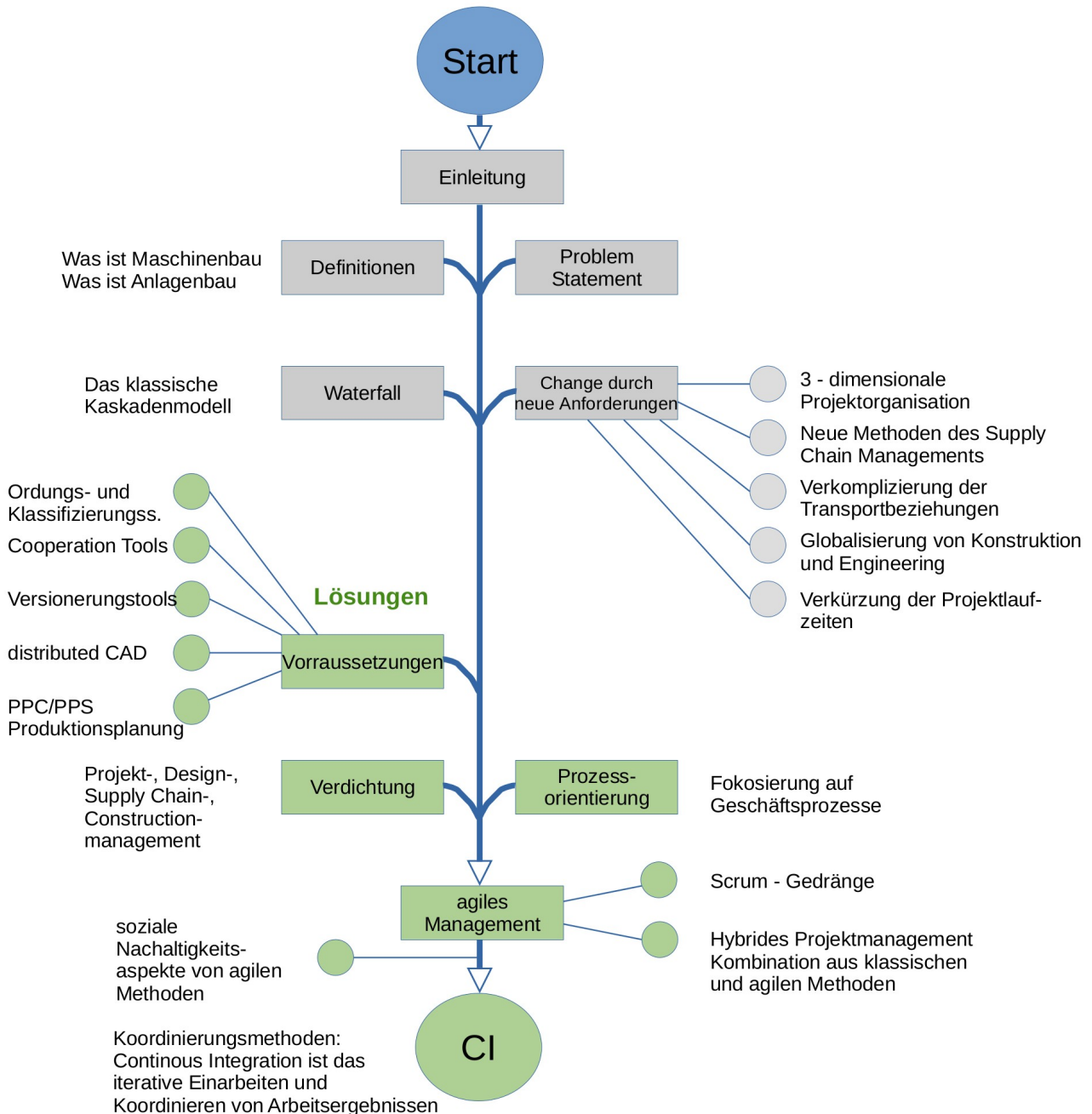


Illustration 2: Darstellung der Arbeit mittels Ishikawa Diagramms



## 2 Aufgabenstellung (Problem Statement)

Als Treiber für die Wandlung vom klassischen Projektmanagement, das eher in der Bauwirtschaft seine Wurzeln hatte, zu moderneren Managementformen, erweisen sich:

- Die Globalisierung der kompletten Wertschöpfungskette inklusive der Dienstleistungsbereiche (Engineering, Konstruktion, Abwicklung). Der Maschinen- und Anlagenbau war immer schon global aufgestellt, allerdings werden die letzten 30 Jahre die international arbeitsteilig erstellten Dienstleistungen anteilig am Gesamtgeschäft immer wichtiger. Dies führt:
  - a) zur örtlichen Aufsplittung, die beteiligten Techniker sind auf der ganzen Welt verteilt und
  - b) zur organisatorischen Zersplitterung; Neben der funktionalen Organisationsstruktur kommt eine regionale dazu.
- Ein allgemeiner Trend zur Steigerung des Komplexitätsgrades, der zum einen durch die fortschreitende technische Entwicklung und der Verbesserung der Informationsverarbeitung und -weitergabe in Geschwindigkeit und Qualität entsteht, zum anderen durch permanenten Druck auf die Lieferzeiten verursacht wird.
- Fortschreitender Einsatz digitaler Werkzeuge, die die Globalisierung des Dienstleistungsbereiches zum „Stand der Technik“ erheben.
- Eine verbesserte Informationsbasis, die praktisch in Echtzeit allen Marktteilnehmern zur Verfügung steht und durch ihre Verfügbarkeit erst neue Marktteilnehmer schafft. Dieser verbesserten Informationsbasis ist auch eine höhere Geschwindigkeit der Informationsverarbeitung und -verbreitung eigen, die verschiedene Geschäftsbeziehungen erst sinnvoll/möglich macht.

Es wird zuerst versucht darzustellen wie, fortschreitende Digitalisierung und Globalisierung bestehende Strukturen im gemischten Maschinen- und Anlagenbau verändern können und folgend mit einer Analogie aus der Softwareindustrie versucht, mögliche Auswirkungen auf die Nachhaltigkeit und dabei im Wesentlichen zum sozialen Aspekt, abzuschätzen.

<der Rest der Seite bleibt aus redaktionellen Gründen frei>

## 2 Aufgabenstellung (Problem Statement)

<diese Seite bleibt aus redaktionellen Gründen frei>

### 3 Das klassische Modell, sequenzielle Vorgehensmodelle

Als Ausgangspunkt für die Strukturierung von Projekten kann die Einteilung in Projektstufen betrachtet werden. Beispielhaft sei hier die Einteilung im Anlagenbau<sup>5</sup> genannt:

- Projektstufe 1: Projektvorbereitung
- Projektstufe 2: Basic Engineering
- Projektstufe 3: Ausschreibung und Vergabe
- Projektstufe 4: Detailed Engineering
- Projektstufe 5: Ausführung
- Projektstufe 6: Projektabschluss

Folgend werden die in den Projektstufen enthaltenen Aktivitäten im Projektstrukturplan aufgelöst.

Beispiel für Projektstrukturplan<sup>6</sup>

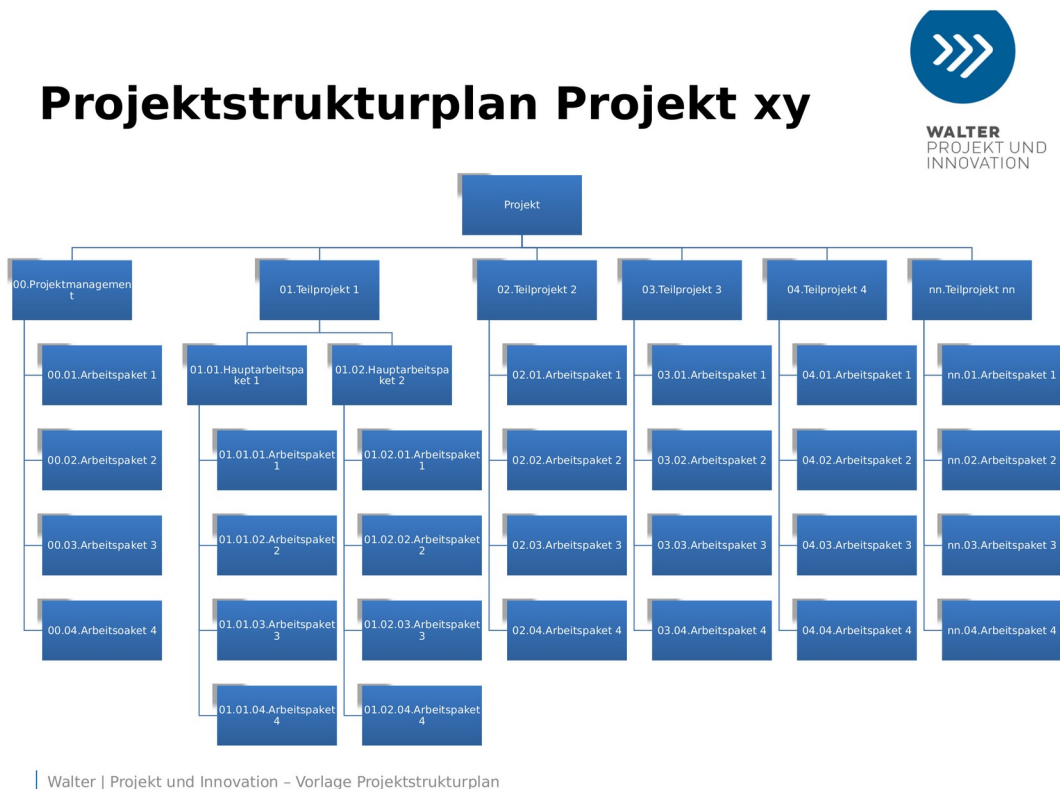


Illustration 3: Projektstrukturplan entsprechend Jörg Walter - Projektmanagement im Maschinenbau

Nach eventueller Detaillierung um eine zusätzliche Ebene (wird oft Vorgangs-/Arbeitspaketebene genannt), der Abschätzung des jeweiligen Zeitbedarfs und der logischen Verknüpfung der Arbeitspakete lässt sich daraus der Projekt-Ablaufplan entwickeln.

5 1.4 Projektstufen, Projektmanagement im Anlagenbau, S 5ff, Alexander Malkwitz, Norbert Mittelstaedt, Jens Bierwisch, Johann Ehlers, Thies Helbig, Ralf Steding, Berlin, 2016[4]

6 Projektstrukturplan, <https://bibliothek.projektmanagement-maschinenbau.de/meine-bibliothek/>, abgerufen am 12. Juni 2021, Pforzheim [5]

### 3 Das klassische Modell, sequenzielle Vorgehensmodelle

#### Beispiel Projekt-Ablaufplan als Gant-Chart<sup>7</sup>

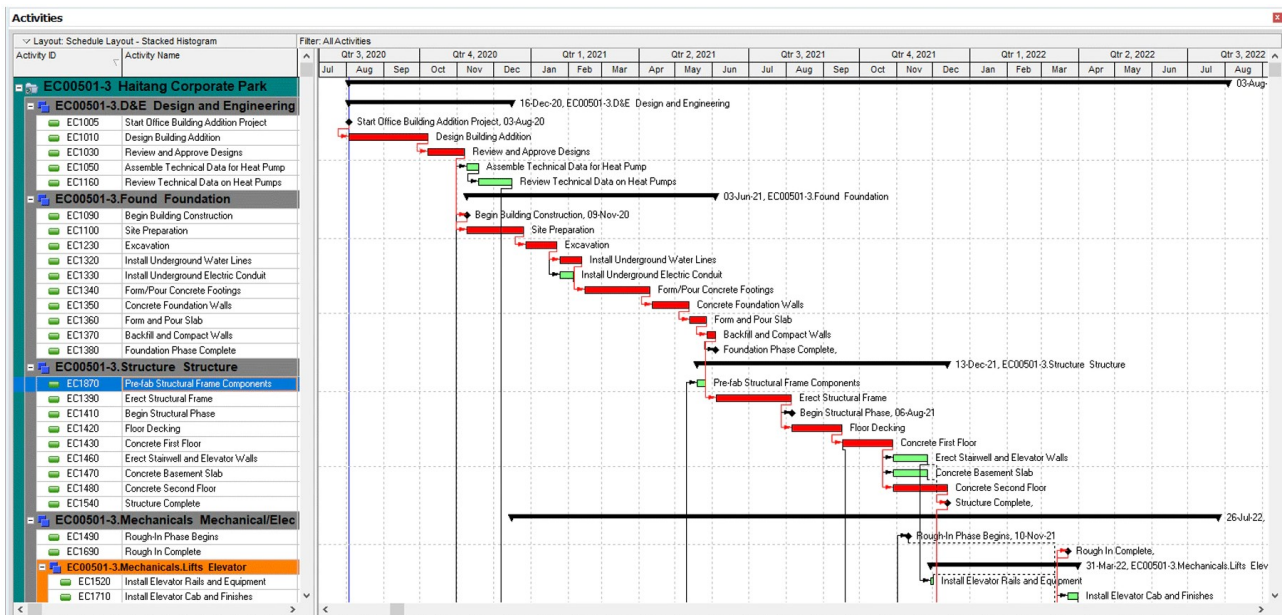


Illustration 4: Primavera P6 - Projekt-Ablaufplan als Gant-Chart

Üblicherweise wird die Ablaufplanung um die Ressourcenplanung (Personal, Material, u.a.) erweitert.

Seine Entstehung hat diese Art des „Kern-Projektmanagements“ als sequenzielle Abarbeitung von technischen Großvorhaben im Bauwesen (technische Großvorhaben der Vergangenheit waren zu einem großen Anteil Bauvorhaben, z.B. Eisenbahnbau), einen Meilenstein erlebte dieses Vorgehensmodell mit der Einführung der PERT (Program Evaluation and Review Technique)<sup>8</sup> Programmtechnik durch das US Navy Special Projects Office im Jahre 1957, aber auch Mondlandungen ließen sich erfolgreich damit managen<sup>9</sup>.

Persönlich wurde ich mit diesem Vorgehensmodell Anfang der 80er Jahre im Maschinen- und Anlagenbau (Voest-Alpine Industriebau) konfrontiert, als Vorbild galten US-amerikanische Bau- und Anlagenbauunternehmen (z.B. Bechtel).

Ein interessanter Aspekt ist, dass die Beschreibung als „sequenzielles Vorgehensmodell“ – allerdings zuerst ohne die Bezeichnung selbst – als solche aus der Softwareindustrie stammt.

<sup>7</sup> Primavera P6 CPM – Zeitplanung, Oracle Primavera P6,

<https://www.oracle.com/at/industries/construction-engineering/primavera-p6/>, abgerufen am 11.4.2022, 1995, 2022 urheberrechtlich geschützt für Oracle Corporation, Redwood Shores, CA 94065, USA [6]

<sup>8</sup> Application of a Technique for Research and Development Program Evaluation, D.G. Macolm, J.H. Roseboom, C.E. Clark, W. Frazer, Booz, Allen and Hamilton, Special Projects Office, US Navy, Copyright of Operations Research by INFORMS, Chicago - Illinois, Washington DC, USA, 1959[7]

<sup>9</sup> Negative Slack in Critical paths, Chapter 15 - Putting It All Together: LC-39 Site Activation, Moonport: A History of Apollo Launch Facilities and Operations, By Charles D. Benson and William Barnaby Faherty, NASA Special Publication-4204 in the NASA History Series, 1978,

<https://www.hq.nasa.gov/office/pao/History/SP-4204/contents.html>,

<https://www.hq.nasa.gov/office/pao/History/SP-4204/ch15-7.html>, abgerufen am 11.4.2022, Steve Garber, NASA History Web Curator, Updated March 26, 2007, Washington, DC 20546-0001, USA [8]

Herbert D. Benington beschreibt in seinem Aufsatz zur „Produktion von umfangreichen Computerprogrammen“<sup>10</sup> inhaltlich ein sequenzielles Vorgehensmodell:

Sequenzielle Vorgehensweise bei der Programmierung<sup>11</sup>

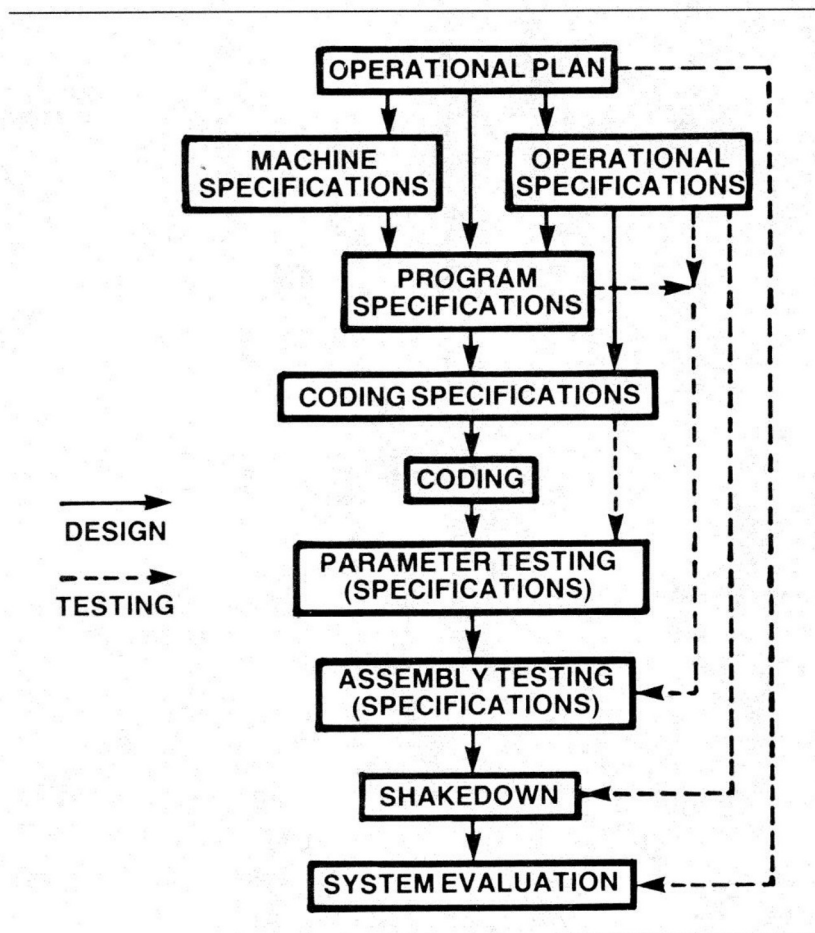


Illustration 5: Arbeitsabschnitte bei der Erstellung von umfangreichen Computerprogrammen

Die weitere Entwicklung der sequenziellen Methoden (darunter insbesondere die Wasserfallmethode) fällt ungefähr in die Zeitspanne der Entwicklung der strukturierten Programmierung<sup>12</sup> als grundlegendes Programmier-Paradigma.

### 3.1 Das Kaskadenmodell oder Wasserfallmethode

Als erstes und bekanntestes sequenzielles Vorgehensmodell im Projektmanagement kann das Kaskaden- oder Wasserfallmodell gelten. Das Modell wurde erstmals 1970 von Winston W. Royce für die Entwicklung großer Softwaresysteme vorgestellt.<sup>13</sup>

10 Production of Large Computer Programs, symposium on advanced programming methods for digital computers sponsored by the Navy Mathematical Computing Advisory Panel and the Office of Naval Research, June 1956, Robert R. Everett, Robert R. Everett, publiziert in: IEEE - Annals of the History of Computing, Seiten 350 – 361, Volume 5, Issue 4, Oct.-Dec. 1983, doi: 10.1109/MAHC.1983.10102[9]

11 Ebenda, Seite 356

12 Structured Programming, O.-J. Dahl, E. W. Dijkstra and C. A. R. Hoare, Academic Press, London, 1972, veröffentlicht in Software – Practice and Experience, Volume 5, 1975[10]

13 Managing the development of large software systems, S 328 -338, Dr. Winston W. Royce, IEEE Wescon, 1970[11]

### 3 Das klassische Modell, sequenzielle Vorgehensmodelle

Schritte bei der Entwicklung großer Computerprogramme<sup>14</sup>

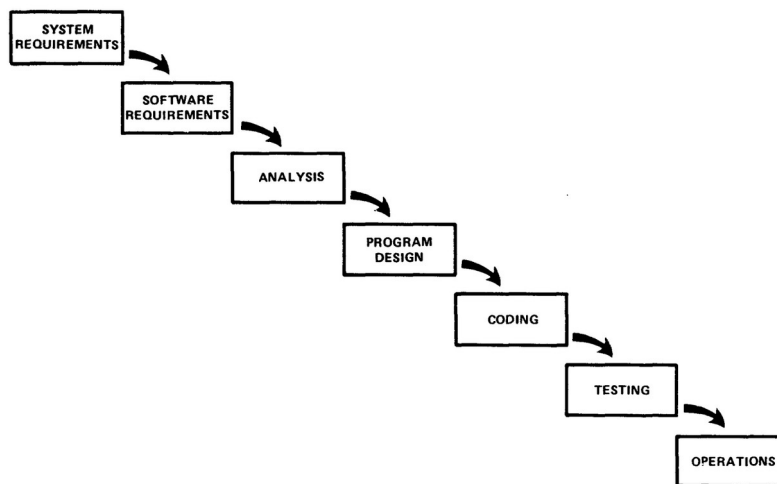


Illustration 6: Schritte bei der Softwareentwicklung nach Royce

Anzumerken ist, dass das Wasserfall-Modell nicht das einzige sequenzielle Projektmanagement Modell ist, darüber hinaus ist an dieser Stelle zumindest noch das V-Modell (Validierungs- und Verifizierungsmodell) als lineares Modell zu erwähnen. Allerdings haben alle linearen (sequenziellen) Modelle die gemeinsame Eigenschaft, dass ein Informationstransfer entgegen der Entwicklungsrichtung nicht stattfindet (nicht vorgesehen ist). Diese Eigenschaft hat bereits Winston W. Royce bei der Vorstellung der „Schritte bei der Entwicklung großer Computersysteme“ bemängelt und hat daher vorgeschlagen, wesentliche Sequenzen von Schritten zu wiederholen.

In der einschlägigen Literatur wird das Wasserfallmodell als „der“ Vertreter des klassischen (linearen) Projektmanagement dargestellt, insbesondere in der Gegenüberstellung zu agilen Projektmanagement–Methoden.

## 3.2 Die Grenzen des Wasserfall-Modells (Waterfall model)

Die Verwendbarkeit eines linearen, sequenziellen Projektmanagement-Modells verschlechtert sich mit Anstieg des Komplexitätsgrades. Dieser Komplexitätsanstieg wird durch eine allgemeine Erhöhung des Komplexitätsgrades (technischer Fortschritt, Geschwindigkeit der Informationsverarbeitung und -verbreitung, neue zusätzliche Umwelteinflüsse), die Verkomplizierung der Organisationsstrukturen, neue Methoden im Supply Chain Management, Verkomplizierung der physischen Transportwege und Parallelaktivitäten mit räumlich verteilten Teilnehmern verursacht.

Darüber hinaus sind in einem rein sequenziellen Projektablauf Iterationen (wenn eine praktikable Lösung nicht bereits beim 1. Durchlauf zu erwarten ist) nicht vorgesehen und es bleibt die Frage der Handhabung von Parallelaktivitäten.

<sup>14</sup> Implementation steps to develop a large computer program for delivery to a customer, managing the development of large software systems, S 329, Dr. Winston W. Royce, IEEE Wescon, 1970[11]

## 4 Das gewandelte Modell und neue Anforderungen

Neue Anforderungen an das Projektmanagement – System lassen sich in den meisten Fällen auf einen Anstieg des Komplexitätsgrades zurückführen, obwohl in vielen Fällen die primäre Ursache anderes gelagert ist. So kann zum Beispiel die Reaktion auf die Reduktion von Lieferzeiten, nämlich das Ineinanderschieben (Parallelisierung) von Aktivitäten zur Erhöhung der Komplexität der Abläufe führen.

### 4.1 Anstieg des Komplexitätsgrades allgemein

Komplexität kann folgendermaßen definiert werden, Zitat aus Gabler Wirtschaftslexikon:<sup>15</sup>

*1. Begriff: Gesamtheit aller voneinander abhängigen Merkmale und Elemente, die in einem vielfältigen, aber ganzheitlichen Beziehungsgefüge (System) stehen. Unter Komplexität wird die Vielfalt der Verhaltensmöglichkeiten der Elemente und die Veränderlichkeit der Wirkungsverläufe verstanden.*

*2. Merkmale: Komplexität ist durch Anzahl und Art der Elemente und deren Beziehungen untereinander bestimmbar. Komplexe Prozesse weisen eine Eigendynamik auf und sind meist irreversibel, sodass Handlungen nicht rückgängig gemacht werden können. Wichtigstes Merkmal komplexer Situationen ist die Intransparenz für den Entscheider: Er hat keine Möglichkeit, das Netzwerk zirkulärer Kausalität intuitiv zu erfassen, keine Möglichkeit exakter Modellierung und exakter Prognosen, er muss mit Überraschungen und Nebenwirkungen rechnen. Der Umgang mit komplexen Systemen erfordert ein hohes Maß an Wissen über die kausalen Zusammenhänge der Systemelemente (Art der Vernetzung) und die Fähigkeit, Komplexität auf wenige Merkmale und Muster zu reduzieren (Komplexitätsreduktion).*

Darüber hinaus wäre als Abgrenzung zu komplizierten Systemen noch festzuhalten, dass sich komplizierte Systeme einfach aus der (großen) Anzahl der Teilelemente definieren und komplexe Systeme durch die Interaktion der Teilelemente untereinander und mit der Umwelt entstehen.

Als allgemeine Indikatoren, die Komplexität anzeigen, sind hier anzuführen<sup>16</sup>:

1. Agentenbasiertheit: Agenten sind Moleküle, Individuen, Abteilungen usw., die untereinander wechselwirken.

---

<sup>15</sup> Komplexität, Gabler Wirtschaftslexikon, Prof. Dr. Eberhard Feess, <https://wirtschaftslexikon.gabler.de/definition/komplexitaet-39259/version-262672>, abgerufen am 6.1.2022 [12]

<sup>16</sup> Wiki Komplexes System, Jens Kowiak, Franz Fester, Claudius Hansch et al, [https://de.wikipedia.org/w/index.php?title=Komplexes\\_System&oldid=217074045](https://de.wikipedia.org/w/index.php?title=Komplexes_System&oldid=217074045), abgerufen am 3.1.2022 [13]

#### 4 Das gewandelte Modell und neue Anforderungen

2. Die Auswirkungen der Wechselwirkungen sind nicht linear: Kleine Unterschiede in den Startbedingungen können zu sehr großen Auswirkungen führen.
3. Emergenz: das Entstehen neuer, nicht vorhergesehener Eigenschaften und Auswirkungen durch das Zusammenwirken der Agenten.
4. Interaktionen: Obwohl die Agenten immer lokal miteinander interagieren, sind die Auswirkungen der Wechselwirkungen global spürbar.
5. Offene Systeme: In dem Sinne, dass ein komplexes System energetisch nicht in einem stabilen Zustand ist; es wird permanent von Energie und Materie (und Informationen) durchströmt und es fließt Energie und Materie (und Informationen) zu und ab.
6. Allerdings können komplexe Systeme durch Selbstorganisation auch unter stetigem Zu- und Abfluss von Energie und Materie (das schließt Informationen ein) einen stabilen Zustand aufrechterhalten.
7. Innere Harmonisierung und Selbstregulation ist die Fähigkeit auf Grund der zur Verfügung stehenden Informationen und entsprechender Verarbeitung derselben, inneres Gleichgewicht und Balance zu stabilisieren.
8. Pfadabhängigkeit bezeichnet die Abhängigkeit nicht nur vom aktuellen Zustand, sondern (möglicherweise) von der (gesamten) Vorgeschichte.
9. Die Existenz von Attraktoren in den meisten komplexen Systemen, die dazu führen, dass das System bestimmte Zustände oder Abfolgen von Zuständen anstrebt. Dieses Streben erfolgt unabhängig von den Anfangsbedingungen.

Kathleen B. Hass erwähnt in ihrem Vortrag „Living on The Edge – Project Complexity Management“<sup>17</sup> wesentliche Komplexitätsfaktoren, Naomi Caietti fasst diese in ihrem Artikel „How to Manage a Complex Project“<sup>18</sup> folgendermaßen zusammen:

1. Detaillierungsgrad: Anzahl der Variablen und Schnittstellen,
2. Ambiguität/Mehrdeutigkeit: mangelnde Erkenntnis Ereignisse und Kausalitäten (Abhängigkeiten) betreffend,
3. Ungewissheit: Unfähigkeit Aktionen im Voraus zu evaluieren,
4. Unvorhersehbarkeit: Unfähigkeit Zukünftiges vorherzusagen,
5. Dynamik: die Geschwindigkeit von Veränderungen,
6. Sozialstruktur: Art und Anzahl der Interaktionen,
7. Wechselbeziehungen: es bestehen viele Wechselwirkungen und Verbindungen.

