

Fakultät für Informatik und Automatisierung
Fachgebiet Rechnerarchitektur und Eingebettete Systeme

DISSERTATION

zur Erlangung des akademischen Grades
Doktoringenieur (Dr.-Ing.)

EIN PETRI-NETZ-TABELLEN-MODELL ZUR ANWENDUNG
IM KLASSISCHEN UND AGILEN PROJEKTMANAGEMENT

M. Sc. Maxi Weichenhain

Matrikel-Nr.: 46100

geb. am 08.07.1985

Gutachter

1. Prof. Dr.-Ing. habil. Wolfgang Fengler, TU Ilmenau
2. Prof. Dr.-Ing. Monika Heiner, TU Cottbus
3. Prof. Dr.-Ing. habil. Herwig Unger, Fernuniversität Hagen

Tag der Einreichung: 01.07.2019

Tag der wissenschaftlichen Aussprache: 17.07.2020

*„Planung ersetzt den Zufall durch den Irrtum.
Nur aus Irrtümern kann man lernen,
nicht jedoch aus Zufällen.“
(Werner Kirsch, LMU München)*

In Erinnerung an meinen geliebten Vater und Stiefvater

Mario Weichenhain & Hans-Joachim Kühn.

Vorwort / Danksagung

Die vorliegende Arbeit entstand im Rahmen meiner Promotion an der TU Ilmenau. Meine langjährige Tätigkeit in einem internationalen Beratungsunternehmen hat mich letztendlich zu dieser Arbeit motiviert. Ich bin während dieser lehrreichen Arbeitsjahre mit einer Vielzahl von Projekten in unterschiedlichen Größenordnungen in Berührung gekommen, die allesamt eine Gemeinsamkeit aufweisen: Projekte werden gleich ihrer Größenordnung immer komplexer und umfangreicher in ihrer Struktur und damit auch die Ausgestaltung der Planung solcher Projekte immer schwieriger und komplexer.

Das Gelingen dieser Arbeit ist auf die Unterstützung sowie Hilfe von unterschiedlichsten Personen zurückzuführen. Ein großer Dank geht an meinen Doktorvater Herrn Dr.-Ing. habil. Fengler für die großartige Unterstützung und hervorragende Betreuung. Seine Begeisterung für wissenschaftliche Forschungen – welche einfach nur anstecken kann – hat mich immer wieder bereichert und motiviert. Die zahlreichen Diskussionen und Anregungen bilden den Grundstein dieser Arbeit. Ebenfalls möchte ich Frau Prof. Dr.-Ing. Heiner und Herr Prof. Dr.-Ing. habil. Unger für die Beratungen und Anregungen danken, die im hohen Maße zur Abrundung dieser Arbeit beigetragen haben. Mein Dank gilt allen Bekannten und Familienmitgliedern für die verständnisvolle Unterstützung meines Vorhabens. Danke!

Auch möchte ich die Gelegenheit nutzen, mich bei all den Studenten zu bedanken, die mit mir zusammen als Betreuerin ihre wissenschaftlichen Arbeiten gemeistert haben. Die Zusammenarbeit hat mir große Freude bereitet und gab mir die Möglichkeit, viele wertvolle Erfahrungen zu sammeln. Die folgenden Arbeiten sind im Rahmen meiner Promotion und unter meiner Anweisung entstanden, welche auch an geeigneter Stelle zitiert sind: Rebecca Rau – Hauptseminar (HS), Ömer Kaya – HS, Projektarbeit und Masterarbeit, Johannes Krähmer – HS, Christian Thein – HS und Masterarbeit, Dominik Gudd – HS, Silvia Kaltwasser – Bachelorarbeit, Matthias Francke – HS und Projektarbeit, Sebastian Schenk – HS, Benjamin Gold – HS, M. Ali Maktabi – HS und Hanna Schott – Masterarbeit. [Rau 2016; Kaya 2016, 2017a, 2017b; Krähmer 2016; Thein 2017, 2016; Gudd 2017; Kaltwasser 2017; Francke 2018, 2017; Schenk 2017; Gold 2017; Maktabi 2019; Schott 2019]

Kurzfassung

Autor: Maxi Weichenhain

Titel: Ein Petri-Netz-Tabellen-Modell zur Anwendung im klassischen und agilen Projektmanagement

Projektmanagement (PM) und Petri-Netze (PN) haben als Gemeinsamkeit, dass Aktivitäten in Abhängigkeit von Ereignissen ausgeführt werden, welche dann selbst wieder Ereignisse erzeugen können. Wenn die Anwendungsdomäne PM und das Modellierungsmittel PN zusammentreffen, besteht die Möglichkeit Synergieeffekte entstehen zu lassen. Um ein Modell des agilen und klassischen PM in ein PN-Modell zu überführen, sind dazu beide Welten zur Modellierung und Simulation zu vereinen. Es existieren durchaus eine Reihe von Ansätzen auf diesem Gebiet, allerdings ohne Berücksichtigung wichtiger Teilaspekte, wie z. B. die der praktischen Anwendbarkeit. Die vorliegende Arbeit legt deshalb einen zyklischen Ablauf zugrunde, der mit strukturierten Tabellen beginnt. Diese kommen dem Vorgehen im praktischen PM entgegen, erreichen aber durch die Systematisierung die Überführbarkeit in das PN-Modell. Das PN-Modell legt eine Variante zugrunde, die aus der Sicht der Anwendungen verschiedene PN-Erweiterungen wie hierarchische, zeitbewertete oder gefärbte Netze unter Verwendung strukturierter Marken aufnimmt. Teilaufgaben werden in hierarchische PN-Konstrukte gekapselt, die anpassbar durch das innere PN, das Verhalten von Objekten des PMs beschreiben. Typische Objekte wie Aktivitäten und Ressourcen sind durch Eigenschaften wie Atomarität, Wiederverwendbarkeit und Zeitverbrauch gekennzeichnet. Wechselwirkungen zwischen Konstrukten werden über PN-Elemente beschrieben. Der gesamte Projektplan wird dadurch aus Konstrukten und deren Beziehungen aufgestellt. Das PN bildet dabei die zentrale Ablauflogik. Die Umsetzung beruht darauf, die Dateneingabe von der Planung und den aktuellen Zwischenständen in der Tabellenstruktur vorzunehmen, und im Hintergrund automatisch das PN zu generieren bzw. zustandsabhängig zu verändern. Dafür wird in der Arbeit ein experimentelles Tool beschrieben. Anschließend sollen die Ergebnisse durchgeführter Simulationen mit einem Standard-PN-Tool erneut in die bestehende Tabelle zurückgeführt werden. Diese grundlegenden Vorteile des PN-Konzeptes können so vollumfänglich ausgenutzt und für den Projektmanager auch ohne PN-Kenntnisse Verwendung finden. Dadurch wird die Methode dynamisch und ist für verschiedenste Projekte universell einsetzbar.

Abstract

Author: Maxi Weichenhain

Title: A Petri Net Table Model Applied to Classic and Agile Project Management

Project Management (PM) and Petri Nets (PN) have in common that activities are carried out depending on events, which can generate events by themselves. The combination of the application domain PM (agile and traditional) and the modelling tool PN may result in advantageous synergy effects. In order to transform a model of PM into a powerful PN model, these two concepts must be merged for mapping and simulation. There are a number of approaches in this area, with no consideration of important aspects such as practical applicability.

The present thesis is therefore based on a cyclic process, starting with structured tables. These are in line with the procedure of practical PM but achieve through systematization the transferability into the PN model. The PN model is based on a powerful model with suitable extensions and necessary interpretations from the field of hierarchical, time-valued or coloured high-level PNs using structured token.

Subtasks are encapsulated in hierarchical PN constructs that describe the behavior of PM objects in an adaptable way through the inner PN. Typical objects such as activities and resources are characterized by properties such as atomicity, reusability and time consumption. Interactions between constructs are described by PN elements. The objects of modelling for the entire project plan are based on constructs with their relationships to each other, with the PN forming the single point of processing logic.

The implementation is based on entering the data from the planning and the current, intermediate statuses into a fixed, defined table structure and to automatically generate the PN in the background or to change it depending on conditions. The paper describes an experimental tool for this purpose. Afterwards, simulations are to be carried out on the basis of the models, and their results are to be incorporated back into the table. The advantages of the PN concept can thus be fully exploited and can be used by the project manager without requiring special PN knowledge. The method is based on a dynamic consideration and can be used universally for a wide range of projects.

Inhaltsverzeichnis

Abbildungsverzeichnis.....	V
Tabellenverzeichnis	VII
Formelverzeichnis.....	IX
Abkürzungsverzeichnis	XI
1 Einleitung, Forschungsfrage und Aufbau der Arbeit.....	1
1.1 Motivation	1
1.2 Forschung	3
1.3 Zielsetzung und Abgrenzung der Arbeit	5
1.4 Aufbau der Arbeit.....	6
2 Projektmodellierung und -management	9
2.1 Überblick von Projektmanagement und deren angrenzenden Bereiche.....	9
2.1.1 Projektmanagement.....	10
2.1.2 Agiles Vorgehen und agile Methoden.....	14
2.1.3 Gegenüberstellung Geschäftsprozess-, Workflowmanagement und diskrete Modelle in Bezug auf das Projektmanagement	16
2.2 Tools für das Projektmanagement	20
2.3 Petri-Netze	22
2.3.1 Klassische Petri-Netze	27
2.3.2 Ausgewählte Erweiterungen zu Höheren Petri-Netzen.....	29
2.4 Projektmanagement und Projektmodellierung mit Petri-Netz-Konzepten.....	35
2.4.1 Existente Petri-Netz-Ansätze des Projektmanagements	36
2.4.2 Nutzen von Petri-Netz-Konzepten im PM-Umfeld.....	38
2.4.3 Zusammenfassung.....	42
2.5 Ziele und Abgrenzung	45
3 Projekte als Gegenstand von diskreter Modellierung	47
3.1 Modellierungsgegenstand	48

3.2	Eigenschaften von Objekten	50
3.2.1	Atomare Einheit.....	51
3.2.2	Informationelle und physische Objekte	52
3.2.3	Objekte mit quantitativen Angaben	55
3.2.4	Bedingungen, Vorgänger und Nachfolger	57
3.3	Objektbeziehungen.....	59
3.4	Modellierung	60
3.4.1	Tabellenstruktur	62
3.4.2	Darstellungsform	64
3.4.3	Anwendung im Projektkreislauf	64
3.4.4	Prospektives Projektcontrolling.....	66
4	Entwicklung eines Petri-Netz-Modells für Projekte.....	69
4.1	Modellierungsgegenstand im Petri-Netz.....	69
4.1.1	Konzept der Konstrukte und Schnittstellen	70
4.1.2	Einbindung der Atomarität	71
4.2	Beschreibung des verwendeten Petri-Netz-Modells	73
4.3	Tabellenstruktur	80
4.4	Modellierung	87
4.4.1	Objektbeziehungen, Bedingungen, Vorgänger und Nachfolger.....	88
4.4.2	Konstrukte und deren Schnittstellen.....	89
4.4.3	Anpassbarkeit, Erweiterbarkeit und Nachweis.....	108
4.5	Strukturierte Objekte (Marken).....	108
4.5.1	Tabellarische Datengrundlage	110
4.5.2	Genereller Aufbau und Verarbeitung strukturierter Marken	112
4.5.3	Regeln zur Anwendung bei der Verarbeitung und Zurückführen der Erkenntnisse in die Tabelle.....	116
4.6	Entwurf und Zugriff auf die Modelle im Projektkreislauf.....	117
4.6.1	Die Netzgenerierung	119
4.6.2	Besonderheit der Ermittlung der korrekten Anfangsmarkierung	120
4.6.3	Besonderheit der verschiedenen Zeitrestriktionen.....	123
4.7	Darstellungsform.....	125
5	Einführung und Aufbau eines Beispielprojektes.....	127
5.1	Allgemeine Vorstellung des zugrundeliegenden Projektes.....	127
5.2	Reales anonymisiertes Praxisbeispiel	131

6 Die Projekt-Petri-Netz-Modell-Erweiterung	135
6.1 Beispiel zur Umsetzung für den weiteren Verlauf der Arbeit	135
6.2 Transformation in die Tabellenstruktur	136
6.3 Erstellung und automatische Generierung des Beispiels.....	138
6.4 Beispieldarstellung als Projekt-Petri-Netz-Modell stark zusammengefasst...	141
6.5 Ausschnitte des generierten Projekt-Petri-Netz-Modells	142
7 Zusammenfassung und Ausblick	145
7.1 Inhalt der Arbeit.....	145
7.2 Möglichkeiten, Grenzen und zukünftige Arbeiten	148
Literaturverzeichnis	XIII
Anhangsübersicht	XXXIII
A Tooleinsatz im Projektmanagement.....	XXXIV
B Weitere Phasen des Projektmanagements	XXXVII
C Tabellengrundlage des Beispielprojektes	XXXVIII
D Vergleich von Geschäftsprozess-, Workflowmanagement und diskrete Modelle mit dem Projektmanagement.....	XXXIX
E Petri-Netze und das Projektmanagement	XLIII
F Detailliertere Information zum anonymisierten Beispiel	XLIX
G Vergleich Publikationen zu Petri-Netz-Ansätzen im Projektmanagement.....	LIV
H Legende für Abbildungen des Projekt-Petri-Netz-Modells	LXIII

Abbildungsverzeichnis

Abbildung 1: Vier Phasen des Projektmanagements	12
Abbildung 2: Einsatz- & Bekanntheitsgrade von Projektmanagementsoftware	21
Abbildung 3: Grobe Einteilung Petri-Netze	23
Abbildung 4: Übersicht klassischer Petri-Netze	25
Abbildung 5: Übersichten der Grundannahmen & der Analyse von Petri-Netzen.....	26
Abbildung 6: Auswahl an Sonderkanten	28
Abbildung 7: Zusammenspiel Vorgänger/Nachfolger ohne zeitliche Interpretation.....	57
Abbildung 8: PDCA-Zyklus	65
Abbildung 9: Konstrukt-Idee	71
Abbildung 10: Auflösung kapazitätsbeschränkter Platz	79
Abbildung 11: Benötigte Informationen und Aufbau der Tabellen.....	83
Abbildung 12: Konstrukte und deren Beziehungen.....	90
Abbildung 13: Modellierung globales Zeitkonzept - Zeit-Konstrukt.....	91
Abbildung 14: Lokale Zeit-Konstrukt	92
Abbildung 15: Verkettung von Sprint-Konstrukten	93
Abbildung 16: Das Sprint-Konstrukt.....	95
Abbildung 17: Aufbau Aufgaben-Konstrukt	97
Abbildung 18: Aufbau Phasen-Konstrukt.....	99
Abbildung 19: Aufbau Ressourcenverwaltung mit Schnittstellen.....	101
Abbildung 20: Fehlerhafte Ressourcenzuordnung	103
Abbildung 21: Aufbau Ressourcengeneration mit Vielfachheit.....	105
Abbildung 22: Aufbau Ressourcengeneration mit Schnittstellen.....	107
Abbildung 23: Ausschnitt & Beispiel - Verwendung einer strukturierten Marke.....	113
Abbildung 24: Beispiel Verwendung strukturierter Marke – stark vereinfacht	115

Abbildung 25: Kreislauf des Projekt-Petri-Netz-Modells.....	118
Abbildung 26: Anfangsmarkierung.....	122
Abbildung 27: Einbindung Zeitrestriktionen und dessen Aussagen	124
Abbildung 28: Grafische Anordnung der Konstrukte	125
Abbildung 29: Schematische Darstellung der Building Blocks.....	129
Abbildung 30: Zusammenhang Building Blocks, Arbeitspakete und Sprints.....	131
Abbildung 31: Grobes Zusammenspiel der Tabellen.....	134
Abbildung 32: Projektbeispiel Fakten.....	135
Abbildung 33: Schritte automatischer Generierung.....	138
Abbildung 34: Reihenfolge Sprinterstellung – abgeänderte Grundlage	139
Abbildung 35: Überblick Programm zur Erstellung des Plan-Ist-Modells	140
Abbildung 36: Projekt-Petri-Netz-Modell Skizze – Beispiel.....	141
Abbildung 37: Beispiel Projekt-Petri-Netz-Modell - Zeit-Konstrukt Gesamtblick	142
Abbildung 38: Beispiel Projekt-Petri-Netz-Modell - Aufgabe und Vorgänger	143
Abbildung 39: Zeit-Konstrukt – Zugriff auf Dauer im Aufgaben-Konstrukt	144
Abbildung 40: Ausschnitt der Komplexität aus dem Projektplan.....	XXXVIII
Abbildung 41: Dreimonatsplanung Arbeitspakete.....	LI

Tabellenverzeichnis

Tabelle 1: Zusammenspiel Projektmanagement zu Petri-Netz.....	81
Tabelle 2: Auszug aus der Tabelle Projektplanung zur Datenpflege von Sprints	85
Tabelle 3: Anfangsmarkierung Zeit-Konstrukt.....	93
Tabelle 4:Tabelle für Ressourcenzuordnung	110
Tabelle 5:Tabelle für Ressourcenbeurteilung	111
Tabelle 6: Bedingungen/Berechnungen Anfangsmarkierung einer Aufgabe	121
Tabelle 7: Projektplanung – Beispiel.....	136
Tabelle 8: Ressourcen und Regenerationsphase – Beispiel.....	137
Tabelle 9: Ressourcenzuordnung und Ressourcenbeurteilung – Beispiel	137
Tabelle 10: Eignung PM-Methoden mit Bezug zur Projektgröße	XXXV
Tabelle 11: Unterschiede zwischen Prozess und Projekt.....	XXXIX
Tabelle 12: Projektplan Beispiel.....	LII
Tabelle 13: Backlog Beispiel.....	LIII
Tabelle 14: Vergleichstabelle Publikationen Petri-Netze in Projektmanagement.....	LV

Formelverzeichnis

Formelzeichen	Bedeutung
Petri-Netz	
P, p_i	Platzmenge, Platz
T, t_i	Transitionsmenge, Transition
F	Kantenmenge (Flussrelation)
(p_i, t_j)	Kante
$V(p_i, t_j)$	Vielfachheit einer Kante
$K(p_i)$	Kapazität eines Platzes
$m(p_i)$	Markierung eines Platzes ohne Kapazität
$m^k(p_i)$	Aktuelle Markierung
$m^{k+1}(p_i)$	Folgemarkierung
m_0	Anfangsmarkierung
sf	Schaltfähig
s	Schalten
t_z	Zeitbereich
$t_z(k)$	Zeitpunkt
Wt_v	Menge von Zeitverzögerungen
UN	Umgebungsnetz
PU_i	<i>Platzunternetz_i</i>
P_{PU_iR}	Randplatz vom <i>Platzunternetz_i</i>
Mengenlehre	
\cap	Schnittmenge
\emptyset	Leere Menge
\cup	Vereinigung
\subseteq	Teilmenge von oder gleich
\mathbb{N}	Natürliche Zahlen
\mathbb{R}	Reelle Zahlen

Aussagenlogik

\rightarrow	Abbildung
$:$	Gilt
$ $	Mit der Eigenschaft
\forall	Für alle
\in	Element von
\wedge, \vee	Logisches Und, Logisches Oder
\exists, \nexists	Es gibt, Es gibt nicht
\leftrightarrow	Äquivalenz

Allgemein

Δ	Delta
$+, -$	Plus, Minus
\cdot	Zahlenprodukt
\times	Mengenprodukt/Kreuzprodukt
∞	Unendlich
\geq, \leq	Größer oder gleich, kleiner oder gleich
\neq	Nicht gleich

Abkürzungsverzeichnis

A

A · Aufgabe

AfZ · Arbeitsfreie Zeit

AZ · Arbeitszeit

B

BB · Building Blocks

BPM · Business Process Management

BPMN · Business Process Model and Notation

C

CPM · Critical Path Method

CPN · Coloured Petri Nets

D

D · Dauer

DE-Modellen · Discrete Event-Modellen

DES · Discrete Event System

E

ED · Enddatum

F

FM · farbige Marke

G

GPM · Geschäftsprozessmanagement

H

HCPN-ST · High Level Coloured Petri-Netze mit
strukturierten Token

HPN · Höhere Petri-Netze

HS · Hauptseminar

I

ID · Identifikationsnummer

IDEF0 · Icam DEFinition for Function Modeling

K

K · Koordinator

L

L/M · Lernen/Modifizieren

M

M · Menge

MA · Mitarbeiter

ML · Meta Language

MS · Microsoft

MZ · Modellzeit

N

n · natürliche Zahl

O

OMG · Object Management Group

P

P · *Projekt*

PDCA · *Plan Do Check Act*

PERT · *Program Evaluation and Review Technique*

PM · *Projektmanagement*

PMO · *Project Management Office*

PN · *Petri-Netze*

PNFL · *Petri-Netz und Fuzzy-Logik*

PNTs · *Project Network Techniques*

PPNM · *Projekt-Petri-Netz-Modell*

Pr/T-Netze · *Prädikat/Transitions-Netze*

P-T-Netz · *Platz-Transitions-Netz*

R

R · *Ressourcen*

RA · *Ressourcenanforderung*

RAS · *Ressourcenallokationsstrategien*

RB · *Ressourcenbereitstellung*

RE · *Ressourcenerneuerung*

RG · *Ressourcengeneration*

RID · *Ressourcen ID*

RP · *Ressourcenpool*

RR · *Ressourcenregeneration*

R-St · *Ressource Stück*

R-V · *Ressource Vielfachheit*

RV · *Ressourcenverwaltung*

RZ · *Ressourcenzuordnung*

S

S · *Sprint*

SD · *Startdatum*

SM · *schwarze Marke*

SPN · *Stochastic Petri Nets*

St · *Stück*

StM · *strukturierte Marke*

SW · *Software*

T

T · *Transition*

TCPN · *Time Coloured Petri Net*

TPG · *Task Precedence Diagramm*

TPN · *Time Petri Net*

U

UML · *Unified Modelling Language*

V

V · *Vielfachheit*

W

WBS · *Work Breakdown Structures*

WfMC · *Workflow Management Coalition*

WfMS · *Workflow-Management-System*

X

XML · *Extensible Markup Language*

Y

YAWL · *Yet Another Workflow Language*

1 Kapitel

Einleitung, Forschungsfrage und Aufbau der Arbeit

Projekte sind so alt wie die Idee selbst. Sie blicken auf eine lange Historie zurück, weit bevor dieser Begriff dafür Verwendung fand. Früher wurden die damit verbundenen Tätigkeiten ohne ein umfangreiches Repertoire aus Methoden, Strukturen, Instrumenten oder Vorgaben ausgeführt. Dies erfolgt heute in den Projektphasen und mittels einer strukturierten Vorgehensweise.