---

<sup>17</sup> Living on the edge: project complexity management. Kathleen B. Hass, Paper presented at PMI® Global Congress 2007—North America Atlanta GA, Project Management Institute - Newtown Square PA, USA, 2007 [15]

<sup>18</sup> How to Manage a Complex Project, Naomi Caietti, ProjectManager.com Inc, Austin USA 2016, <https://www.projectmanager.com/blog/manage-a-complex-project>, abgerufen am 9.1.2022 [14]



### 4.1.1 Messung des Komplexitätsgrades – der Darnall-Preston Komplexitätsindex (DPCI)

Russell Darnall und John M. Preston definieren in ihrem Buch „Project Management from Simple to Complex“ einen Index, der den Komplexitätsgrad eines Projektes abbildet<sup>19</sup>. Diese Vergleichszahl, genannt DPCI, setzt sich aus 4 Teilbereichen zusammen, nämlich:

- externe Einflussgrößen, darunter fallen die Projektgröße, die Projektdauer und die verfügbaren Ressourcen,
- interne Einflussgrößen, wie zum Beispiel Klarheit über die Projektziele und den Umfang des Projektes, die Komplexität der Organisation und die Konformität mit den Interessen Anspruchsberechtigter (Stakeholder Conformity),
- technologische Einflussgrößen – die Neuheit der Technologie und die Vertrautheit der Projektteam-Mitglieder mit den zu Grunde liegenden Technologien,
- Umwelteinflüsse wie rechtliche, kulturelle, politische und ökologische Rahmenbedingungen.

DCPI Klassifizierungen reichen von 1, 1, 1, 1 (extern klein, kurze Dauer und alle Ressourcen verfügbar, intern niedrige Komplexität, Technologie einfach, Umwelt simpel) bis zu 4, 4, 4, 4 (extern groß, lange Dauer und nicht alle Ressourcen verfügbar, intern hochkomplex, komplexe Technologie und schwierige Umweltbedingungen).

Ein wichtiger Kritikpunkt an dieser Vorgangsweise bei Maschinen- und Anlagenbauprojekten ist, dass die damit erreichbare Einschätzung nur einen Durchschnitt über ein Gesamtprojekt liefert. So können weite Bereiche eines Projektes aus ganz einfachen, unkomplexen „Wasserfall“ Vorgängen bestehen, relativ kleine aber für den Gesamterfolg kritische Aktivitäten aber hochkomplex sein. Daher ist die von Wysocki vorgeschlagene Methode der Zerlegung in „Swimlanes“ und Einzelaktivitäten (siehe Kapitel 5.2 Prozessorientierung) und danach die Einschätzung der Einzelaktivitäten entsprechend ihrem Verhalten nach Project Management Life Cycle (PMLC) Modell in der Projektlandschaft (siehe 5.4.2.4.1 eine intuitive Sichtweise der Projektlandschaft: Klassifizierung nach Eindeutigkeit von Zielen und Lösungsweg) wahrscheinlich zielführender.

## 4.2 Spezielle Treiber des Komplexitätsgrades

Neben den allgemeinen Faktoren, die einen Anstieg des Komplexitätsgrades hervorrufen können, gibt es spezielle, die insbesondere im kombinierten Maschinen- und Anlagenbau zutreffen.

---

<sup>19</sup> Darnall-Preston Complexity Index, Kapitel 2.3 Complex Systems and the Darnall-Preston Complexity Index, Project Management from Simple to Complex, S 51 ff, Russell Darnall und John M. Preston, Minneapolis, USA, 2016(copyright) [16]

### 4.2.1 3-dimensionales Projektmanagement

Obwohl für die Organisation der Kommunikationsstrukturen nicht wirklich ausschlaggebend, lässt sich die explosionsartige Komplizierung der Kommunikationsstrukturen mit Hilfe des Vergleichs von Organisationschemata andeuten.

Beispiel für klassische Projektorganisation

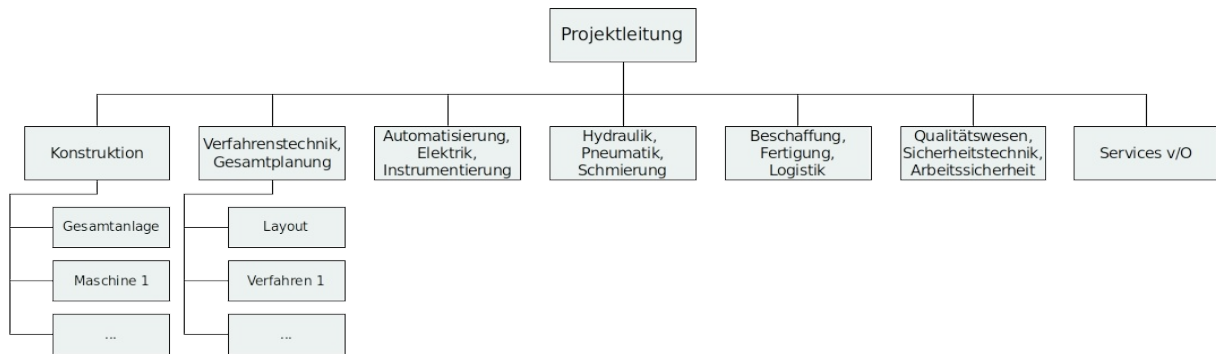


Illustration 7: klassische Projektorganisation im gemischten Maschinen- und Anlagenbau

Ein verändertes Modell unter Einbeziehung von verteilten Örtlichkeiten der Wertschöpfungsstufen (Wertschöpfungsketten) kann man als "3-dimensionales Projektmanagement" darstellen. Die 3 Dimensionen sind somit:

- Örtlichkeiten,
- Verfahren und
- Wertschöpfungsstufen.

Das organisatorische Ergebnis lässt sich folgendermaßen darstellen:

dreidimensionale Projektmatrix

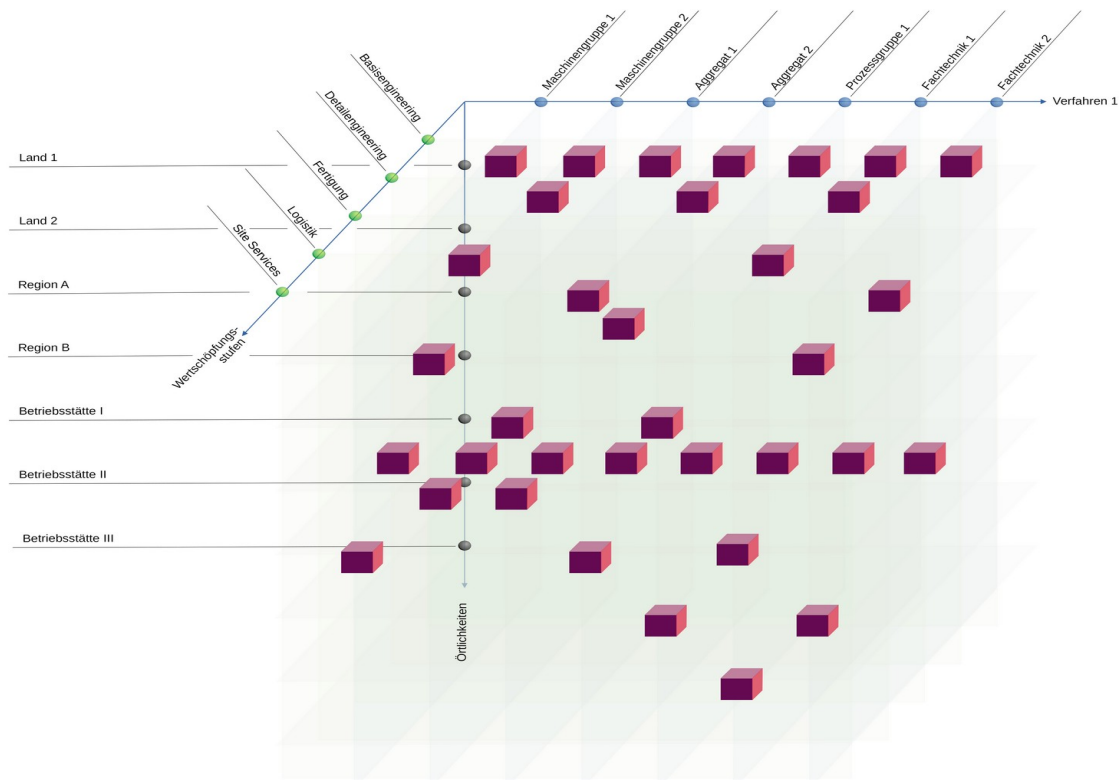


Illustration 8: dreidimensionale Projektorganisation

## 4 Das gewandelte Modell und neue Anforderungen

Fügt man die Projektleitung hinzu, so stellt man fest, dass das Projektmanagement quasi schon in der 4. Dimension sitzt. Das Projektmanagement hat Zugriff auf alle 3 Dimensionen der Projektorganisation.

### Die Projektleitung in der dreidimensionalen Projektmatrix

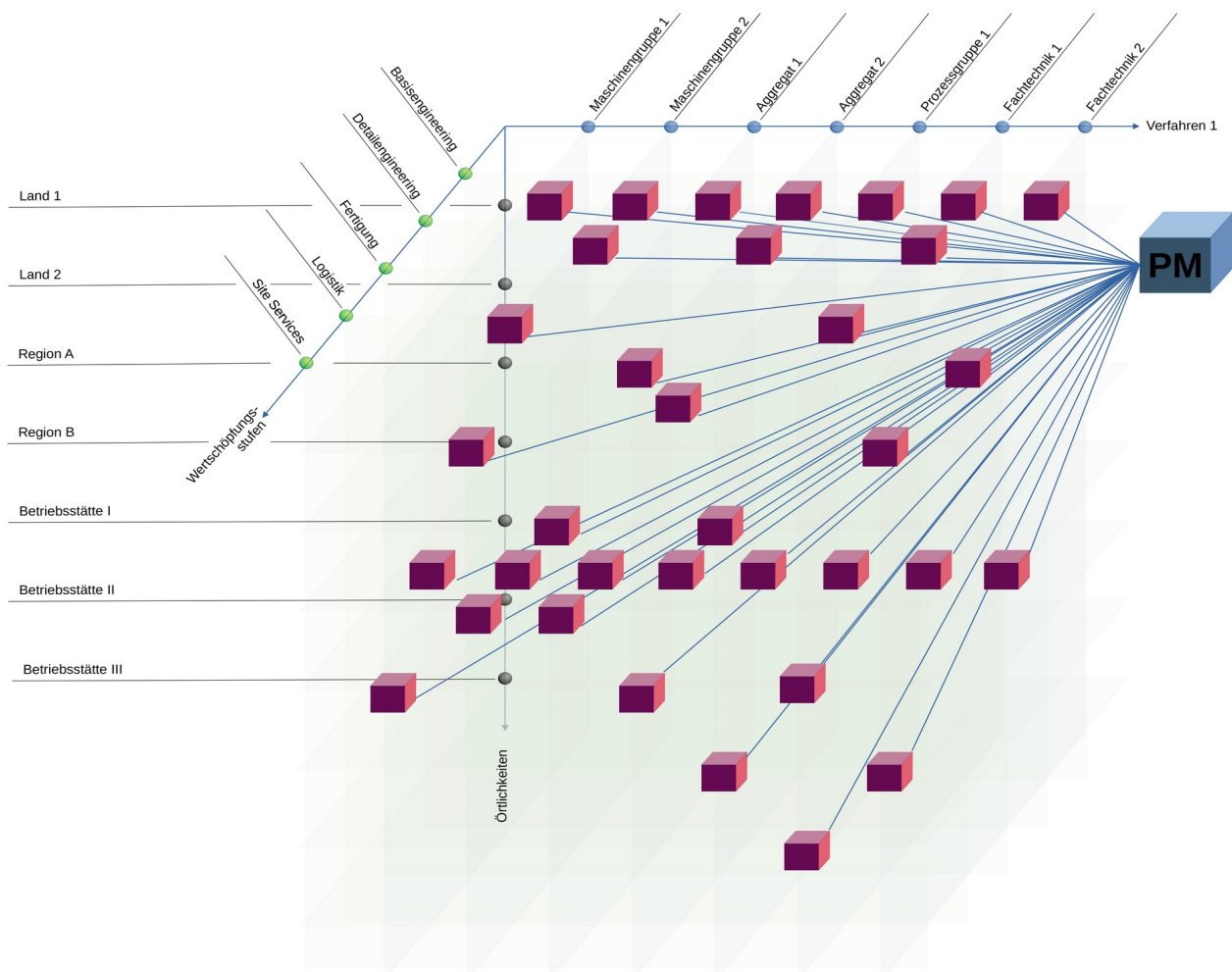


Illustration 9: Projektleitung und dreidimensionale Projektmatrix

Leicht nachzuvollziehen, dass im Vergleich zur klassischen Projektorganisation die Kommunikationsbeziehungen komplizierter werden. Schwierig wird es insbesondere dann, wenn die Linien- oder Funktionsmanager (die oben als kleine Kugeln dargestellten Instanzen) im Projektgeschehen mitmischen: das kann bis zur vollständigen Paralyse der Projektabwicklung führen.

### 4.2.2 Neue Methoden des Supply Chain Managements

Die Beschaffungs-, Fertigungs- und Kooperationsstrategien haben sich geändert, die Grenzen z.B. zwischen Beschaffungs- und Fertigungsstrategien verschwimmen. Das hat gravierende Einflüsse auf die Kommunikationsstrukturen und vor allem auf die Komplexität des Projektes. Als Faustregel kann gelten: Obwohl die Einführung neuer Methoden des Supply Chain Managements oft als Maßnahme der Komplexitätsreduktion gedacht, führen sie in der Praxis praktisch immer zu einer Komplexitätssteigerung.

## 4 Das gewandelte Modell und neue Anforderungen

Die Übersicht über das Instrumentarium der Supply Chain Management Methoden wurde aus den "Fallstudien zum Internationalen Management"<sup>20</sup> entnommen und für den gemischten Maschinen- und Anlagenbau angepasst.

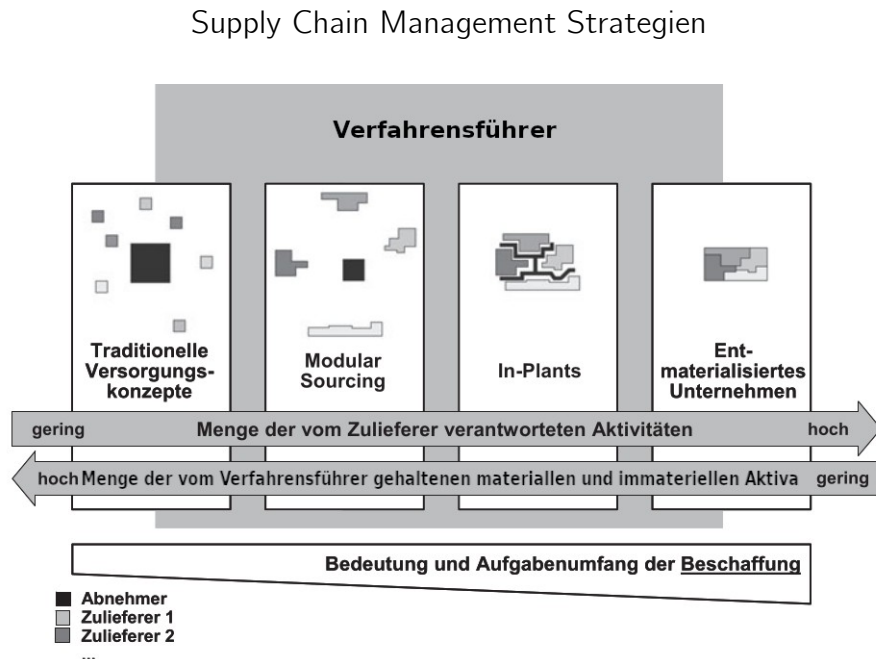


Illustration 10: Strategien des Supply Chain Managements nach Zentes et al

### 4.2.2.1 Traditionelle Beschaffungskonzepte

Traditionelles Beschaffungsmanagement bedeutet Optimierung der Beschaffungseffizienz entsprechend nach geringen Kosten, geringem Risiko und/oder geringem Beschaffungsaufwand.

Die Beschaffungsstrategie wird entsprechend nach single-, dual-, multiple-, local-, global Sourcing Modellen optimiert. Allen traditionellen Beschaffungskonzepten ist gemein, dass die Systemverantwortung oder Teilsystemverantwortung beim Auftraggeber bleibt.<sup>21</sup>

### 4.2.2.2 Modulare Beschaffung

Neben Fertigungsaufgaben werden auch Konstruktion und/oder Engineering durch den Lieferanten übernommen. Neben den technischen Schnittstellen entstehen kaufmännische-, vertragsrechtliche- und haftungsrechtliche Schnittstellen.

### 4.2.2.3 In-Plants

Die Verlagerung von Aktivitäten zu (Sub)Lieferanten kann natürlich weitergehen. Bei In-Plants stellt der Auftraggeber nur mehr die Montagehalle zur Verfügung oder die Montagen werden durch den Lieferanten auf der Baustelle durchgeführt, der Lieferant führt bereits die

<sup>20</sup> Beschaffungsmanagement bei Fairchild Dornier von Michael EBig und Tobias Buck, Fallstudien zum Internationalen Management, Grundlagen – Praxiserfahrungen – Perspektiven, Seite 259, Joachim Zentes, Bernhard Swoboda, Dirk Morschett, Gabler Verlag - Springer Fachmedien Wiesbaden GmbH, 3. Auflage, Wiesbaden 2008, [17]

<sup>21</sup> BWL-Lexikon, Beschaffungsstrategien, <https://www.bwl-lexikon.de/wiki/beschaffungsstrategien/>, abgerufen am 12.12.2020, [18]

Montageleistung aus. Auf jeden Fall führt der Lieferant in In-Plant Beziehungen Engineering und Entwicklungsaufgaben aus.

#### 4.2.2.4 Entmaterialisiertes Unternehmen

Die Verantwortlichkeiten können noch weiter weitergegeben werden, z.B. wenn die Montage mit den erforderlichen Überwachungsleistungen bei einem besonderen Montagelieferanten in dessen Werkshallen stattfindet. Es entstehen somit virtuelle, grenzenlose, fraktale und sogenannte entmaterialisierte Industrieunternehmen.<sup>22</sup>

Ausprägungsgrad der Virtualität nach Eßig<sup>23</sup>

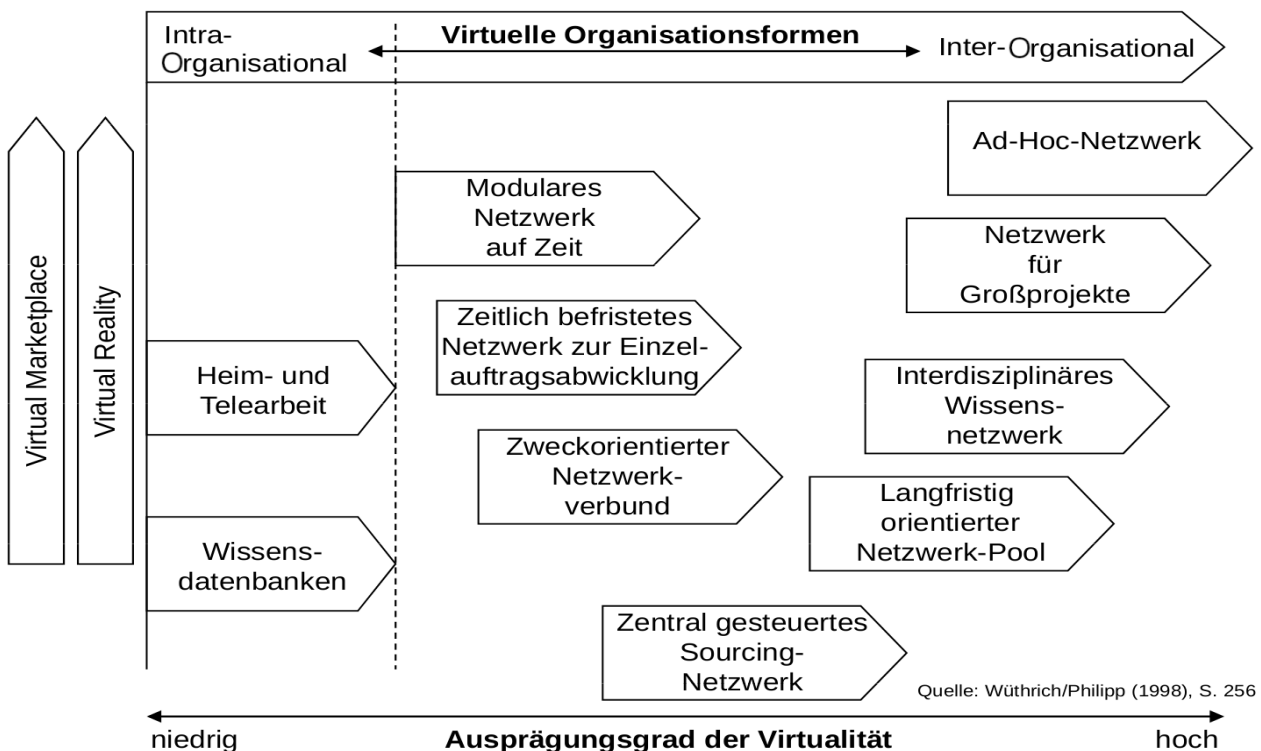


Illustration 11: Grad der Virtualität nach Eßig

Allerdings sollte man sich im Klaren sein, dass derartige entmaterialisierte Unternehmen im internationalen Anlagenbau keine Neuigkeit sind; ein Verfahrensführer, der ausschließlich Dienstleistungen erbringt und alle physischen Leistungen durch Lieferanten erbringen lässt und dabei als Generalunternehmer (als EPC – Engineering, Procurement, Construction<sup>24</sup>) auftritt, ist nicht neu und nicht unüblich.

Stellt man sich die dreidimensionale Projektmatrix aus Abbildung 3 um derartige "Verklumpungen" wie modulare Beschaffung, In-Plants und entmaterialisierte Unternehmen vor, wird das Ergebnis insbesondere in Bezug auf die Informations- und Kommunikationsstruktur sehr komplex. Tatsächlich kann in einer konkreten Projektsituation

<sup>22</sup> Vorlesungsskript: Supply Chain Management I, Universität der Bundeswehr München, Professur für Materialwirtschaft und Distribution, Univ. Prof. Dr. Michael Eßig, 3.3 Virtuelle, grenzenlose, fraktale und entmaterialisierte Industrieunternehmen, Seite 83, München, 2021 [19]

<sup>23</sup> Ebenda, Seite 84

<sup>24</sup> Turnkey contracting, Turnkey contracting under the FIDIC Silver Book, Seite 2ff, Jonathan Hosie, London, November 2007 [20]

## 4 Das gewandelte Modell und neue Anforderungen

nicht von vornherein klar sein, wo die kritischen Informationspfade liegen; schlimmer noch, es ist manchmal nicht klar, dass derartige Informationspfade überhaupt erforderlich sind (typisches Beispiel: Risikomanagement über Unternehmensgrenzen hinweg).

### 4.2.3 Die Verkomplizierung der physischen Transportbeziehungen

Die physischen Liefer- und Transportbeziehungen haben sich in den letzten Jahrzehnten gewandelt, zur Veranschaulichung:

Die klassischen Lieferbeziehungen

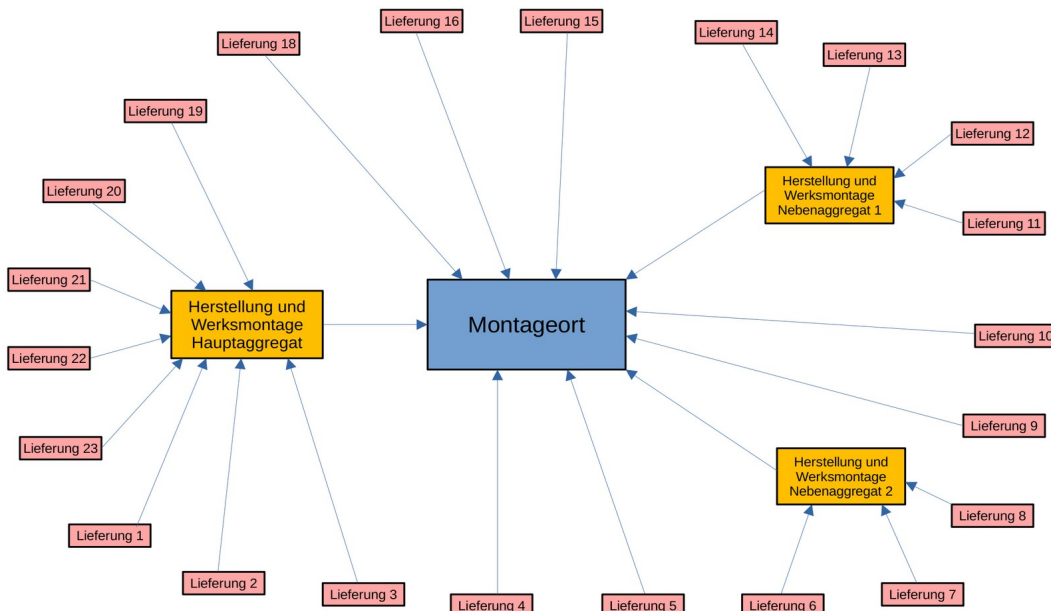


Illustration 12: Warenströme in klassischen Lieferbeziehungen

Die verkomplizierten Lieferbeziehungen

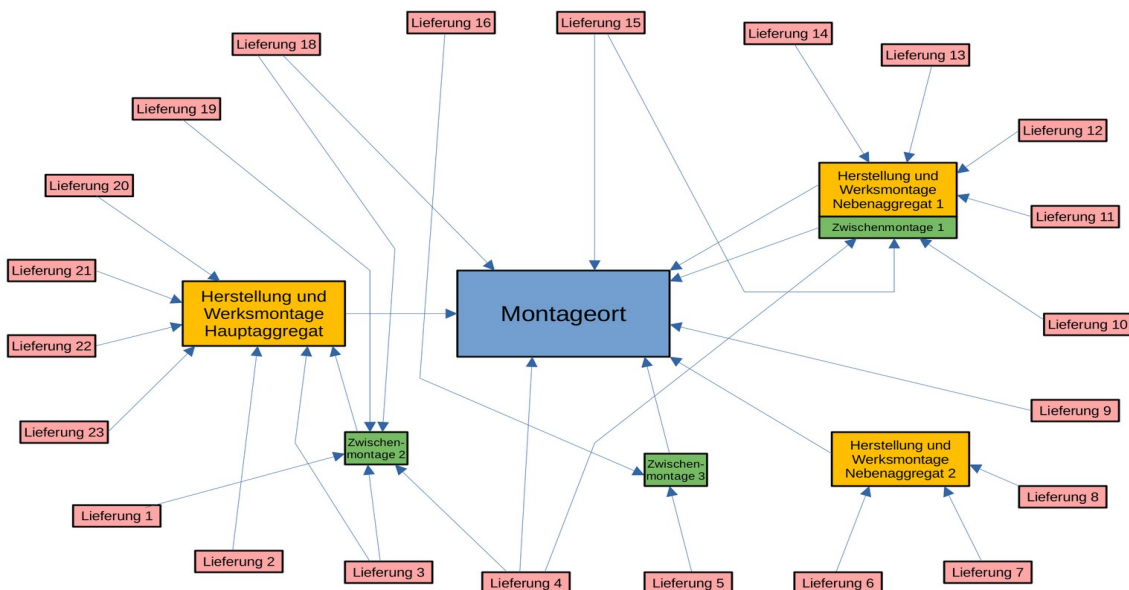


Illustration 13: Warenströme bei komplexen Lieferbeziehungen

Es werden zusätzliche Zwischenstationen (z.B. Zwischenmontage) eingeführt, die in der Logistikplanung berücksichtigt werden müssen.

### 4.2.4 Das Koordinierungsproblem paralleler Aktivitäten bei gleichzeitiger geographischer Dislozierung der Akteure

Um Durchlaufzeiten zu reduzieren (siehe auch 5.4.2.3 Zeitersparnis durch Ineinanderschieben sequenzieller Arbeiten) versucht man Aktivitäten nicht mehr sequenziell sondern, zumindest teilweise, parallel ablaufen zu lassen.<sup>25</sup>

Prinzip der Parallelisierung

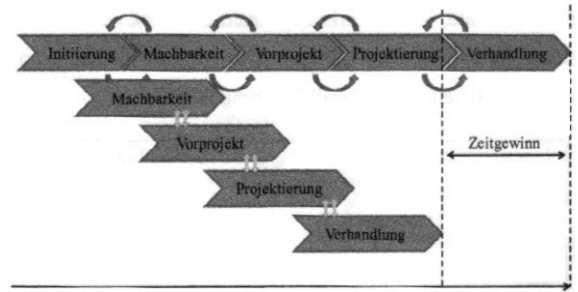


Illustration 14: Prinzip der Parallelisierung, Stärk in Anlehnung an B. Prasad

Betrachtet man die Vorgangsweise mit einem Projektstrukturplan wäre folgende Struktur denkbar:

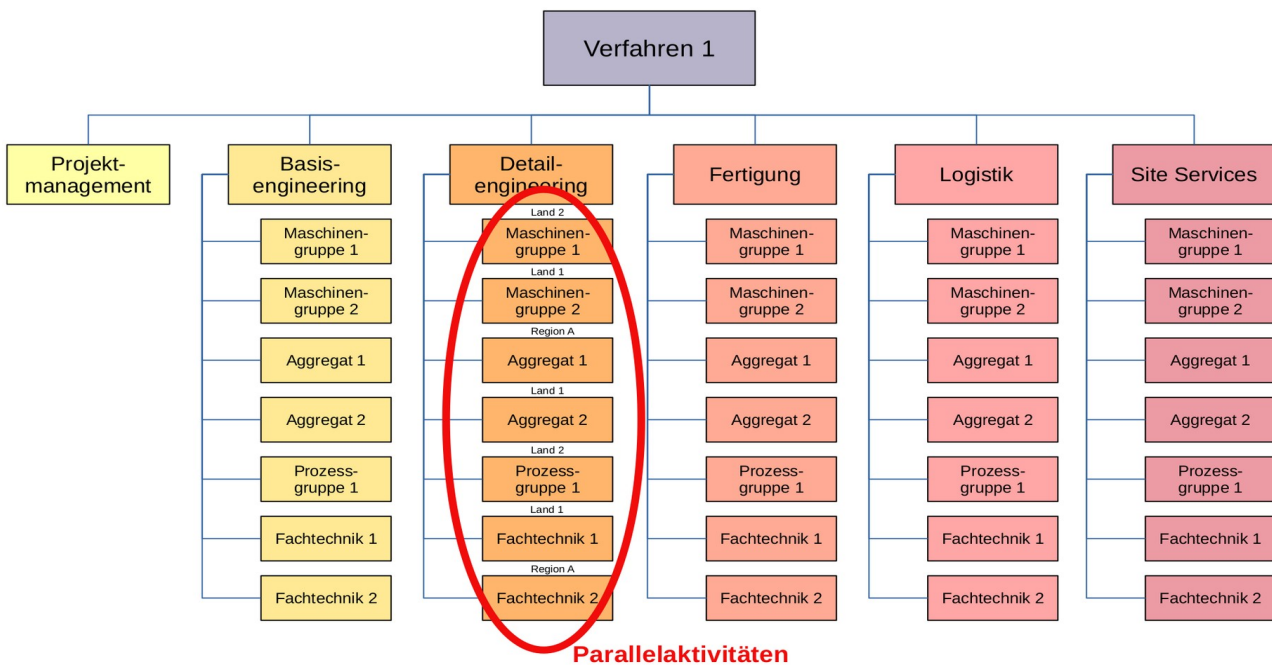


Illustration 15: Projektstrukturplan und Parallellaktivitäten

Der Gesamtablauf des Projektes wird also mit einer Mischform aus kaskadierten (Waterfall) Methoden, also traditionelles Projektmanagement (TPM) und agilen Methoden bewältigt.

<sup>25</sup> 4.1.2.2 Parallelisierung, Agiles Projektmanagement im Anlagen- und Maschinenbau, Seite 59ff, Johannes Stärk, Dissertation Universität Hohenheim, 2016 [21]

## 4 Das gewandelte Modell und neue Anforderungen

Gründe für die Einführung von Parallelaktivitäten:

- Zeiteinsparung: man schiebt ursprünglich serielle Aktivitäten ineinander um die benötigte Gesamtzeit zu reduzieren.
- Gleichzeitige Entwicklung wünschenswert, um enge Koordination zwischen beteiligten Disziplinen zu ermöglichen.
- Iterative und nicht sequenzielle Ansätze erwünscht, weil das Arbeitsergebnis nicht linear vorausplanbar ist.
- Iterative Ansätze können zu einem früheren Abliefern von brauchbaren Arbeitsergebnissen führen; die erste für alle Projektbeteiligten brauchbare Lösung kommt früher als das "komplett fertige" Arbeitsergebnis.

Erschwernis- und komplexitätserhöhend wirkt sich örtliche Trennung der beteiligten Disziplinen oder sogar einzelner Beteiligter bei dieser parallelen/agilen Abarbeitung der Arbeitsinhalte aus.

Allerdings kann als gute Nachricht für den gemischten Anlagen- und Maschinenbau gewertet werden, dass das Koordinierungsproblem in der Softwareindustrie längst gelöst ist, stammen doch die Bezeichnungen "kaskadiert" (waterfall) und "agil" aus der Softwarebranche<sup>26</sup>. Dabei wurden Mechanismen entwickelt, die die parallele Bearbeitung von Programmabschnitten ermöglichen und über regelmäßige Zusammenführung und Auflösen von Schnittstellenproblemen und folgende Funktionskontrolle die Koordination paralleler Entwicklungsarbeit ermöglichen. Man stelle sich zum Beispiel die Lösung von Kollisionsproblemen in der Rohrleitungsplanung des Anlagenbaus als Analogie zur Auflösung von Schnittstellenproblemen bei der Softwareentwicklung vor.

---

<sup>26</sup> Agile vs. Waterfall: Differences in Software Development Methodologies - The Ultimate Reference for Project Managers, Jose Maria Delos Santos, <https://project-management.com/agile-vs-waterfall/>, abgerufen am 14. July 2021 [22]



# 5 Lösungsansätze

Die Summe, der in Kapitel 4. genannten neuen Anforderungen führt dazu, dass zumindest in Teilbereichen die Aufgaben durch traditionelle Projektmanagement (TPM) Methoden – oder auch Wasserfall/Kaskaden Methoden genannt – nicht mehr sinnvoll lösbar sind.

Zur Lösung dieser Defizite bieten sich agile/iterative Vorgehensmodelle (APMs) an oder Mischformen aus traditionellem Projektmanagement und agilem Projektmanagement, nämlich hybrides Projektmanagement. APMs und hybrides Projektmanagement sind Methoden des komplexen Projektmanagements (CPM).<sup>27</sup>

## 5.1 Voraussetzungen

### 5.1.1 Ordnungs- und Klassifizierungssystem

Für die eindeutige Identifizierung eines Bauteils (=Betriebsmittel-Kennzeichnung) ist nach Whitepaper des Projektes “Assistenzsysteme für die Prozessindustrie auf Basis von cyber-physischen Produktionssystemen” des Bundesministeriums für Bildung & Forschung eine brauchbare Betriebsmittelkennzeichnung erforderlich<sup>28</sup>. Das betrifft die komplette Lebensdauer eines Bauteils, einer Anlage, Maschine vom frühen Basisengineering über die Stücklistenauflösung, Bestell- und Transportwesen, vor Ort Leistungen, Service und Kundendienst, Dokumentation und Ersatzteilwesen. Die Betriebsmittelkennzeichnung ist mit der DIN EN 81 346 genormt<sup>29</sup>.

Das Ordnungs- und Klassifizierungssystem sollte man als Selbstverständlichkeit ansehen, ist es aber leider nicht. Das Fehlen (oder die fehlende Durchgängigkeit) kann zu kritischen Situationen führen, steigert aber auf jeden Fall die Ineffizienz.

### 5.1.2 Cooperation tools

Eine Vielzahl von Werkzeugen haben die letzten Jahre die Arbeit von Gruppen mit örtlich getrennt arbeitenden Mitgliedern erleichtert, davon sind zwei der Werkzeuge wesentlich bei der Überbrückung räumlicher Distanzen:

- Leicht verwaltbare (im Sinne von Vergabe von Rechten) und über das Internet erreichbare Verzeichnisse und Verzeichnisstrukturen zur Speicherung von Arbeitsergebnissen aller Art. Zu Bedenken ist hierbei insbesondere, dass Teile des Teams nicht der gleichen Organisation angehören (Kunde, Lieferanten, Konsortialpartner, freie Mitarbeiter), somit scheidet in vielen Fällen die Einbindung in eine firmeneigene Verzeichnisstruktur via VPN Kanal auf Grund von

---

<sup>27</sup> Part III - Complex Project Management, Effective Project Management (Traditional, Agile, Extreme, Hybrid), Seite 353, Robert K. Wysocki, Indianapolis, USA, 2019 [23]

<sup>28</sup> Whitepaper: Betriebsmittelkennzeichnung in der Prozess- und Anlagenbau Industrie, Management Summary, Seite 7, Dr S. Adler , A. Eckstein, S. Skrytutskyi , M. Eisentraeger , S. Moeser, Fraunhofer IFF, Magdeburg, 2017 [24]

<sup>29</sup> Ebenda, Seite 15

Sicherheitsüberlegungen aus.

Allerdings sind Cloudlösungen wie z.B. ShareFile® der Citrix Systems Inc. verfügbar, ein firmeneigener oder angemieteter FTP Server leistet solche Dienste aber auch (schon seit den 1990er Jahren). Bezüglich Versionierung von gemeinsam bearbeiteten Datenbeständen siehe Versionierungstools.

- Videotelefonie, Videokonferenzen mit Präsentationsmöglichkeiten wie z.B. Skype® oder Microsoft Teams®. Im Besonderen sei an dieser Stelle auf freie und Open Source Lösungen wie z. B. Jitsi oder Jami verwiesen. Jami ist eine Audio/Video/Conferencing Software mit integrierten Instant Messenger, die Internetseite von Jami befindet sich hier: <https://jami.net/><sup>30</sup>.

Bewusst habe ich an dieser Stelle auf "übliche" Werkzeuge zur Organisation von Zusammenarbeit wie z.B. Email, Messenger Dienste, Terminpläne, elektronische ToDo-Listen (Tasks) und dergleichen mehr, verzichtet, da ich diese als Standard in der Projektkommunikation betrachte.

### 5.1.3 Versionierungstools

In der Praxis kommen Versionierungstools meist im Paket mit den im Kapitel "Cooperation Tools" erwähnten Dokumenten-Verzeichnissen. Ein Beispiel dafür ist das Doxis4® der SER Group, Die Internetseite für Doxis4® ist hier:

<https://www.sergroup.com/en/document-management.html><sup>31</sup>

Allerdings haben derartige Systeme teilweise Limitierungen bei der Überwindung von Internetbarrieren. So kann der Datentransfer von und nach (VR) China teilweise sehr mühsam werden.

Darüber hinaus existieren anwendungsspezifische Versionierungstools wie z.B. der AutoDesk Vault, ein Datenmanagement Werkzeug, das direkt mit dem 3D CAD Programm AutoDesk Inventor® verbunden ist.

Als Alternative ist ein Weg ähnlich wie bei der Softwareentwicklung möglich: die Verwendung von firmeneigenen Servern oder Cloudlösungen als Speicherort der Datenbasis und der Einsatz von frei erhältlichen Versionierungstools wie GIT (siehe <https://git-scm.com/>)<sup>32</sup> oder Monotone (siehe <https://www.monotone.ca/>)<sup>33</sup>.

### 5.1.4 CAD Systeme, die verteiltes Arbeiten erlauben

Es existieren CAD Systeme, die disloziertes, simultanes Arbeiten mehrerer Benutzer am gleichen Projekt unterstützen. Ein Beispiel dafür ist wiederum der AutoDesk Inventor®.

---

30 Jami Audio/Video/Conferencing, <https://jami.net>, abgerufen am 27.7.2021, Montreal, Canada [25]

31 Document management with Doxis4®, <https://www.sergroup.com/en/document-management.html>, abgerufen am 27.7.2021, Bonn, Deutschland [26]

32 Git – distributed-even-if-your-workflow-isnt, <https://git-scm.com/>, abgerufen am 26.4.2022, Software Freedom Conservancy, Brooklyn New York, USA, 2005 [28]

33 Monotone is a free distributed version control system, <https://www.monotone.ca/>, abgerufen am 26.4.2022, California, USA [27]

Eine Alternative dazu ist z.B. Solid Edge® von Siemens Digital Industries Software. Die Internetseite von Solid Edge® ist hier: <https://solidedge.siemens.com/de/><sup>34</sup>.

Betreffend Verfahrenstechnik sind hier COMOS von Siemens sowie Smap-3D anzuführen.

### 5.1.5 PPC/PPS (Produktionsplanung und Steuerung)

Im Zentrum der Überlegungen stehen hier die termingerechte Übergabe und Einplanung der im Projekt generierten BOM (Material-/Stücklisten) in die Produktionsplanung und Steuerung, also auf der Ebene 3 des PERA (Purdue Enterprise Reference Architecture)<sup>35</sup> Konzeptes. Genau genommen liegt hier ein Datentransfer zwischen Ebenen 2 und 3 im Dienstleistungsbetrieb mit den Ebenen 3 der Fertigungsbetriebe vor.

Die Besonderheiten bei der Abarbeitung in Betrieben mit örtlich verteilten Planungs- und Fertigungsbereichen sind:

- Fertigungsaufträge unter Umgehung der Verkaufsbereiche abzuarbeiten.
- Diese „externen Aufträge“ treffen auf ein Gemisch von Fertigungsstrategien: Make to Order, Assembly to Order, Make to Stock, die auf einem teilweise gemeinsamen Maschinenpark abzuwickeln sind.

Ein durchgängiges Modell, das die Aufgaben der Produktionsplanung und -Steuerung unter den vorgenannten Prämissen ohne die Zuhilfenahme manueller Hilfsmittel und informeller Kontakte leisten kann, ist mir leider nicht bekannt.

Bei der Neukonzeption derartiger Systeme kann auf bewährte Methoden und Modelle wie zum Beispiel das CIMOSA (Computer Integrated Manufacturing Open System Architecture)<sup>36</sup> zurückgegriffen werden.

---

34 Solid Edge 2D/3D-CAD-System, <https://solidedge.siemens.com/en/>, abgerufen am 27.7.2021, Plano, Texas, USA [29]

35 Purdue Enterprise Reference Architecture, Kapitel 25, S 183 ff, How to survive in the jungle of Enterprise Architecture Frameworks, Jaap Schekkermann, Victoria, Canada 2004[30]

36 CIMOSA process model for enterprise modelling, Kosanke, K., Vernadat, F.B., Zelm, M., 1997, in: Goossenaerts, J., Kimura, F., Wortmann, H. (eds) Information Infrastructure Systems for Manufacturing. DIISM 1996. IFIP — The International Federation for Information Processing. Springer, Boston, MA, USA [31]

## 5.2 Prozessorientierung

Die oberflächliche Betrachtung von Vertrieb und Abwicklung von Maschinen- und Anlagenbauprojekten aus der Sicht der ablaufenden Geschäftsprozesse, kann zu folgendem Ergebnis kommen:

Grobanalyse der Prozesse im kombinierten Maschinen- und Anlagenbau

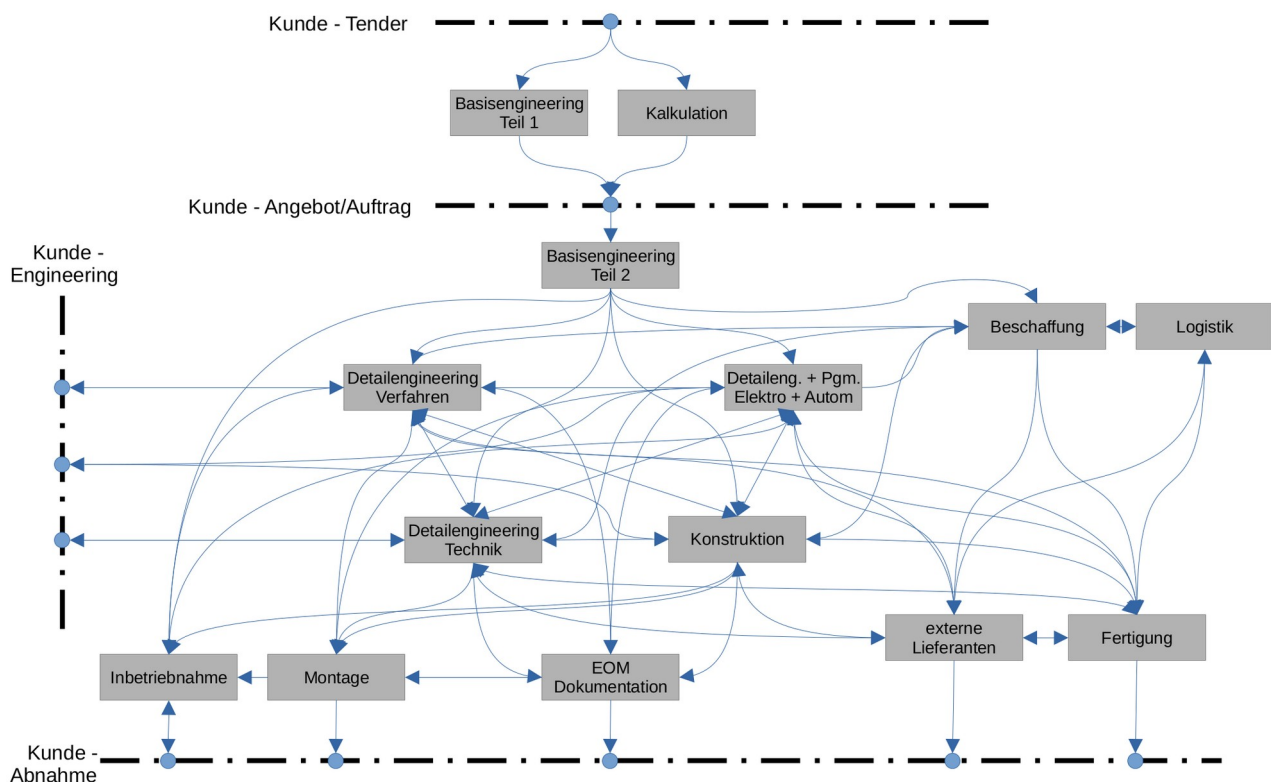


Illustration 16: Prinzipanalyse der Prozesse im kombinierten Maschinen- und Anlagenbau

Die Abbildung soll veranschaulichen, dass ein entsprechender Lösungsansatz nicht hilfreich bei der Verbesserung des Verständnisses der ablaufenden Prozesse und deren Interdependenzen ist. Zur weiteren Auflösung der Aktivitäten und Abhängigkeiten gibt es die Möglichkeit der Atomisierung (Auflösung der Aktivitäten in immer kleinere Subaktivitäten), allerdings darf bezweifelt werden, dass damit Übersichtlichkeit und Klarheit bezüglich der wesentlichen Zusammenhänge und Abhängigkeiten verbessert würde.

Robert K. Wysocki schlägt an dieser Stelle die Zerlegung des Gesamtprojektes in wesentliche Kombinationen/Abfolgen von Teilprozessen vor. Die Betonung liegt auf „wesentlich“. Wysocki bezeichnet diese Teilprozesse als „Swim Lanes“<sup>37</sup> in Zusammenhang mit der Entwicklung sogenannter „ECPM“ (Effective Complex Project Management) Prozess-Flussdiagramme. Das ECPM – Rahmenkonzept ist eine Sammlung von Methoden zur Auflösung von komplexen Projektmanagementaufgaben.<sup>38</sup>

<sup>37</sup> Imbedding ECPM in Traditional Project Management, Chapter 14 – Hybrid Project Management Framework, Effective Project Management – Traditional, Agile, Extreme, Hybrid, S 433, Robert K. Wysocki, Indianapolis USA, 2019 [23]

<sup>38</sup> Ebenda, Background of the Effective Complex Project Management Framework, Seite 415 ff

## 5 Lösungsansätze

Das Ablaufdiagramm dazu siehe Anlage 2.

wesentliche Teilprozesse als Swimlanes<sup>39</sup>

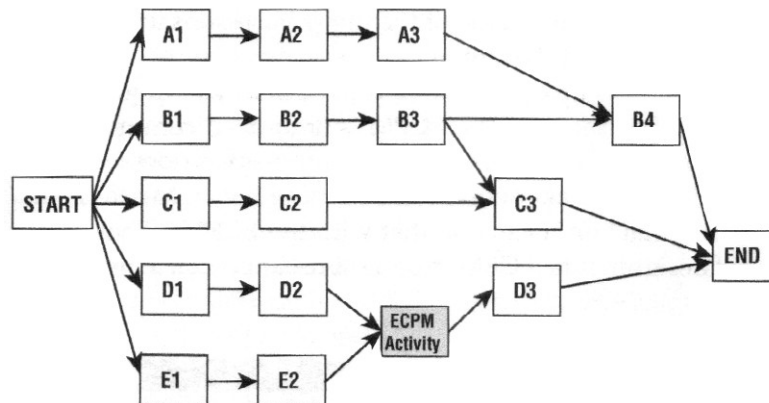


Illustration 17: Swimlanes nach Wysocki (Vorrangdiagramm)

Auf den Maschinen- und Anlagenbau umgelegt ergeben sich 3 grundlegende Abfolgen von Teilprozessen.

### 5.2.1 Swimlanes bei Maschinenbau-Komponenten

Als Verallgemeinerung kann der Ablauf der Geschäftsprozesse zur Entwicklung, Auslieferung und Inbetriebnahme von Maschinenbau-Komponenten folgendermaßen dargestellt werden.

Prozessablauf Maschinenbau

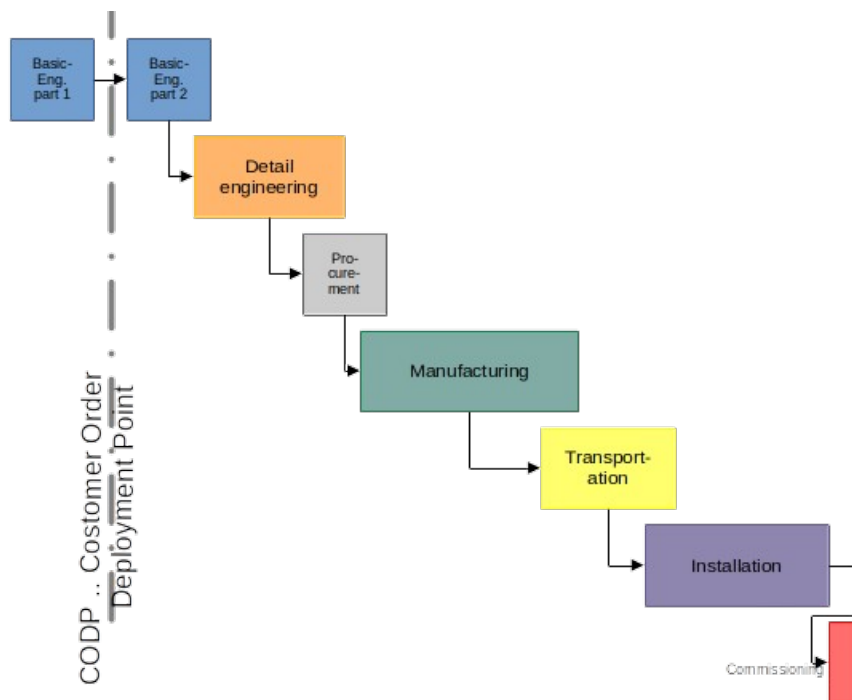


Illustration 18: Maschinenbau Swimlane

<sup>39</sup> Imbedding ECPM in Traditional Project Management, Chapter 14 – Hybrid Project Management Framework, Effective Project Management – Traditional, Agile, Extreme, Hybrid, S 433, Robert K. Wysocki, Indianapolis USA, 2019 [23]

## 5 Lösungsansätze

In einem realen Anlagen-/Maschinenbau Projekt werden dabei mehrere ähnlich aufgebaute Geschäftsprozesse dieser Art parallel ablaufen. Werden dabei mehrere Teilmaschinen zu einer Gesamtmaschine verknüpft, kann unter bestimmten Voraussetzungen schon dadurch ein komplexes Koordinierungsproblem entstehen, das agile oder extreme Vorgehensmethoden erfordert.

Die Darstellung als Gantt-Diagramm wurde verwendet, um die Überlappung der verschiedenen Aktivitäten darzustellen. Zum Beispiel beginnen die Beschaffungsaktivitäten während der Konstruktionsarbeiten, es kann nicht abgewartet werden bis die Konstruktionsarbeiten beendet sind. In vielen Teilen sind Beschaffungsaktivitäten für den Abschluss der Konstruktionsarbeiten zwingend notwendig, zum Beispiel bei technische Details von Zukaufteilen.

### 5.2.2 Swimlanes im Anlagenbau-Projektteil

Ähnlich wie bei Maschinenbau-Komponenten lassen sich Swimlanes des Anlagenbau-Projektteils darstellen.

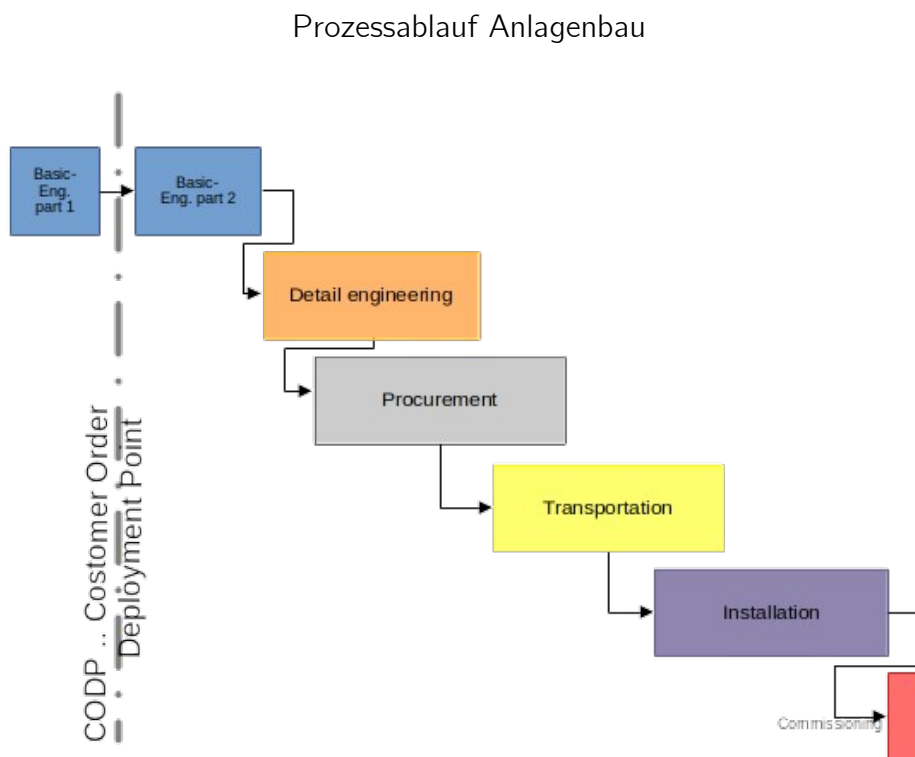


Illustration 19: Anlagenbau Swimlane

Die wichtigsten Unterschiede zu den Maschinenbau-Komponenten sind:

- die Fertigung fehlt im Vergleich zum Maschinenbau,
- die Konstruktion (Detailengineering) benötigt mehr Zeit, es werden im Wesentlichen Unikate geplant,
- die Montagearbeiten benötigen mehr Zeit.

### 5.2.3 Swimlanes von Sonderbauteilen im kritischen Pfad

Die Besonderheit dieser Sonderform ist, dass auf Grund der langen Beschaffungszeiten die Beschaffungsaktivitäten schon vor Auftragserhalt (CODP) einsetzen müssen. Diese Beschaffungsaktivitäten vor Auftragserhalt müssen darüber Klarheit bringen, ob ein potenzieller Auftrag innerhalb der geforderten Lieferzeit erfüllt werden kann. Im Grunde wurde diese Vorgangsweise durch ständige Kürzung der Gesamt-Lieferzeit verursacht, sodass letztendlich der kritische Pfad mit Anlieferung von Vormaterialien, Fertigung, Transport und Installation weniger Komponenten die mögliche Gesamt-Lieferzeit bestimmt. Das Risiko der parallel ablaufenden Fertigstellung der Konstruktionsarbeiten wird dabei bewusst in Kauf genommen.