Zurückblickend können der Bau der Pyramiden in Ägypten vor über 4500 Jahren oder zeitgenössische Baukonzepte in die Vorhaben von beachtlicher Größe, Innovationskraft und Aufwand als erfolgreiche Projekte eingereiht werden. Heutzutage begegnen wir Projekten in allen Lebensbereichen, angefangen bei der Forschung, Wissenschaft, Politik oder Softwareeinführungen. Die Geschichte des modernen Projektmanagements (PM), so wie wir es heute kennen, hat seinen Anfang wesentlich später gefunden. Als zwei der bedeutendsten Pioniere aus dem frühen 20. Jahrhundert können Frederick Taylor mit seinen ausführlichen arbeitswissenschaftlichen Studien und Henry Gantt (Gantt-Diagramm) genannt werden. [vgl. Litke 2005a, 8 ff]

1.1 Motivation

Egal in welchem Bereich, ist die Frage doch immer die Gleiche: Wie führt dies zu einem Projekt und dem dazugehörigen Projektmanagement? Alle haben gemeinsam, dass ein Ziel mit erklärter Problemstellung verfolgt wird. Für dessen Lösung ist meistens fachübergreifende Kenntnis erforderlich, welche außerhalb des Routineablaufs erfolgt. Projekte sind an einen eingeschränkten finanziellen und personellen Rahmen sowie an eine Kosten- und Erfolgskontrolle gebunden. [vgl. Litke 2005a, 8 ff]

In den letzten Jahren spielt der Nutzen von digitalen Technologien und die damit einhergehende erhoffte Verbesserung des Geschäftes eine immer größere Rolle. Die Wandlungen, gefolgt von ihren Herausforderungen bezüglich Zeitersparnisse, Kostenreduzierungen, Regulierungen, steigende Kundenanforderungen, gesetzlichen Vorschriften, wie auch jegliche Art von Optimierungen müssen immer wieder aufs Neue geplant und erfolgreich umgesetzt werden. Dies verlangt eine ständige Verbesserung und

Digitalisierung des Projektmanagements. Diese Veränderungen sind auch dem dynamischen Charakter und dem fortschreitenden Wandel hin zum agilen Projekt geschuldet. Für die Bewältigung von Anforderungen, egal welcher Natur, bildet das Projektmanagement eine wichtige Grundlage. Aber was genau bedeutet Management und wo kommt es überall vor? Es ist nicht nur die fachlich definierte Projektabwicklung isoliert zu betrachten. Ein gut funktionierendes Projektmanagement bildet die Bausteine für die Tätigkeiten zur Erhaltung und Verbesserung der Wettbewerbsfähigkeit eines jeden Unternehmens. Der Einsatz von Projektmanagementmethoden hat einen hohen Abdeckungsgrad erzielt, was aus der Vielzahl des Angebotes an Methoden und Hilfsprogrammen (Tools) auf dem Markt ersichtlich ist. Hierbei ist SCRUM eines der erfolgreichsten agilen Methoden im Bereich des Projektmanagements. Es versteht sich allerdings lediglich als ein Framework und liefert keine Tools zum Handling und zur Umsetzung. Unabhängig von der Auswahl der Projektmanagementmethode ist ein Tooleinsatz unabdingbar. Eine Beobachtung ist, dass mehrere Projektmanager oder Personen für das Project Management Office (PMO) tätig sind, aber an der Toolunterstützung oft gespart wird. Im täglichen Projektalltag ist zu erkennen, dass Excel und PowerPoint aufgrund ihrer weiten Verbreitung für jegliche Tätigkeiten herangezogen werden, was eine Studie von techconsult belegt. [vgl. Groß 2014; vgl. Corcoran 2019]

Laut [Gobert 2008] besteht am Markt ein Überangebot von mehr als 300 Software-Tools, welches die Entscheidungen für eine branchenspezifische Software (SW) – optimiert auf Projektmanagement – erschwert. Hinzukommen verschiedenste Erweiterungen (beispielsweise Add-ons/-ins), welche die Microsoft (MS) Office® Funktionalitäten auf einzelne Projektmanagementbereiche spezifisch erweitern. Simulationsmöglichkeiten sind damit noch nicht gegeben. Dies bedeutet, dass damit die Lücke der tabellarischen Grundlage mit einem im Hintergrund gelegten Mechanismus zur Simulation nicht geschlossen und somit auch kein anderes Qualitätsniveau bei ausgewählten Projektmanagementaktivitäten erreicht werden kann. [vgl. Kaltwasser 2017, 2]

Die Studie kommt zu der Erkenntnis, dass vor allem Microsoft Excel einen Einsatzgrad aufweist, der den Schluss zulässt, dass es nicht nur vielfach Anwendung findet, sondern dazu zahlreich ergänzend neben Spezial- oder Branchenlösungen eingesetzt wird. Die Gründe können vielfältiger Natur sein, wie zum Beispiel Einarbeitungszeit, Budget, Auswahlprozess oder der Austausch von Dateien. [vgl. Groß 2014; vgl. Kaltwasser 2017, 2]

1.2 Forschung

Eine umfassende Analyse im Rahmen des Chaos Report der Standish Group, in dem zwischen 1994 und 2015 mehr als 50.000 Projekte weltweit aus verschiedenen Branchen und Projekten, angefangen von kleinen Verbesserungen bis hin zu umfangreichen Systemerneuerungen ausgewertet wurden, kam zu folgendem Ergebnis: [Hastie und Wojewoda 2015]

- Im Jahr 2015 sind lediglich 29 % aller durchgeführten Unternehmensprojekte als voller Erfolg (in time und in budget, Verwirklichung des vollen Funktionsumfanges) bezeichnet worden.
- 71 % der Projekte verfehlten die geplanten Zielvorgaben mindestens in einer der drei oben genannten Dimensionen oder es kam sogar zu einem Abbruch.

Die Projektmodellierung als Teilaufgabe des Projektmanagements, ist ein Grundpfeiler für eine erfolgreiche Umsetzung eines Projektes. Das primäre Ziel der Projektmodellierung begründet sich in der Erstellung eines Projektmodells, welches die zur Schaffung eines gemeinsamen Verständnisses und die Ausgangsbasis zur Bearbeitung und Analyse nachgelagerter Aufgaben innerhalb des Projektmanagements darstellt. Je präziser dies ausfällt, desto besser können Projektaktivitäten aufeinander abgestimmt und Auswirkungen bei Abweichungen analysiert werden. Der Aufwand eines vollumfänglichen Projekt-Tools für qualitativ hochwertige Modelle ist immens und somit kosten- und zeitintensiv, gemessen an allen enthaltenden Kosten (Installation, Lizenz, Lernaufwand, Kommunikationskosten durch Verbreitung, Verbreitungsgrad, gemeinsame Bearbeitung von Dokumenten, intuitiv und selbsterklärend, usw.). Es stellt zudem keine Garantie für eine komplette Abdeckung der Bedürfnisse des jeweiligen Projektmanagements und des zugrundeliegenden Projektes dar. Ebenfalls ist damit noch

nicht abgedeckt, dass der sich daraus ergebene Aufwand auf auftretende Eventualitäten eines Projektes zu berücksichtigen ist und in einem Modell verarbeitet werden muss. Dabei wird sowohl das Eintreffen von unerwarteten Ausnahmen bzw. Fehlern oder Verzögerungen nicht vollumfänglich im Rahmen von der Projektmanagementsoftware abgedeckt. [vgl. Deng, Wendt, und Rolstadås 1998, 79; vgl. Thein 2017, 1 ff]

Gerade weil diese Komplexität bei der geeigneten Toolunterstützung besteht, wird die Frage nach dem Tool oftmals nicht eingehend, nicht ausreichend oder gar nicht beleuchtet. Hier beweist Microsoft Office®, insbesondere Excel, dass sich eine große Verbreitung und der Bekanntheitsgrad über viele andere Kriterien hinwegsetzt. Ebenfalls sind zugrunde gelegte Logiken, enthalten in der Software der Tools im PM, nicht zentral abgebildet und Änderungen können so nicht effizient erfolgen. Dies ist immer mit Änderungen am Funktionsumfang oder mit Erweiterungen verbunden und erfordert einen erheblichen Aufwand.

Etwaige anfallende Störungen, die gerade durch den neuen dynamischen Charakter moderner Projekte und deren Vordringen mittels agiler Methoden auftreten können, sind nur schwer, die resultierenden Beeinträchtigungen der Abläufe dagegen nicht planbar. Konzepte von Petri-Netzen (PN) werden zwar in der Literatur als potenziell vielversprechende Methode zur Modellierung dynamischer Projekte angesehen, aber eine aktive Integration oder Forschung, um sie als eine Methode der Projektmodellierung im Projektmanagement zu etablieren, wurde bis heute nicht hinreichend unternommen. In den letzten Jahren gab es eine Reihe von wissenschaftlichen Publikationen, die sich mit der Verwendung eines Modells (basierend auf dem theoretischen Modell eines Petri-Netzes) auseinandergesetzt haben. Zu einer Ausgestaltung oder sogar Marktpräsenz dieser Systeme ist es bislang nicht gekommen. [vgl. Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000, 517; vgl. Ashok Kumar und Ganesh 1998, 49; vgl. Thein 2017, 1 ff]

Die Fähigkeit zur Simulation und Modellierung der Projektumgebung in den IT-Methoden in Form von Modellen, muss demzufolge als ein wesentlicher Wettbewerbsfaktor angesehen werden. Dabei ist sicherzustellen, dass der Modellgedanke bestehen bleibt und somit die Basis für weitere Verwendungsmöglichkeiten aus dem Umfeld des Modells potenziell nutzbar sind. So ergibt sich die Frage, ob der Einsatz von

Konzepten aus den Bereichen des Petri-Netzes zur Steigerung der Erfolgsquote von Projekten beiträgt, indem dieses bei der Modellierung bzw. Konstruktion von Projektmodellen eingesetzt wird. Ein wesentlicher Grund für die niedrige Erfolgsquote bei Projektumsetzungen ist die große Anzahl an Störungen, was auf den dynamischen Charakter moderner Projekte zurückzuführen ist. [vgl. Deng, Wendt, und Rolstadås 1998, 79; vgl. Thein 2017, 1 ff]

1.3 Zielsetzung und Abgrenzung der Arbeit

Unter der Maßgabe der Effizienz bekommt die Modellierung und Analyse auf der Grundlage von tabellarischen Steuerstrukturen eine immense Bedeutung. Neben der reinen Kostenreduktion sind die Kundenanforderungen (z. B. Reaktionsfähigkeit auf veränderte Situationen sowie Einhaltung und Überwachung der Planung) nicht zu vernachlässigen. Es muss das Ziel sein, die Projektplanung zu simulieren.

Einen Lösungsansatz dafür bietet die Modellierung einer simulierten Projektplanung. Hierbei soll auf die in der Praxis erprobten theoretischen Grundlagen des Petri-Netzes als Modellierungsansatz zurückgegriffen werden. Um dies zu gewährleisten, muss die Basis für eine geforderte schnelle Auswertung und Simulation der Projektplanung sichergestellt sein. Dies gilt auch für deren Umsetzung als Abfolge (Grundlage hierfür bietet das Modell) und die praktische tabellarische Struktur (wie Kosten zur Erstellung bzw. kontinuierliche Anpassung des Modells und dessen Überführung in vorgesehene PM-Tools). Diese Arbeit beruht auf dem Konzept, dass die Modellierungsmethodik dem Projektmanager selbst nicht in Erscheinung treten soll, sondern die gewohnten tabellarischen Strukturen in geeigneter Form als Ausgangspunkt heranzieht.

Der Lösungsansatz der simulierten Projektplanung ist in der Forschungsfrage begründet, welche sich wie folgt ableiten lässt:

Ziel ist es, eine Analyse und Formalisierung auf der Grundlage einer Tabellenstruktur für die Bereiche des agilen und klassischen Projektmanagements zu ermöglichen. Dies soll durch eine Integration geeigneter Forschungsfelder aus dem Umfeld der Petri-Netz-Strukturen resultieren. Ebenfalls wird damit ein Simulationskontext geschaffen, der im Lebenszyklus der Planung anhand von gepflegten Tabellen aktuelle Informationen zur Verfügung stellt.

An diese modulare und simulierbare Projektmodellierung auf der theoretischen Grundlage eines Petri-Netzes wird die Anforderung der Atomarität sowie die Abbildung von hierarchischen Konstruktionen des darzustellenden Ablaufs gestellt. Weiterhin muss die Formalisierung von Schritten und Objekten im Prozessablauf für eine geeignete Modellierung erfolgen. Transformationsregeln aus den formalisierten Tabellen, einschließlich Hierarchiestrukturen, müssen zur Erstellung geeigneter erweiterter Petri-Netze resultieren.

Ferner erfolgt eine Einschätzung der Möglichkeiten und Leistungsfähigkeit der entwickelten Methodik mit einer vorgenommenen Verallgemeinerung des Projektcharakters.

Durch eine Verminderung der administrativen Aufwände sowie einen zeit- und realitätsnahen simulierten Ablauf kann durch eine Steigerung der Produktivität, der Reaktionszeit und der Analysefähigkeit ein erheblicher Mehrwert eintreten.

Neben der Einbindung von praktischen Erfahrungen muss dieses Konzept den Anforderungen eines modernen Projektmanagements Genüge tragen. Für eine praxisnahe Gestaltung werden die Nachweise der methodischen Schritte durch eine Modellierung und Simulation an einem Beispiel eines weltweit agierenden Beratungsunternehmens aus verschiedenen Projekten veranschaulicht.

Diese Arbeit beschränkt sich auf den methodischen Ansatz und der somit geschaffenen Möglichkeit der Umsetzung. Es ist nicht das Ziel, die Eignung von Petri-Netz-Konzepten zur Bearbeitung aller Aufgaben des Projektmanagements zu bewerten. Viel mehr wird sich diese Arbeit auf Teilbereiche mit einem fiktiven Beispiel im Rahmen der Steuerung und Planung beziehen, die erzielten Ergebnisse vollumfänglich beleuchten und auf Praxistauglichkeit untersuchen.

1.4 Aufbau der Arbeit

Im Rahmen der Dissertation wird zur Einführung eine nähere Betrachtung des Projektmanagements in Bezug auf das Workflowmanagement und Geschäftsprozessmanagement im zweiten Kapitel vorgenommen. Das zweite Kapitel gibt ebenfalls einen Einblick in die vorhandene Toollandschaft des Projektmanagements. Die

thematischen Grundlagen der Petri-Netze und dessen Einsatzgebiete in verschiedenen Forschungsfragen zum Zwecke der Projektmodellierung werden auf deren Notwendigkeit und Vorteilhaftigkeit untersucht. Das dritte Kapitel identifiziert die notwendigen Objekte und Informationen, um die Projektmodellierung in den Kontext der diskreten Modellierung zu setzen. Hierbei wird sich der Fragestellung der Atomarität zugewendet und die geeignetste Ebene der benötigten Modellierungsobjekte identifiziert.

Das folgende Kapitel vier zielt darauf ab, die aus Kapitel drei identifizierten Anforderungen auf das PN zu übertragen und als ein Modell umzusetzen. Die Idee der Konstrukte zur Abbildung dieser Anforderungen wird näher erläutert und an einzelnen dafür entwickelten Konstrukten aufgezeigt. In dem hier vorgestellten Ansatz werden auf der Grundlage von Projekt-Objekten sogenannte Konstrukte identifiziert. Diese bilden den Modellierungsgegenstand und erfahren eine ausführliche Betrachtung in diesem Kapitel. Ein weiteres Ziel ist die Beschreibung des zu verwendeten PN-Modells und der Tabellenstruktur zur Dateneingabe. Abschließend wird der Kreislauf von der Dateneingabe, über die Modellierung, bis zur Gewinnung von Erkenntnissen beleuchtet.

Das fünfte Kapitel verdeutlicht die praktische Relevanz anhand eines anonymisierten Praxisbeispiels eines Informationssystemprojektes. Auszüge dieses Praxisbeispiels werden im anschließenden sechsten Kapitel unter Anwendung der identifizierten Modellierungsmethoden sowie einer Anreicherung formaler Syntax und Semantik und unter entsprechender Berücksichtigung der Anforderung einer Simulation dargestellt. Das abschließende siebte Kapitel betrachtet die Möglichkeiten und Grenzen der vorgestellten und simulierten Petri-Netz-Projektmodellierung, würdigt die Arbeit kritisch und gibt einen Ausblick auf offene Forschungsfragen.

2^{Kapitel} Projektmodellierung und -management

Dieses Kapitel dient dazu, ein grundlegendes Verständnis für Methoden und Definitionen in Bezug auf das Projektmanagement und der Petri-Netze herzustellen. Petri-Netze sind die gewählte Methode, um Prozessabläufe dynamisch darzustellen. Somit eignen sie sich ebenfalls als Grundlage für die Darstellung von Geschäftsprozessen. In der Theorie werden Petri-Netze den nebenläufigen Prozessen zugeordnet. Sie finden aber auch außerhalb des Software Engineerings vielerlei Einsatzmöglichkeiten, wie beispielsweise bei Workflow-Netzen. [vgl. Bauske 1999, 72; vgl. Scharfe 2013, 29]

Um das einheitliche Verständnis zu fördern, werden anfangs einzelne Begriffe des Projektmanagements, Geschäftsprozessmanagements und Workflows definiert und voneinander abgegrenzt. Anschließend wird der Stand der Technik bei Petri-Netzen aufgegriffen. Im letzten Abschnitt steht die Adaptivität von Petri-Netzen und dessen Nutzen im Vordergrund. Im Rahmen dieser Arbeit wird Adaptivität als Anpassbarkeit verstanden.

Ist das Projektmanagement so unterschiedlich zum Geschäftsprozessmanagement, oder finden sich viele Gemeinsamkeiten, welche den Ansatz bekräftigen, PN zur Modellierung von Projektmanagement als Alternative heranzuziehen? In der Praxis kommen häufig Modelle für die Beschreibung von Geschäftsprozessen zum Einsatz. Die Abstraktion eines betrachteten Ausschnitts unter Identifikation der wesentlichen Aspekte und deren Beziehungen, wird als Modell bezeichnet. [vgl. Fleischmann u. a. 2018, 19 f]

2.1 Überblick von Projektmanagement und deren angrenzenden Bereiche

Projektmanagement wurde in den letzten Jahrzehnten immer ausführlicher und vielfältiger in der Literatur behandelt. Dabei haben sich viele einzelne Themengebiete verselbstständigt und zu einer inflationären Entwicklung der verschiedensten Bereiche im Management geführt. Burghardt hat die einzelnen Managementbereiche in eine Einordnung zum Projektmanagement gebracht.

Im Rahmen dieser Dissertation wird zur Einführung eine nähere Betrachtung des Projektmanagements in Bezug auf das Workflowmanagement und Geschäftsprozessmanagement vorgenommen. Diese Teilbereiche in Verbindung mit Projektmanagement werden im späteren Verlauf auf eine angepasste Unterstützung mit dem theoretischen Modell eines Petri-Netzes untersucht. Abschließend wird eine Gegenüberstellung und Abgrenzung der nachfolgenden Begriffe gegeben. [vgl. Burghardt 2012, 16]

Der dynamische Charakter der heutigen Projekte ruft immer mehr die agilen Projektmanagementmethoden in den Focus. Das später gewählte Beispiel eines Informationssystemprojektes wird einen gemixten Ansatz vom klassischen und agilen Projektmanagement anvisieren, und somit die Komplexität an das zu simulierende und modellierende Modell erweitern. Es handelt sich um ein auswertungsorientiertes Informationssystemprojekt im Bereich eines Data-Warehouse-Systems.

2.1.1 Projektmanagement

Für die Definition des Projektmanagements hat der Projektbegriff einen maßgeblichen Einfluss. Ein **Projekt** ist ein Vorhaben, dem ein **eindeutiges Ziel** zugrunde liegt, welches vor Projektbeginn feststeht und allen Beteiligten bekannt ist. Es erfordert nicht nur eine **Begrenzung** in Bezug auf **Zeit, finanzielle Mittel, Ressourcen (R)**, sondern auch eine **inhaltliche** Einschränkung. Diese ist in Form von **Projektanforderungen** definiert, ohne die ein Projekt kein **eindeutiges Ziel** hat und somit keinem **Endergebnis** zusteuern kann. Zudem ist jedes Projekt **individuell** und weist meist eine **hohe Komplexität** auf. [vgl. Mangold 2011, 22]

Die Aufgabe des Projektmanagements lässt sich dahingehend in der Planung der Abläufe und Zeiten, Überwachung der Einhaltung von Plänen und des steuernden Eingreifens bei Abweichungen, die durch vielfältige Wechselwirkungen oder Einschränkungen vor und während eines Projektes eintreten können, ableiten. Projektdurchführung und Projektmanagement sind parallel zusammenarbeitende Prozesse. [vgl. Jakoby 2015, 6]

Das Projektmanagement gilt als Oberbegriff aller „planenden, überwachenden, koordinierenden und steuernden Maßnahmen [...], die für Um- oder Neugestaltung von [...] Problemlösungen erforderlich sind“. [Kuster u. a. 2011, 8]

Das Projektmanagement beeinflusst und bestimmt maßgeblich die verschiedenen Phasen eines Projektes. Im Rahmen des Projektmanagements kann eine Einteilung in vier wesentliche **Teilprozesse** erfolgen. Angefangen bei der Definition und Gründung des Projektes (**Initialisierung**), über die Planung aller Aktivitäten, bis hin zu den Arbeitszeiten und Ergebnissen (**Planung**). Dies bildet die Grundlage für den Projektstart und damit einhergehend die Steuerung der Projektdurchführung (**Steuerung**) bis hin zum Abschluss des Projektes durch die Validierung der erreichten Lösung (**Abschluss**). [vgl. Jakoby 2015, 7] Im weiteren Verlauf wird sich diese Dissertation auf die zwei Phasen der Planung und Steuerung beschränken.

Die wesentliche Aufgabe in der **Planungsphase** besteht darin, auf Grundlage von bereits vorhandenen Daten, die sich ergebenen Projektaufgaben hinreichend tief zu zerlegen und in einem Plan zusammenzustellen (beispielsweise anhand eines Projektstrukturplanes). Darauf aufbauend kann anschließend eine genaue Aufwandsschätzung, Termin- und Kapazitätsplanung resultieren. Im Anschluss kann die Personalplanung, Planung der Projektberichterstattung und -dokumentation geschehen. Nach erfolgter Planung können Aussagen in Bezug auf die Wirtschaftlichkeitsanalyse mit der Projektkostenplanung vorgenommen werden. Ein genauer Plan über Ablauf und Rahmenbedingungen des Projektes (Soll) ist das Ergebnis jener Phase. Dies bildet die Grundlage für die nun folgende Durchführung des Projektes. [vgl. Heilmann 2003, 7; vgl. Rau 2016, 3 f]

Der ständige Vergleich zwischen den festgelegten Soll-Daten und den tatsächlichen Ist-Daten ist die Kernaufgabe während der **Projektsteuerung**. Ziel ist es, präventiv mögliche Abweichungen so früh wie möglich zu erkennen, und nach erfolgter Analyse die gegensteuernden Maßnahmen einzuleiten. Liegen gravierende Abweichungen vor, müssen aufgrund der Annahme, dass der ursprüngliche Plan nicht mehr eingehalten und umgesetzt werden kann, Anpassungen vorgenommen werden. [vgl. Heilmann 2003, 7; vgl. Rau 2016, 3 f]

Dabei gilt es immer, die drei in sich oftmals im Widerspruch stehenden Größen Termin, Kosten und Ressourcen zu überwachen. Die Literatur greift die Abhängigkeit und Beziehung von diesen drei zentralen Eckpunkten des Projektes immer wieder auf. Sie bilden die Erfolgskriterien und werden für die Risikoanalyse herangezogen. Die direkte Beziehung zueinander signalisiert die Beeinflussung, die sich gegenseitig ergibt. Wird

eine Größe verändert, so wirkt sich dies direkt auf die anderen beiden aus. Wird beispielsweise angestrebt, das Projekt in kürzerer Zeit jedoch bei gleichbleibendem Umfang zu gewährleisten, müssen mehr Ressourcen hinzugezogen werden, welche ihrerseits weitere Kosten erfordern. [vgl. Litke und Kunow 2006, 16; vgl. Kaltwasser 2017, 10 f]

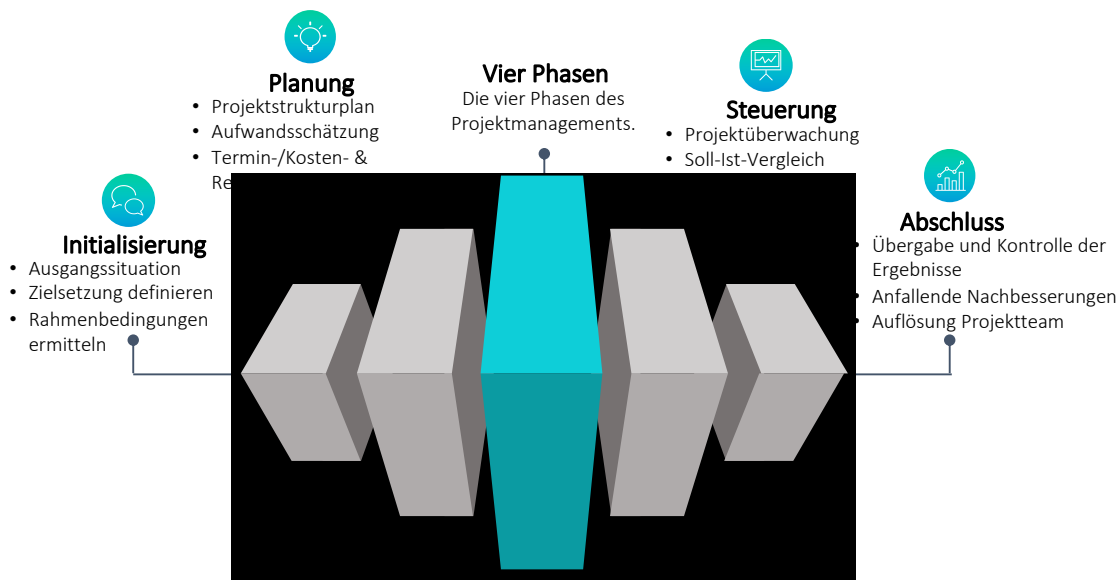


Abbildung 1: Vier Phasen des Projektmanagements [vgl. Jakoby 2015, 6]

Zur Ausführung der Projektsteuerung müssen Kennzahlen, wie geplante und effektive Dauer der einzelnen Projektphasen und des Gesamtprojektes, die geplanten und effektiven Kosten sowie der prozentuale Erfüllungsgrad miteinander ins Verhältnis gesetzt und ausgewertet werden. Diese Kontrollen müssen oftmals zeitkritisch und präventiv erfolgen, um noch positiv in das Projektgeschehen eingreifen zu können. Im Rahmen der Ressourcenkontrolle ist vor allem der Personaleinsatz immer eine schwer zu planende und zu analysierende Komponente. Das Auseinanderlaufen von Soll- und Istwerten kann darauf beruhen, dass Erfahrungswerte fehlen, nichtzutreffende Annahmen getroffen wurden oder sogar eine unrealistische Aufwandsschätzung zugrunde liegt. Aus verschiedenen Gründen stehen im Projektalltag oftmals nicht die vereinbarten Ressourcen zum vereinbarten Zeitpunkt oder Zeitraum zur Verfügung. Dies erfordert zeitkritischen Handlungsbedarf, welcher eine präventive Analyse des Projektes voraussetzt. [vgl. Kuster u. a. 2011, 163; vgl. Rau 2016, 3 f]

Neben den vier Phasen ist die Durchführung von Risiko-, Qualitätsmanagement und der Steuerung der Informationsflüsse ebenfalls als wichtiger Bestandteil im Projektmanagement integriert. Eine kurze Beschreibung ist in Anhang B zu finden.

Um die vorgestellten Phasen im Projektmanagement und die jeweils vorgestellten Methoden gut unterstützen zu können, gibt es natürlich auch die verschiedensten **Tools**, auf die nun im Folgenden eingegangen wird. Die **Vielfalt** für die gebotene Unterstützung in der Planung und Steuerung von Projekten mit Programmen ist **groß**. Den Anfang machen die klassischen Office-Werkzeuge, welche durch ihre umfangreichen Funktionen auch für diverse Aufgaben im Projektmanagement nutzbar sind. So kann die Anschaffung und die Schulung für Spezialsoftware vermieden werden. Die Vielfalt erstreckt sich über spezielle Tools, welche die Erledigung von kleineren Teilbereichen oder Erweiterungen von beispielsweise Office-Werkzeugen im Focus haben, bis hin zu eigenständigen Programmen, welche den Projektmanagementlebenszyklus weitestgehend abdecken oder sogar in eine projektübergreifende/unternehmensweite Planung resultieren. [vgl. Jakoby 2015, 171 ff; vgl. Rau 2016, 7; vgl. Heilmann 2003, 10]

Zusammenfassend zur Toolunterstützung kann hervorgehoben werden, dass die IT-Unterstützung im Projektmanagement einen entscheidenden Beitrag zur Datenanalyse, projektübergreifende Auswertungen, Zwang zur konsequenten Planung und Strukturierung, präventive Eingriffe und natürlich die Überwachung des Projektfortschrittes hat. Ab einer gewissen Projektgröße ist ein Projektmanagement ohne den Einsatz von unterstützender Software nicht mehr möglich und somit unabdingbar. Unabhängig davon ist es schwer, alle Schritte mit Hilfe einer Software abzudecken. So müssen beispielsweise Projektziele, Projektstrukturen, Vorgänge, Abhängigkeiten und Verantwortungen zunächst inhaltlich festgelegt werden, bevor eine Übertragung in das ausgewählte Tool stattfinden kann. Das ausgewählte Tool kann nur die Funktionalitäten und Modellierungsmethoden abbilden, für welche Einsatzzwecke es vorgesehen ist. Software und deren darunterliegende Modelle sind eng miteinander verknüpft. Somit ist die durchgeführte Projektmanagementaktivität am Ende immer abhängig von der Qualität der gegebenen Datengrundlage und deren Verarbeitung in der Modellierungsmethode. [vgl. Kuster u. a. 2011, 158 ff; vgl. Rau 2016, 7 f]

2.1.2 Agiles Vorgehen und agile Methoden

Das Tätigkeitsfeld um das Projektmanagement wird immer komplexer. Hier sei nur eine kurze Aufzählung möglicher Herausforderungen genannt: Es ist schwierig, alles im Detail im Voraus zu planen, und auf Veränderungen oder auf unvorhergesehene Ereignisse zu reagieren. Aber was stellt die größte Herausforderung für die meisten Projektmanager dar? Es ist der schon vorher genannte Gegensatz von Zeit, Kosten und Qualität, ohne das gesteckte Projektziel und den Projektscope aus den Augen zu verlieren. Das agile Projektmanagement könnte die Antwort auf die angesprochenen Probleme darlegen. [vgl. Komus und Kuberg 2015; vgl. Allisy-Roberts u. a. 2006; vgl. Weichenhain und Alexander 2018]

Der Beginn des agilen Projektmanagements ist in der Softwareentwicklung zu finden und hat sich in den letzten Jahren auf andere Bereiche der Industrie ausgeweitet. Immer wieder besteht die Herausforderung darin, in einem sich rasch verändernden Projekt schnell und anpassungsfähig zu agieren. Studien haben gezeigt, dass sich Kombinationen von agilen Entwicklungen mit dem traditionellen Projektmanagement am Markt bewährt haben, obwohl diese beiden Projektmanagementmethoden so gegensätzlich sind und sich grundlegend voneinander unterscheiden. Das Ergebnis einer Studie belegt, dass die Mehrheit der agilen Methoden selektiv (25 %) oder als hybride Kombination (39 %) Anwendung finden. [vgl. Komus und Kuberg 2015; vgl. Allisy-Roberts u. a. 2006; vgl. Weichenhain und Alexander 2018]

SCRUM ist eines der erfolgreichsten agilen Methoden im Bereich des Projektmanagements. Laut einer durchgeführten Studie aus 2017, repräsentiert SCRUM eine Marktabdeckung von 58 % im Vergleich zu anderen agilen Methoden. SCRUM wurde von Ken Schwaber und Jeff Sutherland in den frühen 1990er Jahren entwickelt. Es fokussiert sich nicht auf die einmalige Lieferung eines Ergebnisses, sondern vielmehr auf Teillieferungen und dessen kontinuierlichen Flusses. Sie konzentrieren sich auf die Erweiterungen und die Verbesserung des Produktes. [vgl. Komus und Kuberg 2015; vgl. Kaack 2017; vgl. Weichenhain und Alexander 2018]

Zunächst ein kurzer Einblick, welcher die Grundidee des agilen Vorgehens laut SCRUM darstellt. Der Grundsatz basiert auf Flexibilität und auf einer kontinuierlichen Anpassung, anstelle von detaillierter Projektplanung. Vorweg zu nehmen ist, dass **SCRUM** keine Projektmanagementmethode ist, sondern ein **Framework** darstellt. Dieses Framework bietet ein Set an Zuständigkeiten, Arbeitsweisen und Ergebnissen zur Steuerung von Entwicklungsarbeiten. Merkmale sind die kontinuierlichen Feedback-Schleifen und die Durchführung von Sprints. Es gibt keinen zu Beginn festgelegten Funktionsumfang. [vgl. Weichenhain und Alexander 2018]

Der Ansatz ist, die Anforderungen in einer Liste (**Backlog**) zu sammeln, pflegen, erweitern und priorisieren. Die Summe aller im Backlog enthaltenen umzusetzenden Anforderungen werden inkrementell, mit laufenden Iterationen, in sogenannten **Sprints** umgesetzt. Das Kernelement der Sprints hat eine festgelegte Dauer von \pm vier Wochen. Mit dieser Dauer wird ein Kompromiss aus einem störungsfreien Ablauf für das Entwicklerteam und einer möglichst kurzen Unterbrechung des Ideenflusses forciert. Der Sprint kann für die kurze Darstellung grob in drei wesentliche Bereiche: **Start**, **Durchführung** und **Abschluss**, eingeteilt werden. Den Anfang macht das **Sprint Planning**. In diesem Meeting erfolgt die Klärung der aufzunehmenden Anforderungen für den nächsten Sprint. Während des Sprints ist das **Daily SCRUM** ein wichtiger Bestandteil des Frameworks. Hierbei handelt es sich um ein kurzes (15-minütiges) Standup Meeting des Entwicklerteams. Kernidee ist, kurz den vergangenen Tag zu resümieren, mögliche Hindernisse des jetzigen Tages zu benennen und ausstehende Aufgaben zu kommunizieren. Am Ende des Sprints erfolgt das **Sprint Review**. Dies trägt die Ergebnisse zusammen. Im Anschluss werden in der Feedbackrunde namens **Retrospektive** die Optimierungspotentiale herausgearbeitet. [vgl. Weichenhain und Alexander 2018]

Agile Methoden fanden ihren Weg ins klassische Projektmanagement, indem sie bestimmte Aufgaben ersetzen bzw. bestimmte Bereiche der klassischen Projektmanagementmethoden erweitern. Somit konnten sie als Ergänzung oder Erweiterung in einer kombinierten Form aus agilen und klassischen Methoden integriert werden. [vgl. Komus und Kuberg 2015; vgl. Kaack 2017; vgl. Weichenhain und Alexander 2018]

2.1.3 Gegenüberstellung Geschäftsprozess-, Workflowmanagement und diskrete Modelle in Bezug auf das Projektmanagement

Ein ereignisdiskretes System (DES – Discrete Event System) ist ein System, bei dem Statusänderungen (Ereignisse) zu bestimmten Zeitpunkten auftreten und Ereignisse keine Zeit benötigen bzw. verbrauchen. Solche Systeme werden mit diskreter Ereignissimulation modelliert. Das Petri-Netz stellt eine spezielle Form dieser dar. [vgl. Varga 2001, 37]

Somit liegt einem diskreten ereignisorientierten Modell eine sprunghafte Zustandsänderung beim Auftreten von Ereignissen zugrunde, während in dem Zeitraum zwischen zwei Ereignissen der Zustand konstant bleibt. Diese Eigenschaft macht eine klare verständliche Struktur möglich, die es gestattet, eine mathematische Formalisierung zu finden. Der Fokus der betrachteten Phasen im Projektmanagement ist vom Charakter her den diskreten Prozessen zuzuordnen, beispielsweise durch abzählbare Einzelteile oder die Unterteilung in atomare Einheiten. Schlussfolgernd sind es dynamische Systeme, deren Zustand sich aufgrund von diskreten Ereignissen ändert. [vgl. Wolf-Ulrich 2005, 22] Petri-Netze werden als theoretische Grundlage von ereignisdiskreten Systemen verwendet und weiterentwickelt. [vgl. Giua und Seatzu 2014, 200]

Die Abbildung einer Gruppe von Elementen, die von der Realität abstrahiert sind und wechselseitige Beziehungen zueinander haben, geschieht mit Modellen. Zur Abbildung und Analyse dieser, sind Modelle mit einer asynchronen Arbeitsweise der Teilprozesse und der Abbildung des Informationsaustausches durch diskrete Ereignisse notwendig. Ein diskretes System ist durch diskrete Variablen, Ereignisse und Beziehungen spezifiziert. Diskret heißt, dass eine Variable verschiedene Werte zu bestimmten Zeitpunkten annehmen kann, aber nicht die Veränderung im Zeitverlauf betrachtet wird. Ereignisdiskrete Systeme sind gekennzeichnet durch Ereignisfolgen. In diesem Sinne stellt jedes Ereignis eine abrupte Änderung eines Eingangs-, Zustands- oder Ausgangssignals dar. [vgl. Lunze 2006, 1 f; vgl. Hrúz und Zhou 2007, 1 f; vgl. Rau 2016, 21]

Im späteren Verlauf der Arbeit werden die Grundlagen, ausgesuchte Erweiterungen und vorgeschlagene Neuerungen bei Petri-Netzen für die Umsetzung der Modellierung vorgestellt.

Im weiteren Verlauf dieses Kapitels wird ein Vergleich der integrativen Bereiche im Projektmanagement vorgenommen. Die einzelnen Managementbereiche des Geschäftsprozess- und Workflowmanagements werden in Anhang D näher erläutert, um sie später zum Projektmanagement abgrenzen zu können. Prozesse und Projekte bilden die zentralen Bestandteile des Prozess- bzw. Projektmanagements. Teilaufgaben werden zu einer Gesamtaufgabe – dem Prozess oder Projekt – zusammengefasst. Demzufolge bestehen Prozesse und Projekte aus Abfolgen von Aktivitäten mit einem sachlogischen Zusammenhang. Einzelne Prozesse und Projekte verfügen über definierte Start- und Endzeitpunkte. Sie verbrauchen Ressourcen und Kompetenzen für die Ausführung. Prozess- und Projektmanagement stehen in einem starken Zusammenhang zueinander. Neben der oben erläuterten Gleichartigkeit ihrer zentralen Aufgaben (Prozess und Projekt) ist dieser Zusammenhang in einer gegenseitigen Unterstützung begründet. [vgl. Becker und Kahn 2012, 6; vgl. Thein 2016, 8 ff; vgl. Margherita 2014, 652]

In [vgl. Rau 2016, 29] wurden die elementaren Bestandteile und Merkmale des Projektmanagements aufgeführt und ihre jeweiligen Entsprechungen, falls vorhanden, in den weiteren drei Bereichen dargestellt. Daraus wird ersichtlich, dass sich die drei Bereiche - Geschäftsprozess-, Workflow- und Projektmanagement - viele Gemeinsamkeiten teilen. Beispielsweise ist die Orientierung in Phasen über alle Betrachtungsgegenstände eine Gemeinsamkeit und kann überall abgebildet werden. Dies führt zu dem Schluss, dass die vier Phasen - Initialisierung, Planung, Steuerung und Abschluss - durch alle ablauforientierten Methodenkonzepte aufgegriffen und unterstützt werden können. Weiterhin kam die Auswertung hinsichtlich diskreter Modelle zu dem Ergebnis, dass die Aufgaben bezüglich der Ressourceneinsatzplanung, der Kostenplanung, der Organisation der Informationsflüsse und des Dokumentenmanagements von den üblichen Discrete Event-Modellen (DE-Modellen) nicht abgebildet werden können. [vgl. Rau 2016, 28 ff]

Im Mittelpunkt des Geschäftsprozessmanagements steht die effiziente Abstimmung von Prozessen. Im Gegensatz zum Projektmanagement kümmert sich das Geschäftsprozessmanagement um die Umsetzung, Koordinierung und Verbesserung von wiederkehrenden Vorgängen, nicht aber um ein einmaliges und zeitlich begrenztes Vorhaben. Wie schon im vorangegangenen Kapitel ausgeführt, ist ein Workflow ein vollständiger oder teilweise automatisierter Geschäftsprozess. Speziell umfasst es den rechnergestützten Anteil und die Ausführungen, womit die beschriebene Businessstrategie umgesetzt wird. [vgl. Wirtschaftslexikon Gabler 2017; vgl. Kaltwasser 2017, 9; vgl. Gadatsch 2013, 36 f; vgl. Scharfe 2013, 13 ff]

Die Abgrenzung zwischen Prozess und Projekt, bzw. zwischen Prozess- und Projektmanagement, war für die Herbeiführung eines Verständnisses für die zugrundeliegenden Begriffe dieser Arbeit wichtig. Im weiteren Teil der Arbeit wird das Projektmanagement in den Fokus der Betrachtung rücken. Es kann festgehalten werden, dass ein starker Zusammenhang zwischen Projekt und Prozess besteht.

Auch, wenn die Überwachung eines Projektes mit dem Workflowmanagement-System (WfMS) möglich wäre, ist der Workflow auf die Strukturabläufe fokussiert und ist ungeeignet, um beispielweise weitere managementrelevante Informationen, wie die Berücksichtigung von Ressourcen, einzubinden. Der bekannte Vertreter Yet Another Workflow Language (YAWL) ist auf feste Strukturen und im Vorfeld bekannte Abläufe ausgelegt. Es fehlen ausführliche Informationen über die in den Strukturen befindlichen Objekte. Somit befindet sich das Modell des Workflows auf dem Niveau der einfach farbigen PN, was ebenfalls beispielsweise die Möglichkeiten der Lebendigkeitsanalyse nicht zur Verfügung stellt. Die Möglichkeiten der Höheren PN (HPN) mit dem Vertreter der strukturierten Marken, ist allerdings von großer Bedeutung und erweitert den Anwendungsbereich ungemein. Deswegen wird ein komplexes höheres PN Anwendung finden, welches auf die Problemstellungen im Projektmanagement zugeschnitten werden kann. Weiterhin ist als Argumentation anzubringen, dass somit eine Formalisierung erreichbar ist und auch durch die Eigenschaften der Lebendigkeitsanalyse weitere Vorteile ausgenutzt werden können.

Gegenüberstellung und Eingrenzung der Konzepte in Bezug auf Tools

Im Rahmen der Tools geht es um die Unterstützung einer spezifischen Aufgabe. Diese kann sich aufgrund der Gemeinsamkeiten durch den Einsatz unterschiedlichster Tools realisieren lassen. Im Anhang A wird der Tooleinsatz im Rahmen des PM näher untersucht. Bezogen auf PM muss nicht nur immer zwingend ein Tool den Anforderungen gerecht werden. Bei der Toolauswahl ist darauf zu achten, dass die Anwendung projektbezogen abgestimmt ist und für jedes Projekt erneut angepasst werden kann. Damit ist die pauschale Verallgemeinerung für eine Vielzahl von Projekten nicht möglich, was ebenfalls den individuellen Charakter eines Projektes widerspiegelt. [vgl. Kaltwasser 2017, 9 f] In der Realität wird der Einsatz von Microsoft Excel oder anderen Applikationen oft hauptsächlich in allen kleinen und mittleren Projekten, dem Einsatz von Projektmanagementsoftware vorgezogen. Microsoft Excel ist mit seinem enormen Umfang in der Lage, eine Vielzahl von Sachverhalten abzubilden.

Sapper [vgl. Sapper 2007, 100] nimmt eine Unterteilungen der einzusetzenden Methoden im Projektmanagement vor (Anhang A). Daraus geht hervor, wie sich in Relation zur Projektgröße auch die Vollständigkeit der anwendbaren Methoden zum Projektmanagement verändert. Dies liegt in den wachsenden Planungs-, Verwaltungs- und Kontrollaufgaben in Relation zur Projektgröße begründet. Der fehlenden Notwendigkeit zum Einsatz von Methoden ist geschuldet, dass bei kleineren Projekten der Verwaltungsaufwand für das Projektmanagement die eigentliche Projektarbeit übersteigen kann. Als weitere Gründe können der Einsatz unpassender Software sowie das Fehlen von unterstützenden Werkzeugen, mangelnder Kompetenzen und Zeit aufgeführt werden. Zum heutigen Stand geben PM-Standards keine Vorgaben in Bezug auf die Wahl des Tools. [vgl. Kaltwasser 2017, 9 ff; vgl. Schenk 2017, 19 ff] Die Vielfältigkeit der am Markt vorhandenen Tools und dessen Einsatzszenarios ist enorm. Im Rahmen dieser Arbeit wird sich nachfolgend auf die Betrachtung des Projektmanagements im Zusammenhang mit der theoretischen Grundlage der Petri-Netze als Vertreter der diskreten Modelle fokussiert.

Projektmodellierung mit Sprachen der Prozessmodellierung sowie PN

In den vorangegangenen Kapiteln wurden die Gemeinsamkeiten verdeutlicht und der fließende Übergang zwischen Projekten und Prozessen auch im Hinblick der Modellierung aufgezeigt. Es wird genauer beleuchtet, ob sich ohne die Betrachtung einer konkreten Sprache, Geschäftsprozessmanagementmethoden grundsätzlich für die Modellierung von Projekten eignen. Dies wurde in der Arbeit von [Thein 2017] mit drei verschiedenen Ansätzen eruiert. Die drei Ansätze waren **(1)** die Ähnlichkeit zwischen Projekten und Prozessen, **(2)** die Prozesse im Projekt und **(3)** die Grenzfälle am Übergang zwischen Projekten und Prozessen. Die Überlegung der Unterteilung in die drei Ansätze werden auch in anderen Literaturen bestätigt. Im Rahmen des Funktionsumfanges wurden die folgenden wesentlichen Objekte identifiziert: Aktivitäten, Ziele, Ressourcen, Zeit und Beziehungen. [vgl. Thein 2017, 20 ff, 35 ff; vgl. Corsten, Corsten, und Gössinger 2008, 121 ff, 170 ff, 186 ff; vgl. Gadatsch 2013, 66 ff; vgl. Hansmann, Laske, und Luxem 2005, 266 ff; vgl. Perković, Šturlić, und Meden 2011, 1587]

2.2 Tools für das Projektmanagement

Der Einsatz von Software zur Unterstützung von PM ist keine klar abgrenzbare Software. Grundsätzlich umfasst sie jegliche Programme, die während der Durchführung von Projekten zum Einsatz kommen können. Das kann vom einfachen E-Mail-Klienten bis hin zu einer datenbankgestützten Multiprojekt-Software reichen. [vgl. Schenk 2017, 20]

Die Aufgaben, welche mit Software begleitet werden sollen, lassen sich analog in den vier Phasen innerhalb von Projekten eingliedern: **Initiierungsphase**, **Planungsphase**, **Durchführungsphase** und die **Abschlussphase**. An all diesen Phasen sind verschiedene Management-Ebenen beteiligt. Abhängig von den Phasen werden unterschiedliche Anforderungen an die zu unterstützende Software gestellt. Dies ist, wie sich schon herausgestellt hat, unabhängig von Projektmanagementstandards, welche i. d. R. keine speziellen Anforderungen hinsichtlich der Nutzung von PM-Software stellen.