Prozessablauf Sonderbauteile

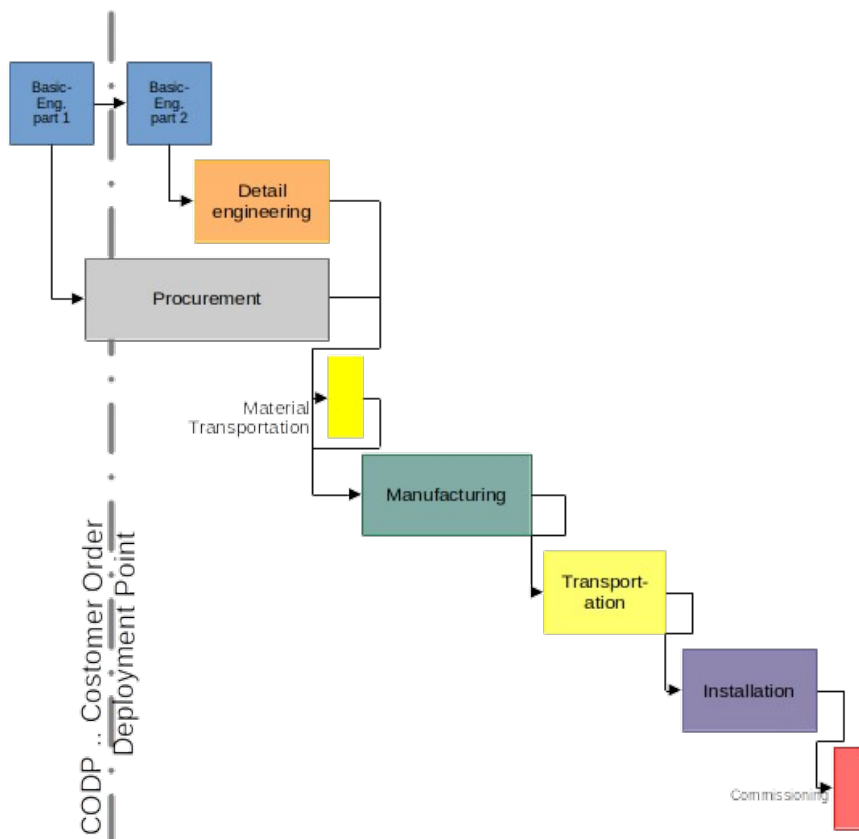


Illustration 20: Sonderbauteile Swimlane

## 5.3 Projektorganisation

Bei der Entwicklung der Projektorganisation (genau genommen der Projekt-Aufbauorganisation) mit zeitgemäßen Modellen, wie z.B. dem „Projektmanagement-Guide“ von Domendos<sup>40</sup> kann man grundsätzlich funktionalen Gesichtspunkten folgen.

<beispielhafte Projektorganisation nach Domendos umseitig>

40 Project organization – function and roles in the project, Projectmanagement.guide, Domendos - Franz Pavlik, <https://projectmanagement.guide/category/projectmanagement/>, abgerufen am 30. Januar 2022, Wien – Österreich [32]



## 5 Lösungsansätze

beispielhafte Projektorganisation<sup>41</sup>



Illustration 21: beispielhafte Projektorganisation nach Domendos - Projectmanagement.guide

Die entsprechende Konzentration der Führungsaufgaben, orientiert an funktionellen Gesichtspunkten, führt zu einer Projektorganisation wie folgt:

Kern-Projektmanagement-Team:

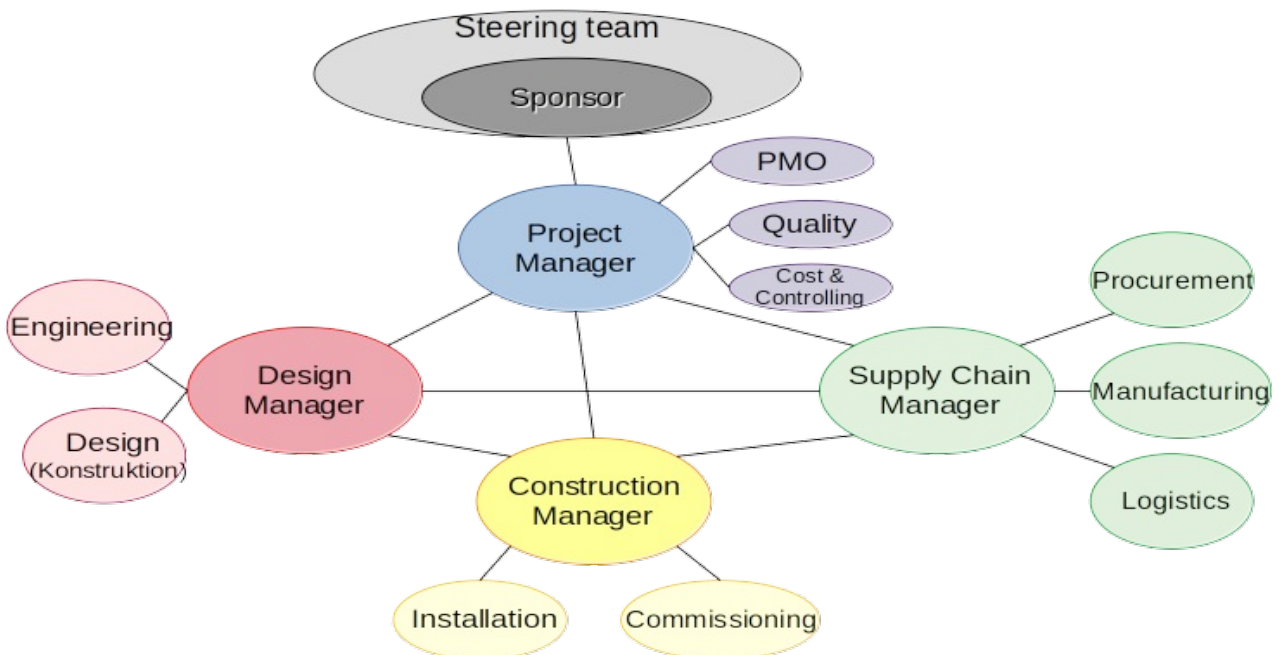


Illustration 22: zeitgemäße Projektorganisation

<sup>41</sup> Example project organization, Project organization – function and roles in the project, Projectmanagement.guide, Domendos - Franz Pavlik, <https://projectmanagement.guide/category/projectmanagement/>, abgerufen am 30. Januar 2022, Wien – Österreich [32]



Bewusst wurde auf die Darstellung einzelner Gewerke verzichtet, so folgt z.B. das Elektrotechnik-/Automatisierungspaket im Wesentlichen den Besonderheiten der Verfahrenstechnik.

Darüber hinaus wäre zu beachten, dass nicht das komplette Projekt-Kernteam zu 100% über die gesamte Projektdauer beschäftigt sein wird. So wird der Design-Manager vermehrt in der 1. Projekthälfte beschäftigt sein, der Construction Manager in der Installationsphase in der 2. Projekthälfte.

### 5.3.1 Verdichtung und Colocation<sup>42</sup>

Unter Colocation versteht man die Vorkehrung, die am meisten aktiven Projektmitglieder an einem Ort zu versammeln um gemeinsame Aktivitäten als Team zu verbessern.

Dieses Ziel<sup>43</sup> steht im diametralen Gegensatz zu der durch Zersplitterung der Aufgaben durch die in Kapitel 4.2.1 (3. Achse, 3-dimensionales Projektmanagement) beschriebenen geographischen Verteilung der in Dienstleistungsbereichen beschäftigten Ingenieure. Darüber hinaus führt die in 4.2.1 beschriebene Zersplitterung zur progressiven Vermehrung der Schnittstellen. Um diesem Schnittstellenproblem entgegenzuwirken und den Colocation-Forderungen nachzukommen, empfiehlt es sich Aufgaben aus den Disziplinen (Wertschöpfungskette, Funktionen [Verfahren, Maschinen]) abzuziehen und in einem Kern-Projektmanagement-Team zu vereinen. Man sollte einem derartigen Kernteam natürlich agile (iterativ, kommunikativ) Vorgangsweisen unterstellen.

Angeführtes Beispiel (5.3) betrifft ein "Turn-Key" Projekt (EPC – Engineering, Procurement, Construction), für eine Anlagenlieferung "nur" mit Unterstützungsleistungen während der Montage kann z.B. der Construction Manager entfallen.

### 5.3.2 Co-Manager in komplexen Projektteams

Für komplexe Projektsituationen (agentenbasiert, Wechselwirkungen nicht linear, Emergenz, global spürbare lokale Interaktionen, Attraktoren, offene Systeme, u.a. siehe 4.1) existiert eine Standardlösung der Teamstruktur, die auf wichtige Funktionalitäten hinweist. Eine Schlüsselposition darin hat neben dem Projekt-Manager ein Co-Projekt-Manager<sup>44</sup>. Neben funktionellen Bereichen mit entsprechender Aufgabenteilung zwischen Projekt-Manager und Co-Projekt-Manager, z.B. getrennte technische und kaufmännische Projektleiter, wird als Vorteil sehr oft der erzielbare Redundanzeffekt angeführt, daher bei Ausfall eines der beiden Manager, kann auf Grund der sehr komplexen Projektmaterie Informationsverlust vermieden werden. Diese Redundanzfunktion kann auch durch Teilung der Aufgaben zwischen z.B. Projektleiter, Design Manager und Supply Chain Manager erreicht werden.

---

42 Colocation, Project resource management, Project Management Body of Knowledge – Pmbok Guide, S340, 6. Auflage, Project Management Institute, Pennsylvania USA, 2017 [33]

43 Co-located CPM Project Teams, Chapter 11 – Complexity and Uncertainty in the Project Landscape, S358 ff, Effective Project Management – Traditional – Agile – Extreme – Hybrid, Robert K Wysocki, Indianapolis USA, 2019 [23]

44 The Complex Project Team, Chapter 4 – What Is a Collaborative Project Team, Ebenda S93ff

Das komplexe Co-Manager Projektteam<sup>45</sup>

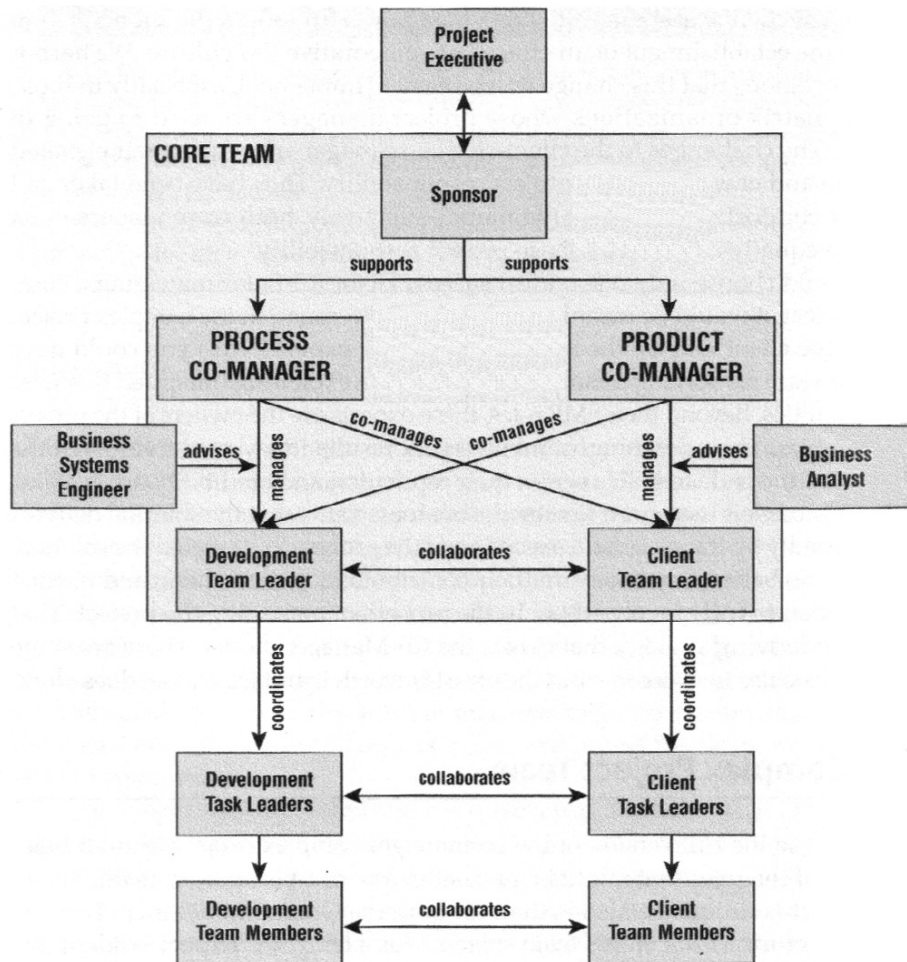


Illustration 23: komplexes Co-Manager Projektteam nach Robert K. Wysocki

Inwieweit ein Projekt einen Co-Manager in Form eines kaufmännischen Projektleiters erfordert, hängt von der allgemeinen rechtlichen Situation, dem Vertragsmodell, der Projektgröße und -dauer, der Projektstruktur (siehe auch „modulare Beschaffung“), den besonderen Eigenschaften des Kunden und der Intensität des Nachtragswesens (Change Order Management) ab.

### 5.3.3 Die Integration der Kunden-Organisation

Eine Besonderheit des komplexen Co-Manager-Projektteams nach Wysocki ist die Integration von Teilen der Kundenorganisation in die Projektorganisation. Dies kann bei Vorliegen bestimmter Voraussetzungen sinnvoll sein. Solche Voraussetzungen sind z.B.:

- Vorliegen der vertraglichen Vereinbarungen,
- der Bestand wird bei komplizierten Umbauprojekten durch Kundenpersonal gemanagt,
- komplizierten, vom Kunden zu planenden Baumaßnahmen.

<Skizze für Projektorganisation mit Kundenbeteiligung umseitig>

<sup>45</sup> The complex co-manager project team, Chapter 4 – What Is a Collaborative Project Team, S94, Effective Project Management – Traditional – Agile – Extreme – Hybrid, Robert K Wysocki, Indianapolis USA, 2019 [23]

komplexes Projektteam mit Kundenbeteiligung

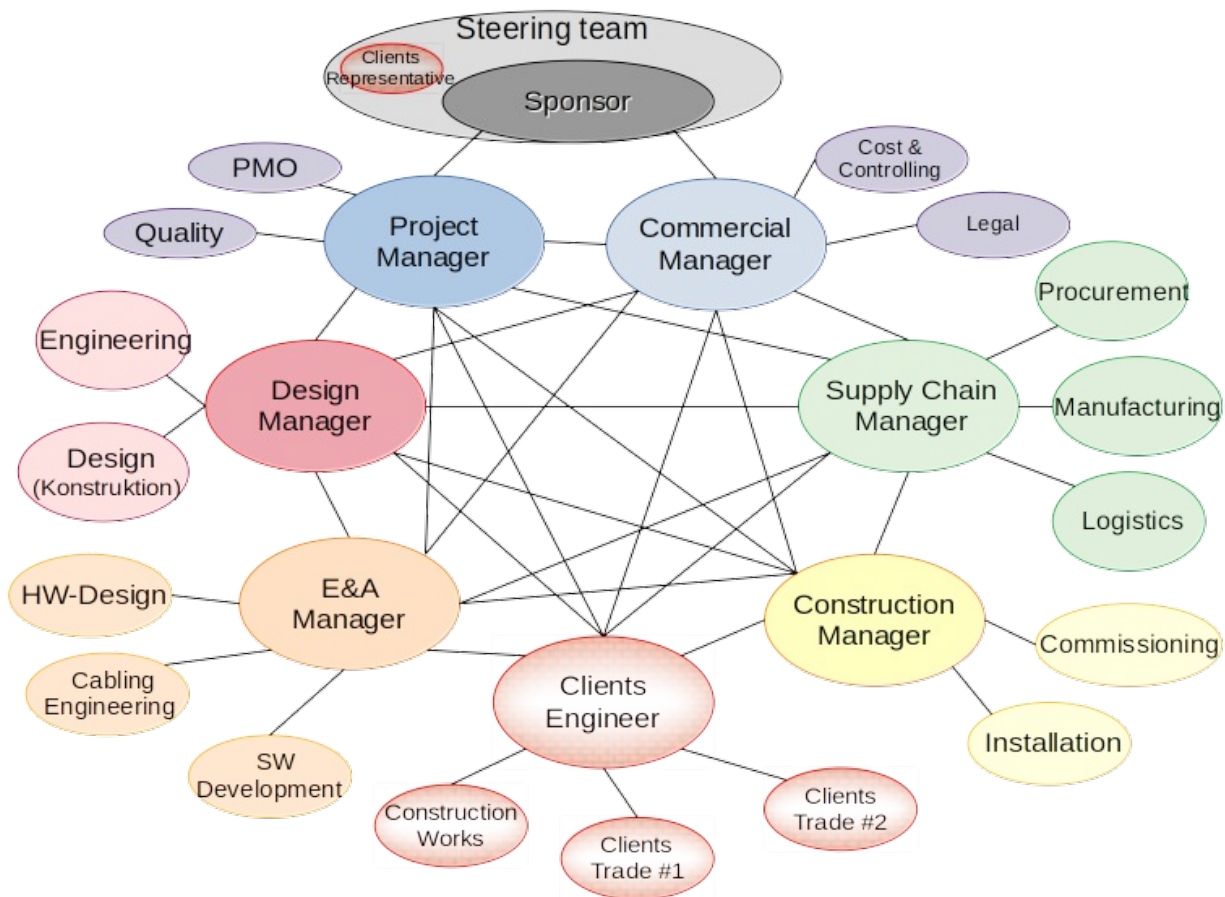


Illustration 24: komplexes Projektteam mit integriertem Kundenteam

Auch im Lenkungskomitee (Steering Committee) sind Kundenvertreter möglich. Ein Beispiel dafür sind sogenannte „Open-Book“ Verträge.

### 5.4 Agile Managementmethoden<sup>46</sup>

Der in Kapitel 4 beschriebene Komplexitätszuwachs, zunehmend unsichere Entscheidungssituationen sowie Markterfordernisse, die wesentlich schnelleres, agiles Handeln erforderten, verlangten einen neuen Zugang zur Abwicklung von Projekten, die agilen Projektmanagementmethoden (APMs) wurden geboren.<sup>47</sup>

APMs zeichnen sich durch empirische Steuerungsmethoden aus, die Entscheidungsfindung orientiert sich an den festgestellten Realitäten im laufenden Projekt. Es sind also regelmäßige und grundlegende Inspektionen erforderlich, die erforderliche Anpassungen unmittelbar in die Projektbearbeitung einbinden.<sup>48</sup>

<sup>46</sup> Agile vs. Waterfall: Differences in Software Development Methodologies - The Ultimate Reference for Project Managers, Jose Maria Delos Santos, <https://project-management.com/agile-vs-waterfall/>, abgerufen am 14.July 2021 [22]

<sup>47</sup> Part III - Complex Project Management, Chapter 11 – Complexity and Uncertainty in the Project Landscape, Effective Project Management (Traditional, Agile, Extreme, Hybrid), Seite 355, Robert K. Wysocki, Indianapolis, USA, 2020 [23]

## 5 Lösungsansätze

APMs sind teamorientiert und darauf ausgerichtet, funktionelle Anwendungen/technische Lösungen zu produzieren, die sich an Kundenanforderungen (in diesem Sinne interne und externe Kunden) orientieren, im Gegensatz zur Kaskadenmethode, die sich an der Spezifikation orientiert. Agilen Methoden ist ein inkrementeller und iterativer Ansatz (regelmäßige Inspektionen und Anpassungen) eigen, womit auf Änderungen und Unvorhergesehenes von vornherein Rücksicht genommen wird.

APMs (auch bezeichnet als iterative PMLC – Project Management Life Cycle Modelle) bestehen aus einer Anzahl von Prozessgruppen (Geschäftsprozesse), die innerhalb einer Iteration sequenziell wiederholt werden und die in eine Rückkoppelungsschleife münden, wenn die Iteration abgeschlossen ist.<sup>49</sup>

Iteratives PMLC Modell<sup>50</sup>:

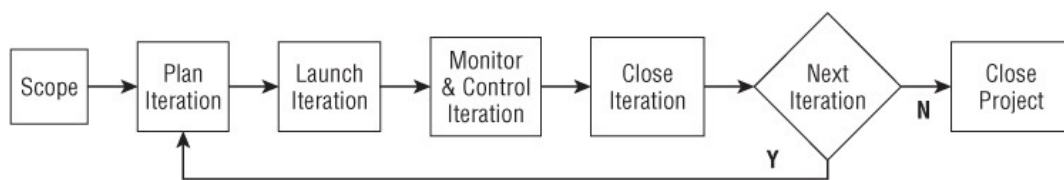


Illustration 25: Iteratives Project Management Life Cycle Modell

Allerdings sind APMs keine Erfindung des Digitalisierungszeitalters, spätestens 1986 führen Takeuchi und Nonaka in ihrem Artikel „The New New Product Development Game“ den Ausdruck „Scrum“ im Zusammenhang mit den Managementanforderungen bei der Produktentwicklung ein.<sup>51 52</sup>

Die APM–Entwicklungsgeschichte lässt sich folgendermaßen veranschaulichen:<sup>53</sup>

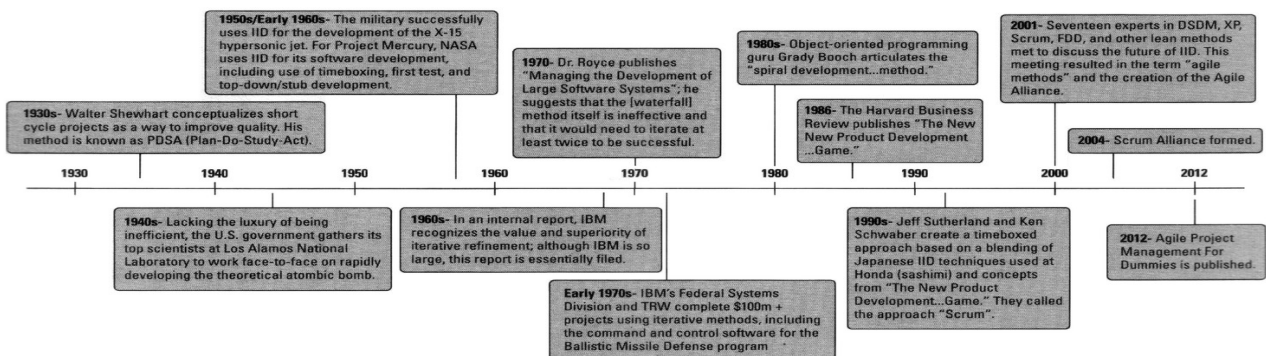


Illustration 26: APM - Entwicklungsgeschichte

48 How agile projects work, Chapter 1: Modernizing Project Management, Seite 15, Agile Project Management for Dummies, Mark C. Layton, New Jersey USA, 2012 [34]

49 Definition of the Iterative PMLC Modell, Part III - Complex Project Management, Chapter 12 – Agile Complex Project Management Models, Effective Project Management (Traditional, Agile, Extreme, Hybrid), Seite 382 ff, Robert K. Wysocki, Indianapolis, USA, 2020 [23]

50 Ebenda Seite 383

51 The New New Product Development Game, Hirotaka Takeuchi und Ikujiro Nonaka, Harvard Business Review, Ausgabe Januar 1986, Brighton Massachusetts USA [35]

52 Introducing Agile Project Management, Chapter 1: Modernizing Project Management, Seite 13, Agile Project Management for Dummies, Mark C. Layton, New Jersey USA, 2012 [34]

53 ebenda Seite 14

Trotzdem haben agile Vorgehensmodelle erst Anfang der 2000er Jahre im Zusammenhang mit Fragen der Software-Erstellung und der Formulierung des „Agile Manifesto“<sup>54</sup> eine entsprechende Formalisierung erhalten, sodass APM gemeinhin mit Softwareprojekten assoziiert wird.

Agil versus Kaskade :<sup>55</sup>

Neben dem inkrementellen und iterativen Ansatz (im Gegensatz zum linearen, sequenziellen Vorgehen bei der Kaskade) sind folgende Unterschiede kennzeichnend:

- APM ist in kurze Abschnitte aufgeteilt – sogenannte Sprints - , die Kaskade beinhaltet große Projektphasen,
- APM fokussiert auf die Zufriedenstellung von Kundenanforderungen, die Kaskade auf Ablieferung des Projektes,
- bei APM werden die Anforderungen kurzfristig (im Extremfall auf Tagesbasis) festgelegt, in der Kaskade nur einmalig am Projektanfang,
- Änderungen der Anforderungen sind bei APM jederzeit möglich, bei der Kaskade versucht man das zu vermeiden,
- Tests und Validierung werden bei APM parallel zur (im Zuge der) Entwicklung durchgeführt, bei der Kaskade nur am Ende des Entwicklungsprozesses,
- Testteams (oder einzelne Personen, die Tests durchführen) nehmen an den Vorgängen zu Änderung von Anforderungen teil, bei der Kaskade ist dies nicht der Fall,
- APM erfordert keinen Projektleiter (Lenkungsfunktionen können durch andere Rollen ausgeführt werden, z.B. Scrum Master, Product Owner), das Kaskadenmodell erfordert den Projektleiter als wesentliche Rolle in allen Projektphasen.

### 5.4.1 Die APM Grundlagen

Die Grundlagen von APM sind im agilen Manifesto<sup>56</sup>, der Erklärung der Wechselbeziehungen und den 12 Prinzipien zusammengefasst.

---

54 The Agile Manifesto, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin , Ken Schwaber, Jeff Sutherland, Dave Thomas, <https://agilemanifesto.org/iso/de/manifesto.html>, abgerufen am 14. July 2021, Tennessee USA, 2001 [36]

55 Agile vs. Waterfall: Differences in Software Development Methodologies - The Ultimate Reference for Project Managers, Jose Maria Delos Santos, <https://project-management.com/agile-vs-waterfall/>, abgerufen am 14.July 2021 [34]

56 The Agile Manifesto, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin , Ken Schwaber, Jeff Sutherland, Dave Thomas, <https://agilemanifesto.org/iso/de/manifesto.html>, abgerufen am 14. July 2021, Tennessee USA, 2001 [36]

### 5.4.1.1 *Das agile Manifesto*

Das agile Manifesto (wie auch die Bezeichnung „agil“) wurde bei bei einem Entwicklertreffen in den USA im Jahre 2001 formuliert. Die Grundsätze des agilen Manifests lauten:<sup>57</sup>

- Individuen und Interaktionen sind wichtiger als Prozesse und Werkzeuge,
- funktionierende Software ist wichtiger als umfassende Dokumentation,
- Zusammenarbeit mit dem Kunden ist wichtiger als Vertragsverhandlung,
- das Reagieren auf Veränderung ist wichtiger als das Befolgen eines Plans.

### 5.4.1.2 *Die Erklärung der Wechselbeziehungen (DOI – Declaration of Interdependence)*

Im Jahre 2005 wurde von einer Gruppe von Softwareentwicklern und Projektleitern ein Wertesystem für APM definiert, das als Zielsystem von APM betrachtet werden kann.<sup>58</sup> Das Wertesystem lautet:

- Erhöhung des Return on Investments durch Fokussierung auf Werteströme (in dem Sinne, dass permanent/kontinuierlich positiver Output produziert wird),
- Ablieferung verlässlicher Ergebnisse durch regelmäßige Interaktion mit den Kunden und Einbindung der Kunden in das „Product-Ownership“ (der Product-Owner ist eine Überwachungs-/Leitungsfunktion in einem APM-Team),
- die Einplanung von Unsicherheiten/Unwägbarkeiten und das Meistern dieser Unsicherheiten/Unwägbarkeiten durch iterative Vorgehensweisen, Antizipation und Anpassungsfähigkeit,
- das Freisetzen von Kreativität und Innovation durch Bekenntnis zu der Tatsache, dass Individuen letztendlich die Quelle von Wertschöpfung sind und die Herstellung von Umgebungsbedingungen die Kreativität und Innovation befördern,
- Leistungssteigerung durch Übernahme von Verantwortung durch die agierende Gruppe und gemeinsame Verantwortung für die Gruppen-Effektivität,
- Verbesserung von Effektivität und Verlässlichkeit durch situationsbedingte Anwendung von spezifischen Strategien, Prozessen und Praktiken.