Für eine optimierte Toolunterstützung zum Projektmanagement können Unternehmen spezielle Softwarelösungen einsetzen. Diese sind darauf ausgelegt, in allen Bereichen zu unterstützen, angefangen bei der Planung bis hin zur Durchführung von Projekten.

Zusammenfassend sowie in Auswertung meiner bisherigen Tätigkeiten kann festgehalten werden, dass ein PM-Tool mit großem Funktionsumfang auch oft eine Einführung oder gar Schulung zur Anwendung benötigt. Die Modelle weisen trotzdem auch ihre Einschränkungen auf und siedeln sich oft im hochpreisigen Bereich an. Im niedrigen Preissektor ist der Funktionsumfang eher eingeschränkt oder auf spezielle Anwendungsgebiete beschränkt.

In der Realität zeigt sich aber oftmals der Einsatz von Microsoft Office® Excel als Alternative für ein auf Projektmanagement spezialisiertes bzw. extra auf das Problem ausgerichtetes Programm. Hierzu das Ergebnis der Studie von techconsult im Auftrag von Microsoft über Einsatzgrade in mittelständischen Unternehmen: In der Studie wurden PM-Tools mit unterschiedlichem Funktionsumfang einbezogen. [vgl. Groß 2014; vgl. Corcoran 2019]

„Während Microsoft Project und SAP Business One eine breite branchenneutrale Palette an Funktionen bieten, die sich nahezu für jedes Projekt eignen, sind viele der anderen Lösungen auf spezielle Einsatzbereiche zugeschnitten.“ [Groß 2014]

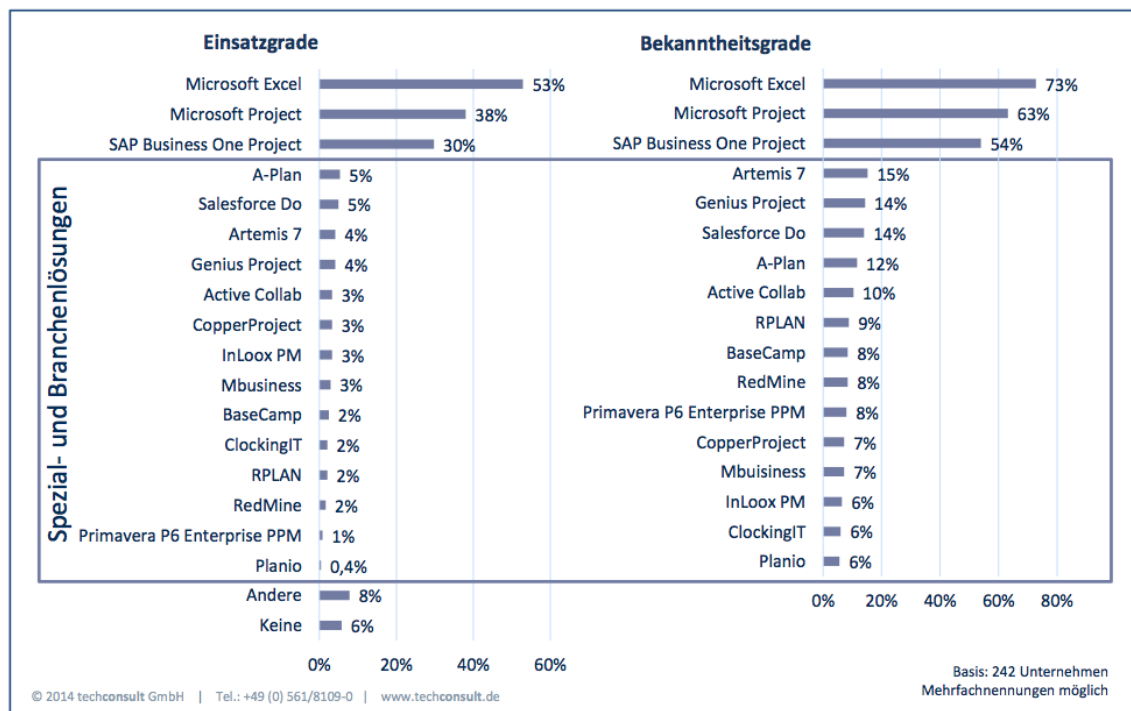


Abbildung 2: Einsatz- & Bekanntheitsgrade von Projektmanagementsoftware [Groß 2014]

„Da Mehrfachnennungen möglich waren, summieren sich die Einsatzgrade auf nahezu 180 %, d. h. das Gros der Unternehmen setzt zwei oder mehr Projektmanagementlösungen ein. Vor allem Microsoft Excel liegt mit Einsatzgraden von über 50 % in einem Bereich, in dem davon auszugehen ist, dass es häufig zusätzlich, z. B. als begleitendes Dokumentations- und Reporting-Tool, neben Spezial- oder Branchenlösungen eingesetzt wird und diese ergänzt. [...] Der Anteil der Mitarbeiter (MA) eines Unternehmens, die mit einer Projektmanagementlösung arbeiten, variiert nur unwesentlich mit der Größe des Unternehmens. In ca. drei Viertel der Unternehmen arbeiten bis zu 20 % der Mitarbeiter mit der Projektmanagementlösung. Dass mehr als die Hälfte der Mitarbeiter mit der Projektmanagementsoftware arbeitet, ist fast ausschließlich in kleineren und mittleren Unternehmen der Fall.“ [Groß 2014]

Wie durch die Studie sowie Erfahrungen erkannt, kommen die Microsoft Office® Produkte durch ihren sehr hohen Verbreitungsgrad im Feld des Projektmanagements immer wieder zum Einsatz. Excel bietet einen sehr gewaltigen Funktionsumfang, der es erlaubt, vielfältigste Einsatzgebiete abzudecken. Das belegt auch die große zur Verfügung stehende Palette von Add-ons, Add-ins oder Plug-ins fürs PM für das Microsoft Office® Paket. Das wiederum zeigt auf, dass die Nachfrage nach einzelnen Erweiterungen ebenfalls vorhanden ist und ihre Berechtigung hat. Eine genauere Auseinandersetzung zu der Aufschlüsselung von Add-ons, Add-ins oder Plug-ins ist in [Kaya 2017a, 7 ff] zu finden. Aus Erfahrung werden auch diese angebotenen Add-ons, Add-ins oder Plug-ins in der Praxis nicht oft verwendet.

Der größte Nachteil hierbei ist die fehlende Simulationsmöglichkeit. Es handelt sich oftmals um eine situativ zugeschnittene Einzellösung, welche durch komplexe, teils schwer zu wartenden sowie fehleranfälligen Formeln und Diagrammen bestehen.

2.3 Petri-Netze

Der nachfolgende Abschnitt soll, beginnend mit den Grundlagen der PN, die nötige Wissensbasis für die später erforderlichen Erweiterungen darstellen. Anschließend werden diese Erweiterungen näher erläutert.

Petri-Netze zeichnen sich dadurch aus, dass sie mathematisch definiert sind und sich zudem durch eine visuelle Notation grafisch darstellen lassen. Wie schon hergeleitet, wird durch PN die Beschreibung von diskreten Systemen und somit die Darstellung von Parallelität und Nebenläufigkeit in diesen Systemen ermöglicht. Das führt dazu, dass ein dynamisches System mit einer statischen Struktur grafisch dargestellt werden kann. Die Darstellung der statischen Struktur wird in Plätzen, Transitionen und Kanten abgebildet, die ein dynamisches Verhalten durch den Einsatz von Marken simulieren können. [vgl. Silva 2013; vgl. Gudd 2017; vgl. Baumgarten 1990; vgl. Weske 2012]

Im weiteren Verlauf dieser Arbeit werden im Rahmen des Projekt-Petri-Netz-Modells (PPNM) die Eigenschaften von zeitbewerteten PN, gefärbten PN und strukturierten Marken aus den Prädikat/Transitions-Netzen (Pr/T-Netze) Verwendung finden. Der Anfang des Kapitels beruht auf der Basis der theoretischen Grundlage von PN.

Petri-Netze können in klassische PN und Höhere PN (high-level PN) unterteilt werden. Beide Arten von Petri-Netzen basieren auf dem gleichen Grundmodell, wobei der höheren Ebene eine bessere Ausdrucksstärke in der Modellierbarkeit zugesprochen wird. Dies ist dadurch bedingt, dass die höhere Ebene mit besseren Strukturierungsmöglichkeiten und Abstraktionsmöglichkeiten ausgestattet ist. [vgl. Jensen 1992, 61; vgl. W.M.P. van der Aalst 1994, 46]

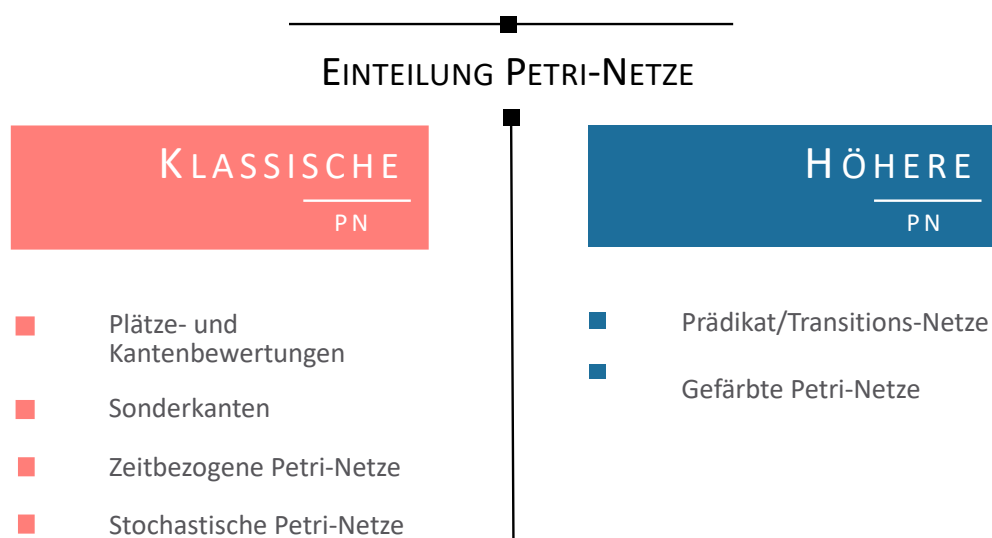


Abbildung 3: Grobe Einteilung Petri-Netze

Zunächst wird kurz auf die Grundlagen eingegangen. Sowie die klassischen Petri-Netze als auch die Höheren Petri-Netze werden in den nachfolgenden Unterkapiteln näher vorgestellt. Im Rahmen der Grundlagen wird sich auf Platz-Transitions-Netz (P-T-Netz) beschränkt. Zu den P-T-Netzen zählen die Erweiterungen der Plätze- und Kantenbewertungen, die Sonderkanten sowie die zeitbezogenen und stochastischen PN.

Ein **Platz-Transitions-Netz** wird durch sechs wesentliche Elemente beschrieben. Ein PN ist somit als ein Tupel definiert, welches aus Plätzen, Transitionen, Kanten, Vielfachheiten und Kapazitäten besteht; $PN = \{P, T, F, V, K, M_0\}$: [vgl. Fengler und Philippow 1991, 51 f; vgl. Gudd 2017, 2 ff]

- *Zustände*, auch **Plätze** genannt, werden als runde Knoten dargestellt (**P** als Menge der Plätze) und beinhalten Marken.
- *Zustandsübergänge*, auch **Transitionen** genannt, sind dargestellt als rechteckige Knoten (**T** als Menge der Transitionen). Durch das Schalten einer Transition wird eine Marke weiterbefördert. Die Eigenschaft der Schaltfähigkeit ermöglicht den Transitionen einen Zustandsübergang. Dies bedeutet, dass durch das Aktivieren einer Transition eine Schaltregel zur Anwendung kommt und eine definierte Vielfachheit von Marken von einer Ausgangsmarkierung zu einer Folgemarkierung übergibt. Eine Transition ist nur dann schaltfähig, wenn die Kapazitäten des Folgeplatzes durch das Schalten nicht überschritten werden.
- **Kanten**, die Plätze und Transitionen miteinander verbinden, werden auch *Flussrelationen* genannt. Das **Kantengewicht/Vielfachheit** bestimmt die Anzahl der beförderten Marken durch die Kanten (**V** als Kantengewicht; **F** als Flussrelation). Der Transport einer definierten Markenanzahl von einem Platz zur Transition entspricht einer sogenannten Vorkante. Wohingegen eine Nachkante eine definierte Markenanzahl von einer Transition zu einem Platz transportiert.
- Plätze werden mit **Token** bzw. **Marken** belegt, um den aktiv eingenommenen Zustand darzustellen (**K** als **Kapazitätsfunktion** der Plätze). Die Kapazitätsfunktion beschränkt die Anzahl der Token pro Platz. Plätze agieren passiv und Marken dynamisch.

- Beim Vorplatz mit der **Anfangsmarkierung** (M_0) wird die anfängliche Markierung von Plätzen und deren Token festgelegt.
- Konstruktionsregeln: Es gibt Strukturen, welche erlaubt und welche verboten sind.

Die nachfolgende Übersicht in Abbildung 4 stellt die wichtigsten Fakten zu Petri-Netzen in einer Übersicht dar. [vgl. Gudd 2017, 5 ff]

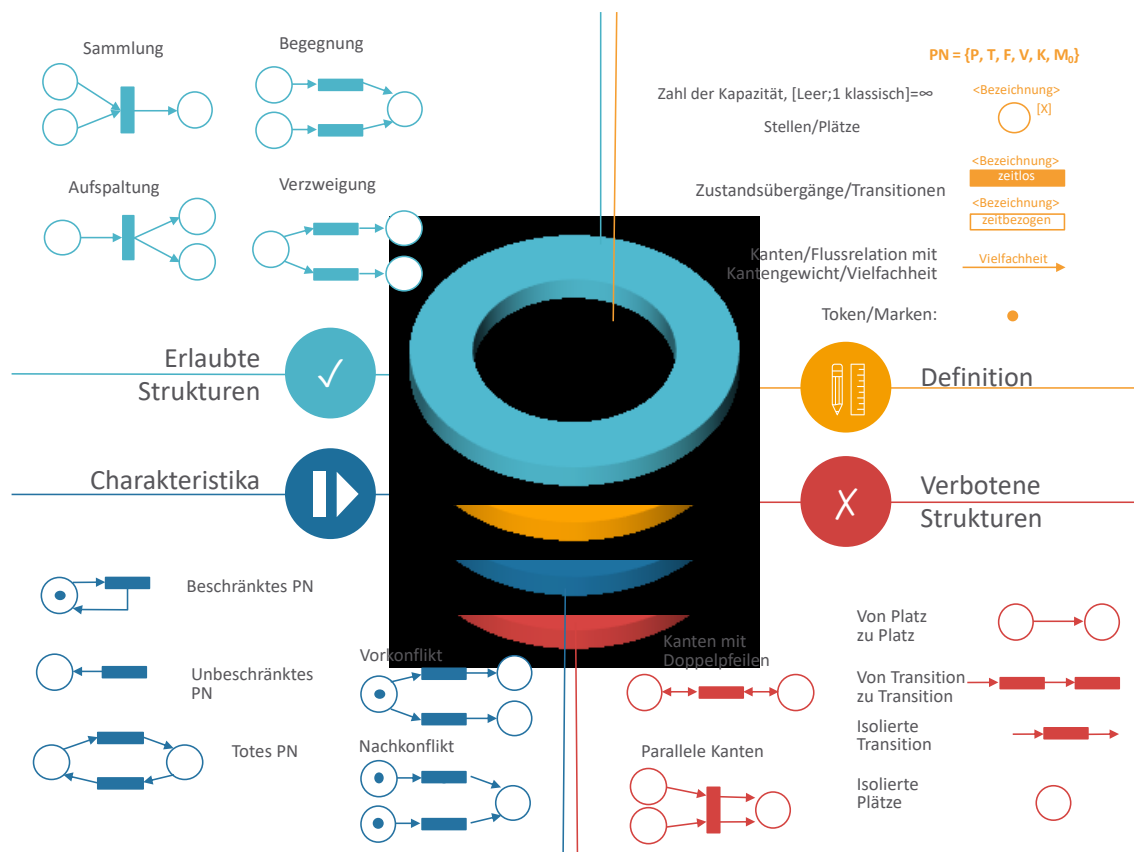


Abbildung 4: Übersicht klassischer Petri-Netze [vgl. Fengler und Philippow 1991; vgl. Baumgarten 1990; vgl. Gudd 2017]

Ebenfalls ist die Abbildung von zeitlich parallelen und nebenläufigen Vorgängen mit Petri-Netzen sehr gut darstellbar. Dies wird umgesetzt, indem sie als erweiterte Zustandsgraphen durch die Verwendung von Token abgebildet werden können. [vgl. Lunze 2006, 397 f]

Auch zur Erstellung der Modelle von Petri-Netzen gilt es darauf abgestimmte Software zu finden, welche die Aufmerksamkeit auf die Modellierung und Abbildung der gewünschten Systeme legt. Diese Modelle sollen, wenn gewünscht und ebenfalls mit der richtigen Software editiert, ausgeführt und getestet werden können. [vgl. Hrúz und Zhou 2007, 91 f; vgl. Rau 2016, 24 ff]

Nachfolgend sind die wesentlichen Grundannahmen und Analysemöglichkeiten kurz übersichtlich dargestellt.

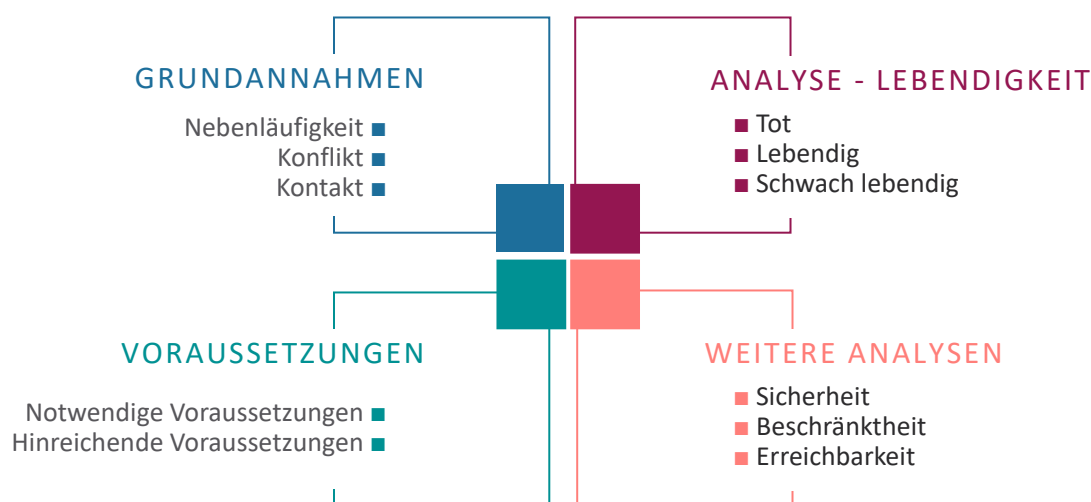


Abbildung 5: Übersichten der Grundannahmen & der Analyse von Petri-Netzen

Auf die dargestellten Fakten, Grundannahmen und Analysemöglichkeiten wird hier nur selektiv näher eingegangen und für weitere Informationen auf die angegebene Literatur verwiesen. Die zwei wichtigsten Eigenschaften sind hierbei **Sicherheit** und **Lebendigkeit**. Die Sicherheit eines PN ist u. a. von dessen **Beschränktheit** abhängig. Daneben sind Erreichbarkeit und Konfliktfreiheit ebenfalls wichtige Eigenschaften. [vgl. Fengler und Philippow 1991, 58; vgl. Baumgarten 1990, 130 ff; vgl. Gudd 2017, 5]

- **Sicherheit:** An einem Platz können sich mehrere Marken sammeln. Es ist unerwünscht, dass die Markenanzahl in einem Platz unbegrenzt zunimmt, oder eine bestimmte Anzahl überschritten wird. Ein Petri-Netz wird als sicher angesehen, wenn bezüglich einer vorgegebenen Markenanzahl x die Erreichbarkeitsmenge keine Markierung enthält, bei der einem Platz mehr als x Marken zugewiesen sind und alle Plätze somit eine unendliche Gesamtkapazität besitzen. [vgl. Abel 2014, 450 ff]

- Beschränktheit sichert zu, dass in keinem Platz des Petri-Netzes eine bestimmte Anzahl von Marken übertroffen wird und somit eine Kontrolle über zu erreichende Markierungen ermöglicht. [vgl. Fengler und Philippow 1991, 58; vgl. Baumgarten 1990, 130 ff; vgl. Gudd 2017, 5]
- Lebendigkeit besteht, wenn alle Transitionen eines Netzes lebendig und somit unter allen Folgemarkierungen aktivierbar sind. Eine nicht aktivierbare Transition wird als tot bezeichnet. Ein Petri-Netz ist tot, wenn alle enthaltenen Transitionen tot sind. Hier gibt es auch Abstufungen der schwachen Lebendigkeit. [vgl. Fengler und Philippow 1991, 58; vgl. Baumgarten 1990, 130 ff; vgl. Gudd 2017, 5]

2.3.1 Klassische Petri-Netze

Nachfolgend wird einzeln auf die Erweiterungen eingegangen, was u. a. durch das Einfügen von Zeit und Sonderkanten geschieht.

Plätze- und Kantenbewertungen

Die Erweiterung um Plätze- und Kantenbewertungen in einem PN hebt den Umstand auf, dass Kapazitätsbeschränkung und Kantengewicht immer gleich eins sein müssen. Dies erlaubt eine beliebig, endlich hohe Festlegung der Kapazitäten an den Plätzen sowie Gewichte an den Kanten. Damit kann festgelegt werden, wie viele Token sich als Maximum auf einem Platz befinden bzw. wie viele Token sich beim Schalten an einer Kante bewegen dürfen. [vgl. Lunze 2006, 3110; vgl. Rau 2016, 24 ff]

Sonderkanten bei Petri-Netzen

Die Einteilung der Sonderkanten kann in zwei Arten vorgenommen werden: einem statischen (Unterlaufestkante, Testkante und Inhibitorkante) und einem dynamischen Block (Setzkante, Lösch- und Rücksetzkante). [vgl. David und Alla 2010, 14; vgl. Gudd 2017, 8 f] Die nachfolgende Abbildung verdeutlicht die Untergliederung. Im weiteren Verlauf spielen alle Sonderkanten eine Rolle und werden in der nächsten Abbildung kurz vorgestellt.

Eine Gruppe bilden die statischen Sonderkanten. Diese beeinflussen die Schaltfähigkeit einer Transition, ohne dabei die Markierung zu verändern. Hierzu zählen Testkante,

Inhibitorikante und Unterlaufsteskante. Die zweite Gruppe bilden die dynamischen Sonderkanten, welche nicht die Schaltfähigkeit beeinflussen, die Markenanzahl aber verändern. Darunter zählen die Setzkante und die Löschkante bzw. Rücksetzkante. [vgl. David und Alla 2010, 14; vgl. Gudd 2017, 9 f] Eine Definition ist im Kapitel 4.2 unter Sonderkanten zu finden.

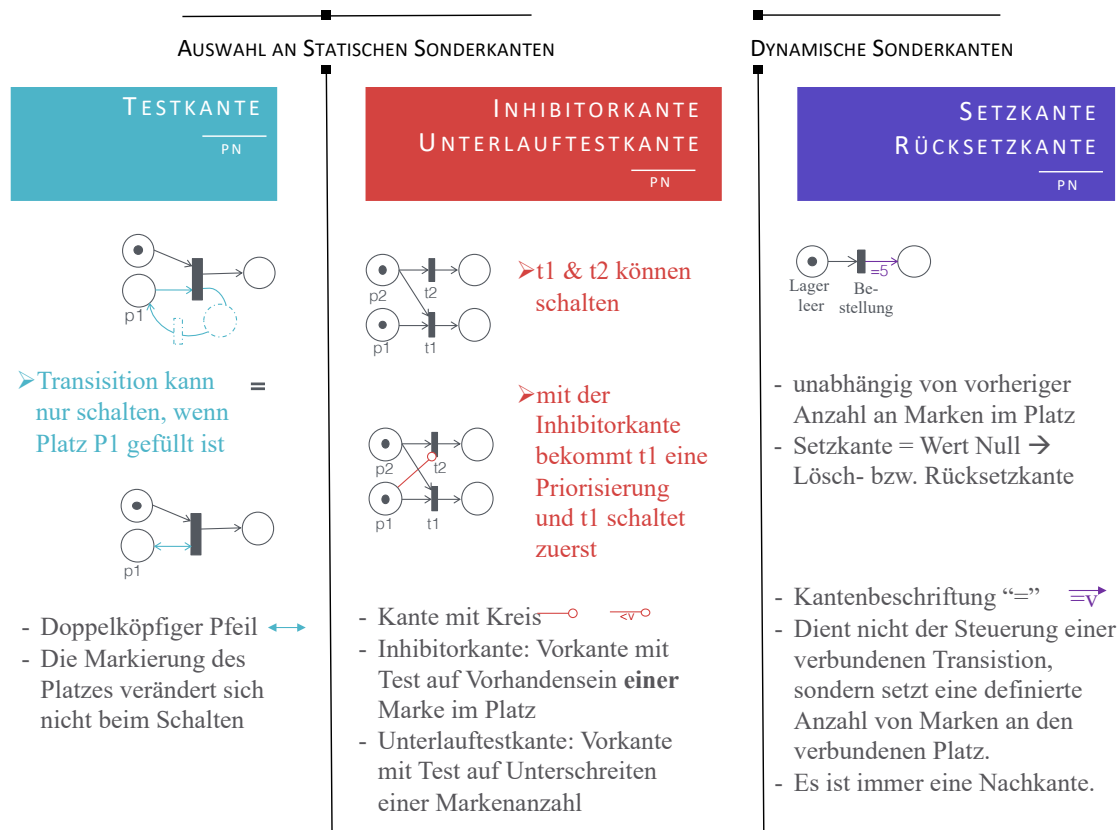


Abbildung 6: Auswahl an Sonderkanten

Zeitbezogene Petri-Netze

Zeitbezogene Petri-Netze (englisch Time Petri Net – TPN) gestatten die Darstellung von zeitlichem Verhalten. Damit wird nicht nur die Aussage was passiert ist, ermöglicht, sondern auch wann etwas passiert. Dies führt neben der quantitativen Analyse auch zu einer qualitativen Analyse des Systems. [vgl. Diaz 2009, 122 ff]

Die Einbindung der Dimension der Zeit kann auf zwei wesentliche Vorgehensweisen unterschieden werden. Entweder nach einer Schaltregel oder nach einem Zeitfortschritt an der Transition.

Bei der Schaltregel werden Preselection- oder Race-Modelle modelliert. Wohingegen bei der Art des Zeitfortschritts eine deterministische Modellierung durch ein Zeitintervall auf einer stochastischen Art und Weise und durch das Nutzen von Fuzzy-Logik ermöglicht wird. [vgl. Silva 2013; vgl. Bause und Kritzinger 1998; vgl. Gudd 2017]

Im Projektmanagement steht bei der Planung und Überwachung die Zeitdauer von Aktivitäten im Vordergrund. Aufgrund dessen wird die Zeit den Transitionen zugeordnet. Im weiteren Verlauf der Arbeit wird das entwickelte Zeitkonzept detaillierter vorgestellt. Findet bei einer Transition eine konstante Zeitdauer von der Aktivierung bis zur Schaltfähigkeit statt (beispielsweise es vergehen seit der Aktivierung drei bis fünf Zeiteinheiten), wird dies der Verzögerung zugeordnet. Wenn ein Zeitintervall an einer Transition angegeben wird, indiziert dies den minimalen und maximalen Zeitpunkt der Transitionsausführung. Das Intervall kann zwischen Null und einer endlichen Zahl liegen. [vgl. Gudd 2017, 12; vgl. Silva 2013, 206; vgl. Diaz 2009, 122]

Durch Nutzen von Fuzzy-Logik und basierend auf der Theorie der unscharfen Mengen (Empfindungen, Einschätzungen und unscharfe Angaben können so in eine mathematische Relation gesetzt werden; Beispiel: Etwas heiß) wird eine entsprechende Logik generiert und mit den Transitionen verbunden. Dies kann durch eine Zeitdauer oder einen stochastischen Prozess ergänzt werden. [vgl. Traeger 1994, 4 f; vgl. Gudd 2017, 12 f; vgl. Zhou und Zain 2016, 9 ff]

2.3.2 Ausgewählte Erweiterungen zu Höheren Petri-Netzen

In diesem Kapitel werden einzelne Erweiterungen vorgestellt, welche auch als Höhere Petri-Netze bekannt sind. Im direkten Vergleich haben HPN die gleiche abstrakte Ausdrucksstärke wie klassische PN, führen aber zu kompakteren und strukturierteren Modellansichten. [vgl. W.M.P. van der Aalst 1994; vgl. Silva 2013; vgl. Gudd 2017, 7]

Das größte Unterscheidungsmerkmal der Höheren PN sind die unterscheidbaren Marken. Im Gegensatz zu den klassischen PN mit anonymen Marken, welche keine Datenstruktur aufweisen. Die unterscheidbaren Marken und die Grundidee der gefärbten PN werden im 4. Kapitel wieder aufgegriffen.

Gefärbte Petri-Netze

Die gefärbten Petri-Netze (englisch Coloured Petri Nets – CPN) wurden von Kurt Jensen eingeführt. Sie stellen eine Erweiterung der Platz-Transitions-Netze dar und besitzen eine eigene Beschreibungssprache CPN ML (meta language). Weitere Merkmale von gefärbten PN sind **(1)** die Möglichkeit, dass sich jedes CPN in ein P-T-Netz umformen lässt und **(2)** die unterscheidbaren Marken (Farbe). **(3)** Ebenfalls, dass Plätze sog. Multimenge der Marken enthalten und **(4)** die Transitionen je nach Farbe ein unterschiedliches Schaltverhalten besitzen. CPN ist eine Sprache für die Modellierung und Validierung von Systemen, bei denen Nebenläufigkeit, Kommunikation und Synchronisation eine große Rolle spielen. Es wird zu den Diskret-Event-Modellen eingeordnet und besteht aus einer Kombination der PN mit den Funktionalitäten der ML Programmiersprache. PN liefern die Grundlage der grafischen Notation und mit ihr einhergehend die Eigenschaften der Nebenläufigkeit, Kommunikation und Synchronisation. Die Standardprogrammiersprache ML ermöglicht die Definition von Datentypen, Datenmanipulation und die Grundlage für kompakte und parametrisierbare Modelle. [vgl. Silberberg, Lask, und Bachmann 2016, 1 ff; vgl. Kleine-Döpke 2013; vgl. Jensen, Kristensen, und Wells 2007, 213; vgl. Jensen und Kristensen 2009, 81 ff]

Ein klassisches PN hat keine unterscheidbaren Marken (Token) und ist von einer flachen Struktur geprägt. Durch den Einsatz von CPN ist es möglich, Datentypen und komplexe Datenmanipulationen vorzunehmen. Dazu wird jeder Marke/Token eine Farbe (Wert) angefügt, welche überprüft und verändert werden kann. Ebenfalls besteht die Möglichkeit eine Hierarchie einzubauen, was die Vorteile von wiederverwendbaren Submodulen, Schnittstellen und deren Semantik mit sich bringt. [vgl. Jensen und Kristensen 2009, 2]

Ein gefärbtes Petri-Netz ist ein 8-Tupel $CPN = \{\Sigma, P, T, F, C, K, E, M_0\}$, mit nachfolgenden Bedingungen: [vgl. Jensen und Kristensen 2009, 2 ff; vgl. Daeumlich 2006]

- Als *Neuerung* besitzt es eine **Menge** Σ von Typen (Farbmengen).
- Es ist ein Netz mit einer Menge von **Plätzen P** und **Transitionen T**. Sie sind disjunkt und werden durch sogenannte Kanten/**Flussrelationen** $F - F \subseteq (PxT) \cup (TxP)$ – verbunden.

- Die *Neuerung* der **Typenfunktion** $C - C: [P \mapsto \Sigma]$ – besteht darin, dass jedem Platz ein Typ zugewiesen wird.
- Transitionen können nun *angepasst* mit Bedingungen belegt werden, auch als **Wächterfunktion** $K(t) - (t \in T)$ – bekannt.
- Die *veränderte Kantenausdrucksfunktion* $E(f) - (f \in F)$ – erlaubt ebenfalls Ausdrücke (z. B. Funktionen, Variablen).
- Auch die **Anfangsmarkierung** $M_0(p) - (p \in P)$ – kann an die jeweiligen Typen *angepasst* werden.

Durch die Erweiterung von einem PN zu einem CPN besteht die Möglichkeit, mehrere gleiche Teile des Netzes zu einer vereinfachten Struktur zusammen zu fassen. Dies wird auch **Faltung** genannt. Damit die Unterscheidung noch erhalten bleibt, werden die Token gefärbt. Rein formal kann man CPN in verschiedenste Komplexitätsstufen definieren. Beispielsweise können Token nicht nur mit einem einzigen Attribut (Attribut Farbe) versehen werden, sondern können auch mehrere verschiedene Attribute definiert und den einzelnen Token auf den Plätzen bzw. Pfeilen zugewiesen (Multimengen) werden. Weiterhin können CPN mit variablen Pfeilgewichtungen Anwendung finden, welche ebenfalls durch eine Faltung hergeleitet werden können. [vgl. Silberberg, Lask, und Bachmann 2016, 1 ff; vgl. Kleine-Döpke 2013; vgl. Jensen und Kristensen 2009, 81 ff]

Zu den typischen Anwendungsgebieten für CPN zählen die Protokolle, Datennetze, eingebettete Systeme sowie der Einsatz in Systemen, in denen Parallelität und Kommunikation herrscht (beispielsweise Prozessmodellierung, Modellierung von Fertigungssystemen und Agentensystemen). [vgl. Jensen, Kristensen, und Wells 2007, 2; vgl. Gudd 2017, 11]

Zeitbewertete Petri-Netze

In den klassischen Petri-Netzen spielt die Abbildung der Zeit bisher keine Rolle. Es ist keine Aussage bezüglich einer Zeit integriert. Ein zeitlicher Rückschluss könnte nur über die mögliche Reihenfolge der Ereignisse erfolgen.

Dabei ist aber auch die Nebenläufigkeit zu beachten, welche eine zeitliche Parallelität aufweist. Im PM spielt die Zeit und vor allem die zeitliche Einplanung aber eine übergeordnete und wichtige Rolle. Dies macht die Modellierung des Zeitverhaltens von Nöten und wird durch das Einbinden vorhandener Ansätze von zeitbewerteten PN erfolgen. [vgl. Starke 1995; vgl. Warmuth 2009]

In der Literatur sind verschiedene Ansätze für zeitbewertete PN zu finden. Diese unterscheiden sich dahingehend, dass sie die Zuordnung der Zeit an unterschiedliche PN Komponente (Plätze, Transitionen oder Marken) platzieren. Diese Ansätze eignen sich für komplexe Probleme, wie die Abbildung des Zeitverbrauchs bei einer Aufgabe. Der Ansatz, die Zeit an Marken zu hängen verfolgt beispielsweise die Abbildung der Verfügbarkeit von Ressourcen. Hier wird die Zeitdauer der einzelnen Aktivität in den Vordergrund gerückt. Bei einer Abbildung der Zeit an den Transitionen, sogenannte zeitbehaftete Transitionen, wird die benötigte Zeit für eine Komponente innerhalb eines Projektes abgebildet. Durch die Entnahme der benötigten Marken im Vorplatz und nach erfolgreicher Erledigung der Generierung im Nachplatz, kann die Abbildung von vordefinierten Zeiträumen sowie wiederverwendbaren Ressourcen bewerkstelligt werden. Es erfordert allerdings, dass alle Transitionen zum Zuordnen zeitlos oder in der Berechnung der Zeit mitberücksichtigt werden müssen. Es ist aber nicht möglich, die aktuelle Zeit oder gar das aktuelle Datum in der Simulation zu repräsentieren, auch ist keine Prüfung auf beispielsweise den frühesten Startzeitpunkt gegeben. [vgl. Schott 2019, 31 f; vgl. Bobbio 1990, 18; vgl. W.M.P. van der Aalst 1994, 48; vgl. Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000, 520]

Strukturierte Marken

Nach den im vorigen Abschnitt vorgestellten gefärbten Netzen, gehören auch die Prädikat/Transitions-Netze zu den Höheren Netzen. Sie verwenden unterscheidbare Marken, welche nicht nur aufgrund ihrer Farben, sondern auch durch ihre Struktur unterscheidbar sind. [vgl. Starke 1990b; vgl. Genrich und Lautenbach 1979, 124 ff]

Bei Pr/T-Netzen wird eine Termsprache zur Beschriftung der Kanten verwendet. High Level Coloured Petri-Netze mit strukturierten Token, kurz HCPN-ST, bauen auf gefärbte und Pr/T-Netze auf. Diese Arbeit beruht auf der Definition von HCPN-ST nach [Al Ali 2010]. Sie sind eine formale Methode und unterscheiden sich durch die Art der

Beschriftung der Kanten. Die HCPN-ST verwenden prädikatenlogische Ausdrücke für die Kantenbeschriftung. [vgl. Al Ali 2010, 8, 51; vgl. Schott 2019, 11 f; vgl. Starke 1990b, 236 ff]

HCPN-ST erweitern die farbigen Netze mit der Definition der Marken. Die Besonderheit liegt darin, dass eine Farbe nicht nur eine Information repräsentiert, sondern sich aus einer Reihe von Informationen zusammensetzen kann. Dadurch sind in den Marken auch komplexe Strukturen abbildbar. Diese unterschiedlichen Informationen können beim Schalten einer Transition durch strukturierte Marken verknüpft werden. Durch die Verwendung solcher Strukturen und deren Syntax können komplexe Prozesse kompakt dargestellt werden. Es stellt eine mächtige Modellierungssprache für unterschiedlichste Abstraktionsebenen dar. [vgl. Hertlein, o. J.; vgl. Schott 2019, 11 f]

Das HCPN-ST ist ein gefärbtes Petri-Netz mit 8-Tupel – $HCPN-ST = \{\text{Farbset}, P, T, F, C, K, V, M_0\}$, mit nachfolgenden Bedingungen: [vgl. Ali, Fengler, und Däne 03.07.2011, 177; vgl. Al Ali 2010, 51 ff; vgl. Schott 2019, 12]

- Als *Neuerung* besitzt es eine **Menge von Farben** (Farbset). Hier wird zwischen zwei Arten unterschieden: Das **elementare Farbset** (endlich nichtleere Menge von Farben) und das **strukturierte Farbset** (kartesisches Produkt, zusammengesetzt von Elementen des elementaren Farbsets).
- Es ist ein Netz mit einer Menge von **Plätzen P** und **Transitionen T**. Diese sind durch sogenannte **Kanten/Flussrelationen F** verbunden. Die **Kante V**, die **Kapazitätsfunktion K** und eine **Anfangsmarkierung M_0** sind ebenfalls zu finden.
- Die *Neuerung* der **Abbildung Typenfunktion C** besteht in diesem Kontext darin, dass jedem Platz eine nichtleere Menge von Farben zugewiesen wird.

Im Gegensatz zu den anonymen Marken repräsentieren strukturierte Marken Objekte mit individuellen Eigenschaften. Plätze und Transitionen können ferner mit Tabellen gekoppelt werden, um komplexe Sachverhalte darzustellen. Eine Transformation von HCPN-ST in farbige Petri-Netze ist möglich. Hierbei kommen Transformationstabellen von strukturierten Marken in farbige Marken zum Einsatz. Dabei wird für jede mögliche

Kombination von Informationen eine eindeutige Farbe generiert. Dies ermöglicht den Einsatz von strukturierten Marken zur Modellierung, auch wenn eine spätere Simulation erwünscht ist. [vgl. Al Ali 2010, 85 ff; vgl. Koschmider und Ried 2005, 67; vgl. Schott 2019, 12]

Vorhandene Ansätze zur Analyse

Petri-Netze beruhen auf einem mathematischen Modell und können nebenläufige Systeme abbilden. Aufgrund dieser Fähigkeit werden sie auch als Verfahren zur Entscheidungsfindung eingesetzt. Dies ist durch die mathematische Grundlage für Analyseverfahren zur Validierung und Verifikation von Prozessen gegeben. Eine Auflistung etlicher Werkzeuge ist in [Moldt, Wagner, und Haustermann o. J.] zu finden. Ein weiterer Ansatz ist die Fuzzy-Logik. Hier gibt es auch schon Ansätze, diese in ein PN zu integrieren. Bei der Fuzzy-Logik handelt es sich um die Theorie der unscharfen Mengenlehre, welche 1965 in den USA entwickelt wurde. Daraus ergibt sich die Möglichkeit, nicht exakte und unvollständige Datensätze, die nur in einer verbalen Form vorliegen, mathematisch zu beschreiben. Dafür wird aus den sprachlich formulierten Regeln mittels der Fuzzy-Logik eine zur Weiterverarbeitung mathematische Beschreibung erzeugt. Viele Probleme aus der praktischen Anwendung können so greifbar und runtergebrochen werden. Sie werden aufgrund ihrer einfachen Berechenbarkeit gerne angewendet. Durch die Fuzzy-Logik wird die dynamische Modellierung von Systemen und die mathematische Ausbildung stark vereinfacht. Ebenfalls müssen die Beschränkungen immer genaue und detaillierte Daten und eine Wissensbasis zur Verfügung haben. Dafür bietet sich der kombinierte Einsatz von Fuzzy-Logik und PN an, diese Beschränkungen zu überwinden. Die Idee besteht darin, dass eine graphische Darstellung basierend auf einem Petri-Netz mit einer leistungsstarken und intuitiven Fuzzy-Logik in der Modellierung verknüpft wird. Durch diesen Ansatz wird eine natürlich sprachliche Beschreibung oder Wenn-Dann-Regeln zugelassen und bestärkt. Es gibt verschiedenste Petri-Netz und Fuzzy-Logik (PNFL)-Ansätze in der Literatur, auf welche an dieser Stelle verwiesen wird. Mit dieser Methode kann qualitatives Wissen und quantitative Modellierung zusammengebracht werden. [vgl. Schröder 2009, 153; vgl. Dimitrov und Korotkich 2010, 254 ff; vgl. Windhager 2013, XV]

2.4 Projektmanagement und Projektmodellierung mit Petri-Netz-Konzepten

Dieses Kapitel beschäftigt sich mit den Anforderungen an ein modernes PM bezüglich bereits bestehender Erweiterungen, basierend auf PN für den Einsatz im PM-Umfeld. Es wird eine beispielhafte Auswahl an PM-Erweiterungen verglichen, welche auf PN-Konzepten aufbauen. Die Auswahl der Konzepte liegt in der Thematik begründet. Die Auswahl der untersuchten Konzepte beschränkt sich auf eine methodische Analyse und schließt somit aus, dass diese bereits in der Praxis Anwendung fanden.

Die Auswahl an Werkzeugen zur Unterstützung der Anwender bei unterschiedlichsten Projektmanagementtätigkeiten ist groß. Es existiert eine Vielzahl von unterschiedlichsten Werkzeugen auf dem Markt. Es gibt Untersuchungen welche aufzeigen, dass schon viele Tools, die nicht aus dem reinen PM-Umfeld kommen, zur Abbildung von Projektmanagementtätigkeiten herangezogen werden können. Der Einsatz von Höheren PN in der Industrie ist etabliert und in verschiedensten Einsatzgebieten zu finden. [vgl. W.M.P. van der Aalst 1994, 45 ff; vgl. Francke 2017, 9 ff]

Die Einleitung bilden einige gefundene Ansätze, welche kurz vorgestellt werden. Anschließend folgt das Aufzeigen der Notwendigkeit von Petri-Netzen sowie etwaige Vor- und Nachteile. [vgl. Thein 2017, 36 f] Dabei beschränkt sich das Kapitel zum einen auf PN-Konzepte und zum anderen insbesondere auf die Methoden zur Ressourcenallokation, Critical Path Method (CPM), PERT (Program Evaluation and Review Technique) und Netzplantechnik. Gründe für diese Abgrenzung der Methoden des Projektmanagements sind die folgenden: Im Fokus dieser Arbeit stehen nicht die hochpreisigen Alleskönner im PM-Umfeld, sondern die im Projektalltag zur Verfügung stehenden Möglichkeiten, welche aus inhaltlichen und wirtschaftlichen Gründen ohne einen professionellen Tooleinsatz oder nur unterstützend agieren. Die Ressourcenallokation wird immer wieder als ein wichtiger entscheidender Faktor für eine erfolgreiche Projektabwicklung genannt. Sie weisen eine hohe praktische Verbreitung auf und sind Bestandteil fast aller Managementaktivitäten im PM. Software-Tools aus dem PM basieren häufig auf denselben formalisierten Methoden. Der Vergleich mit einem erweiterten Scope würde daher kaum neue Erkenntnisse bringen.

2.4.1 Existente Petri-Netz-Ansätze des Projektmanagements

Es wurde eine Reihe von signifikanten Publikationen über den Einsatz von Petri-Netzen im Projektmanagement oder angrenzenden Themengebieten betrachtet. Die interessantesten Forschungen mit einem ganzheitlichen Bezug zu PM werden zur Übersicht in einer Tabelle im Anhang G vorgestellt und anschließend werden beispielhaft einige ausgesuchte Forschungen zusammenfassend beschrieben. Im Rahmen jenes Kapitels wird sich auf eine Übersicht und eine zusammenfassende Schlussfolgerung hinsichtlich des Einsatzes im Rahmen des PM beschränkt.