---

57 The Agile Manifesto, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Ken Schwaber, Jeff Sutherland, Dave Thomas, <https://agilemanifesto.org/iso/de/manifesto.html>, abgerufen am 14. July 2021, Tennessee USA, 2001 [36]

58 DOI - Declaration of Interdependence, <https://web.archive.org/web/20180127094805/http://www.pmdoi.org/>, abgerufen am 16. November 2021, David Anderson, Sanjiv Augustine, Christopher Avery, Alistair Cockburn, Mike Cohn, Doug DeCarlo, Donna Fitzgerald, Jim Highsmith, Ole Jepsen, Lowell Lindstrom, Todd Little, Kent McDonald, Pollyanna Pixton, Preston Smith, Robert Wysocki, 2005 [37]

### 5.4.1.3 Die 12 Prinzipien von APM

Die Autoren des agilen Manifests haben die Grundsätze (wie in 5.4.1.1 beschrieben) in zwölf Prinzipien des agilen Projektmanagements konkretisiert, Zitat:<sup>59</sup>

- *Unsere höchste Priorität ist es, den Kunden durch frühe und kontinuierliche Auslieferung wertvoller Software zufrieden zu stellen.*
- *Heiße Anforderungsänderungen selbst spät in der Entwicklung willkommen. Agile Prozesse nutzen Veränderungen zum Wettbewerbsvorteil des Kunden.*
- *Liefere funktionierende Software regelmäßig innerhalb weniger Wochen oder Monate und bevorzuge dabei die kürzere Zeitspanne.*
- *Fachexperten und Entwickler müssen während des Projektes täglich zusammenarbeiten.*
- *Errichte Projekte rund um motivierte Individuen. Gib ihnen das Umfeld und die Unterstützung, die sie benötigen und vertraue darauf, dass sie die Aufgabe erledigen.*
- *Die effizienteste und effektivste Methode, Informationen an und innerhalb eines Entwicklungsteams zu übermitteln, ist im Gespräch von Angesicht zu Angesicht.*
- *Funktionierende Software ist das wichtigste Fortschrittsmaß.*
- *Agile Prozesse fördern nachhaltige Entwicklung. Die Auftraggeber, Entwickler und Benutzer sollten ein gleichmäßiges Tempo auf unbegrenzte Zeit halten können.*
- *Ständiges Augenmerk auf technische Exzellenz und gutes Design fördert Agilität.*
- *Einfachheit – die Kunst, die Menge nicht getaner Arbeit zu maximieren – ist essenziell.*
- *Die besten Architekturen, Anforderungen und Entwürfe entstehen durch selbstorganisierte Teams.*
- *In regelmäßigen Abständen reflektiert das Team, wie es effektiver werden kann und passt sein Verhalten entsprechend an.*

### 5.4.2 Motivation agile Vorgehensmodelle im Maschinen- und Anlagenbau einzusetzen

#### 5.4.2.1 Enge Kooperation erwünscht

Es ist ein enger, permanenter Datenaustausch zwischen den Beteiligten im Planungsprozess erwünscht. Die Arbeitsergebnisse werden nicht formalisiert am Ende der Bearbeitung an den nächsten Beteiligten weitergegeben, sondern permanent und informal ausgetauscht. Die Beteiligten werden zu Teammitgliedern im Scrum-Prozess und sitzen "sozusagen im selben Zimmer" auch wenn sie nur virtuell verbunden sind.

---

<sup>59</sup> The Agile Manifesto, Prinzipien hinter dem Agilen Manifest, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Ken Schwaber, Jeff Sutherland, Dave Thomas, <http://agilemanifesto.org/iso/de/principles.html>, abgerufen am 14. July 2021, Tennessee USA, 2001 [38]



5.4.2.2 *Zeitersparnis durch früheres Abliefern von Arbeitsergebnissen*

Die Illustration zur theoretischen Zeitersparnis wurde mittels Anpassung der Grafik aus Prozessverständnis<sup>60</sup> von Thorsten Phillip Klein erstellt.

Theoretische Zeitersparnis bei Verwendung von agilen Vorgehensmodellen:

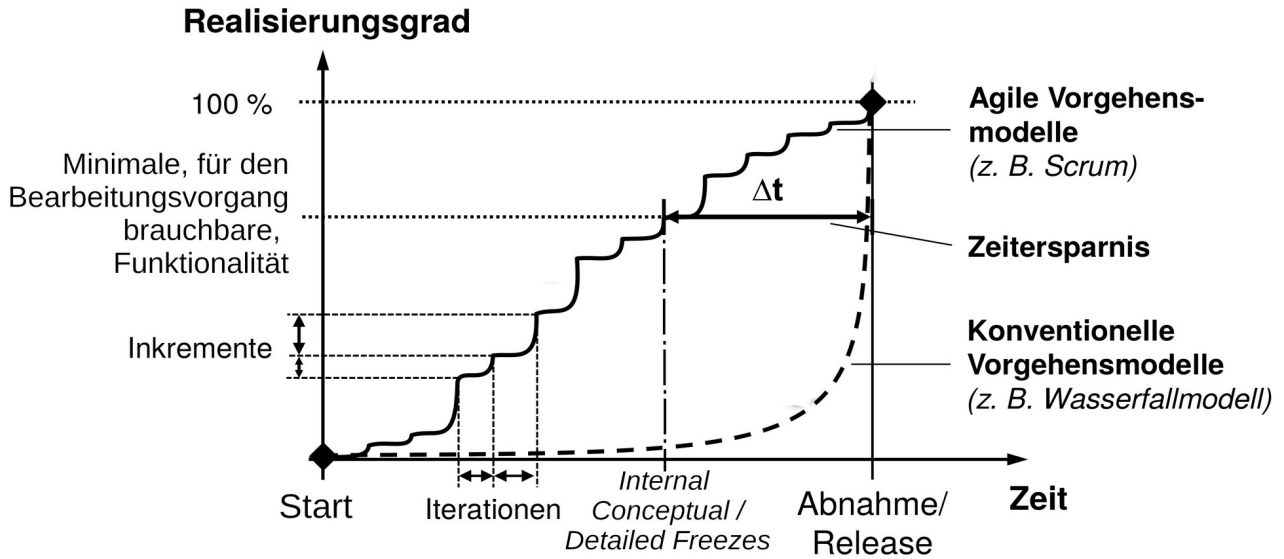


Illustration 27: potentielle Zeitersparnis bei einzelnen Aktivitäten unter Anwendung agiler Methoden

Der "Design Freeze" wird also aufgeweicht in interne konzeptionelle oder Detaillierungs-"Freezes".<sup>61</sup>

<der Rest der Seite bleibt aus redaktionellen Gründen frei>

60 3.3.2.1 Prozessverständnis, Agiles Engineering im Maschinen- und Anlagenbau, Seite 47, Dissertation von Thorsten Phillip Klein, München 2015 [39]

61 Definition of design freeze, The role of design freeze in product development, Seite 2ff, Tido Eger, Claudia Eckert and P John Clarkson, International conference on engineering design, ICED05, Melbourne 2005 [40]



## 5 Lösungsansätze

Insbesondere interessant wird diese Betrachtungsweise bei hintereinander folgenden Aktivitäten (Beispiel 3 hintereinander folgende Aktivitäten):

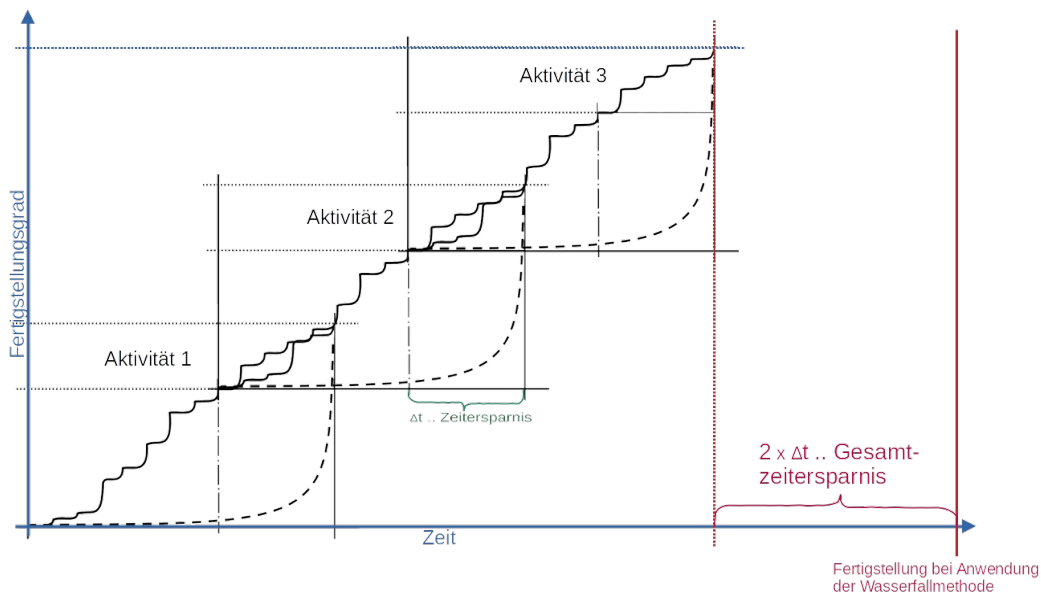


Illustration 28: potentielle Zeitersparnis bei hintereinander folgenden Aktivitäten

Es werden also "unfertige Ergebnisse" abgeliefert, die aber entsprechend den Anforderungen der weiteren Verwendung genügen.

### 5.4.2.3 Zeitersparnis durch "Ineinanderschieben" sequenzieller Arbeiten

Man könnte versucht sein eine Sequenz von Aktivitäten, die bei der zweiten (Folge-)Aktivität die Arbeitsergebnisse der ersten Aktivität voraussetzt, dadurch zu verkürzen, dass die Aktivitäten parallel stattfinden und die Übergabe oder Koordinierung der Ergebnisse dermaßen stattfindet, dass sich die Akteure während der Erstellung ihrer Arbeitsergebnisse laufend abstimmen und somit ein koordiniertes Gesamtprodukt zustande bringen.

<der Rest der Seite bleibt aus redaktionellen Gründen frei>

Grundgedanke:

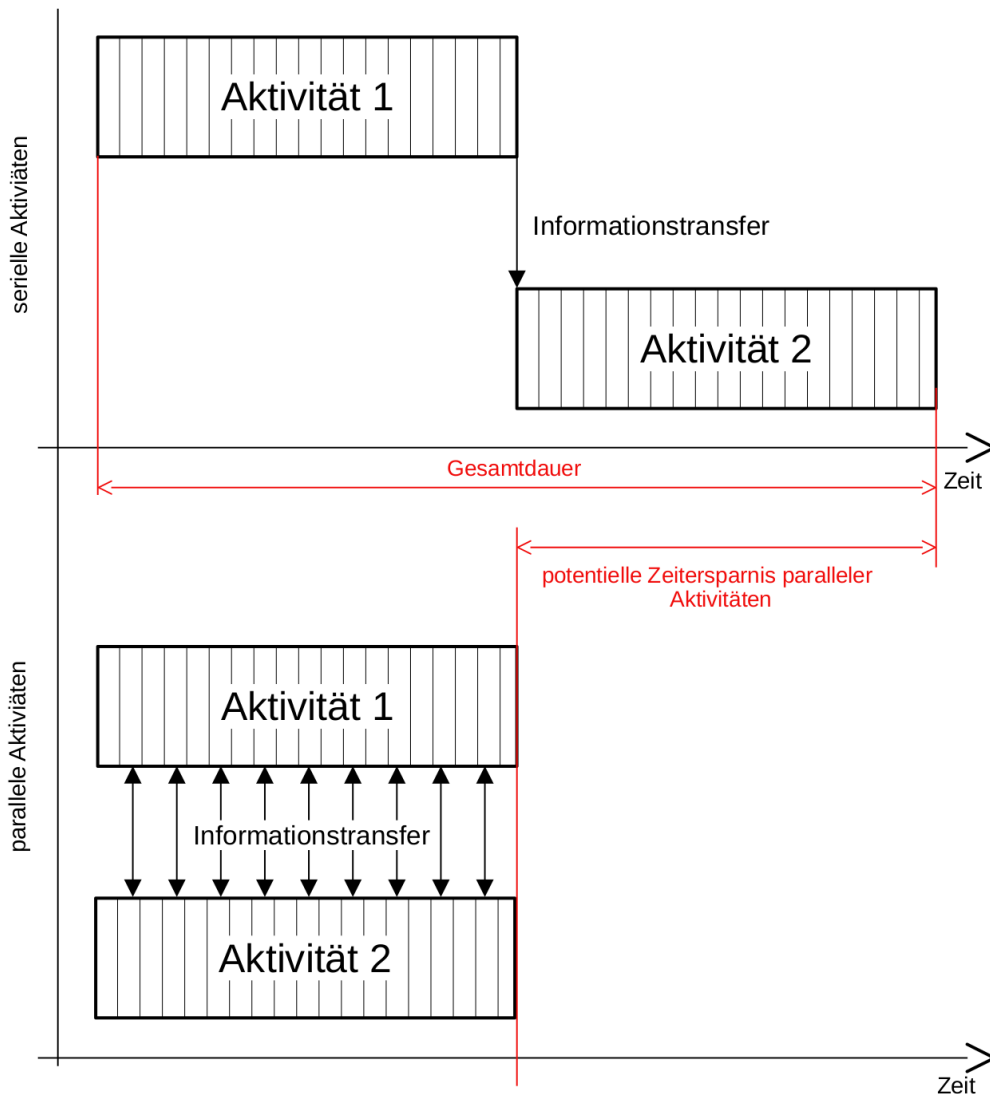


Illustration 29: potentielle Zeitersparnis durch Ineinanderschieben sequenzieller Aktivitäten

Diese Methode wird nicht bei allen Aktivitäten möglich sein, auch wird nicht die komplette Zeitersparnis realisierbar sein; Aktivitäten werden zum Beispiel einen gewissen Vorlauf benötigen. Allerdings ist diese Vorgangsweise gerade bei vielen ingenieurtechnischen Arbeiten (Softwareerstellung kann man, wenn man will, dazu zählen) bestens geeignet gute Arbeitsergebnisse bei kurzen Bearbeitungszeiten zu erzielen. Diese Sichtweise führt direkt zum teamorientierten, iterativen Arbeiten; es ist sozusagen die "atomistische Sicht" agiler Vorgehensmodelle.

#### 5.4.2.4 *das Arbeitsergebnis ist nicht linear vorausplanbar*

Obwohl als Unsicherheit und Risikofaktoren unerwünscht, so treten in der Praxis bei der Planung zukünftigen Handels neben rein kommerziellen Unschärfen auch technische Unsicherheitsfaktoren auf, die ein stufenweises und iteratives Vorgehen wünschenswert machen. Diese lassen sich folgendermaßen zusammenfassen:

## 5 Lösungsansätze

- Change Prozesse: Es werden in einem laufenden Prozess neue (technologische) Verfahren, neue Partner (siehe 4.2.2 modulare Beschaffung), ablauf-organisatorische Änderungen oder neue technische Entwicklungen eingeführt.
- Die technische Lösung muss erst erarbeitet werden: das findet im Grunde genommen bei jeder verfahrenstechnischen Anlage statt, die in der kompletten technischen Ausprägung erst während der Projektbearbeitung durchgeplant wird. Allerdings variiert der Grad der planerischen Unschärfe, Gründe dafür sind z.B.:
  - die projektierte Lösung ist schwer skalierbar, es müssen z.B. während der Projektbearbeitung noch genauere Untersuchungen angestellt werden, ob die geforderten Parameter mit einem Aggregat erreicht werden oder ein zweites, zusätzliches Aggregat erforderlich wird,
  - bei Arbeitsstart sind noch nicht alle wesentlichen Einflussparameter und Randbedingungen bekannt. Dies tritt vor allem bei Umbau- und Erneuerungsmaßnahmen auf – Rückbaumaßnahmen sind des Öfteren ein Unsicherheitsfaktor – insbesondere bei laufendem Betrieb. Es ist in solchen Fällen sehr oft nicht möglich bei Projektstart auf genaue Untersuchungsergebnisse zurückzugreifen, die genaue Analyse der Einflussparameter und Randbedingungen gehört zum Projekt,
  - man befindet sich mit seinem Projekt inmitten eines größeren Projektes, das wiederum Veränderungsmöglichkeiten an den Schnittstellen zum eigenen Projekt hervorbringt.

Zusammenfassend sollte man sich an dieser Stelle bewusst werden, dass trotz aller Vorkehrungen (Risikoanalyse, -management, zeitliche und finanzielle Contingencies, ..) für das zukünftige Projektgeschehen ein nicht zu unterschätzender Anteil an Entscheidungen unter Ungewissheit (Entscheidungsfindung bei Fehlen von Wahrscheinlichkeits-Vorstellungen<sup>62</sup>) zu fällen sind.

Einer ähnlichen Betrachtung liegt die Kategorisierung von Projekten oder Teilprojekten entsprechend dem Ausmaß der Unsicherheiten der Zieldefinition gegenüber dem Ausmaß der Unsicherheiten des Lösungsweges zu Grunde.

---

62 2. Grundbegriffe der Entscheidungsforschung, c) Entscheidungsmodelle, Industriebetriebslehre: Entscheidungen im Industriebetrieb, S55 ff, Edmond Heinen, 7. Auflage, Wiesbaden 1983 [41]

5.4.2.4.1 Eine intuitive Sichtweise der Projektlandschaft <sup>63</sup>

Die Projektlandschaft

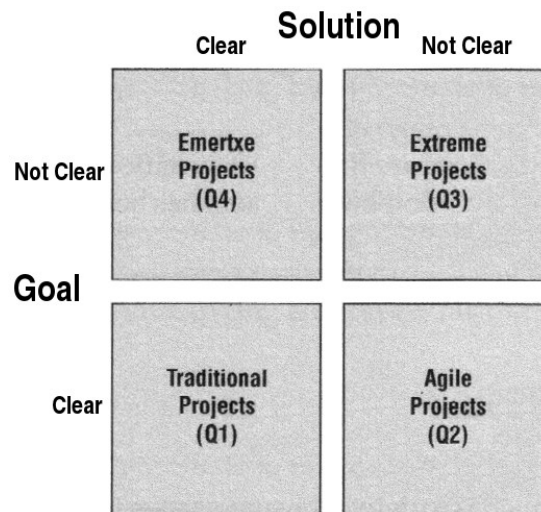


Illustration 30: Die vier Quadranten der Projektlandschaft

Erläuterung: Emergxe (gesprochen „Imerts“) im 4. Quadranten ist kein Tippfehler, sondern entstand durch die Buchstabierung von „Extreme“ Projects von hinten nach vorne und bezeichnet damit extreme Projekte, allerdings anders herum: Nämlich das Ergebnis ist schon verfügbar, die Anwendung ist aber noch nicht klar. Standardbeispiel dafür ist die RFID Technologie.

Komplexitätserhöhung und Unsicherheitsfaktoren führen ein Projekt in Richtung „extrem komplexer Projektmanagementaufgaben“ (xPMs).

Auswirkungen von Komplexitätsanstieg und Unsicherheit auf die Projektlandschaft<sup>64</sup>

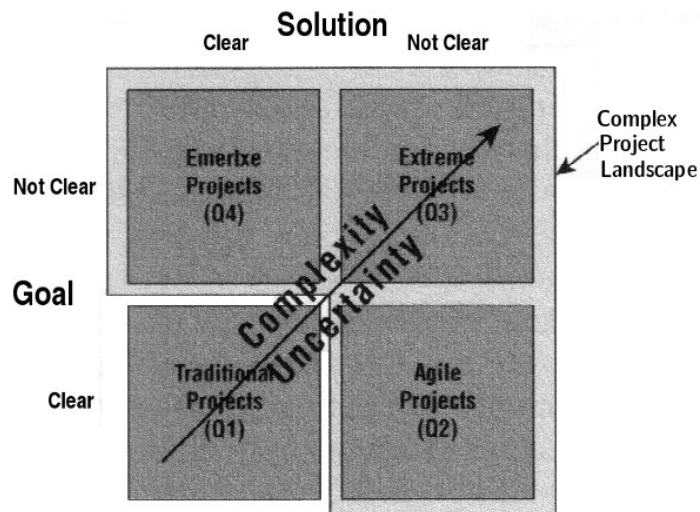


Illustration 31: Auswirkungen von Komplexitätsanstieg und Unsicherheit

Obwohl die Verwendung von Instrumenten des xPMs, insbesondere Organisationsfragen (siehe Kapitel Verdichtung), ohne weiteres sinnvoll ist, wird man im Maschinen- und Anlagenbau

<sup>63</sup> An Intuitive View of the Project Landscape, Chapter 1 – What is a Project, Effective Project Management (Traditional, Agile, Extreme, Hybrid), Seite 8 ff, Robert K. Wysocki, Indianapolis, USA, 2020 [23]

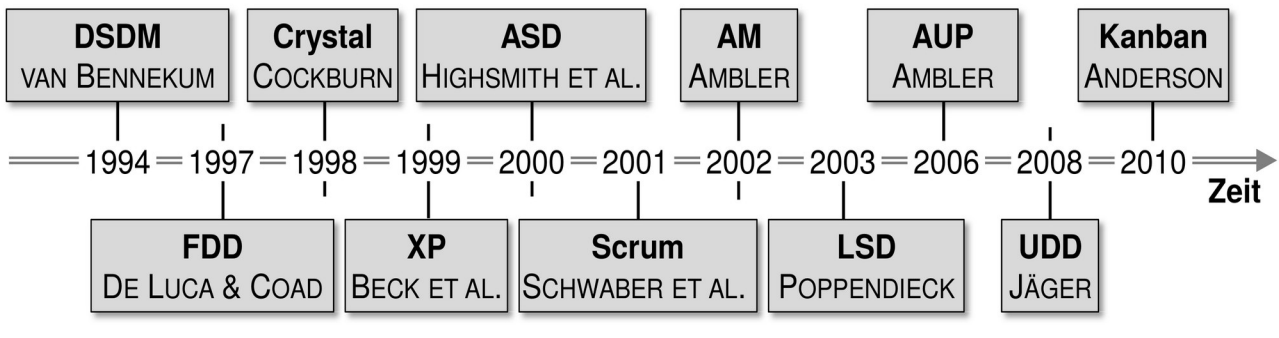
<sup>64</sup> Ebenda, Chapter 11 – Complexity and Uncertainty in the Project Landscape, S 362

versuchen sich bei der Lösung der Aufgabenstellungen auf den 1. Quadranten (TPMs) und den 2. Quadranten (APMs) zu konzentrieren.

### 5.4.3 Übersicht und Entwicklungsgeschichte agiler Vorgehensmodelle

Thorsten Philipp Klein hat in seiner Dissertation eine Übersicht über die wichtigsten, zur Zeit gängigen agilen Methoden/Vorgehensmodelle erarbeitet.

Zeitschiene zu Entwicklung der wichtigsten agilen Vorgehenstechniken<sup>65</sup>



Legende:

<b>Abkürzung</b>
<b>BEGRÜNDER</b>

- AUP – Agile Unified Process
- AM – Agile Modeling
- ASD – Adaptive Software Development
- DSDM – Dynamic System Development Method
- FDD – Feature Driven Development
- LSD – Lean Software Development
- UDD – Usability Driven Development
- XP – Extreme Programming

Illustration 32: Zeitlicher Ablauf der Entwicklung agiler Vorgehenstechniken

<Übersicht umseitig>

<sup>65</sup> Agile Vorgehensmodelle in der Softwareentwicklung, Dissertation zum Thema Agiles Engineering im Maschinen- und Anlagenbau, Seite 44, Thorsten Philipp Klein, IWB - Institut für Werkzeugmaschinen und Betriebswissenschaften, TU München, 2015 [39]

Übersicht agile Vorgehensmodelle<sup>66</sup>

Jahr	Name	Kurzbeschreibung
1994	<b>Dynamic Systems Development Method (DSDM)</b> (DSDM Consortium 2015)	Framework für die Anwendung eines prototypischen Vorgehensmodells zur schnellen Entwicklung von Anwendungen unter Berücksichtigung definierter Prinzipien
1997	<b>Feature Driven Development (FDD)</b> (COAD ET AL. 1999)	Fünfphasige Prozessabfolge mit Best Practices für die iterative Entwicklung von Software mit Fokus auf die Phasen Design und Building
1998	<b>Crystal</b> (COCKBURN 1998)	Anwendungsspezifische Auswahl und Anpassung eines Prozessmodells anhand der Kriterien Teamgröße und Projektkritikalität
1999	<b>Extreme Programming (XP)</b> (BECK 1999)	Synthese an Ideen und Ansätzen aus bestehenden Methoden für Planungs- und Entwicklungsaktivitäten für Software
2000	<b>Adaptive Software Development (ASD)</b> (HIGHSMITH 2000)	Adaptiver Lebenszyklus für Softwareentwicklung inklusive Entwicklungsphilosophie für Änderungsmanagement
2001	<b>Scrum</b> (SCHWABER & BEEDLE 2001)	In der Softwareentwicklung verbreitet eingesetztes Framework für Systementwicklungen und Projektmanagement
2002	<b>Agile Modeling (AM)</b> (AMBLER 2002)	Werte, Prinzipien und Methoden zur effizienten Modellierung in Kombination mit agilen Vorgehensmodellen
2003	<b>Lean Software Development (LSD)</b> (POPPENDIECK & POPPENDIECK 2003)	Überführung des Lean-Gedankens aus der Produktion (Toyota Produktionssystem) und der IT in sieben Prinzipien zur Softwareentwicklung
2006	<b>Agile Unified Process (AUP)</b> (AMBLER 2006)	Hybrider Modellierungsansatz des Rational Unified Process mit agiler Softwareentwicklung
2008	<b>Usability Driven Development (UDD)</b> (JAEGER 2008)	Iterativer Entwicklungsprozess mit Fokus auf die Benutzbarkeit eines Systems
2010	<b>Kanban</b> (ANDERSON 2010)	Softwareentwicklung unter Berücksichtigung von Durchlaufzeiten, Engpässen etc., vergleichbar mit der gleichnamigen Methode zur Produktionsprozesssteuerung

Table 1: Übersicht der wichtigsten agilen Vorgehensmodellen (Herausgeber der ursprünglichen Literatur in Klammern)

<sup>66</sup> Agile Vorgehensmodelle in der Softwareentwicklung, Dissertation zum Thema Agiles Engineering im Maschinen- und Anlagenbau, Seite 45, Thorsten Philipp Klein, IWB - Institut für Werkzeugmaschinen und Betriebswissenschaften, TU München, 2015 [39]

Das wichtigste und bekannteste daraus ist das “Scrum” Management.

### 5.4.4 Scrum (Deutsch Gedränge)

Scrum Management wurde im wesentlichen von Ken Schwaber und Mike Beedle<sup>67</sup> formuliert, wobei nur wenige Fixpunkte (genannt: The Core) definiert sind, nämlich:

- 4 Ereignisse
- 3 Artefakte
- 3 Rollen

Die zum Funktionieren erforderlichen Regeln sind im Scrum Guide beschrieben.

Der Scrum Guide<sup>68</sup> wurde von Ken Schwaber und Jeff Sutherland entwickelt und enthält die Definitionen von Scrum und eine kurze Beschreibung zur Scrum. Eine komplette Version des im Internet unter der „Attribution Share-Alike Liense“ frei verfügbaren Dokumentes befindet sich in Anlage 1.

Der prinzipielle Ansatz von Scrum als agile Methode ist empirisch, inkrementell und iterativ, der Langfristplan wird “Product Backlog” genannt, Detailpläne “Sprint Backlog”. Diese Detailpläne gelten jeweils für den nächsten Bearbeitungsschritt (Sprint). Sprints dauern üblicherweise 14 Tage.

Rollen: Als Hauptverantwortlicher fungiert der “Product Owner”<sup>69</sup>, er ist die Schnittstelle von den Stakeholdern(Kunden[externe, interne], Anwendern, Management) zum “Team”<sup>70</sup> von Entwicklern. Eine besondere Rolle spielt der “Scrum Master”<sup>71</sup>, obwohl nicht Teil des Entwicklerteams gibt er die Regeln für das Team vor, er hat eine beratende, dienende Funktion.

Als “Artefakte”<sup>72</sup> sind der Product Backlog (Anforderungen), das Backlog Refinement (Prozess zur Weiterentwicklung des Backlogs), der Sprint Backlog (der Plan für das nächste Projektinkrement – den Sprint) und das Product Increment (die Summe aller Backlog Einträge) zu nennen.

Die “Ereignisse”<sup>73</sup> beinhalten das Sprint Planning (Planung des nächsten Projektinkrementes), den Daily Scrum (tägliche kurze Abstimmung), den Sprint Review zur Überprüfung des Inkrements mit der eventuellen Anpassung des Backlogs und die Sprint Retrospektive am Ende des Sprints mit der Überprüfung der eigenen Arbeitsweise.