Vergleich der Petri-Netz-Ansätze des Projektmanagements

Ansätze im PM-Umfeld welche auf PN-Konzepte basieren, haben sich schon seit den 1980er Jahren als ein relevantes wissenschaftliches Feld etabliert. Dabei wurde in den Jahren bis 2000 in dem Themenfeld immer wieder versucht, die Projektzeit durch einen optimaleren Ressourceneinsatz oder Fortschrittsmanagement zu optimieren. Die Untersuchungen wurden auch im Bereich der Softwareentwicklung und das Handling deren Artefakte ausgedehnt. Ein Projekt durchlebt mehrere Zustände. Hier wurde versucht die historischen Methoden mit PN zu erweitern. Auch der Einsatz von stochastischen PN (SPN), Berechnung von Durchführungskosten und -zeiten, Fuzzy-Logik, sowie unvorhergesehenen Ereignissen wurden durch den Einsatz von PN mit dessen Vorteilen in das PM eingebracht. [vgl. Lee, Lu, und Lin 1994; vgl. Chang u. a. 1998; vgl. Kumar und Ganesh 1998; vgl. Jongwook Kim, Desrochers, und Sanderson 1995; vgl. Deng, Wendt, und Rolstadås 1998; vgl. Lee, Lu, und Lin 1994; vgl. Belhe und Kusiak 1993; vgl. Chun-suk Park, Gang-soo Lee, und Jung-mo Yoon 1993]

Seit 2000 waren die Ressourcenoptimierung und das Ressourcenmanagement ein häufiges und vielfältiges Untersuchungsgebiet. Alternative Pfade und die Simulation sind dabei oft aufgegriffen worden, um das PM zu bereichern. Dabei wurde das Multiprojektumfeld mit der Planung mehrerer gleichzeitiger Projekte vereinzelt hervorgehoben. Gegenseitige Abhängigkeiten und Beziehungen hatten einen Einfluss auf zu simulierende Projektergebnisse. Ebenfalls wurde die Thematik von unerwarteten Problemen und dem dynamischen Charakter von Projekten durch PN-Methoden betrachtet. Aber auch die Terminplanung, Zeiten und Kostenoptimierung, grafische

Aufbereitung und Analyse wurden betrachtet. Eine Arbeit [Cohen und Zwikael 2008] hat sich der Eignung von PN für den Einsatz im PM gewidmet. Ein oft gewähltes Mittel war TPN und CPN oder weitere PN-Erweiterungen. [vgl. Cohen und Zwikael 2008; vgl. Cheng u. a. 2013; vgl. Xing-cun Sun u. a. 2013; vgl. Davidrajuh 2012; vgl. Aloini, Dulmin, und Mininno 2012; vgl. Mejía und Montoya 2010; vgl. Yu, Huang, und Wang 2009; vgl. Wu u. a. 2009; vgl. W.-B. Hu u. a. 2012; vgl. Lin, Tai, und Hu 2013; vgl. Yen-Liang Chen, Ping-Yu Hsu, und Yuan-Bin Chang 2008; vgl. Kumanan und Raja 2008; vgl. Shi, Li, und Chai 2007; vgl. Haji und Darabi 2007; vgl. Raja und Kumanan 2007; vgl. Kao, Hsieh, und Yeh 2006; Kao u. a. 2006; Sawhney, Mund, und Chaitavatputtiporn 2003; Prashant Reddy, Kumanan, und Krishnaiah Chetty 2001; Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000]

In den letzten fünf Jahren war das Thema der Ressourcenoptimierung dominierend in den gefundenen Publikationen. Dabei wurde versucht, die Ressourcenzuweisung und den kritischen Pfad mit PN zu modellieren, simulieren und als Ergebnis zu optimieren. Dabei wurden beispielsweise die PN Erweiterungen wie TPN, SPN und CPN herangezogen. [vgl. Lin und Dai 2014; vgl. W. Hu und Wang 2014; vgl. Kuzenkov, Zebzeev, und Gromakov 2014; vgl. Niño Mora u. a. 2015; vgl. Liu u. a. 2016; vgl. Bevilacqua, Ciarapica, und Giovanni 2018] Ebenfalls wurde schon versucht, die Problematik der wiederverwendbaren Ressource mit in die Modellierung zu integrieren. Hier kamen sogenannte Puffer-Plätze zum Einsatz, welche die Ressource wieder zurückgeführt haben. Im Rahmen dieser Arbeit wird dessen Rückführung und Generierung betrachtet. [Kumanan und Raja 2008]

Auch bei einer Untersuchung 2015 für die Animations- und Videospiegelindustrie wurde eine Vielzahl kommerzieller Softwarepakete (kostenlos oder kostenpflichtig) für das Projektmanagement untersucht. Eine der am häufigsten verwendeten Software für das Ressourcenmanagement ist unter anderem MS Project®, da die funktionale Plattform und die Kompatibilität mit dem Office-Paket den Benutzern die Nutzung des Systems erleichtern. Darüber hinaus gestattet die grafische Komponente die Verfolgung der Aktivitäten und Ressourcen durch Gantt-Diagramme. Das MS Project® ermöglicht im Voraus eine Ressourcenzuteilung, die vom Benutzer manuell vorgenommen wird, wenn das Projekt auf der Plattform erstellt wird. Dann kann einer Ressource gleichzeitig zwei oder mehr Aufgaben zugewiesen werden. Folglich gibt es eine Überbelegung von

Ressourcen, die nicht zulässig ist. In diesem Fall arrangiert dieses Tool eine automatische Umverteilung von Ressourcen durch Verdrängung einiger Aktivitäten, die am Konflikt beteiligt sind. Die Aktivitäten sind nach ihrem Vorrang in Reihe angeordnet. Daher verschwindet der Konflikt, aber die Projektabschlusszeit erhöht sich. Darüber hinaus erlauben sowohl die manuelle Zuteilung als auch die Umverteilung von Ressourcen keine Zuteilung nach Rollen. Ebenso ist die Implementierung von Optimierungsmethoden zur Projektschätzung nicht möglich. Dieselbe Situation tritt bei der Software TeamWork.com und ProjectManager.com auf, obwohl diese Software nicht nur dieselben Vorteile wie MS Project® bietet, sondern auch die Ressourcenverwaltung, die Verfolgung der Projektfortschritte und die Planung ermöglicht. Sie teilen sich aber auch Schwächen in Bezug auf die Ressourcenzuteilung und Optimierungsmethoden. [vgl. Niño Mora u. a. 2015]

Ein Autor [Mejía u. a. 2016] hat ein allumfassendes Framework zur Modellierung, Ablaufplanung und Simulation von Projekten mit **(1)** neu entwickelten PN-Erweiterung, **(2)** zwei PN-Muster und **(3)** einem Graphen-Such-Algorithmus untersucht. Dabei wurde als Ausgangsbasis ein Business Process Model and Notation (BPMN) in ein PN überführt, um darauf die Analyse und Erkenntnisse für die erneute Planung zu gewinnen. Auch Ansätze aus der Multidomänenmatrix in Verbindung mit CPN, stochastische PN, Terminplanung und die Untersuchung des optimalen Weges sind präsent. [vgl. Topic und Jevtic 2019; vgl. Parida u. a. 2018; vgl. Mejía u. a. 2016; vgl. V. A. Jeetendra, Kumanan, und O.V. Krishnaiah 1998; vgl. Qu und Wang 2015] In 2018 wurde im Ansatz von [Bevilacqua, Ciarapica, und Giovanni 2018] das Runterbrechen einzelner Aktivitäten durch einen Projektstrukturplan zur Vorbereitung der PN-Überführung vorgenommen. Hierbei wurden keinerlei Kriterien oder Methoden definiert. In Bezug auf das Projektplanungsproblem gibt es wenige Arbeiten in der Literatur, die Petri-Netze sowohl für die Modellierung als auch für die Optimierung verwenden. [Mejía u. a. 2016]

2.4.2 Nutzen von Petri-Netz-Konzepten im PM-Umfeld

PM-Methoden zählen zu den meist genutzten Management-Tools in der Industrie. Die gestellten Anforderungen an ein erfolgreiches PM hängen von der Art und Beschaffenheit des zugrundeliegenden Projektes ab. [vgl. Badiru 1995, 38 ff] Die meisten Projekte finden in einem dynamischen Umfeld statt, welche verschiedensten Unsicherheiten unterliegen.

Unsicherheiten beziehen sich insbesondere auf Ressourcen und Zeiten, was häufigere Anpassungen und Überarbeitungen des aufgestellten Plans nach sich ziehen. In der Praxis wird oftmals auf die Einarbeitung geringfügiger Abweichungen verzichtet, bzw. auf die Einarbeitung maßgeblicher Abweichungen beschränkt. Die daraus resultierende Unvollständigkeit stellt eine potenzielle Fehlerquelle dar und kann sich auf die Qualität auswirken. Ein weiteres Problem ist die schlechte Anwendbarkeit von agilen Vorgehensmodellen und den daraus ständig resultierenden Änderungen. [vgl. Mejía u. a. 2016, 19]

Das führt zu Anforderungen an das PM, welche oft durch das klassische PM und besonders durch die Netzplantechnik nicht abgedeckt werden können. Diese Methoden werden seit den 1940er Jahren eingesetzt und erfuhren seitdem keine angemessene Anpassung. Teilweise führt es dazu, dass moderne Projekte aus einer statischen Perspektive betrachtet werden und deren dynamischer Charakter somit unberücksichtigt bleibt. Neben Unsicherheiten betrifft dies auch unvorhergesehene Störungen und Ausnahmefälle, wie auch alle anderen zustandsverändernden Ereignisse. Weitere Gründe sind die begrenzte Verständnishilfe für Projektbeteiligte oder die mangelnde Überwachung des Projektfortschritts ohne Analyse der Performance im Projekt. Ebenfalls werden die Eigenschaften realer Projekte je nach Anwendungsgebiet nur unzureichend erfasst und dies führt zu dem Schluss, dass klassische PM-Methoden nur bedingt bzw. gar nicht zur Bewältigung der folgenden Aufgaben geeignet sind: Modellierung Ressourcenverbrauch und -nivellierung, Anpassung Projektplanung und Ablauf, automatisches Projekt-Monitoring und -Controlling in Echtzeit, Abbildung mehrerer paralleler Vorgänge, dynamische grafische Visualisierungen, analytische Fähigkeiten und Simulationsfähigkeiten. Im Anhang E ist eine detailliertere Aufzählung zu finden. [vgl. Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000, 517 ff; vgl. Thein 2017, 37 f; vgl. Xing-cun Sun u. a. 2013, 515; vgl. Yen-Liang Chen, Ping-Yu Hsu, und Yuan-Bin Chang 2008, 151; vgl. Ashok Kumar und Ganesh 1998, 50 ff; vgl. Jongwook Kim, Desrochers, und Sanderson 1995, 266 ff; vgl. Lung-chun Liu und Horowitz 1989, 1282]

Das Projektmanagement wurde als ein zukünftiges Teilgebiet identifiziert, in dem die Stärken des PN bezüglich der dynamischen Repräsentation und Monitoring durch alle Phasen eines Projektes zum Einsatz kommen können. Viele Forschungen haben PN schon als Management-Tool untersucht. [vgl. W.M.P. van der Aalst 1994]

Vorteile von Petri-Netz-Konzepten

Im Artikel [Cohen und Zwikael 2008] wurde das Feld des PM untersucht und festgestellt, dass die meisten Projektmanager Projekte mit Ressourceneinschränkungen und Unsicherheiten planen müssen. Umfangreiche, langfristige und komplexe Projekte mit großer Unsicherheit und begrenzter Ressourcenverfügbarkeit stellen die einfachen Modellierungswerkzeuge des Projektmanagements wie Gantt-Diagramme, CPM und PERT in Frage. Als Beispiel wurde angeführt, dass PERT ungenaue Informationen liefert, welche die Schätzungen oft als nicht hilfreich erweisen. Ein weiteres Beispiel ist die Überarbeitung der Modellierung: Ein bestimmter Prozentsatz der Aufgaben enthält Fehler und muss wiederholt werden. Keines der obigen Modellierungsverfahren kann einen solchen Nacharbeitsprozess genau modellieren. In der Ausarbeitung des Artikels wurde ebenfalls festgestellt, dass die meisten PN-Simulationen typischerweise das sich wiederholende Verhalten von Aufgaben oder Prozessen betrachten, und nur wenige Artikel sich auf die Kombination von PN und Projektplanung konzentrieren. [Cohen und Zwikael 2008]

Das PN kann als visuelle Kommunikationshilfe, ähnlich wie ein Ablauf-, Block- oder Netzwerkdiagramm genutzt werden. Der wesentliche Unterschied hinsichtlich der PN besteht in der Präsenz der Token, die für Simulationen von dynamischen Verhalten sowie gleichzeitigen und zeitversetzten Aktivitäten stehen. Dies ermöglicht die Darstellung von Systemverhalten durch die Einbindung von Zustandsgleichungen, Algebra und mathematischen Gleichungen. [vgl. Kumanan und Raja 2008, 1742 ff]

Die gestiegenen Anforderungen aus dem Projektalltag und die aufgezeigten Unzulänglichkeiten erfordern neue Methoden. Erweiterte PN-Konzepte können diesem Anspruch gerecht werden. Sie verfügen über weitaus umfangreichere Funktionen und können zum anderen mit relativ geringem Aufwand um neue Funktionen ergänzt werden. Die Vorteile des Einsatzes von PN werden durch folgende Aspekte begründet: [vgl. Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000, 519; vgl. Kumar und Ganesh 1998, 55]

Im Anhang E wird ausführlich auf die Vorteile der PN im PM-Umfeld eingegangen und an dieser Stelle nur eine kurze Zusammenfassung dieser wiedergegeben. **(1)** PN basieren

auf einem mächtigen und gut verständlichen Formalismus und einer formal-mathematischen Grundlage. Dies ermöglicht es, alle dynamischen Systemzustände darzustellen. Die Modellierung des Ablaufes inkl. erreichter Fortschritte sowie die enthaltene Logik kann zur Analyse aufbereitet werden. **(2)** Der breite Funktionsumfang (Analyse, Simulation, etc.) ermöglicht vielfältige Einsatzzwecke entlang der Phasen im PM. **(3)** Die Vielzahl vorhandener PN-Erweiterungen können für das Projekt herangezogen werden. Darunter zählen beispielsweise: die dynamische Abbildung von Zustandsänderungen, Abbildung von konkurrierenden, gemeinsamen, teilweisen und substituierten Ressourcenverbrauch durch Aktivitäten, Abbildung von Konflikten und Engpässen. **(4)** Bei der Modellierung können Vereinfachungen oder Zusammenfassungen ohne Informationsverlust, durch die Hilfe von Teilnetzen, dargestellt werden. **(5)** Echtzeitüberwachung und -analyse des Projektfortschrittes. **(6)** Verwirklichen von Echtzeitanpassungen und Echtzeitüberarbeitungen während der Projektplanung. **(7)** Existierende Projektmodellierungsmethoden und Techniken wie Gantt-Chart, Critical Path Method, PERT, Work Breakdown Structures (WBS) und Project Network Techniques (PNTs) mangelt es an einer formalen Beschreibung. **(8)** Analyse von beispielsweise Wechselwirkungen, Schäden, Konflikten, oder Ersatzmöglichkeiten. **(9)** Ressourcenumverteilung nach Optimierungsmethoden werden von PM-Tools fast nicht unterstützt. **(10)** Professionelle Tools sind auf ihren ausgelegten, wenn auch umfangreichen, Funktionsumfang beschränkt und nicht von außen veränderbar. Ein formaler Ansatz, beruhend auf formalisierten Tabellen, kann wiederum erweitert und auf das Problem angepasst werden. [vgl. W.M.P. van der Aalst und van Hee 1996, 25; vgl. Mejía u. a. 2016, 191; vgl. Aichele 2006, 113; vgl. Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000, 517; vgl. Deng, Wendt, und Rolstadås 1998, 84; vgl. Thein 2017, 101; vgl. V. A. Jeetendra, Kumanan, und O.V. Krishnaiah 1998; vgl. Kumanan und Raja 2008, 1742 ff; vgl. Bevilacqua, Ciarapica, und Giovanni 2018, 58 ff; vgl. Niño Mora u. a. 2015, 3]

Nachteile von Petri-Netz-Konzepten

Neben den genannten Vorteilen geht der Einsatz von PN auch mit Nachteilen einher. Abhängig vom Projektumfeld bzw. der konkreten Problemstellung, muss eine Abwägung erfolgen. Folgende wesentliche Aspekte konnten identifiziert werden. [vgl. Thein 2017, 40 ff] Im Anhang E wird ausführlich auf die Nachteile der PN im PM-Umfeld

eingegangen und an dieser Stelle nur eine kurze Zusammenfassung dieser wiedergegeben. **(1)** PN-Modelle von komplexen Projekten können zu großen Umfängen und hoher Komplexität neigen. **(2)** Die Modellierung und Bearbeitung mit Hilfe von PN kann, selbst bei kleinen und einfachen Projekten, manuell schwer durchgeführt werden. **(3)** Die Kombination von anderen Methoden mit PN ist erst seit 2000 hinreichend gestartet und noch ein junges Forschungsfeld (objektorientierte Modellierungsmethoden, Projektstrukturpläne und Balkendiagramme, Modellierungssprache IDEF0 (Icam DEFinition for Function Modeling), Netzplantechnik oder dem Konzept der Fuzzy-Logik). [vgl. Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000, 516; vgl. Zapf und Heinzl 2000, 36; vgl. Kumar und Ganesh 1998, 55; vgl. Deng, Wendt, und Rolstadås 1998, 82 ff]

2.4.3 Zusammenfassung

Um den Anforderungen des heutigen PM gerecht zu werden, ist der Bedarf an neuen Methoden begründet. Dies bewerkstelligt eine angemessene Beachtung des dynamischen und komplexen Charakters heutiger Projekte. PN bringen neben einem formal-mathematischen Formalismus auch einen breiten, anpassbaren und erweiterbaren Funktionsumfang mit sich. Weiterhin kann der Einsatz in Echtzeit und skaliert mit dem Projektumfang erfolgen. Die Nachteile müssen dabei stets abgewogen werden und erfordern ihrerseits eigene Erweiterungen in der Zukunft. Diese sind in hohem Aufwand, der zwingenden Notwendigkeit einer SW-Unterstützung und der mangelnden Integration mit anderen Modellierungsmethoden zu sehen. [vgl. Thein 2017, 42 ff]

Unzulänglichkeiten von herkömmlichen PM-Tools sind unter anderem: keine automatische Neuplanung von Aktivitäten, keine angemessene Konfliktlösungsmöglichkeit im Rahmen der Ressourcenpriorisierung, keine Darstellung von Ressourcenwechselwirkungen, keine Ursachenforschung für verspätete Prozesse von Aktivitäten sowie keine Hilfe bei anteiliger Auslastung/Zuteilung gegenseitiger Exklusivität und Ressourcenersatz. [vgl. Kumanan und Raja 2008, 1742 ff] Ebenfalls ist immer wieder zu beobachten, dass in vielen Projekten auf Excel und/oder PowerPoint zurückgegriffen wird. Vorlagen werden jedes Mal neu entworfen und an die neuen Projektgegebenheiten angepasst. Dieses Vorgehen erfordert immer einen erheblichen Anfangsaufwand, ganz abgesehen von der entstandenen Komplexität, fehlender

Wiederverwendbarkeit durch ständige Anpassungen und Fehleranfälligkeit. Der Grund wieso sich dies in vielen Projekten etabliert hat, ist darauf zurückzuführen, dass nur ausgewählte Teilbereiche des PM herausgegriffen werden, Microsoft Office® einen sehr hohen Verbreitungsgrad hat und das Erlernen einer neuen Software inkl. Lizenzkosten entfällt.

Als eines der wichtigsten Vorteile kann herausgearbeitet werden, dass ein einheitlicher Zugang der zugrunde gelegten Logik im PM durch das PN gewährleistet werden kann. Die Abbildung von Änderungen kann so effizient und zentral erfolgen. Unabhängig vom Tooleinsatz sind Änderungen am Funktionsumfang oder Erweiterungen immer mit viel Aufwand verbunden, können nicht zentral erfolgen und sind teilweise nicht möglich, abhängig vom Tool. Es kann festgehalten werden, dass zahlreiche Erweiterungen und Modifikationen existieren, um PN für bestimmte Applikationen anzupassen. Es wurde noch nicht hinreichend untersucht, wie ein konsistentes Framework (Rahmengerüst) zur Modellierung und Analyse von festgestellten/gegebenen Projektmerkmalen gestaltet werden kann, was Vorgängerbeziehungen oder Priorisierungen, sequentielle Ausführungen, Konflikte, Nebenläufigkeiten, Aspekte der Synchronisierung, Zusammenschlüsse, Alternativwege oder dynamische Aspekte beleuchtet. [vgl. W.M.P. van der Aalst 1994]

Zusammenfassend kann aus den betrachteten Publikationen erkannt werden, dass folgende Problemstellungen oft aufgeführt werden: wachsende Umfänge, Unsicherheiten, Anforderungen und Komplexität. Die Bedeutung von Planung, Monitoring, Controlling und Echtzeitanalyse parallel zum Projektgeschehen wächst zunehmend. Ebenfalls entsteht dadurch das Bedürfnis nach einer dynamischen Projektplanung und speziell nach einer dynamischen Erstellung des Ablaufplanes und Ressourcensteuerung. Die klassischen Tools zum PM werden diesen gesteigerten Ansprüchen nicht mehr gerecht und der Bedarf an einer realitätsnahen Modellierung von Projekten muss gedeckt werden. Es wird stets erwähnt, dass die Eigenschaften von PN für eine realitätsnahe und dynamische Modellierung ihren Nutzen begründen (z. B. dynamisch, nebenläufig, nichtdeterministisch, formal-mathematisch). Bei der Bewältigung der Probleme sind folgende Ansätze ein häufig gewähltes Mittel: Automatisierung auf Basis eines PN-Modells, bestehende PN-Erweiterungen weiter zu entwickeln, den Einsatz von PN-Mustern für vielfach wiederkehrende

Problemsituationen und die Verknüpfung von PN-Konzepten mit bestehenden Methoden, Projektmodellen und IT-Systemen. Es wurde kein formeller Ansatz für die Überführung des PM in eine tabellarische Struktur gefunden, um diese für ein PN zu verwenden. [vgl. Thein 2017, 48; vgl. Weichenhain und Fengler 2019a]

Als Resümee wird auf die wohl drei wichtigsten Gründe eingegangen. Als Erstes wird auf die formale Semantik trotz grafischer Natur der PN, gefolgt von zustandsbasierten und ereignisbasierten Abbildungen sowie abschließend auf die Fülle an Analysetechniken zur Untersuchung von PNs eingegangen. [vgl. Kaya 2017b, 41 ff] Die Modellierung von Logik kann ungeachtet der grafischen Natur von PNs dank der formalen Semantik in eine formale als auch grafische Sprache dargelegt werden. Unter anderem kann so auch ohne IT-Affinität ein Ablauf logisch dargestellt oder durch die Eindeutigkeit anderweitigen Interpretationen vermieden sowie Systemstillständen, Blockaden, etc. erkannt werden. [vgl. W. M. P. Aalst 1998, 174; vgl. Kaya 2017b, 41 ff]

Durch eine zustandsbasierte Abbildung können nicht nur einzelne Schritte, sondern auch lokale Zustände oder Teilmarkierungen modelliert werden. Ein Zustandsübergang wird explizit durch die Weitergabe einer Marke abgebildet. Dies ermöglicht beispielsweise die Unterscheidung zwischen einer Freigabe und Ausführung einer Tätigkeit. Ebenfalls ist das Schalten einer Transition ereignisbasiert. [vgl. W. M. P. Aalst 1998, 174; vgl. Kaya 2017b, 41 ff]

Als Letztes soll die Fülle an Analysetechniken eines PN durch den mitgebrachten Formalismus und somit einhergehender einfacher Repräsentation des abzubildenden Sachverhaltes vorteilhaft hervorgehoben werden. Die prägnante operationelle Semantik mit dem Formalismus ermöglicht eine unmittelbare Simulation und Anwendung der Analysetechniken. Beispielsweise Überdeckungsgraphen, Platz-Invarianten, verteilte Abläufe, Leistungsbewertungen zur Ressourcenauslastung oder Durchlaufzeiten sowie der Prozess-Mining Algorithmus, welche eine Überprüfung erlauben, ob der modellierte Prozess in der realen Welt auch zur Anwendung kommt. Näheres enthalten die angegebenen Quellen. Die Markierungen halten alle eingetretenen Zustände fest. Das ermöglicht eine Analyse und Fehlerursache von jeglichen erwünschten und unerwünschten Zuständen. [vgl. Kaya 2017b, 42 ff; Wil M. P. van der Aalst 2011, 2013]

2.5 Ziele und Abgrenzung

Dieser Abschnitt zeigt den Ansatz und die Abgrenzung dieser Arbeit. Die Ziele und Abgrenzungen werden im Gegensatz zu Kapitel 1.3 genauer dargelegt und anhand des vorliegenden Kapitels näher betrachtet. Die Ziele sind vielfältig und sollen der Verwirklichung eines angestrebten Ansatzes einer simulierten Projektplanung und dessen Modellierung zur Ausübung von einzelnen Aktivitäten im Rahmen des Projektmanagements zur Planung und Steuerung beitragen. Ebenfalls begründen sie den Umfang der Arbeit. Der Lösungsansatz der simulierten Projektplanung ist in der Forschungsfrage begründet, die sich aus den folgenden Punkten ableitet:

(1) Ziel ist eine Methode für das Projektmanagement, welche aus einer formalisierten Eingabe und einer Transformation in eine weitgehend formalisierte algorithmische Struktur besteht. Diese ist definiert aus Konstrukten, die durch Variabilität an konkreten Managementaufgaben und -situationen anpassbar sind. Komplexe vollumfängliche PM-Tools haben einen anderen Focus: Sie sind weitgehend universell ausgelegt und versuchen damit alle möglichen konkreten Aufgaben und Situationen abzudecken. Das kann bedingt durch das zugrunde gelegte Prinzip nur eingeschränkt möglich sein. Der hier verfolgte Ansatz ermöglicht im Gegensatz dazu, gekennzeichnet durch eine integrierte variierbare Algorithmik, auf Basis eines theoretisch fundierten Modellierungsmittels eine toolbasierte Managementunterstützung. Weiterhin gibt es auch wirtschaftliche und inhaltliche Kriterien, die einen professionellen Tooleinsatz aus inhaltlichen und wirtschaftlichen Gründen nicht bevorzugen.

(2) Analyse und Formalisierung einer tabellarischen Struktur dient als Grundlage für eine Modellierung einer simulierten Projektplanung. Dafür soll eine Tabellenform als standardisierte Basis definiert werden, welche als Grundlage für einen Algorithmus zur Überführung in ein PN herangezogen werden kann. Somit wird ermöglicht, dass die Datenbasis auf einer vielseitig bekannten und benutzten Grundlage aufbaut. Die definierte Struktur soll den Anforderungen für eine Überführung in ein PN gerecht werden und integriert geeignete Forschungsfelder von PN-Strukturen. Damit einhergehend, soll eine Untersuchung zu Forschungsfeldern der schon vorhandenen Erweiterungen von PN zur Abbildung von PM-Tätigkeiten erfolgen. Die damit verbundenen Vorteile müssen identifiziert werden.

(3) Eine modulare und simulierbare Projektmodellierung auf der theoretischen Grundlage eines Petri-Netzes muss auf einem geeigneten Abstraktionslevel beruhen. Im Zuge dessen soll auch das Zusammenspiel modellbasierter Anforderungen beleuchtet werden.

(4) Es soll die Abbildung von hierarchischen Konstruktionen eines Ablaufs auf die Mengen der formalisierten Tabellenstruktur erfolgen.

(5) Geeignete Erweiterungen und Interpretationen von PN-Modellen sind zu identifizieren. Aufbauend müssen Transformationsregeln aus der formalisierten Tabelle einschließlich der zusätzlichen Hierarchieinformationen durch geeignete erweiterte PN aufgestellt werden.

(6) Schaffung eines Simulationskontextes der Petri-Netz-Modelle für die Zeitpunkte vor Projektbeginn (initiale Planung) und während des Projektes (aktualisierte Projektzustände).

(7) Im Verlauf der Arbeit soll dies anhand eines für den Zweck reduzierten Beispielprojektes (Auszug aus einem realen Projekt mit vollem Umfang) mit einem Algorithmus in ein PN überführt werden. Was darauf abzielt, den Nachweis aller methodischen Schritte durch Modellierung und Simulation eines aus der Praxis stammenden realen Beispielprojektes zu verifizieren. Das bestärkt die angestrebte Verallgemeinerung des Projektcharakters. Das Modell soll dem Anspruch genügen, dass es allgemeiner Natur ist und somit formalisiert werden kann.

Zusammenfassend ist das Ziel dieser Arbeit die Untersuchung, ob sich Konzepte, basierend auf dem theoretischen Modell eines Petri-Netzes, zur formalen Modellierung von Projekten und dessen verschiedenen Projektkreisläufen eignen. Die Arbeit beschränkt sich auf den methodischen Ansatz und der somit geschaffenen Möglichkeit der Umsetzung. Auf die Umsetzung eines vollumfänglichen Beispiels wird verzichtet. Es ist ebenso nicht das Ziel, die Eignung von Petri-Netz-Konzepten zur Bearbeitung aller Aufgaben des Projektmanagements zu untersuchen. Es wird sich auf die zwei Phasen Planung und Steuerung konzentriert, mit dem Schwerpunkt auf die Ressourcenplanung.

3 ^{Kapitel} Projekte als Gegenstand von diskreter Modellierung

Das Projektmanagement und das Konzept des Petri-Netzes unterscheiden sich voneinander und somit müssen für die Modellierung des Projektmanagements deren komplexen Aufgaben abstrahiert werden. Es muss, wie bei Petri-Netzen, eine Zerlegung des gesamten Modells in Teilsysteme inkl. der Wechselwirkungen und deren Komponenten geschehen. Eine Zusammenführung ist nur dann möglich, wenn ein gemeinsamer Detailierungs- und Abstraktionsgrad gefunden werden kann. Im Petri-Netz wird die Komplexität in die Verbindung der Teilsysteme gesteckt, das Verhalten der einzelnen Teilsysteme ist hingegen einfach gehalten. [vgl. Reisig und Desel 2014, 172; vgl. Schott 2019, 20 f]

Das Projektmanagement ist nicht wie ein PN in ein Modell unterteilt, es bestehend aus Teilsystemen. Das Kapitel identifiziert die Anforderungen an eine Abbildung des PM durch PN und definiert eine geeignete gemeinsame Abstraktionsebene. Diese Abstraktionsebene wird als Grundlage dienen und auf kleinste abzubildende Objekte und deren Komponenten ausgelegt sein. [vgl. Schott 2019, 20 f]

Den Modellierungsgegenstand bilden die Projekt-Objekte mit deren Operationen und Beziehungen zueinander. Zur Gewährleistung einer formalen und einheitlichen Darstellung ist es zudem notwendig, die Eigenschaften von Objekten zu klassifizieren und deren Anforderungen abzuleiten. Auf dieser Grundlage kann eine formale Tabellenstruktur und somit eine Methode zur Überführung und Adaption des PM definiert werden. Da es sich um eine dynamische Betrachtung handelt, ist ebenfalls der Kreislauf dieser formalen Modellierung als Grundlage der entwickelten Methode darzustellen.

Der Begriff der Formalisierung wird in dieser Arbeit in dem Kontext verwendet, dass möglichst präzise Definitionen zum Vorgehen der Transformation eines PM in ein PN gegeben werden sollen. In dieser Arbeit wird die Definition der Formalisierung nach der Auffassung von [Busch 1998] adaptiert, einen Sachverhalt in eine Form bringen bzw.

eine Form vorgeben, um so das Problem darstellen zu können. Dabei kann eine Formalisierung auf unterschiedlichste Art erfolgen, beispielsweise eine Form in Gestalt einer Formel, sprachlicher Formulierung oder Struktur. Obendrein muss stets die Formalisierung reproduzierbar und somit eindeutig sein. Bei der hier angestrebten Formalisierung wird immer Wert auf die praktische Veranschaulichung gelegt. [vgl. Raabe u. a. 2012, 112 f; vgl. Busch 1998, 69 f]

3.1 Modellierungsgegenstand

Bei der Modellierung von Projekten mit Petri-Netz-Konzepten muss die Adaption von Zustandselementen und Ereignissen aus dem PN ins PM übertragen werden. Dafür wird der Begriff **Projekt-Objekt** eingeführt und in Verbindung mit diskreten Modellen näher betrachtet.

Bei den Projekt-Objekten handelt es sich um **Aktivitäten, Ziele, Ressourcen** (alle Arten von Ressourcen, wie z. B. Rohstoffe, Personal, Organisationseinheiten oder Fähigkeiten), **Zeit** und **Beziehungen**. Die Projekt-Objekte stehen miteinander in Beziehung. Grundlegend kann die Aussage getroffen werden, dass die Aktivitäten die anderen Projekt-Objekte entweder als Eingangsgröße, Verbrauch oder Ausgangsgröße benutzen. Als eine Aktivität werden Vorgänge verstanden, welche sich wiederum zu Aktivitätsketten zusammenschließen lassen. [vgl. Weichenhain, Fengler, und Streitferdt 2019]

Im Projektmanagement werden Aktivitäten gewöhnlich auf der Basis von Vorgängen oder gar Teilprojekten definiert. Im Rahmen des PPNM wird hier der Begriff **atomare Aktivitätsphase** eingeführt. Auch diese verändern durch Ereignisse (Start der Aktivität, gesetzte Ziele, Beziehungen und Auswirkungen zu anderen Aktivitäten sowie verbrauchte Ressourcen und Zeiten) ihren Status. Eine Aktivität wird in **Phasen (Design und Entwicklung)** unterteilt. Durch diese Unterteilung müssen keine verschiedenen Aktivitäten, sondern lediglich die zu durchlaufenden Phasen einer Aktivität definiert werden. Gerade bei Informationssystemprojekten ist beispielsweise das Design einer Funktion von der Entwicklung im Hinblick auf dessen Eingriffsmöglichkeiten zu trennen. Demzufolge ist die atomare Aktivitätsphase dahingehend unterteilt, dass abgeschlossene Ereignisse zu einer Statusänderung führen.

Aktivitäten stehen in unmittelbarem Zusammenhang mit den anderen Projekt-Objekten. Im ersten Teil soll auf die Aktivität an sich eingegangen werden und im späteren Verlauf auf deren atomare Eigenschaft. Vorgänge und Teilprojekte stellen im Rahmen dieser Definition **Aktivitätsketten** (Abfolgen / Aneinanderreihungen von Aktivitäten) dar. Die Darstellung der Aktivitätsketten kann, sofern sinnvoll, zusammengefasst bzw. vereinfacht erfolgen. Es führt vor allem bei der grafischen Darstellung und manuellen Bearbeitung zu Vorteilen. Bei der Zusammenfassung muss beachtet werden, dass bei der Verdichtung von mehreren Aktivitäten zu einer Aktivität nur eine Vereinfachung gegeben ist, wenn die ausgewählten Aktivitäten nicht durch unterschiedliche Objekte bedient werden. Beispielsweise stellt eine Zusammenfassung von Aktivitäten mit unterschiedlichen Ressourcen keine Vereinfachung eines Projektmodells dar. In eine schon gestartete Aktivität kann nicht mehr eingegriffen werden. Im Gegensatz dazu kann in die Abfolge der Aktivitäten ein Eingreifen, manuell oder automatisiert, geschehen. Der manuelle Eingriff ist dadurch gekennzeichnet, dass ein Projektverantwortlicher eine Entscheidung trifft und dann manuell in den Prozess eingreift. Wohingegen bei einem automatisierten Prozess der Eingriff mit Hilfe einer vorab definierten Entscheidungsregel (basierend auf einer ausreichenden Datenbasis) erfolgt. Damit stellt der automatisierte Eingriff keine Unterbrechung eines Prozesses dar, sondern ist Bestandteil eines automatisierten Prozessablaufes. [vgl. Thein 2017, 33 f; vgl. Weichenhain, Fengler, und Streitferdt 2019]

Ziele definieren das Erzeugen eines bestimmten Outputs pro Vorgang oder eine Aktivitätskette. Somit ist das Erreichen eines definierten Zustands das Ziel einer Aktivität und ebenfalls der Auslöser zum Start einer anderen Aktivität, was dem Grundgedanken eines PN somit entspricht. Ebenfalls kann daraus der Rückschluss gezogen werden, dass sich das **Projektziel** aus dem Erreichen einer Abfolge von Zielen zusammensetzt. **Ressourcen** sind verbrauchbare oder wiederverwendbare Betriebsmittel bzw. Mitarbeiter, die zur Abarbeitung einer Aktivität erforderlich sind. Einer Aktivität werden **Zeiten** gesetzt, welche die Dauer der Aktivität und deren Fristen als Zeitpunkte definiert. [vgl. Thein 2017, 32 ff; vgl. Schott 2019, 23 f; vgl. Weichenhain, Fengler, und Streitferdt 2019]

Wie diese Projekt-Objekte miteinander in Beziehung stehen, wird in einem späteren Abschnitt in diesem Kapitel näher beleuchtet. Dabei wird jedes Projekt-Objekt durch dessen **Eigenschaften** und dem dadurch funktionsspezifischen **Verhalten, auch mögliche Operationen im Zusammenspiel der Beziehungen**, charakterisiert.

3.2 Eigenschaften von Objekten

Dieser Abschnitt beschäftigt sich mit den Eigenschaften von Projekt-Objekten. Wie aus den vorangegangenen Abschnitten ersichtlich, bedarf es zur Modellierung die Projekt-Objekte der Aktivität, Zeit, Ressourcen, Ziele und dessen Beziehungen zueinander. Für die Interaktion, also dessen Beziehungen, sind die jeweiligen Eigenschaften von großer Bedeutung.

Eine der wichtigsten Eigenschaften ist die der **Atomarität**. Auf ihrer Grundlage können die zwei Konzepte des Projektmanagements und des Petri-Netzes auf fest definierten Zustandselementen vereint werden. Hier ist die **Aktivitätsphase** für dessen Definition ausschlaggebend. [vgl. Weichenhain, Fengler, und Streitferdt 2019]

Die Anordnung von Aktivitäten spiegelt den logischen Ablaufplan der Abarbeitung wider. Dabei ist nicht zwingend die zeitliche Abfolge ausschlaggebend, sondern die Anordnungsbeziehung und Abhängigkeit zwischen den Aktivitäten. Demzufolge müssen alle **Vorgänger und Nachfolgerbeziehungen** bekannt und zu berücksichtigen sein. Hierbei können logische Abläufe an **Bedingungen** und durch dessen **Wiederverwendbarkeit** beeinflusst werden. [vgl. Weichenhain, Fengler, und Streitferdt 2019]

Für die Ausführung von Aktivitäten werden **informationelle und physische Objekte** benötigt. Dabei muss unterschieden werden, ob es sich um einen Input oder Output einer Aktivität handelt. Weiterhin ist innerhalb der Konzepte das Verhalten der **Zeit** (Dauer und Datumsangaben) und dessen **Verbrauch** ein ausschlaggebender Taktgeber. Hierbei handelt es sich um einen Teil der **Objekte mit quantitativen Angaben**. Die Steuerung und Kontrolle der laufenden Projekte dienen der Unterstützung des Projektmanagements bei der Realisierung und Kontrolle des Projekterfolges. [vgl. Aichele 2006, 168]

Eine beispielhafte Umsetzung dieser Thematik erfolgt in Kapitel 6, basierend auf ausgewählten Eigenschaften und mit eingeschränktem Scope auf Grundlage des vorgestellten Beispielprojektes von Kapitel 5.

3.2.1 Atomare Einheit

Für die Transformation von Projektmanagementaufgaben in simulierbare diskrete Modelle mittels der theoretischen Grundlage eines Petri-Netzes, müssen die Anforderungen der Atomarität erfüllt werden. Nachfolgend wird die Atomarität zunächst klassifiziert und die notwendigen Anforderungen für die vorgesehene Anwendung abgeleitet. In den nach den Vorgaben des Autors erstellten und betreuten Masterarbeiten von [Schott 2019] und [Thein 2017] wird auf die Atomarität detailliert eingegangen, auf der hier aufgebaut wird.

Das Konzept eines Petri-Netzes basiert auf der Modellierung von den kleinsten möglichen Zustandselementen und deren Ereignissen, die den Status dieser Zustandselemente verändern. Dieser Grundgedanke muss auf das PM übertragen sowie eine geeignete Abstraktionsebene gefunden werden. Diese Abstraktionsebene muss der Anforderung gerecht werden, dass sie alle relevanten Daten und Abläufe berücksichtigt, aber trotzdem abbildbar bleibt und nicht in ihren Einzelheiten versinkt. Es gilt dabei den Grad zwischen ‚so genau wie nötig‘ und ‚so einfach abbildbar wie möglich‘ zu finden. Dafür erfolgt die Einführung der **atomaren Einheit**, welche nachfolgend erläutert wird.

Die atomare Einheit als Eigenschaft über alle fünf definierten Projekt-Objekte (Aktivität, Ziel, Ressource, Zeit und Beziehungen) zu legen, würde die Zweckmäßigkeit nicht erfüllen und den Detaillierungsgrad zu sehr ausweiten. Der Grund dafür liegt darin, dass es sich bei der atomaren Einheit um eine kleinstmögliche Unterteilung handelt. Das Objekt Zeit wird anhand eines Zeitkonzeptes betrachtet. Beziehungen hingegen ergeben sich zwischen den einzelnen Projekt-Objekten und werden durch deren Granularität beeinflusst. Das Ziel einer Aktivitätsphase ist an dessen Umfang und dem möglichen Output gebunden und somit werden die Definitionen dieser drei Projekt-Objekte nicht auf eine atomare Einheit festgelegt. Grundsätzlich kann die Definition der atomaren Einheit aus dem Projektmanagement oder dem Petri-Netz entstehen. Das PM ist dabei ausschlaggebend für die atomare Aktivitätsphase, wohingegen das PN die atomare

Ressourcenzuordnung fordert. Dies verdeutlicht, dass eine Abstimmung der Abstraktionsebenen beider Konzepte von Nöten ist. Beginnend mit der atomaren Aktivitätsphase wird auch der Grundgedanke der atomaren Eigenschaft erläutert. Darauf aufbauend wird sich bei Ressourcen auf diese Eigenschaften berufen und sich auf die atomare Ressourcenzuordnung konzentriert.

Übertragen auf das PPNM ist eine **atomare Aktivitätsphase** der kleinste mögliche Aktivitätsschritt, respektive der kleinste mögliche Vorgang, welcher ohne eine vorgesehene Eingriffsmöglichkeit abgearbeitet werden kann. Das ist insofern wichtig, damit ein Level der Darstellung gefunden wird, wobei keinerlei Managementaktionen (Eingriff, Stoppen, Anhalten oder Ändern von Aktivitäten) einen relevanten Mehrwert erzeugen. Umgekehrt bedeutet dies im normalen Projektalltag, dass sobald eine atomare Aktivitätsphase gestartet wurde, sie bis zum Ende durchlaufen muss. Somit sind feste Punkte zum Eingreifen für das Projektmanagement gegeben. [vgl. Reisig und Desel 2014, 172]

Im Umgang mit atomaren Aktivitätsphasen müssen Regeln identifiziert werden, Eingriffsmöglichkeiten sind allerdings begrenzt und genau definiert. Nach dem Start einer atomaren Aktivitätsphase ist keine Steuerung mehr möglich, gleichwohl ob manuell oder automatisiert. Das führt zur Schlussfolgerung, dass ausschließlich vor dem Start und/oder nach dem Ende einer atomaren Aktivitätsphase ein Eingriff möglich ist. Eine Ausnahme ist das Stoppen einer atomaren Aktivitätsphase. Es führt zu einem Abbruch, und es werden alle gesetzten Ziele in der atomaren Aktivitätsphase verworfen. Eine erneute Durchführung muss demnach wieder angestoßen und im besten Fall kann auf diese Weise mit einer Verzögerung und Projektplananpassung fortgefahren werden.

3.2.2 Informationelle und physische Objekte

Informationelle und physische Objekte können als **Input** oder **Output einer Aktivität** definiert werden. Dabei ist auch auf deren **Wiederverwendbarkeit** einzugehen. Von den fünf Projekt-Objekten gehören Ziele, Ressourcen und die Zeit zu den informationellen (Daten) und physischen (Dinge) Objekten. Diese werden um alle Arten von Ressourcen ergänzt. Bei der Art von Objekten hat die Wiederverwendbarkeit einen Einfluss auf die Abbildung und Verwendung innerhalb der Aktivitäten. Bei der **Zeit** handelt es sich rein

um Daten. Hier besteht ein besonderer Anspruch bei der quantitativen Angabe, welche im nächsten Unterabschnitt thematisiert wird.

Der **Output** einer Aktivität kann einer anderen Aktivität als **Input** dienen. Ein Ziel wird immer als Output verstanden und kann Daten oder Dinge umfassen. Eine Ressource hingegen wird immer als Input verstanden und ist nach der Benutzung auf ihre Wiederverwendbarkeit zu prüfen. Genauso kann es zu einer Modifizierung der **Ressourceneigenschaften** kommen, welche sich durch das Lernen als Aneignung von Wissen durch den Einsatz der Ressource ergibt. Auch hier handelt es sich um Daten und Dinge.

Bei der **Ressourcenzuordnung** wird eine generelle Unterteilung in Input oder Output mit etwaigen Modifikationen vernachlässigt. Der Detailierungsgrad wäre zu groß und wird deswegen hier nicht weiter betrachtet. Dies kann Gegenstand weiterer Untersuchungen darstellen und muss auch bei der Datenerfassung und Tabellendefinition mit Berücksichtigung finden. Unterdessen wird die Annahme getroffen, dass eine erfolgreiche Vorphase die erforderlichen Inputs in Form von Ressourcen als Ergebnis der Phase liefert, beispielsweise ist das Produkt eines erfolgreichen Designs ein Fachkonzept.

Im Weiteren wird unter einer Ressource eine allgemeingültige Definition verstanden. Unter einer **Ressource** ist ein Bestand von etwas zu verstehen, was für einen bestimmten Zweck eingesetzt werden soll. Deswegen wird im Rahmen dieser Arbeit die Ressource als Oberbegriff für mögliche Inputs bzw. weitergereichte Outputs angesehen. Die Einteilung der Ressourcen ist nach deren Zweckmäßigkeit vorzunehmen. Die Gründe können vielfältiger Natur sein, wie beispielsweise Quelle der Ressource, Einsatzbedingung, Qualitätsanforderungen, physische Objekte, zur Datenhaltung vorgesehen und vielerlei mehr. Die Ressourcen werden im Rahmen dieser Arbeit in zwei Arten unterteilt: **Daten und Dinge**. Daten stellen jegliche Art von Informationen dar (z. B. Fachkonzepte, Dokumente, etc.). Somit stellt das Informationsobjekt auch eine Ressource dar, und die Vergegenständlichung ist grundsätzlich durch die Abstraktionsebene (schriftlich, mündlich, Wissen) gegeben. Dinge hingegen umfassen physische Objekte aller Arten (z. B. Mitarbeiter, Hardware, etc.). [vgl. Reisig und Desel 2014, 174; vgl. Schott 2019, 6 ff; vgl. Thein 2017, 34 f] Eine Besonderheit bei der **menschlichen Ressource** ist, dass auch das Eintreffen von beispielsweise folgenden

Gegebenheiten berücksichtigt werden muss: Urlaub, Ausfall, Halbwissen oder Leistungsabfall. Ebenfalls muss die Einschränkung machbar sein, dass ein Mitarbeiter, als Vertreter der menschlichen Ressource, nur während der Arbeitszeit zur Bearbeitung der Aufgabe zur Verfügung steht. Daraus ergibt sich die Schlussfolgerung, dass es **keine Aktivität ohne Ressource** gibt.