---

67 Agile Software Development with Scrum (Agile Project Management with Scrum), Ken Schwaber, Mike Beedle, Prentice Hall, Upper Saddle River, USA, 2007 [42]

68 Scrum Guides, Scrumguide.org, <https://scrumguides.org/>, abgerufen am 18.11.2021, Ken Schwaber, Jeff Sutherland, November 2020 [43]

69 Der Product Owner, Agile Software Development with Scrum (Agile Project Management with Scrum), Ken Schwaber, Mike Beedle, Seite 55ff, Prentice Hall, Upper Saddle River, USA, 2007 [42]

70 Das Team, ebenda Seite 103ff,

71 Der ScrumMaster, ebenda Seite 27ff,

72 Scrum-Artefakte, ebenda Seite 10ff,

73 Anhang A – Regeln, ebenda, Seite 135ff

Natürlich ist die Methode zur Verwendung in der Maschinen- und Anlagenbauplaung anzupassen so wie praktisch viele Unternehmen, die diese Technik anwenden, die im Scrum Guide vorgegebenen Elemente für Ihre Bedürfnisse zuschneiden.

### 5.4.5 Die Integration von kaskadierten und agilen Methoden führt zum hybriden Projektmanagement

Die Zusammenführung der Teilprozesse nach den Vorstellungen von Wysocki (siehe Kapitel 5.2 – Prozessoptimierung), wie in Abbildung/Illustration 17 dargestellt, ergibt unter Berücksichtigung der 3 Haupttypen von Swimlanes (5.2.1 bis 5.2.3) nachfolgend dargestelltes Ergebnis.

#### 5.4.5.1 Die Zusammenführung der Swimlanes

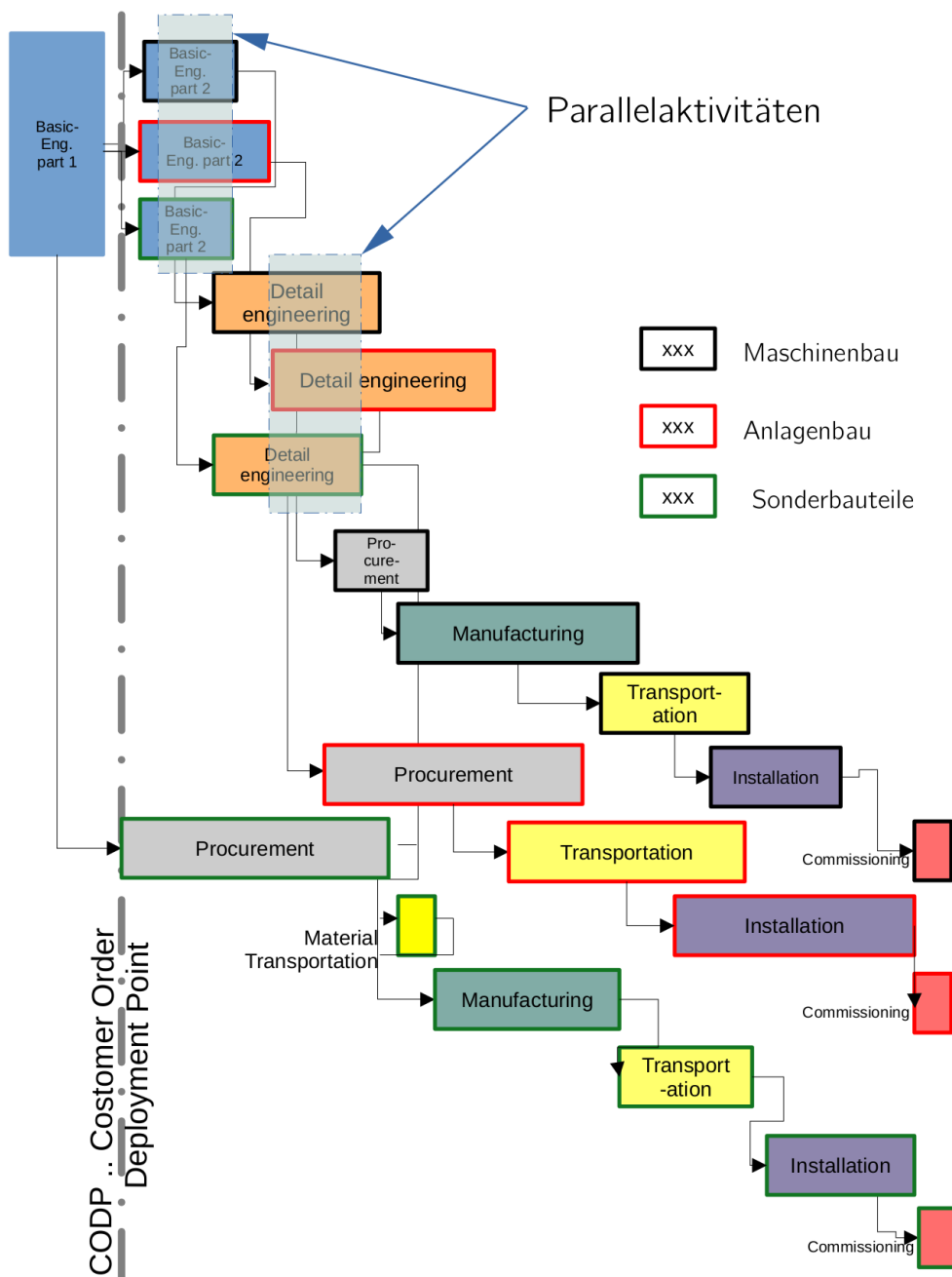


Illustration 33: Die Zusammenführung der wesentlichen Ablauftypen



## 5 Lösungsansätze

Dabei werden Parallelaktivitäten, die aber Koordinierung untereinander erfordern, ersichtlich. Diese Konfiguration weist auf komplexe Projektsituationen hin (Ziel ist klar, Lösungsweg nicht). Bei der Auswahl einer adäquaten Methode zur Abarbeitung dieser komplexen Aktivitäten empfiehlt es sich, die Vorgangsweisen des Project Management Life Cycle Models anzuwenden.<sup>74</sup>

### 5.4.5.2 Die grundsätzlichen Möglichkeiten entsprechend des PMLC (Project Management Life Cycle Model)

Unter PMLC – Project (Management) Life Cycle, Projektlebenszyklus – versteht man die Abfolge der Phasen, die ein Projekt (oder Projektabschnitt) durchläuft vom Start bis zur Beendigung<sup>75</sup>. Sie bestehen z.B. typischerweise aus Definitionsphase, Planungsphase, Teambildung, Kommunikation und Überwachung (laufen anderen Phasen parallel), Bewertung und Abschluss.<sup>76</sup>

Allerdings interessiert hier nicht der gesamte Lebenszyklus eines Projektes sondern ein isolierter Teilablauf für den eine geeignete Vorgangsweise gesucht wird. Dieser Teilablauf wird sozusagen wie ein eigenes Projekt behandelt, mit Definition, Start- und Ausführungsphase. Die Vorgangsweise ist analog zu der in 5.4.2.4.1 beschriebenen Projektlandschaft, angewandt auf einzelne Projekt-Teilbereiche.

Auswahl der PMLC Modelle nach der Position in der Projektlandschaft<sup>77</sup>

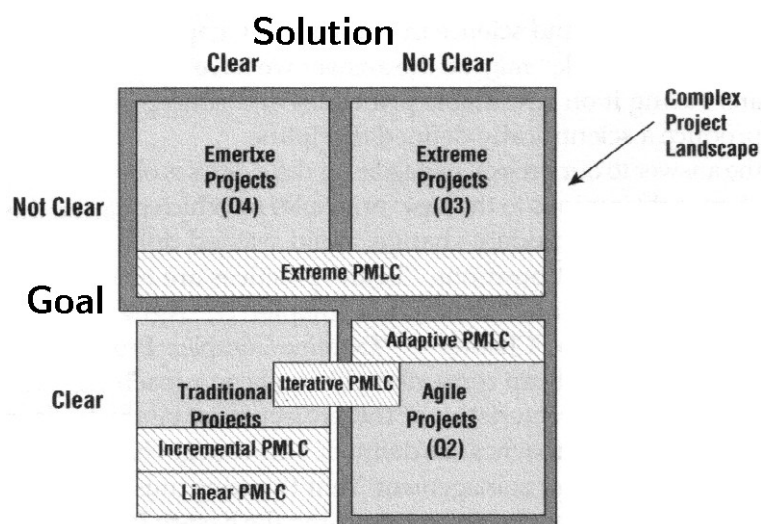


Illustration 34: Das PMLC Modell wird von der Position in der Projektlandschaft bestimmt

74 A Robust Hybrid PMLC Model, Chapter 14: Hybrid Project Management Framework, Effective Project Management (Traditional, Agile, Extreme, Hybrid), Seite 407 ff, Robert K. Wysocki, Indianapolis, USA, 2020 [23]

75 Part 3 – Glossary, Project Management Body of Knowledge – Pmbok Guide, S716, 6. Auflage, Project Management Institute, Pennsylvania USA, 2017 [33]

76 Essential project activities, Project skills, S14ff, Sam Elbeik und Mark Thomas, Woburn – Massachusetts, USA, 1998 [44]

77 Part 1 – Understanding the Project Management Landscape, Chapter 2: What is Project Management ?, Effective Project Management (Traditional, Agile, Extreme, Hybrid), Seite 39, Robert K. Wysocki, Indianapolis, USA, 2020 [23]

Folgend die 5 grundsätzlichen Projektmanagement–Lebenszyklus Konzepte:<sup>78</sup>

### 5.4.5.2.1 Das lineare PMLC Modell

Dabei handelt es sich um eine Implementierung der Wasserfall-Methode; einzelne Arbeitsschritte – Pakete – werden sequenziell abgearbeitet.

### 5.4.5.2.2 Das inkrementelle PMLC Modell

Es ist ähnlich wie das lineare Modell, allerdings wird das Ergebnis stückweise abgeliefert; z.B: Entwurf, vorläufiges Ergebnis, finales Ergebnis. Dementsprechend teilt sich der Arbeitsablauf in Inkremente auf. Zum Ende eines Inkrements wird überprüft, ob das Zwischenergebnis ausreicht, um das nächste Inkrement zu starten. Die inkrementelle Methode wird auf jeden Fall zu den traditionellen Projektmanagementmethoden (TPM) gezählt, weil das anzustrebende Endergebnis vollkommen klar ist.

### 5.4.5.2.3 Das iterative PMLC Modell

Ist man sich über eine oder mehrere Einzelheiten der anzustrebenden Lösung nicht im Klaren, dann sollte man ein iteratives Modell anwenden. Dazu gehören Scrum und DSDM (Dynamic Systems Development Method); siehe auch 5.4.3 – Übersicht agile Methoden. Als Unterschied zum inkrementellen Ansatz wird hier nach jedem Arbeitsschritt aus den erarbeiteten Bestandteilen eine Demonstrationslösung zusammengebaut und mit dem Kunden/Entscheider (z.B. Process Owner) darüber entschieden, ob das Ergebnis brauchbar ist und falls ja, dieses dann dem Prototypen einverleibt und dann wird zum nächsten Schritt übergegangen. Dieses Modell wird schon als agile Vorgehensmethode bezeichnet.

Es kann in komplexen Problemlösungssituationen im Anlagen- und Maschinenbau zur Anwendung kommen, insbesondere wenn mehrere Disziplinen gleichzeitig (parallel) an einer bestimmten Lösung arbeiten.

### 5.4.5.2.4 Das adaptive PMLC Modell

Es sollte angewendet werden, wenn wesentliche Teile der anzustrebenden Funktionalität unklar sind, im Extremfall sehr wenig über die zu findende Lösung bekannt ist. Die anzuwendenden Methoden beinhalten Ideenfindungsmodelle, Problemdefinitionsmethoden udgl. und haben mit der konkreten/auftragsbezogenen Projektarbeit im Anlagen- und Maschinenbau eher nichts zu tun.

### 5.4.5.2.5 Das extreme PMLC Modell

Diese Kategorie von Modellen findet eher in der Forschungs- und Entwicklungswelt Anwendung, wenn Ziel und Lösung weitgehend unbekannt und nur nebulöse Vorstellungen davon existieren, was eigentlich zu entwickeln ist. Als dafür verwendete Methode wird oft INSPIRE (INitiate, SPeculate, Incubate, REview)<sup>79</sup> angeführt. Es ist ebenfalls in der konkreten/auftragsbezogenen Projektarbeit des Anlagen- und Maschinenbaus nicht anzutreffen.

---

<sup>78</sup> Part 1 – Understanding the Project Management Landscape, Chapter 2: What is Project Management ?, Effective Project Management (Traditional, Agile, Extreme, Hybrid), Seite 39, Robert K. Wysocki, Indianapolis, USA, 2020 [23]

<sup>79</sup> Part 3 – Complex Project Management, Chapter 15: Comparing TPM and CPM Models, Ebenda S518 ff

### 5.5 Koordinierungsmethoden

Für parallele und agile Bearbeitungsmethoden sind Koordinierungstechniken erforderlich, die Arbeitsergebnisse abstimmen. Diese Koordinierungstechniken sind insbesondere dann wichtig, wenn Mitglieder eines (z.B.) Scrum-Teams an verschiedenen Orten sitzen und sich die jeweiligen Arbeitsergebnisse untereinander beeinflussen, z.B. Kollisionen bei der Verrohrungsplanung. Diese Koordinierungstechniken setzen auf agilen Projektmanagement-Methoden entsprechend dem PMLC (Project Management Life Cycle Models – siehe 5.4.5.2) auf.

Eine mögliche Koordinierungsmethode wurde als agile Technik aus der Methode der “Continuous Integration” der Software-Entwicklung abgeleitet. Nach Martin Fowler ist Continuous Integration eine Software Entwicklungspraxis, bei der Entwickler ihre Arbeit regelmäßig in das Gesamtprojekt integrieren.<sup>80</sup>

Bei der Software-Entwicklung spricht man von zumindest täglicher Integration in das Gesamtprojekt, im Maschinen- und Anlagenbau hat sich eine Periodendauer von einer Woche bis 14 Tage als sinnvoll erwiesen.

Ein Iterationsschritt läuft bei Design- und Konstruktionsarbeiten unter Anwendung von Continuous Integration folgendermaßen:

- Nach Ablauf der vorgegebenen Bearbeitungszeit liefern alle Beteiligten ihre Zwischenergebnisse beim “Konstruktionskoordinator” ab.
- Falls die Arbeitsergebnisse nicht schon im Format des “Repository systems” vorhanden sind (z.B. Files für den AutoDesk Inventor®), werden diese in ein Austauschformat für 3D Konstruktionsdaten konvertiert. Ein typischer Vertreter für ein 3D Austauschformat sind STEP (Standard for the Exchange of Product Data) Dateien entsprechend ISO 10303-21:2016(Industrial automation systems and integration — Product data representation and exchange — Part 21: Implementation methods: Clear text encoding of the exchange structure)<sup>81</sup>.
- Sämtliche Arbeitsergebnisse werden so in ein aktuelles Modell zusammengebaut.
- Dieses Modell wird in ein komprimiertes Format überführt, das allen Beteiligten zur Verfügung steht. Als Beispiel für ein komprimiertes Format ist das AutoDesk “Navisworks®” (\*.nwd) Format hier angeführt; Navisworks® ist ein Programmpaket zur 3D Entwicklungsüberprüfung, das auf ein stark komprimiertes Dateiformat zurückgreift, so werden zum Beispiel Modelle aus dem AutoDesk Inventor® von 12 GByte auf kleiner 500 MByte geschrumpft. Obwohl das Programmpaket kostenpflichtig ist, steht eine Testversion für 30 Tage kostenfrei zur Verfügung. Der

---

80 Continuous Integration, <https://martinfowler.com/articles/continuousIntegration.html>, abgerufen am 25.7.2021, Martin Fowler, Thoughtworks, 200 E. Randolph 25<sup>th</sup> floor, Chicago, IL 60601-6501, USA , 2006 [45]

81 STEP-file, ISO 10303-21 -- Industrial automation systems and integration -- Product data representation and exchange -- Part 21: Implementation methods: Clear text encoding of the exchange structure, Digital Preservation at the Library of Congress , <https://www.loc.gov/preservation/digital/formats/fdd/fdd000448.shtml>, abgerufen am 26.7.2021 [46]

## 5 Lösungsansätze

offizielle Internetauftritt befindet sich hier:

<https://www.autodesk.com/products/navisworks/overview><sup>82</sup>. Dieses Modell steht allen Beteiligten für die weitere Bearbeitung für den nächsten Iterationsschritt zur Verfügung.

- Inwieweit die Kollisionsprüfung als Analogie zum Build-Prozess bereits beim Zusammenbau des Modells stattfindet, oder erst im nächsten Iterationsschritt, ist Frage der Arbeitsorganisation im konkreten Projekt. Zur Erklärung: Der Build-Prozess bei der Softwareentwicklung ist der Vorgang aus einem geänderten Projektstamm (Repository) wieder ein lauffähiges Programm zu erzeugen.

Continuous Integration, früher referenziert als “Booch Method” wurde im Zuge der Entwicklung von Werkzeugen zur objektorientierten Programmierung von Grady Booch erstmals 1992 vorgestellt.<sup>83</sup>

Der Gleichschritt bei der Einführung der Vorgehensmodelle zur Einführung von Programmiermethoden (Engineeringmethoden) sei an dieser Stelle erwähnt:

Strukturierte Programmierung → Wasserfall-Modell

Objektorientierte Programmierung → Agile Management-Methoden

---

82 Navisworks® - 3D model review software for architecture, engineering, and construction, <https://www.autodesk.com/products/navisworks/overview>, abgerufen am 26.7.2021 [47]

83 Chapter 6 – The Process, Seite 272 ff, Object-oriented analysis and design with applications, 3. Ausgabe (2007), Grady Booch, Redwood City, USA, 1994 [48]

# 6 Nachhaltigkeitsaspekte

Nachdem das Thema Projektmanagement im Maschinen- und Anlagenbau rein dienstleistungsbezogen, quasi als Hilfsdisziplin zur Erstellung von physischen Werten (Maschinen- und Anlagen) zu betrachten ist, kann man davon ausgehen, dass die erstellten Maschinen und Anlagen sich durch Veränderung der Vorgehensmodelle nicht verändern. Geht man weiters davon aus, dass die Managementmethode selbst (vom Energieverbrauch der Datenübertragungen abgesehen) keine ökologischen Auswirkungen hat, schließe ich ökologische Aspekte aus den Betrachtungen zur Nachhaltigkeit aus.

## 6.1 Ökonomische Nachhaltigkeit

Die Änderung der Managementmethoden (Einführung agiler Methoden) ist sicherlich mit dem Ziel "Erhöhung der Effektivität, Senkung der Kosten" getrieben. Somit kann zweifelsfrei die Verbesserung der ökonomischen Nachhaltigkeit als Kernziel betrachtet werden.

Besondere Aufmerksamkeit sollen in diesem Kontext Globalisierungsfragen gewidmet werden:

Globalisierungstendenzen im Maschinen- und Anlagenbau entstehen durch eine entsprechende Strategie zur Erreichung von Wettbewerbsvorteilen (Kostenvorteilen). Daraus ergibt sich ein gewisser Druck den Fertigungsbereichen auch Dienstleistungsbereiche folgen zu lassen.

Dessen ungeachtet gibt es analog zur Softwareindustrie Tendenzen, die Leistungserbringung im Dienstleistungsbereich per se zu globalisieren. Treiber dieser Tendenzen sind u.a.:

- preiswertere Ingenieursleistungen in anderen Ländern/Regionen,
- Einsatz freier/ungebundener Mitarbeiter, Stichwort digitale Nomaden,
- Spezialisierung von Dienstleistungsunternehmen (Z.B. Konstruktionsbüros) auf bestimmte Aufgabengebiete und damit effektivere Leistungserbringung.

## 6.2 Soziale Nachhaltigkeit

Francesco Albarosa und Rafael Valenzuela Musura entwickelten in ihrer Masterarbeit ein Modell zur Einschätzung der Auswirkungen von agilen Projektmanagementmethoden auf die soziale Nachhaltigkeit.<sup>84</sup>

Soziale Nachhaltigkeit wurde in dieser Studie mit den Faktoren Gesundheit, Einflussmöglichkeiten, Kompetenz, Sinnhaftigkeit, Vertrauen, Selbstorganisation gemessen.

Das Ergebnis der Studie ist, dass agile Projektmanagementmethoden insbesondere einen positiven Einfluss auf die Faktoren Gesundheit, Einflussmöglichkeiten und die Fähigkeit neue Kompetenzen zu erwerben, haben. Darüber hinaus wurde festgestellt, dass agile

---

<sup>84</sup> Social Sustainability Aspects of Agile Project Management, Francesco Albarosa, Rafael Valenzuela Musura, Seite 67, Umea University, Sweden, 2016 [49]

## 6 Nachhaltigkeitsaspekte

Projektmanagementmethoden im Kontext dieser Studie keinen negativen Einfluss auf die oben genannten 5 Faktoren zur Einschätzung der sozialen Nachhaltigkeit haben. Somit können agile Projektmanagementmethoden als Vorgehensmodelle betrachtet werden, die die soziale Nachhaltigkeit fördern.

Grafik zum Modell von Francesco Albarosa und Rafael Valenzuela Musura umseitig ..

<der Rest dieser Seite bleibt aus redaktionellen Gründen frei>

## 6 Nachhaltigkeitsaspekte

Modell zur Einschätzung der Auswirkungen von agilen Projektmanagementmethoden auf die soziale Nachhaltigkeit nach Francesco Albarosa und Rafael Valenzuela Musura.<sup>85</sup>

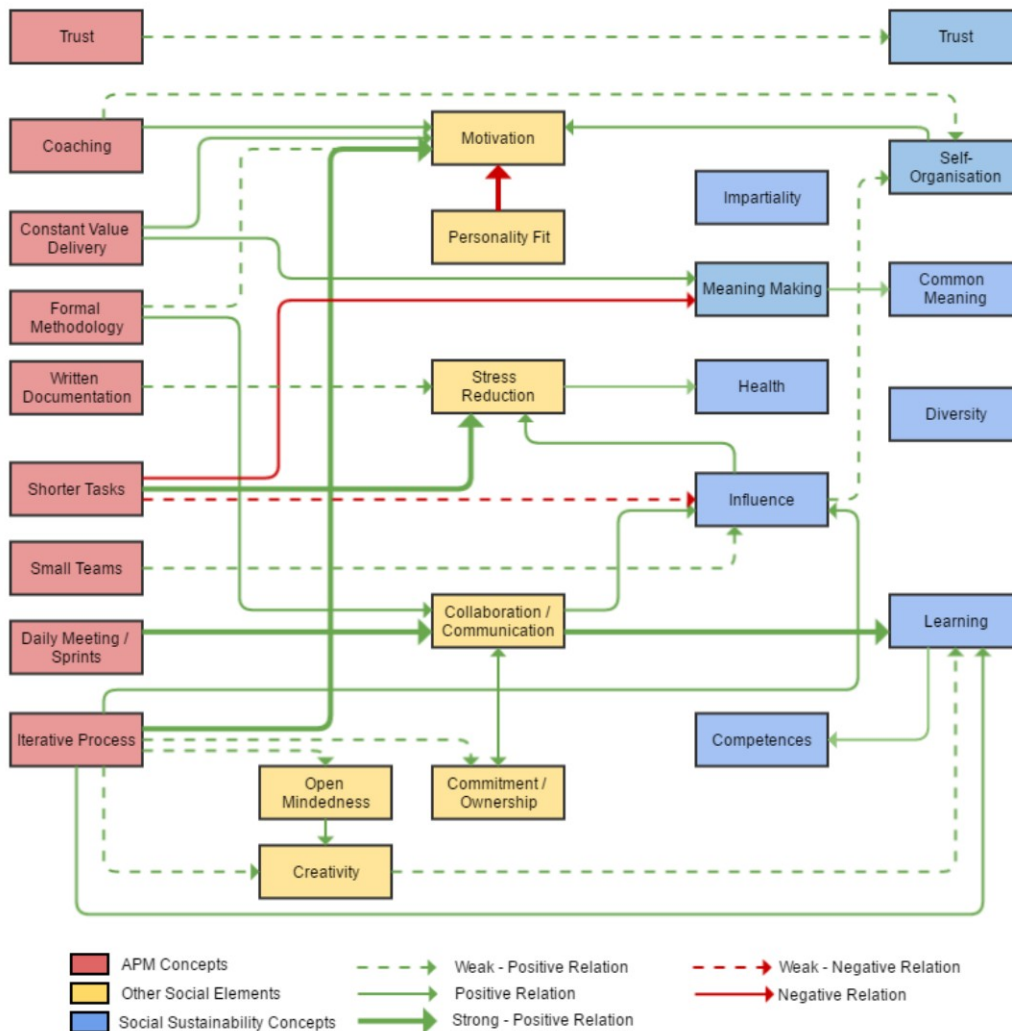


Illustration 35: Modell zur Einschätzung der Auswirkungen von agilen Projektmanagementmethoden auf die soziale Nachhaltigkeit nach Francesco Albarosa und Rafael Valenzuela Musura

85 5.3 The Model: A Relations Network Diagram, Social Sustainability Aspects of Agile Project Management, Francesco Albarosa, Rafael Valenzuela Musura, Seite 59, Umea University, Sweden, 2016 [49]

## 6 Nachhaltigkeitsaspekte



## Index

12 Prinzipien.....	35
Agent.....	11
agile Manifesto.....	33, 34
agiles Projektmanagement.....	
APM.....	21, 31, 32, 33, 34, 35, 41
Anlagenbau.....	1
Attraktor.....	12
Beschaffungskonzept.....	16
5.1.4 CAD.....	22
3D CAD.....	22
Change Prozess.....	39
Cloudlösung.....	22
Co-Manager.....	29
Colocation.....	29
Complex Project Management.....	
CPM.....	21
Cooperation Tool.....	21
Darnall Preston Complexity Index.....	
DPCI.....	13
Declaration Of Interdependence, Erklärung der Wechselbeziehungen.....	
DOI.....	34
Design Freeze.....	36
Detaillierungs-Freeze.....	36
digitale Nomaden.....	49
dreidimensionale Projektmatrix.....	14
Komplexitätsreduktion.....	15
Dynamic Systems Development Method.....	
DSDM.....	46
ECPM.....	
Effective Complex Project Management.....	24
Emergenz.....	12
Entmaterialisiertes Unternehmen.....	17
Entscheidungen unter Ungewissheit.....	39
EPC – Engineering, Procurement, Construction.....	17, 29
Extreme Project Management.....	
xPM.....	40
FTP Server.....	22
GIT.....	22
Globalisierung.....	5
hybrides Projektmanagement.....	21, 44
In-Plant.....	16, 17
Innere Harmonisierung.....	12

## Index

Interaktion.....	12, 34
Iteratives PMLC Modell.....	21, 32
Kaskadenmodell.....	
Kaskadenmethode.....	21, 32, 33
Kern-Projektmanagement-Team.....	28
komplexes System.....	11
Komplexitätsanstieg.....	10
Komplexitätsgrad.....	11, 13
Lebensdauer.....	21
Maschinen- und Anlagenbau.....	2
Maschinenbau.....	1
modulare Beschaffung.....	17
Modulare Beschaffung.....	16
Monotone.....	22
Nachhaltigkeit.....	49
ökologische Nachhaltigkeit.....	49
Ökonomische Nachhaltigkeit.....	49
Soziale Nachhaltigkeit.....	49
Offenes System.....	12
Open Source.....	22
Ordnungs- und Klassifizierungssystem.....	21
Betriebsmittelkennzeichnung.....	21
Betriebsmittelkennzeichnung.....	21
parallele Aktivitäten.....	
Parallelaktivität.....	19, 20
Pfadabhängigkeit.....	12
physische Transportbeziehungen.....	18
5.1.5 PPC.....	23
PPS.....	23
Produktionsplanung und Steuerung.....	23
Projektmanagement.....	5
Projektmanagement Lebenszykluskonzept.....	
PMLC.....	45
Project Management Life Cycle.....	13
Projektorganisation.....	27
Projektstrukturplan.....	7, 19
Prozessorientierung.....	24
Risikomanagement.....	18
Scrum.....	43, 46
Scrum Guide.....	43
Selbstorganisation.....	12, 49
Selbstregulation.....	12
sequenzielle Arbeiten.....	
serielle Aktivitäten.....	37
sequenzielles Vorgehensmodell.....	7, 8
serielle Aktivität.....	20
Soziale Nachhaltigkeit.....	49
Supply Chain Management.....	15

## Index

Swimlane.....	13, 24, 25, 26
Traditional Project Management.....	
TPM.....	21, 41, 46
Verdichtung.....	29
Kern-Projektmanagement-Team.....	29
Versionierungstools.....	22
3.2 Wasserfall.....	10
Kaskadenmodell.....	9
Waterfall Model.....	10, 19, 20
Wertschöpfungsstufe.....	14



## Literaturverzeichnis

- 1: Joerg Walter, Projektmanagement im Maschinenbau, abgerufen: 2020, URL = <https://www.projektmanagement-maschinenbau.de/>
- 2: Prof Dr Marion Steven, Multiple Betriebsgroessenvariation, abgerufen: 2018, URL = <https://wirtschaftslexikon.gabler.de/definition/multiple-betriebsgroessenvariation-38975/version-262395>
- 3: J C Wortmann, Efficiency of manufacturing systems, 1983, Plenum Press, 233 Spring Street, New York 10013, USA
- 4: Alexander Malkwitz, Norbert Mittelstaedt, Jens Bierwisch, Johann Ehlers, Thies Helbig, Ralf Steding, Projektmanagement im Anlagenbau, 2016, Springer Verlag, Heidelberger Platz 3, 14197 Berlin, Germany
- 5: Jörg Walter, Projektmanagement im Maschinenbau, abgerufen: 2021, URL = <https://bibliothek.projektmanagement-maschinenbau.de/>
- 6: Joel Koppelman, Primavera P6 CPM – Zeitplanung, Oracle Primavera P6, abgerufen: 2022, URL = <https://www.oracle.com/at/industries/construction-engineering/primavera-p6/>
- 7: D.G. Macolm, J.H. Roseboom, C.E. Clark, W. Frazer, Application of a Technique for Research and Development Program Evaluation, 1959
- 8: Charles D. Benson and William Barnaby Faherty, Negative Slack in Critical paths, Chapter 15 - Putting It All Together: LC-39 Site Activation, Moonport: A History of Apollo Launch Facilities and Operations, 1978, 2007
- 9: Herbert D. Benington, Robert R. Everett, Robert R. Everett, Production of Large Computer Programs, June 1956, 1983
- 10: O.-J. Dahl, E. W. Dijkstra, C. A. R. Hoare, Structured programming, 1972
- 11: Dr Winston W Royce, Managing the development of large software systems, 1970
- 12: Prof. Dr. Eberhard Feess, Komplexitaet, abgerufen: 2018, URL = <https://wirtschaftslexikon.gabler.de/definition/komplexitaet-39259/version-262672>
- 13: Jens Kowiak, Franz Fester, Claudius Hansch et al, Wiki komplexes System, abgerufen: 2022, URL = [https://de.wikipedia.org/w/index.php?title=Komplexes\\_System&oldid=217074045](https://de.wikipedia.org/w/index.php?title=Komplexes_System&oldid=217074045)
- 14: Naomi Caietti, How to Manage a Complex Project, abgerufen: 2016, URL = <https://www.projectmanager.com/blog/manage-a-complex-project>
- 15: Kathleen B. Hass, Living on the edge: project complexity management. Paper presented at PMI® Global Congress 2007— North America, 2007

## Literaturverzeichnis

- 16: Russell Darnall und John M. Preston, Project Management from Simple to Complex, 2016, University of Minnesota, Saylor Academy Open Textbooks, 330 Wulling Hall, 86 Pleasant Street S.E., Minneapolis, MN 55455, USA
- 17: Joachim Zentes, Bernhard Swoboda, Dirk Morschett, Fallstudien zum Internationalen Management, Grundlagen – Praxiserfahrungen – Perspektiven, 2008, Gabler, GWV Fachverlage GmbH, Wiesbaden, Germany
- 18: Kevin Schroer (responsible for the content), Beschaffungsstrategien, abgerufen: 2020, URL = <https://www.bwl-lexikon.de/wiki/beschaffungsstrategien/>
- 19: Univ. Prov. Dr. Michael Essig, Vorlesungsscript: Supply Chain Management I, 2021
- 20: Jonathan Hosie, Turnkey contracting under the FIDIC Silver Book, 2007
- 21: Johannes Staerk, Agiles Projektmanagement im Anlagen- und Maschinenbau, 2016
- 22: Jose Maria Delos Santos, Agile vs. Waterfall: Differences in Software Development Methodologies, abgerufen: 2021, URL = <https://project-management.com/agile-vs-waterfall/>
- 23: Robert K. Wysocki, Effective Project Management, Traditional, Agile, Extreme, Hybrid, 2019, John Wiley & Sons, Inc, 10475 Crosspoint Boulevard, Indianapolis, IN 46256, USA
- 24: Dr. S. Adler, A. Eckstein, S. Skrytutskyi, M. Eisentraeger, S. Moeser, Whitepaper, Betriebsmittelkennzeichnung in der Prozess- und Anlagenbau Industrie, 2017, IFF Fraunhofer, Sandtorstraße 22, 39106 Magdeburg, Germany
- 25: Cyrille Béraud (CEO), Jami Audio/Video/Conferencing, abgerufen: 2021, URL = <https://jami.net/>
- 26: Dr. John Bates (CEO), Document management with Doxis4® , abgerufen: 2021, URL = <https://www.sergroup.com/en/document-management.html>
- 27: Alex Queiroz, Anthony Cooper, ...., Monotone is a free distributed version control, abgerufen: 2021, URL = <https://www.monotone.ca/>
- 28: Junio Hamano and others, GIT-distributed-is-the-new-centralized (version control), abgerufen: 2021, URL = <https://git-scm.com/>
- 29: Ola Rollen (CEO Intergraph), Solid Edge 2D/3D-CAD-System, abgerufen: 1995, URL = <https://solidedge.siemens.com/en/>
- 30: Jaap Schekkermann, How to survive in the jungle of Enterprise Architecture Frameworks, 2004, Trafford Publishing, Suite 6E, 2333 Government St., Victoria, B.C.V8T 4P4, Canada
- 31: Kurt Kosanke, François B. Vernadat & M. Zelm , CIMOSA process model for enterprise modelling, 1997
- 32: Franz Pavlik (CEO), Project organization – function and roles in the project, abgerufen: 2022, URL = <https://projectmanagement.guide/category/projectmanagement/>
- 33: PMI, PMBOK Guide - A Guide to the Project Management Body of Knowledge, 2017, Project Management Institute, 14 Campus Boulevard, Newton Square, Pennsylvania 19073-3299 USA

## Literaturverzeichnis

- 34: Mark C. Layton, Agile Project Management for Dummies, 2012, John Wiley & Sons Inc, 111 River Street, Hoboken, New jersey 07030-5774, USA
- 35: Hirotaka Takeuchi and Ikujiro Nonaka, The New New Product Development Game, 1986
- 36: Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin , Ken Schwaber, Jeff Sutherland, Dave Thomas, The Agile Manifesto, abgerufen: 2001, URL = <https://agilemanifesto.org/iso/de/manifesto.html>
- 37: David Anderson, Sanjiv Augustine, Christopher Avery, Alistair Cockburn, Mike Cohn, Doug DeCarlo, Donna Fitzgerald, Jim Highsmith, Ole Jepsen, Lowell Lindstrom, Todd Little, Kent McDonald, Pollyanna Pixton, Preston Smith, Robert Wysocki, DOI - Declaration of Interdependence, abgerufen: 2005, URL = <https://web.archive.org/web/20180127094805/http://www.pmdoi.org/>
- 38: Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin , Ken Schwaber, Jeff Sutherland, Dave Thomas, The Agile Manifesto, Prinzipien hinter dem Agilen Manifest, abgerufen: 2001, URL = <http://agilemanifesto.org/iso/de/principles.html>
- 39: Thorsten Phillip Klein, Agiles Engineering im Maschinen- und Anlagenbau, 2015
- 40: Tido Eger, Claudia Eckert and P John Clarkson, the role of design freeze in product development, 2005
- 41: Edmund Heinen, Industriebetriebslehre, Entscheidungen im Industriebetrieb, 1983, Betriebswirtschaftlicher Verlag Dr Th Gabler GmbH, Wiesbaden, Germany
- 42: Ken Schwaber, Mike Beedle, Agile Software Development with Scrum, 2007
- 43: Ken Schwaber, Jeff Sutherland, Scrum Guides, abgerufen: 2020, URL = <https://scrumguides.org/>
- 44: Sam Elbeik und Mark Thomas, Project skills, 1998, Routledge , Woburn – Massachusetts, USA
- 45: Martin Fowler, Continous Integration, abgerufen: 2006, URL = <https://martinfowler.com/articles/continuousIntegration.html>,
- 46: ISO 10303-21, STEP-file - Industrial automation systems and integration -- Product data representation and exchange -- Part 21: Implementation methods: Clear text encoding of the exchange structure, abgerufen: , URL = <https://www.loc.gov/preservation/digital/formats/fdd/fdd000448.shtml>
- 47: Autodesk, Inc, Navisworks® - 3D model review software for architecture, engineering, and construction, abgerufen: 2021, URL = <https://www.autodesk.com/products/navisworks/overview>,
- 48: Grady Booch, Robert A. Maksimchuk, Michael W. Engle, Bobbi J. Young PhD, Jim Conallen, Kelli A. Houston, Object-Oriented Analysis and Design with Applications, 2007, Addison-Wesley, Pearson Education Inc., 75 Arlington Street, Suite 300, Boston, MA 02116, USA

## Literaturverzeichnis

49: Francesco Albarosa, Rafael Valenzuela Musura, Social Sustainability Aspects of Agile Project Management, 2016



## Anlagen

Der Scrum Guide von Ken Schwaber & Jeff Sutherland

III

Das ECPM Rahmenkonzept – Ablaufdiagramm von Robert K. Wysocki

XXXI



Ken Schwaber & Jeff Sutherland

# Der Scrum Guide

Der gültige Leitfaden für Scrum: Die Spielregeln

November 2020

## Der Sinn des Scrum Guides

Wir haben Scrum in den frühen 1990er-Jahren entwickelt. Wir haben die erste Version des Scrum Guides im Jahr 2010 geschrieben, um Menschen auf der ganzen Welt dabei zu helfen, Scrum zu verstehen. Wir haben den Guide seitdem durch kleine, funktionale Aktualisierungen weiterentwickelt. Wir stehen gemeinsam dahinter.

Der Scrum Guide enthält die Definition von Scrum. Jedes Element von Scrum dient einem bestimmten Zweck, der für den Gesamtwert und die mit Scrum erzielten Ergebnisse wesentlich ist. Den Kern oder die Grundideen von Scrum zu verändern, Elemente wegzulassen oder den Regeln von Scrum nicht zu folgen, verdeckt Probleme, begrenzt den Nutzen und macht Scrum im Zweifel sogar nutzlos.

Wir verfolgen den zunehmenden Einsatz von Scrum in einer stetig komplexer werdenden Welt. Wir sehen demütig, wie Scrum in zahlreichen Bereichen komplexer Arbeit über die Softwareentwicklung hinaus - wo Scrum seine Wurzeln hat - eingesetzt wird. Während sich die Verwendung von Scrum weiter verbreitet, wird diese Arbeit von Entwickler:innen, Forscher:innen, Analyst:innen, Wissenschaftler:innen und anderen Spezialist:innen getan. Wir verwenden das Wort „Developer:innen“ in Scrum nicht, um jemanden auszuschließen, sondern um zu vereinfachen. Wer auch immer vom Einsatz von Scrum profitiert, soll sich angesprochen fühlen.

Beim Einsatz von Scrum können Muster, Prozesse und Erkenntnisse angewendet und entwickelt werden, die zum Scrum-Rahmenwerk passen, wie es in diesem Dokument beschrieben ist. Ihre Beschreibung geht über den Zweck des Scrum Guides hinaus, da sie kontextabhängig sind und sich, je nachdem wie Scrum eingesetzt wird, stark unterscheiden. Solche Taktiken für die Anwendung innerhalb des Scrum-Rahmenwerks variieren stark und werden an anderer Stelle beschrieben.

Ken Schwaber & Jeff Sutherland, November 2020

© 2020 Ken Schwaber and Jeff Sutherland

This publication is offered for license under the Attribution Share-Alike license of Creative Commons, accessible at <http://creativecommons.org/licenses/by-sa/4.0/legalcode> and also described in summary form at <http://creativecommons.org/licenses/by-sa/4.0/>. By utilizing this Scrum Guide, you acknowledge and agree that you have read and agree to be bound by the terms of the Attribution Share-Alike license of Creative Commons.

Scrum-Definition .....	3
Scrum-Theorie.....	3
Transparenz.....	4
Überprüfung.....	4
Anpassung .....	4
Scrum-Werte.....	4
Scrum Team .....	5
Developer:innen .....	6
Product Owner:in.....	6
Scrum Master:in.....	7
Scrum Events.....	8
Der Sprint .....	8
Sprint Planning.....	9
Daily Scrum .....	10
Sprint Review .....	10
Sprint Retrospective.....	10
Scrum-Artefakte.....	11
Product Backlog .....	11
Commitment: Produkt-Ziel .....	12
Sprint Backlog .....	12
Commitment: Sprint-Ziel .....	12
Increment.....	13
Commitment: Definition of Done .....	13
Schlussbemerkung .....	14
Danksagungen.....	14
Personen .....	14
Scrum-Guide-Historie.....	14
Änderungen des Scrum Guides von 2020 im Vergleich zu 2017 .....	14
Übersetzung.....	16
Änderungshistorie der Übersetzung.....	16
Übersetzungs-Glossar .....	17

## Scrum-Definition

Scrum ist ein leichtgewichtiges Rahmenwerk, welches Menschen, Teams und Organisationen hilft, Wert durch adaptive Lösungen für komplexe Probleme zu generieren.

Kurz gesagt fordert Scrum, dass ein:e Scrum Master:in ein Umfeld fördert, in dem

1. ein:e Product Owner:in die Arbeit für ein komplexes Problem in ein Product Backlog einsortiert;
2. das Scrum Team aus einer Auswahl dieser Arbeit innerhalb eines Sprints ein wertvolles Increment erzeugt;
3. das Scrum Team und dessen Stakeholder:innen die Ergebnisse überprüfen und für den nächsten Sprint anpassen;
4. diese Schritte wiederholt werden.

Scrum ist einfach. Probiere es so aus, wie es ist, und finde heraus, ob seine Philosophie, Theorie und Struktur dabei helfen, Ziele zu erreichen und Wert zu schaffen. Das Scrum-Rahmenwerk ist absichtlich unvollständig und definiert nur die Teile, die zur Umsetzung der Scrum-Theorie erforderlich sind. Scrum baut auf der kollektiven Intelligenz der Personen auf, die es anwenden. Anstatt den Menschen detaillierte Anweisungen zu geben, leiten die Regeln von Scrum ihre Beziehungen und Interaktionen.

Innerhalb des Rahmenwerks können verschiedene Prozesse, Techniken und Methoden eingesetzt werden. Scrum umhüllt bestehende Praktiken oder macht sie überflüssig. Scrum macht die relative Wirksamkeit des aktuellen Managements, der Umgebung und der Arbeitstechniken sichtbar, so dass Verbesserungen vorgenommen werden können.

## Scrum-Theorie

Scrum basiert auf Empirie und Lean Thinking. Empirie bedeutet, dass Wissen aus Erfahrung gewonnen wird und Entscheidungen auf der Grundlage von Beobachtungen getroffen werden. Lean Thinking reduziert Verschwendung und fokussiert auf das Wesentliche.

Scrum verwendet einen iterativen, inkrementellen Ansatz zur Optimierung der Vorhersagbarkeit und zur Risikokontrolle. Scrum setzt auf Personengruppen, die gemeinsam über alle Fähigkeiten und Fachkenntnisse verfügen, um die Arbeit zu erledigen und solche Fähigkeiten im Bedarfsfall zu teilen oder zu erwerben.

Scrum kombiniert vier formale Events zur Überprüfung und Anpassung innerhalb eines umspannenden Events, des Sprints. Diese Events funktionieren, weil sie die empirischen Scrum-Säulen Transparenz, Überprüfung und Anpassung implementieren.

## Transparenz

Der sich entwickelnde Prozess und die entstehende Arbeit müssen sowohl für diejenigen sichtbar sein, die die Arbeit ausführen, als auch für diejenigen, die die Arbeit empfangen. Bei Scrum basieren wichtige Entscheidungen auf dem wahrgenommenen Zustand seiner drei formalen Artefakte. Artefakte, die wenig transparent sind, können zu Entscheidungen führen, die den Wert mindern und das Risiko erhöhen.

Transparenz ermöglicht Überprüfung. Eine Überprüfung ohne Transparenz ist irreführend und verschwenderisch.

## Überprüfung

Die Scrum-Artefakte und der Fortschritt in Richtung der vereinbarten Ziele müssen häufig und sorgfältig überprüft werden, um potenziell unerwünschte Abweichungen oder Probleme aufzudecken. Um bei der Überprüfung zu helfen, bietet Scrum eine Kadenz in Form seiner fünf Events.

Überprüfung ermöglicht Anpassung. Überprüfung ohne Anpassung wird als unsinnig betrachtet. Scrum Events sind darauf ausgerichtet, Veränderungen zu bewirken.

## Anpassung

Wenn einzelne Aspekte eines Prozesses von akzeptablen Grenzen abweichen oder wenn das resultierende Produkt nicht akzeptabel ist, müssen der angewandte Prozess oder die produzierten Ergebnisse angepasst werden. Die Anpassung muss so schnell wie möglich erfolgen, um weitere Abweichungen zu minimieren.

Die Anpassung wird schwieriger, wenn die beteiligten Personen nicht bevollmächtigt (empowered) sind oder sich nicht selbst managen können. Von einem Scrum Team wird erwartet, dass es sich in dem Moment anpasst, in dem es durch Überprüfung etwas Neues lernt.

## Scrum-Werte

Die erfolgreiche Anwendung von Scrum hängt davon ab, dass die Menschen immer besser in der Lage sind, fünf Werte zu leben:

**Commitment, Fokus, Offenheit, Respekt und Mut**

Das Scrum Team committet sich, seine Ziele zu erreichen und sich gegenseitig zu unterstützen. Sein primärer Fokus liegt auf der Arbeit des Sprints, um den bestmöglichen Fortschritt in Richtung dieser



Ziele zu bewirken. Das Scrum Team und dessen Stakeholder:innen sind offen in Bezug auf die Arbeit und die Herausforderungen. Die Mitglieder des Scrum Teams respektieren sich gegenseitig als fähige, unabhängige Personen und werden als solche auch von den Menschen, mit denen sie zusammenarbeiten, respektiert. Die Mitglieder des Scrum Teams haben den Mut, das Richtige zu tun: an schwierigen Problemen zu arbeiten.

Diese Werte geben dem Scrum Team die Richtung vor, was seine Arbeit, seine Handlungen und sein Verhalten betrifft. Die Entscheidungen, die getroffen werden, die Schritte, die unternommen werden, und die Art und Weise, wie Scrum angewendet wird, sollten diese Werte stärken und nicht schwächen oder untergraben. Die Mitglieder des Scrum Teams lernen und erforschen die Werte, während sie in den Events und mit den Artefakten von Scrum arbeiten. Wenn diese Werte durch das Scrum Team und die Menschen, mit denen es arbeitet, verkörpert werden, werden die empirischen Scrum-Säulen Transparenz, Überprüfung und Anpassung lebendig und bauen Vertrauen auf.

## Scrum Team

Der zentrale Bestandteil von Scrum ist ein kleines Team von Menschen, ein Scrum Team. Das Scrum Team besteht aus einem:einer Scrum Master:in, einem:einer Product Owner:in und Developer:innen. Innerhalb eines Scrum Teams gibt es keine Teilteams oder Hierarchien. Es handelt sich um eine geschlossene Einheit von Fachleuten, die sich auf ein Ziel konzentrieren, das Produkt-Ziel.

Scrum Teams sind interdisziplinär, d.h. die Mitglieder verfügen über alle Fähigkeiten, die erforderlich sind, um in jedem Sprint Wert zu schaffen. Sie managen sich außerdem selbst, d.h. sie entscheiden intern, wer was wann und wie macht.

Das Scrum Team ist klein genug, um flink zu bleiben und groß genug, um innerhalb eines Sprints bedeutsame Arbeit fertigzustellen, üblicherweise 10 oder weniger Personen. Im Allgemeinen haben wir festgestellt, dass kleinere Teams besser kommunizieren und produktiver sind. Wenn Scrum Teams zu groß werden, sollten sie in Erwägung ziehen, sich in mehrere zusammengehörende Scrum Teams zu reorganisieren, die sich alle auf dasselbe Produkt konzentrieren. Daher sollten sie sich Produkt-Ziel, Product Backlog und Product Owner:in teilen.

Das Scrum Team ist umsetzungsverantwortlich (responsible) für alle produktbezogenen Aktivitäten: Zusammenarbeit mit den Stakeholder:innen, Verifikation, Wartung, Betrieb, Experimente, Forschung und Entwicklung und alles, was sonst noch erforderlich sein könnte. Es ist von der Organisation so aufgebaut und befähigt, dass es seine Arbeit selbst steuert. Das Arbeiten in Sprints mit einer nachhaltigen Geschwindigkeit verbessert den Fokus und die Kontinuität des Scrum Teams.

Das gesamte Scrum Team ist ergebnisverantwortlich (accountable), in jedem Sprint ein wertvolles, nützliches Increment zu schaffen. Scrum definiert drei spezifische Ergebnisverantwortlichkeiten innerhalb des Scrum Teams: Developer:innen, Product Owner:in und Scrum Master:in





## Developer:innen

Developer:innen sind jene Personen im Scrum Team, die sich der Aufgabe verschrieben haben, jeden Sprint jeden Aspekt eines nutzbaren Increments zu schaffen.

Die spezifischen Fähigkeiten, die von den Developer:innen benötigt werden, sind oft breit gefächert und variieren je nach Arbeitskontext. Die Developer:innen sind jedoch immer ergebnisverantwortlich dafür,

- einen Plan für den Sprint zu erstellen, das Sprint Backlog;
- Qualität durch die Einhaltung einer Definition of Done einzubringen;
- täglich ihren Plan zur Erreichung des Sprint-Ziels anzupassen; und
- sich wechselseitig als Expert:innen zur Verantwortung zu ziehen.

## Product Owner:in

Der:die Product Owner:in ist ergebnisverantwortlich für die Maximierung des Wertes des Produkts, der sich aus der Arbeit des Scrum Teams ergibt. Wie dies geschieht, kann je nach Organisation, Scrum Team und Individuum sehr unterschiedlich sein.

Der:die Product Owner:in ist auch für ein effektives Product-Backlog-Management ergebnisverantwortlich, das Folgendes umfasst:

- das Produkt-Ziel zu entwickeln und explizit zu kommunizieren;
- die Product-Backlog-Einträge zu erstellen und klar zu kommunizieren;
- die Reihenfolge der Product-Backlog-Einträge festzulegen; und
- sicherzustellen, dass das Product Backlog transparent, sichtbar und verstanden ist.

Der:die Product Owner:in kann die oben genannten Arbeiten selbst durchführen oder die Umsetzungsverantwortung an andere delegieren. Unabhängig davon bleibt der:die Product Owner:in ergebnisverantwortlich.

Damit der:die Product Owner:in Erfolg haben kann, muss die gesamte Organisation seine:ihre Entscheidungen respektieren. Diese Entscheidungen sind im Inhalt und in der Reihenfolge des Product Backlogs sowie durch das überprüfbare Increment beim Sprint Review, sichtbar.

Der:die Product Owner:in ist eine Person, kein Gremium. Der:die Product Owner:in kann die Bedürfnisse vieler Stakeholder:innen im Product Backlog berücksichtigen. Diejenigen, die das Product Backlog ändern möchten, können dies tun, indem sie versuchen, den:die Product Owner:in zu überzeugen.



## Scrum Master:in

Der:die Scrum Master:in ist ergebnisverantwortlich für die Einführung von Scrum, wie es im Scrum Guide definiert ist. Er:sie tut dies, indem er:sie allen dabei hilft, die Scrum-Theorie und -Praxis zu verstehen, sowohl innerhalb des Scrum Teams als auch in der Organisation.

Der:die Scrum Master:in ist ergebnisverantwortlich für die Effektivität des Scrum Teams. Er:sie tut dies, indem er:sie das Scrum Team in die Lage versetzt, seine Praktiken innerhalb des Scrum-Rahmenwerks zu verbessern.

Scrum Master:innen sind echte Führungskräfte, die dem Scrum Team und der Gesamtorganisation dienen.

Der:die Scrum Master:in dient dem Scrum Team auf unterschiedliche Weise, unter anderem dadurch,

- die Teammitglieder in Selbstmanagement und interdisziplinärer Zusammenarbeit zu coachen;
- das Scrum Team bei der Fokussierung auf die Schaffung von hochwertigen Increments zu unterstützen, die der Definition of Done entsprechen;
- die Beseitigung von Hindernissen (impediments) für den Fortschritt des Scrum Teams zu bewirken; und
- sicherzustellen, dass alle Events von Scrum stattfinden, positiv und produktiv sind und innerhalb der Timebox bleiben.

Der:die Scrum Master:in dient dem:der Product Owner:in auf unterschiedliche Weise, unter anderem dadurch,

- bei der Suche nach Techniken zur effektiven Definition des Produkt-Ziels und zum Product-Backlog-Management zu helfen;
- dem Scrum Team dabei zu helfen, die Notwendigkeit klarer und präziser Product-Backlog-Einträge zu verstehen;
- bei der Etablierung einer empirischen Produktplanung für ein komplexes Umfeld zu helfen; und
- die Zusammenarbeit mit Stakeholder:innen nach Wunsch oder Bedarf zu fördern (facilitate).

Der:die Scrum Master:in dient der Organisation auf unterschiedliche Weise, unter anderem dadurch,

- die Organisation bei der Einführung von Scrum zu führen, zu schulen und zu coachen;
- Einführungen von Scrum in der Organisation zu planen und zu empfehlen;
- Mitarbeitende und Stakeholder:innen beim Verständnis und bei der Umsetzung eines empirischen Ansatzes für komplexe Arbeit zu unterstützen; und
- Barrieren zwischen Stakeholder:innen und Scrum Teams zu beseitigen.

## Scrum Events

Der Sprint ist ein Container für alle anderen Events. Jedes Event in Scrum ist eine formelle Gelegenheit, Scrum-Artefakte zu überprüfen und anzupassen. Diese Events sind speziell darauf ausgelegt, die erforderliche Transparenz zu ermöglichen. Wenn ein Event nicht wie vorgeschrieben durchgeführt wird, verpasst man die Gelegenheit, zu überprüfen und anzupassen. Events werden in Scrum verwendet, um Regelmäßigkeit zu schaffen und die Notwendigkeit von Meetings, die in Scrum nicht definiert sind, zu minimieren. Optimalerweise werden alle Events zur selben Zeit und am selben Ort abgehalten, um die Komplexität zu reduzieren.

## Der Sprint

Sprints sind der Herzschlag von Scrum, wo Ideen in Wert umgewandelt werden.

Es sind Events mit fester Länge von einem Monat oder weniger, um Konsistenz zu schaffen. Ein neuer Sprint beginnt unmittelbar nach dem Abschluss des vorherigen Sprints.

Alle Arbeiten, die notwendig sind, um das Produkt-Ziel zu erreichen, einschließlich Sprint Planning, Daily Scrums, Sprint Review und Sprint Retrospective, finden innerhalb der Sprints statt.

Während des Sprints

- werden keine Änderungen vorgenommen, die das Sprint-Ziel gefährden würden;
- nimmt die Qualität nicht ab;
- wird das Product Backlog nach Bedarf verfeinert; und
- kann der Scope (Inhalt und Umfang) geklärt und mit dem:der Product Owner:in neu vereinbart werden, sobald mehr Erkenntnisse vorliegen.

Sprints ermöglichen Vorhersagbarkeit, indem sie mindestens jeden Kalendermonat eine Überprüfung und Anpassung der Fortschritte in Richtung eines Produkt-Ziels gewährleisten. Wenn der Horizont eines Sprints zu lang ist, kann das Sprint-Ziel hinfällig werden, die Komplexität kann steigen und das Risiko kann zunehmen. Kürzere Sprints können eingesetzt werden, um mehr Lernzyklen zu generieren und das Risiko von Kosten und Aufwand auf einen kleineren Zeitrahmen zu begrenzen. Jeder Sprint kann als ein kurzes Projekt betrachtet werden.

Es gibt verschiedene Vorgehensweisen, um den Fortschritt vorherzusagen, wie Burn-Down-Charts, Burn-Up-Charts oder Cumulative-Flow-Diagramme. Diese haben sich zwar als nützlich erwiesen, ersetzen jedoch nicht die Bedeutung der Empirie. In komplexen Umgebungen ist unbekannt, was passieren wird. Nur was bereits geschehen ist, kann für eine vorausschauende Entscheidungsfindung genutzt werden.

Ein Sprint könnte abgebrochen werden, wenn das Sprint-Ziel obsolet wird. Nur der:die Product Owner:in hat die Befugnis, den Sprint abubrechen.