Wiederverwendbarkeit

Die Eigenschaft der Wiederverwendbarkeit bezieht sich nur auf das Projekt-Objekt **Ressourcen**. Dazu wird die Wiederverwendbarkeit der Ressourcen in zwei Gruppen eingeteilt, die wiederverwendbaren oder verbrauchbaren Ressourcen.

Wenn eine Ressource **verbraucht** und nach ihrem Einsatz nicht mehr zur Verfügung steht, handelt es sich um nicht erneuerbare Ressourcen. Dazu zählen beispielsweise die zu verarbeitende Materialien oder Rohstoffe. Zur Einplanung dieser muss die zur Verfügung stehende Anzahl mit angegeben bzw. berücksichtigt werden. Im Gegensatz dazu stehen die **erneuerbaren**, auch **wiederverwendbaren** Ressourcen. Dazu zählen nicht nur Arbeitsmaterialien, sondern auch die menschliche Arbeitskraft. Diese Art von Ressourcen steht nicht zur Verfügung, wenn sie mit der Abarbeitung einer Aktivität beschäftigt sind. Nach der Fertigstellung können sie unter Einbeziehung von Regenerationszeiten (Urlaub, Krankheit, Wartung) wiederverwendet werden. Die Regenerationszeit kann auch Null betragen, wie beispielsweise bei Arbeitsmaterialien oder benötigten Konzepten. Auch kann bei der Rückführung der Ressource eine Modifikation oder Anreicherung erfolgen. [vgl. Reisig und Desel 2014, 174; vgl. Schott 2019, 6 ff; vgl. Thein 2017, 34 f]

Ebenfalls ist die Information (Daten) als wiederverwendbar oder verbrauchbar einzuordnen. Dabei wird in dem Kontext angenommen, dass sie entweder modifiziert und/oder wiederverwendet wird oder unbegrenzt zur Verfügung steht. Damit wäre eine Information nie verbrauchbar. Eine Information ist immer nur mit der Anwendung einer weiteren Ressource sinnvoll – beispielsweise das Wissen des Mitarbeiters oder die Niederschrift des Wissens in ein Konzept – und sie unterliegt ebenfalls der Einteilung der in Verbindung stehenden Dingen.

Ressourceneinheit

Einheiten geben den Satz der Zeit oder Stück einer Ressource an, welcher einer Aufgabe zugeordnet ist. Im Projekt gibt es zwei Arten von Einheiten: Ressourcen vom Typ **Mengeneinheit** (Stück) und vom Typ **Zeiteinheit** (ein Vielfaches der zugrunde gelegten Zeit).

Wenn die Angabe der Einheit in Stück erfolgt, muss diese Ressourcenart der Aufgabe gleich komplett zur Verfügung stehen. Die Dauer der Abarbeitung einer Aufgabe wird auch durch ein Vielfaches des zugrunde gelegten Zeitkonzeptes angegeben. Dabei wird im Netz pro Takt eine Einheit dessen verbraucht. Ebenfalls wurde festgelegt, dass die Abarbeitung der angegebenen Dauer nicht am Stück, sondern zwischen einer Zeitspanne erfolgen kann. Das muss auch auf die Ressourcen, welche ebenfalls als Vielfaches der Zeit zur Verfügung stehen (dies betrifft hauptsächlich die Ressource des Mitarbeiters) bei der Modellierung mitberücksichtigt werden. Es lässt die Schlussfolgerung zu, dass die Zuordnung dieser Ressourceneinheit sich ebenfalls pro Zeittakt ereignet. Dies beruht darauf, dass keine Ressourcenreservierung erfolgen kann, wenn nicht auch ein Abarbeitungsschritt passiert. Eine Aktivitätsphase muss somit nicht am Stück abgearbeitet werden und eine atomare Ressourcenzuordnung pro Arbeitsschritt (Typ Zeiteinheit) und pro Aktivitätsphase (Typ Stück) wird benötigt.

3.2.3 Objekte mit quantitativen Angaben

Es gibt verschiedene quantitative Angaben, welche je nach Kategorie als natürliche oder reelle Zahlen zur Aufnahme in der Tabelle verwendet werden können. Es wird generell eine Einteilung in sechs Gruppen vorgenommen: Datum, Dauer, eindeutiger Schlüssel in Form einer Identifikationsnummer (ID), Mengenangaben, Bedingungen und der Erfüllungsgrad.

Beim Datum muss die Angabe in einem einheitlichen Datumsformat geschehen. Die quantitativen Angaben bei der Dauer müssen dem Vielfachen der zugrunde gelegten Zeit, aus dem später vorgestellten Zeitkonzept, entsprechen. Die ID unterliegt einem Schema, welches es erlaubt, beim Einsatz von strukturierten Marken auf Informationen zuzugreifen. Näheres dazu im Kapitel 4.

Die Mengenangaben sind in Form einer natürlichen Zahl vorzunehmen. Beispielsweise wird bei der Ressourcenzuordnung die Zuordnung einer halben Ressource über die Angabe der Dauer erfolgen. Im Gegensatz dazu können die Angaben bei Bedingungen dem Wertebereich der reellen Zahlen entsprechen. Beim späteren Beispiel im 5. Kapitel wird ein Erfüllungsgrad eingeführt, der hier Anwendung findet. Dieser wird auf fünf Zustände (0, 25, 50, 75, 100) reduziert und somit als diskret angesehen.

Zeitverbrauch und Zeitverhalten

Eine Aktivität wird durch eine Zeitvorgabe definiert und eingeplant. Diese Zeitvorgabe kann als Zeitspanne (Dauer), einzelne Zeitpunkte oder fixe Zeitpunkte angegeben werden. Was sich darin äußert ob die Dauer allein, die Dauer mit fixen Zeitpunkten (fixer Start, fixes Ende, fixer Start und Ende) oder die Dauer mit variablen Zeitpunkten (frühester Start und spätestes Ende) angegeben wird. Dies unterteilt die Angaben in eine dauerorientierte oder terminorientierte Zeitvorgabe. Bei einer dauerorientierten Zeitvorgabe muss immer die Angabe einer Dauer mitgegeben werden. Wohingegen bei einer terminorientierten Zeitvorgabe dies nicht notwendig ist, beispielsweise bei einem fixen Start- und Endtermin. Im Rahmen dieser Arbeit wird von einer dauerorientierten Zeitvorgabe mit variablen Zeitpunkten ausgegangen. Dies geschieht auf der Grundlage der Flexibilität des hier vorgestellten Projektmanagementansatzes. Der agile Anspruch verlangt ebenfalls eine gewisse zeitliche Flexibilität im Rahmen eines Sprints (selbstorganisierte Entwicklerteams). Die dauerorientierte Zeitvorgabe in Kombination mit den Angaben dem frühesten Start und Ende lässt unterschiedliche Restriktionen bei der Setzung der Zeitangaben zu. [vgl. Königsberger und Schloß 2014]

Dabei werden zwei Typen von Zeitvorgaben unterschieden. Einerseits die Angabe von Zeitpunkten mittels eines Datums (Typ 1), welche einen Zeitraum definiert. Andererseits die Angabe einer Zeitdauer (Typ 2). Die Datumsangabe beruht dabei auf ein frühestes Startdatum und ein spätestes Enddatum. Das Startdatum gibt bekannt, wann mit der Bearbeitung der Aktivität begonnen werden kann, schließt einen späteren Start jedoch nicht aus. Das späteste Enddatum legt hingegen eine Frist fest, zu der die Aktivität spätestens abgearbeitet sein muss. Mit dieser Methode wird die Bearbeitung der Aktivität innerhalb dieser zwei Datumsangaben gesetzt. Die Dauer der eigentlichen Bearbeitung der Aktivität wird ebenfalls mit angegeben. Durch die Aneinanderreihung der einzelnen

Zeitvorgaben wird der zeitliche Ablauf und die Gesamtzeit des Projektes definiert und bildet eine wichtige Komponente in der Projektplanung. Dies verdeutlicht die Bedeutung eines Zeitkonzeptes. [vgl. Schott 2019, 22]

3.2.4 Bedingungen, Vorgänger und Nachfolger

Ein diskretes Modell ermöglicht es komplexe Systeme zu beschreiben, dessen Elemente in logischen Beziehungen zueinander stehen. Hierbei spielt die Theorie der Vorgänger und Nachfolger eine wichtige Rolle. Auch im Projektmanagement ist die Bestimmung von Vorgänger und Nachfolger essentiell. [vgl. Kiencke 2009, 25 f] Bei dem gewählten gemischten Ansatz der klassischen und agilen PM-Methoden ist dieser Aspekt von Bedeutung und ist genauer zu betrachten. Mit einer allgemeinen Festlegung benötigter Eigenschaften soll sichergestellt werden, dass angefangen beim Projektmanager bis zur Generierung des PN, dieser Ansatz bewahrt und korrekt eingeplant wird. Der Zusammenhang und die daraus resultierenden Vorgänger-/Nachfolgerbeziehungen sind in der Abbildung 7 dargestellt.

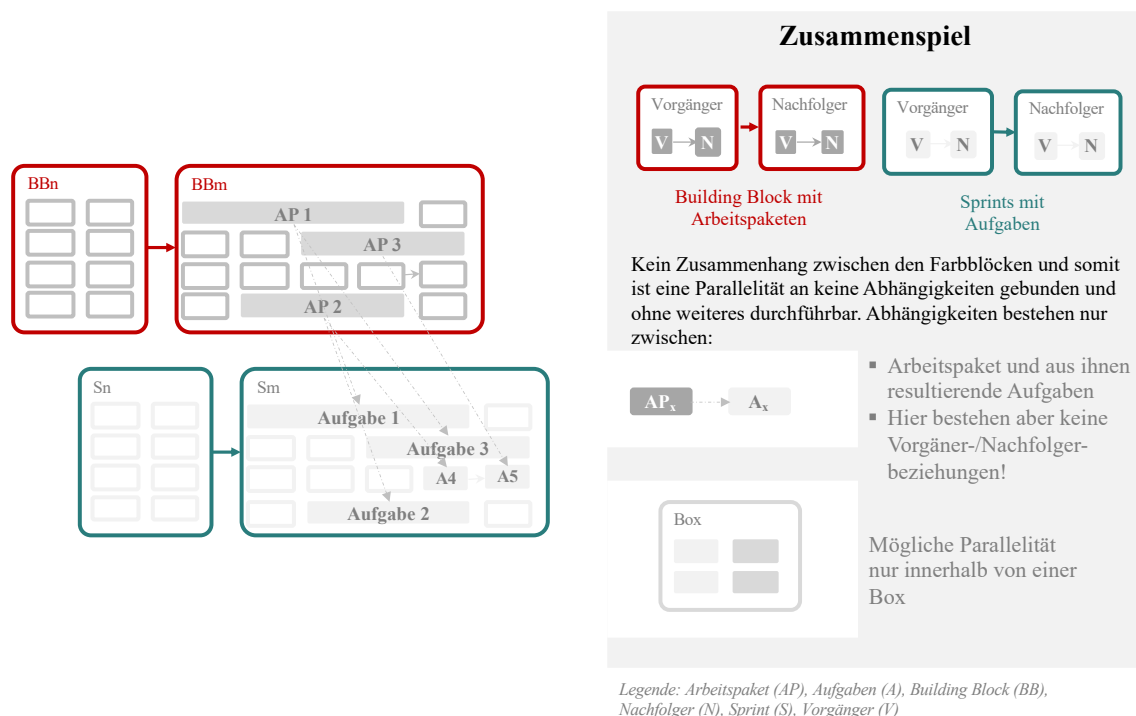


Abbildung 7: Zusammenspiel Vorgänger/Nachfolger ohne zeitliche Interpretation

Wie in Abbildung 7 aufgezeigt, gibt es zwei wesentliche Rahmenkonzepte – auch als Box bezeichnet – die Building Blocks und die Sprints. Zwischen ihnen besteht grundsätzlich kein Zusammenhang und somit wird eine unabhängige Parallelität und Durchführung möglich. Zwar entstehen aus den Arbeitspaketen der Building Blocks die Aufgaben, welche aber vorher nicht bekannt sind. Ebenfalls ist die Sprintzusammensetzung dynamisch, da diese von den Erkenntnissen aus den Arbeitspaketen abhängig ist. Ein Sprint mit seinem Aufgabenumfang kann daher immer erst kurzfristig geplant werden. Natürlich muss es eine grobe Planung geben, welche bereits bei der Reihenfolge der Abarbeitung der Arbeitspakete eingeflossen ist und somit bei der Sprintzusammensetzung wieder Berücksichtigung findet oder sich im Rahmen von Vorgänger-/Nachfolgerbeziehungen äußert. Zusammenfassend kann festgehalten werden, dass eine mögliche Parallelität von Elementen, die in Abhängigkeit zu einander stehen, nur innerhalb einer Box besteht.

Dabei wurden die klassischen und agilen Aspekte auf eine allgemeine Ebene abstrahiert und mit flexiblen Konzepten und deren Kommunikation untersucht. Der Zusammenhang kann durch folgende Aussagen näher beschrieben werden: **(1)** Die Zuordnung von Ressourcen und die eigentliche Abarbeitung von Aktivitäten findet nur auf der Ebene der Arbeitspakete und Aufgaben statt. Die Building Blocks und die Sprints sind zur Rahmenbildung da und stellen sicher, dass die hier erwähnten Anforderungen und Vorgänger/Nachfolger eingehalten werden. **(2)** Die Abarbeitung der Building Blocks und Sprints haben keine Abhängigkeiten zueinander und erfolgen parallel. **(3)** Während der Abarbeitung der einzelnen Arbeitspakete werden Aufgaben definiert. Diese werden über das Instrument der Backlog den Sprints zugeordnet. **(4)** Die Abarbeitung der Arbeitspakete und Aufgaben gliedert sich in die Ressourcenzuordnung sowie der Angabe der zeitlichen Dauer und die eigentliche Bearbeitung. **(5)** Sprints und Building Blocks sind immer nur sequentiell und nie parallel. Somit besteht zwischen den Sprints eine vereinfachte Vorgänger-Nachfolger-Beziehung. Vereinfacht aus dem Grund, weil ein Sprint immer nur jeweils einen Vorgänger und/oder Nachfolger haben kann. **(6)** Eine vereinfachte Vorgänger-Nachfolger-Beziehung ist auch innerhalb der Aufgaben zu finden. Hier wurde definiert, dass sich eine Aufgabe in einen Design- und einen Entwicklungsabschnitt gliedert. In der Designphase werden die fachlichen Anforderungen in ein technisches Konzept umgewandelt. Somit werden während der

Designphase die Konzepte zur späteren Umsetzung der Aufgabe entworfen, welche in der Entwicklungsphase umgesetzt werden. Der grundsätzliche Ablauf dieser zwei Phasen ist identisch. In den folgenden Betrachtungen wird deswegen auf diese Unterteilung nicht detailliert eingegangen und alle Anforderungen und Eigenschaften gelten für beide Phasen. Die Phase der Entwicklung folgt dabei immer auf die Phase des Designs. Aus diesem Grund besteht zwischen den beiden Phasen ebenfalls eine vereinfachte Vorgänger-Nachfolger-Beziehung. **(7)** Building Blocks und Sprint fungieren als Box für die Gruppierung von Arbeitspaketen bzw. Aufgaben. **(8)** Nur in den Boxen ist eine parallele Abarbeitung möglich. Ebenfalls können hier Vorgänger-Nachfolger-Beziehungen bestehen. Diese zwei Merkmale können zutreffen, müssen aber nicht. [vgl. Schott 2019, 22 ff]

3.3 Objektbeziehungen

In diesem Unterkapitel wird auf das Zusammenspiel mit atomaren Aktivitätsphasen und den weiteren Projekt-Objekten (Ziele, Ressourcen, Zeit und Beziehungen) eingegangen. Die nachfolgenden Ausführungen basieren auf der vom Autor betreuten und weiterentwickelten Masterarbeit von [Thein 2017].

Die zueinander in Beziehung stehenden Projekt-Objekte erfordern eine Modellierung auf gleicher Ebene. Das Zusammenspiel der Projekt-Objekte findet demzufolge auf der atomaren Ebene, dem kleinstmöglichen Zustandselement, statt. Beispielsweise kann die Durchführung einer einzelnen atomaren Aktivitätsphase nicht allein das Ziel des gesamten Projektes realisieren. Erst die erfolgreiche Durchführung bzw. Abfolge vieler Aktivitäten führt zur Verwirklichung des gesteckten Projektzieles. Die atomare Aktivitätsphase bildet die Basis und definiert somit das kleinstmögliche Zustandselement. Die anderen Projekt-Objekte werden durch den jeweiligen Gebrauch mit der Aktivität bestimmt, mit Ausnahme der Ressourcenzuordnung. [vgl. Reisig und Desel 2014, 174]

Ziele und **Ressourcen** stehen im direkten Zusammenhang mit Aktivitäten. Das Eintreffen eines Zustandes, was mit der Erreichung eines Zieles gleichzusetzen ist, ist an die Durchführung einer Aktivität gekoppelt und bildet ebenfalls den Ausgangspunkt von neuen Aktivitäten. Die Durchführung einer Aktivität verbraucht Ressourcen, um ein Ziel

zu erreichen. Einer Aktivität werden in der Ressourcenplanung, als Teil der Projektplanung, die benötigten Ressourcen zugewiesen. Eine korrekte Ressourcenzuordnung und deren Überwachung sind von großer Bedeutung für die erfolgreiche Abarbeitung der Aktivitäten, damit die angegebenen Zeitanforderungen eingehalten werden können. [vgl. Reisig und Desel 2014, 174; vgl. Schott 2019, 6 ff]

Die **Zeit** ist eine wesentliche Komponente bei der Abbildung von Projekten, deren Funktionsweise sowie deren Ablauf. Alle Projekt-Objekte stehen mit der Zeit in einer Verbindung (Durchführungsdauer von Vorgängen, Eintrittszeitpunkte von Zielen, Verwendungsdauer von Ressourcen, Puffer und der kritische Pfad).

Die **Beziehungen** beschreiben die Zusammenhänge zwischen den Projekt-Objekten. Es kann festgehalten werden, dass Projekt-Objekte auf atomarer Ebene teilbar sind und somit die wichtigen Modellierungseigenschaften diskreter Systeme erfüllen. Durch die Zusammenhänge zwischen den Projekt-Objekten ist eine Modellierung auf der gleichen atomaren Ebene sichergestellt. Auch der Eingriff in den Projektablauf kann durch einen manuellen oder automatischen Eingriff erfolgen, was eine Anpassung der Abfolge der atomaren Aktivitätsphasen und deren Verbrauch/Verbindungen mit sich zieht. [vgl. Reisig und Desel 2014, 174]

In Kapitel 5 wird das zugrundeliegende Projektbeispiel mit all den für die Modellierung relevanten Informationen vorgestellt. Dafür werden die Projekt-Objekte nach den hier dargestellten Regeln definiert.

3.4 Modellierung

Die zugrundeliegende Methode der Modellierung muss einfach und verständlich sein. Die grafische Darstellungsform sollte sich an dem zu modellierenden System orientieren. Ebenfalls sollte die Eingabe der Daten und dessen Auswertung formalisiert und einfach erfolgen. [vgl. Zimmermann 1997, 10 ff]

Die Darstellung der Projektplanung soll weiterhin in Tabellenform geschehen. Dies gewährleistet eine einfache und PN-unabhängige Grundlage und der Projektmanager kann auch ohne PN-Kenntnisse und mit einem beliebigen Tabellenkalkulationstool alle erforderlichen Angaben aufnehmen. Dies soll aber nicht willkürlich stattfinden, sondern

es soll eine formale Methode für die Tabellenstruktur festgelegt werden. Dabei wird die Tabellengröße in ihrer Breite reduziert und der Fokus liegt auf der reinen Dateneingabe. Mit einer formalen Tabellenstruktur kann anschließend die Überwachung und Abbildung des Projektes durch ein PN umgesetzt werden. Auf die Anforderungen an die formalisierte Tabelle wird im Kapitel 3.4.1 näher eingegangen. [vgl. Schott 2019, 20 f]

Die Modellierungsmethode darf nicht auf spezielle Projektstrukturen eingeschränkt werden und muss somit erweiterbar und weitestgehend unabhängig sein. Dies wird dadurch gewährleistet, dass sowohl klassische als auch agile Projektstrukturen berücksichtigt wurden. [vgl. Zimmermann 1997, 10 ff]

Zusätzlich zu den Anforderungen aus dem PM bringt der agile Ansatz – hier der Spezialfall eines gemixten Ansatzes dieser zwei PM-Methoden – ebenfalls Anforderungen an das formalisierte Modell mit. Bei einem agilen PM steht vor allem die Flexibilität im Vordergrund. Dies bedeutet, dass auch die Erfassung der Informationen einfach und flexibel gestaltet sein muss. Beispielsweise kann sich eine Sprintzusammensetzung vor dem Sprintbeginn noch ändern, was eine Änderung der abzuarbeitenden Reihenfolge bedeutet. Die Abbildung von Änderungen soll effizient und einfach und nicht durch hinterlegte Formeln oder Logiken in der Tabelle erfolgen. Somit muss das Modell dies berücksichtigen. Durch die kurzen Zyklen müssen auch die Simulation und mit ihr die Ergebnisse der Simulation einfach zurückzuführen sein. Ebenfalls muss es machbar sein, mit den Prognosewerten zu jeder Zeit zu spielen. Gerade bei einem agilen Ansatz ist die Ausbalancierung der Ressourcenanforderung entscheidend. Deswegen wird dieses Themengebiet auch bei der späteren Umsetzung in den Fokus gerückt. Ebenfalls müssen dafür Vorgänger und Nachfolger von Aufgaben – wenn erforderlich und Abhängigkeiten bestehen – festgehalten und im Modell mit geprüft werden. Weiterhin muss die Aufnahme aller wichtigen Informationen durch die Tabellenstruktur sichergestellt sein. Dies bedeutet, dass die wichtigsten Informationen aus der Backlogliste zu integrieren sind oder gar die Struktur, um alle Informationen zu erweitern sind. Wobei später für das Modell nur die entscheidenden Informationen Beachtung finden sollen. Aus diesem Grund sollen keine doppelten Datensätze in den Tabellen aufzufinden sein. Somit bedarf es keiner einzuhaltenden Synchronisation. Dabei handelt es sich beispielsweise um die Aufnahme von allen Aufgaben – egal ob zu implementieren oder nicht. Die Tabelle muss ebenfalls die Information verarbeiten, ob

eine Aufgabe schon zur Implementierung freigegeben worden ist und wenn ja, in welchen