## Sprint Planning

Das Sprint Planning initiiert den Sprint, indem es die für den Sprint auszuführenden Arbeiten darlegt. Dieser resultierende Plan wird durch die gemeinschaftliche Arbeit des gesamten Scrum Teams erstellt.

Der:die Product Owner:in stellt sicher, dass die Teilnehmenden vorbereitet sind, die wichtigsten Product-Backlog-Einträge zu besprechen, und wie sie dem Produkt-Ziel zuzuordnen sind. Das Scrum Team kann zu Beratungszwecken auch andere Personen zur Teilnahme am Sprint Planning einladen.

Das Sprint Planning behandelt die folgenden Themen:

Thema Eins: Warum ist dieser Sprint wertvoll?

Der:die Product Owner:in schlägt vor, wie das Produkt im aktuellen Sprint seinen Wert und Nutzen steigern könnte. Das gesamte Scrum Team arbeitet dann zusammen, um ein Sprint-Ziel zu definieren, das verdeutlicht, warum der Sprint für die Stakeholder:innen wertvoll ist. Das Sprint-Ziel muss vor dem Ende des Sprint Plannings finalisiert sein.

Thema Zwei: Was kann in diesem Sprint abgeschlossen (Done) werden?

Im Gespräch mit dem:der Product Owner:in wählen die Developer:innen Einträge aus dem Product Backlog aus, die in den aktuellen Sprint aufgenommen werden sollen. Das Scrum Team kann diese Einträge während dieses Prozesses verfeinern, was Verständnis und Vertrauen erhöht.

Die Auswahl, wie viel innerhalb eines Sprints abgeschlossen werden kann, kann eine Herausforderung darstellen. Je mehr die Developer:innen jedoch über ihre bisherige Leistung, ihre zukünftige Kapazität und ihre Definition of Done wissen, desto sicherer werden sie in ihren Sprint-Vorhersagen sein.

Thema Drei: Wie wird die ausgewählte Arbeit erledigt?

Für jeden ausgewählten Product-Backlog-Eintrag planen die Developer:innen die notwendige Arbeit, um ein Increment zu erstellen, das der Definition of Done entspricht. Dies geschieht oft durch Zerlegung von Product-Backlog-Einträgen in kleinere Arbeitseinheiten von einem Tag oder weniger. Wie dies geschieht, liegt im alleinigen Ermessen der Developer:innen. Niemand sonst sagt ihnen, wie sie Product-Backlog-Einträge in Increments von Wert umwandeln sollen.

Das Sprint-Ziel, die für den Sprint ausgewählten Product-Backlog-Einträge und der Plan für deren Lieferung werden zusammenfassend als Sprint Backlog bezeichnet.

Das Sprint Planning ist zeitlich beschränkt auf maximal acht Stunden für einen einmonatigen Sprint. Bei kürzeren Sprints ist das Event in der Regel kürzer.

## Daily Scrum

Der Zweck des Daily Scrums besteht darin, den Fortschritt in Richtung des Sprint-Ziels zu überprüfen und das Sprint Backlog bei Bedarf anzupassen, um die bevorstehende geplante Arbeit zu justieren.

Das Daily Scrum ist ein 15-minütiges Event für die Developer:innen des Scrum Teams. Um die Komplexität zu reduzieren, wird es an jedem Arbeitstag des Sprints zur gleichen Zeit und am gleichen Ort abgehalten. Falls der:die Product Owner:in oder der:die Scrum Master:in aktiv an Einträgen des Sprint Backlogs arbeitet, nimmt er:sie als Developer:in teil.

Die Developer:innen können Struktur und Techniken beliebig wählen, solange ihr Daily Scrum sich auf den Fortschritt in Richtung des Sprint-Ziels fokussiert und einen umsetzbaren Plan für den nächsten Arbeitstag erstellt. Das schafft Fokus und fördert Selbstmanagement.

Daily Scrums verbessern die Kommunikation, identifizieren Hindernisse, fördern die schnelle Entscheidungsfindung und eliminieren konsequent die Notwendigkeit für andere Meetings.

Das Daily Scrum ist nicht die einzige Gelegenheit, bei der die Developer:innen ihren Plan anpassen dürfen. Sie treffen sich oftmals während des Tages für detailliertere Diskussionen zur Anpassung oder Neuplanung der restlichen Arbeit des Sprints.

## Sprint Review

Zweck des Sprint Reviews ist es, das Ergebnis des Sprints zu überprüfen und künftige Anpassungen festzulegen. Das Scrum Team stellt die Ergebnisse seiner Arbeit den wichtigsten Stakeholder:innen vor, und die Fortschritte in Richtung des Produkt-Ziels werden diskutiert.

Während des Events überprüfen das Scrum Team und die Stakeholder:innen, was im Sprint erreicht wurde und was sich in ihrem Umfeld verändert hat. Auf der Grundlage dieser Informationen arbeiten die Teilnehmenden gemeinsam daran, was als Nächstes zu tun ist. Auch kann das Product Backlog angepasst werden, um neue Möglichkeiten wahrzunehmen. Das Sprint Review ist ein Arbeitstermin, und das Scrum Team sollte vermeiden, es auf eine Präsentation zu beschränken.

Das Sprint Review ist das vorletzte Event des Sprints und ist für einen einmonatigen Sprint auf maximal vier Stunden zeitlich beschränkt. Bei kürzeren Sprints ist das Event in der Regel kürzer.

## Sprint Retrospective

Der Zweck der Sprint Retrospective ist es, Wege zur Steigerung von Qualität und Effektivität zu planen.



Das Scrum Team überprüft, wie der letzte Sprint in Bezug auf Individuen, Interaktionen, Prozesse, Werkzeuge und seine Definition of Done verlief. Die überprüften Elemente variieren oft je nach Arbeitsdomäne. Annahmen, die das Team in die Irre geführt haben, werden identifiziert und ihre Ursprünge erforscht. Das Scrum Team bespricht, was während des Sprints gut gelaufen ist, auf welche Probleme es gestoßen ist und wie diese Probleme gelöst wurden (oder auch nicht).

Das Scrum Team identifiziert die hilfreichsten Änderungen, um seine Effektivität zu verbessern. Die wirkungsvollsten Verbesserungen werden so schnell wie möglich in Angriff genommen. Sie können sogar in das Sprint Backlog für den nächsten Sprint aufgenommen werden.

Die Sprint Retrospective schließt den Sprint ab. Sie ist für einen einmonatigen Sprint auf maximal drei Stunden beschränkt. Bei kürzeren Sprints ist das Event in der Regel kürzer.

### Scrum-Artefakte

Die Artefakte von Scrum repräsentieren Arbeit oder Wert. Sie sind dafür ausgelegt, die Transparenz von Schlüsselinformationen zu maximieren. So haben alle, die sie überprüfen, die gleiche Grundlage für Anpassungen.

Jedes Artefakt beinhaltet ein Commitment, um sicherzustellen, dass Informationen bereitgestellt werden, welche Transparenz und Fokus verbessern, um den Fortschritt messbar zu machen:

- Für das Product Backlog ist es das Produkt-Ziel.
- Für das Sprint Backlog ist es das Sprint-Ziel.
- Für das Increment ist es die Definition of Done.

Diese Commitments dienen dazu, Empirie und die Scrum-Werte für das Scrum Team und seine Stakeholder:innen zu verstärken.

### Product Backlog

Das Product Backlog ist eine emergente, geordnete Liste der Dinge, die zur Produktverbesserung benötigt werden. Es ist die einzige Quelle von Arbeit, die durch das Scrum Team erledigt wird.

Product-Backlog-Einträge, die durch das Scrum Team innerhalb eines Sprints abgeschlossen (Done) werden können, gelten als bereit für die Auswahl in einem Sprint-Planning-Event. Diesen Transparenzgrad erlangen sie in der Regel durch Refinement-Aktivitäten. Das Refinement des Product Backlogs ist der Vorgang, durch den Product-Backlog-Einträge in kleinere, präzisere Elemente zerlegt und weiter definiert werden. Dies ist eine kontinuierliche Aktivität, wodurch weitere Details wie Beschreibung, Reihenfolge und Größe ergänzt werden. Die Attribute variieren oft je nach Arbeitsumfeld.



Die Developer:innen, die die Arbeit erledigen werden, sind für die Größenbestimmung umsetzungsverantwortlich. Der:die Product Owner:in kann die Developer:innen beeinflussen, indem er:sie dabei unterstützt, die Product-Backlog-Einträge zu verstehen und Kompromisse einzugehen.

## Commitment: Produkt-Ziel

Das Produkt-Ziel beschreibt einen zukünftigen Zustand des Produkts, welches dem Scrum Team als Planungsziel dienen kann. Das Produkt-Ziel befindet sich im Product Backlog. Der Rest des Product Backlogs entsteht, um zu definieren, „was“ das Produkt-Ziel erfüllt.

*Ein Produkt ist ein Instrument, um Wert zu liefern. Es hat klare Grenzen, bekannte Stakeholder:innen, eindeutig definierte Benutzer:innen oder Kund:innen. Ein Produkt kann eine Dienstleistung, ein physisches Produkt oder etwas Abstrakteres sein.*

Das Produkt-Ziel ist das langfristige Ziel für das Scrum Team. Das Scrum Team muss eine Zielvorgabe erfüllen (oder aufgeben), bevor es die nächste angeht.

## Sprint Backlog

Das Sprint Backlog besteht aus dem Sprint-Ziel (Wofür), den für den Sprint ausgewählten Product-Backlog-Einträgen (Was) sowie einem umsetzbaren Plan für die Lieferung des Increments (Wie).

Das Sprint Backlog ist ein Plan von und für die Developer:innen. Es ist ein deutlich sichtbares Echtzeitbild der Arbeit, welche die Developer:innen während des Sprints zur Erreichung des Sprint-Ziels ausführen wollen. Folglich wird das Sprint Backlog während des gesamten Sprints immer dann aktualisiert, wenn mehr gelernt wurde. Es sollte genügend Details beinhalten, damit sie ihren Fortschritt im Daily Scrum überprüfen können.

## Commitment: Sprint-Ziel

Das Sprint-Ziel ist die einzige Zielsetzung für den Sprint. Obwohl das Sprint-Ziel ein Commitment der Developer:innen ist, bietet es Flexibilität in Bezug auf die genaue Arbeit, die erforderlich ist, um es zu erreichen. Das Sprint-Ziel schafft auch Kohärenz und Fokus und ermutigt somit das Scrum Team, zusammen statt in separaten Initiativen zu arbeiten.

Das Sprint-Ziel wird während des Sprint-Planning-Events erstellt und dann zum Sprint Backlog hinzugefügt. Während die Developer:innen innerhalb des Sprints arbeiten, behalten sie das Sprint-Ziel im Gedächtnis. Wenn sich herausstellt, dass die Arbeit von ihren Erwartungen abweicht, arbeiten sie mit dem:der Product Owner:in zusammen, um den Umfang des Sprint Backlogs innerhalb des Sprints zu verhandeln, ohne das Sprint-Ziel zu beeinflussen.



## Increment

Ein Increment ist ein konkreter Schritt in Richtung des Produkt-Ziels. Jedes Increment ist additiv zu allen vorherigen Increments und gründlich geprüft, um sicherzustellen, dass sie alle zusammen funktionieren. Um einen Mehrwert zu erzielen, muss das Increment verwendbar sein.

Innerhalb eines Sprints kann mehr als ein Increment erstellt werden. Deren Summe wird im Sprint Review vorgestellt, womit Empirie unterstützt wird. Ein Increment könnte jedoch auch schon vor Ende des Sprints an die Stakeholder:innen geliefert werden. Das Sprint Review sollte niemals als Barriere zur Lieferung von Wert angesehen werden.

Arbeit kann nicht als Teil eines Increments betrachtet werden, solange sie nicht der Definition of Done entspricht.

### Commitment: Definition of Done

Die Definition of Done ist eine formale Beschreibung des Zustands des Increments, wenn es die für das Produkt erforderlichen Qualitätsmaßnahmen erfüllt.

In dem Moment, in dem ein Product-Backlog-Eintrag die Definition of Done erfüllt, wird ein Increment geboren.

Die Definition of Done schafft Transparenz, indem sie allen ein gemeinsames Verständnis darüber vermittelt, welche Arbeiten als Teil des Increments abgeschlossen wurden. Wenn ein Product-Backlog-Eintrag nicht der Definition of Done entspricht, kann es weder released noch beim Sprint Review präsentiert werden. Stattdessen wandert es zur zukünftigen Berücksichtigung in das Product Backlog zurück.

Wenn die Definition of Done für ein Increment Teil der Standards der Organisation ist, müssen alle Scrum Teams diese als Mindestmaß befolgen. Wenn sie kein Organisationsstandard ist, muss das Scrum Team eine für das Produkt geeignete Definition of Done erstellen.

Die Developer:innen müssen sich an die Definition of Done halten. Wenn mehrere Scrum Teams an einem Produkt zusammenarbeiten, müssen sie eine gemeinsame Definition of Done definieren und sich alle daran halten.





## Schlussbemerkung

Scrum ist kostenlos und wird in diesem Guide angeboten. Das Scrum-Rahmenwerk, wie es hier beschrieben wird, ist unveränderlich. Es ist zwar möglich, nur Teile von Scrum zu implementieren, aber das Ergebnis ist nicht Scrum. Scrum existiert nur in seiner Gesamtheit und funktioniert gut als Container für andere Techniken, Methodiken und Praktiken.

## Danksagungen

### Personen

Von den tausenden Personen, die zu Scrum beigetragen haben, sollten wir diejenigen hervorheben, die zu Beginn von besonderer Bedeutung waren: Jeff Sutherland arbeitete mit Jeff McKenna und John Scumniotales, und Ken Schwaber arbeitete mit Mike Smith sowie Chris Martin - und sie alle arbeiteten zusammen. Viele weitere haben in den folgenden Jahren einen Beitrag geleistet. Ohne deren Hilfe wäre Scrum nicht so ausgefeilt, wie es heute ist.

### Scrum-Guide-Historie

Ken Schwaber und Jeff Sutherland haben Scrum gemeinsam zum ersten Mal auf der OOPSLA Konferenz 1995 präsentiert. Diese Präsentation dokumentierte im Kern die Erkenntnisse, die Ken und Jeff in den vorherigen Jahren bei der Anwendung von Scrum gewonnen hatten, und stellte die erste formale Veröffentlichung der Definition von Scrum dar.

Der Scrum Guide dokumentiert Scrum, wie es von Jeff Sutherland und Ken Schwaber über 30 Jahre lang weiterentwickelt wurde, gewachsen ist und aufrechterhalten wurde. Andere Quellen liefern Muster, Prozesse und Erkenntnisse, die das Scrum-Rahmenwerk ergänzen. Diese können zur Steigerung der Produktivität, des Werts, der Kreativität und der Zufriedenheit mit den Ergebnissen führen.

Die vollständige Geschichte von Scrum wird an anderer Stelle beschrieben. Um die ersten Orte zu würdigen, an denen es erprobt wurde und sich bewährt hat, erkennen wir Individual Inc., Newspaper, Fidelity Investments und IDX (jetzt GE Medical) als solche an.

## Änderungen des Scrum Guides von 2020 im Vergleich zu 2017

### Noch weniger vorschreibend

Im Laufe der Jahre wurde der Scrum Guide etwas vorschreibender. Die Version von 2020 zielte darauf ab, Scrum wieder zu einem minimal ausreichenden Rahmenwerk zu machen, indem die vorschreibende Sprache entfernt oder abgeschwächt wurde. Zum Beispiel wurden die Daily-Scrum-Fragen entfernt, die Sprache zur Beschreibung der Attribute von Product-Backlog-Einträgen und Retro-Items im Sprint Backlog abgeschwächt, der Abschnitt zum Sprint-Abbruch gekürzt und mehr.

### Ein Team, fokussiert auf ein Produkt

Ziel war es, das Konzept eines separaten Teams innerhalb eines Teams zu beseitigen, das zu einem „Stellvertretenden-“ oder „wir und sie“-Verhalten zwischen dem:der PO und dem Dev Team geführt hat. Es gibt jetzt nur noch ein Scrum Team, das sich auf dasselbe Ziel fokussiert und drei verschiedene Ergebnisverantwortlichkeiten hat: PO, SM und Developer:innen.

### Einführung des Produkt-Ziels

Der Scrum Guide von 2020 führt das Konzept eines Produkt-Ziels ein, um das Scrum Team auf ein größeres, wertvolles Ziel auszurichten. Jeder Sprint sollte das Produkt näher an das übergeordnete Produkt-Ziel heranbringen.

### Eine Heimat für das Sprint-Ziel, die Definition of Done und das Produkt-Ziel

Frühere Scrum Guides beschrieben das Sprint-Ziel und die Definition of Done, ohne ihnen wirklich eine Identität zu geben. Sie waren nicht wirklich Artefakte, aber waren diesen in gewisser Weise zugeordnet. Mit der Erweiterung um das Produkt-Ziel bietet die Version von 2020 mehr Klarheit darüber. Jedes der drei Artefakte enthält nun „Commitments“ ihnen gegenüber. Für das Product Backlog ist dies das Produkt-Ziel, das Sprint Backlog hat das Sprint-Ziel, und das Increment hat die Definition of Done (jetzt ohne Anführungszeichen). Sie existieren, um Transparenz und Fokus hinsichtlich des Fortschritts jedes Artefakts zu fördern.

### Selbstmanagement statt Selbstorganisation

Frühere Scrum Guides bezeichneten Development Teams als insofern selbstorganisierend, als dass sie wählen konnten, wer die Arbeit erledigt und wie gearbeitet wird. Mit einem stärkeren Fokus auf das Scrum Team betont die Version von 2020 ein selbstmanagendes Scrum Team. Dieses wählt, wer die Arbeit erledigt, wie und woran gearbeitet wird.

### Drei Sprint-Planning-Themen

Zusätzlich zu den Sprint-Planning-Themen „Was“ und „Wie“ legt der Scrum Guide von 2020 den Schwerpunkt auf ein drittes Thema: „Wofür“. Dieses bezieht sich auf das Sprint-Ziel.

### Allgemeine Vereinfachung der Sprache für ein breiteres Publikum

Der Scrum Guide von 2020 legt einen Schwerpunkt auf die Eliminierung redundanter und komplexer Aussagen sowie die Beseitigung aller verbleibenden Rückschlüsse auf IT-Arbeit (z.B. Test, System, Design, Requirements, etc.) gelegt. Der Scrum Guide umfasst jetzt weniger als 13 Seiten [Anm. d. Übersetzer:innen: Im Deutschen sind es bis zur Danksagung 14 geworden.].





## Übersetzung

Dieser Guide wurde von der englischen Originalversion, bereitgestellt von Ken Schwaber und Jeff Sutherland, übersetzt. Hierzu beigetragen haben:

2020: Silke von der Bruck, Sabine Canditt, Jan Gretschuskin, Eva Gysling, Martin Hoppacher, Björn Jensen, Ralph Jocham, Dominik Maximini, Wolf Dieter Moggert, Peter Schmidt, Boris Steiner

2017: Sabine Canditt, Jan Gretschuskin, Eva Gysling, Martin Hoppacher, Ralph Jocham, Dominik Maximini, Stefan Mieth, Wolf Dieter Moggert, Pascal Naujoks, Urs Reupke, Anna Rudat, Andreas Schliep, Harald Schlindwein, Peter Schmidt, Boris Steiner

2016: Jan Gretschuskin, Dominik Maximini, Pascal Naujoks, Boris Steiner, Jürgen Halstenberg, Ralph Jocham, Wolf Dieter Moggert, Patrick Koglin, Harald Schlindwein

2013: Jan Gretschuskin, Dominik Maximini, Pascal Naujoks, Sabrina Roos, Andreas Schliep, Wolfgang Wiedenroth

2011: Dominik Maximini, Andreas Schliep, Ulf Schneider, Wolfgang Wiedenroth

### Kontaktinformationen

Name der Übersetzungsgruppe:	GermanScrumTranslators
Bewerbungen werden angenommen unter:	<a href="https://www.linkedin.com/groups/4071080/">https://www.linkedin.com/groups/4071080/</a>
Primärer Ansprechpartner / Product Owner:	Dominik Maximini
Kontaktadresse:	<a href="mailto:translation@valuerise-consulting.de">translation@valuerise-consulting.de</a>

### Änderungshistorie der Übersetzung

Version 1.0:	Komplette Neuübersetzung des Scrum Guides, basierend auf der Version von 2016.
Version 1.1:	Klarstellung im Abschnitt „Das Scrum Team“: Interdisziplinäre Teams sind im Idealfall nicht von Personen außerhalb des Teams abhängig. Klarstellung im Abschnitt: „Anwendungen von Scrum“: Frequenz von Auslieferungen regelmäßig und mehrfach täglich.
Version 1.2:	Klarstellung im Abschnitt: „Sprint Review“: Der Scrum Master agiert als Servant Leader und nicht nur als Organisator des Events.
Version 2.0:	Überarbeitung der Begrifflichkeiten und Ergänzung des verwendeten Glossars. „Sinn machen“ wurde in „Sinn ergeben“ geändert.
Version 3.2:	Komplette Neuübersetzung anlässlich der Veröffentlichung des Scrum Guide 2020.
Version 3.3:	Einarbeitung des ersten Feedbacks nach der Neuübersetzung.

## Übersetzungs-Glossar

Version 3.3, Stand 22.12.2020

Übersetzungs-Grundregeln, an die wir uns gehalten haben:

- Feststehende Ausdrücke von „Core Scrum“ (Accountabilities, Artefakte, Events) behalten wir Englisch bei, alle anderen werden übersetzt (Beispiel: Sprint Retrospective).
- Folgt ein deutsches Wort auf eines oder mehrere englische, so wird gemäß der deutschen Rechtschreibregeln „durchgekoppelt“ (Beispiel: Sprint-Backlog-Einträge).
- Wir nutzen wie im Original inklusive Formulierungen („gendern“). Wir erreichen das Screenreader-freundlich, indem wir Artikel und Varianten mit Doppelpunkt verknüpfen. Das gilt auch für feststehende Ausdrücke (Beispiele: ein:e Product Owner:in oder die Developer:innen).
- Wir benutzen überwiegend auch bei englischen Ausdrücken die deutsche Grammatik, z.B. den deutschen Genitiv (Beispiel: nicht „innerhalb des Sprint“ sondern „innerhalb des Sprints“).

Englischer Begriff	Deutscher Begriff
accountable	ergebnisverantwortlich
adaptation	Anpassung
Commitment	Commitment
Courage	Mut
cross-functional	interdisziplinär
Daily Scrum	Daily Scrum
Definition of Done	Definition of Done
the Developer the Developers	der:die Developer:in; die Developer:innen
Done	Done
empiricism	Empirie
event	Event
feedback	Feedback
Focus	Fokus
forecast	Vorhersage
framework	Rahmenwerk
Increment	das Increment
inspect & adapt	Überprüfung und Anpassung
inspection	Überprüfung
Lean Thinking	Lean Thinking
Openness	Offenheit
process framework	Prozessrahmenwerk
product	Produkt
Product Backlog	das Product Backlog
Product Backlog item	Product-Backlog-Eintrag
Product Goal	Produkt-Ziel
the Product Owner	der:die Product Owner:in
refinement	Refinement
Respect	Respekt

Anlage 1: Der Scrum Guide von Ken Schwaber & Jeff Sutherland

responsible	umsetzungsverantwortlich
Scrum Artifacts	Scrum-Artefakte
Scrum	Scrum
Scrum Events	Scrum Events
Scrum Guide	Scrum Guide
the Scrum Master	der:die Scrum Master:in
Scrum Team	Scrum Team
self-management	Selbstmanagement
Sprint Backlog	das Sprint Backlog
Sprint Backlog item	Sprint-Backlog-Eintrag
Sprint Goal	Sprint-Ziel
Sprint Planning	das Sprint Planning
Sprint progress	Sprint-Fortschritt
Sprint Retrospective	die Sprint Retrospective
Sprint Review	das Sprint Review
Stakeholder	Stakeholder:innen
timebox	die Timebox
timeboxed	zeitlich beschränkt
value	Wert
Transparency	Transparenz



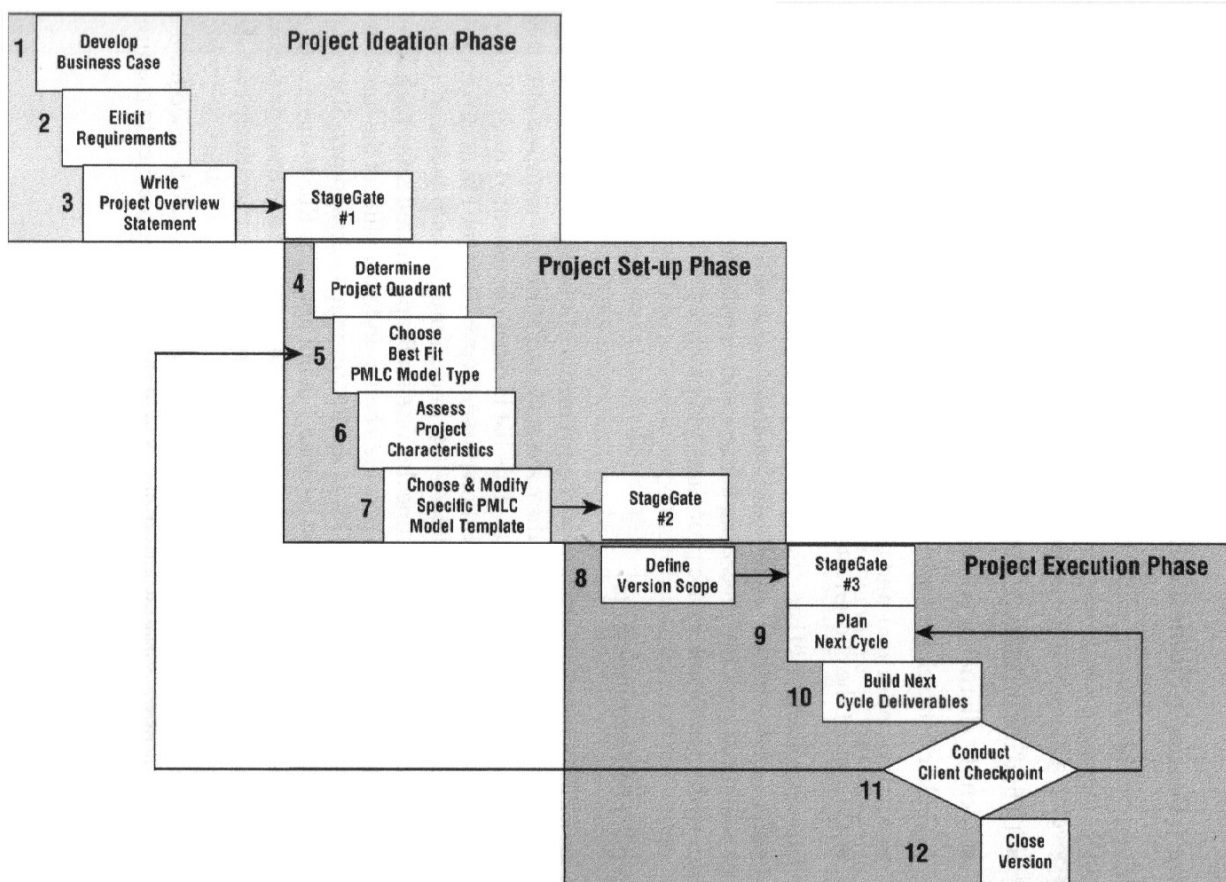
# Anlage 1: Der Scrum Guide von Ken Schwaber & Jeff Sutherland

< diese Seite bleibt aus redaktionellen Gründen frei >

Anlage 2: Das ECPM Rahmenkonzept – Ablaufdiagramm von Robert K. Wysocki

Zur Ermittlung der Struktur von komplexen Geschäftsprozessen bei agilen (Quadrant 2), extremen (Quadrant 3) und „emertxe“ (Quadrant 2) Anforderungen entwickelte Wysocki das ECPM (Effective Complex Project Management) Rahmenkonzept<sup>86</sup>.

## Ablaufdiagramm zum ECPM Rahmenkonzept



86 ECPM Process Flow Diagram, Chapter 14 – Hybrid Project Management Framework, Effective Project Management – Traditional, Agile, Extreme, Hybrid, S 417 ff, Robert K. Wysocki, Indianapolis USA, 2019

< diese Seite bleibt aus redaktionellen Gründen frei >



## Selbstständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe.

Stellen, die wörtlich oder sinngemäß aus Quellen entnommen wurden, sind als solche kenntlich gemacht.

Diese Arbeit wurde in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegt.

Linz, den 22. Mai 2022

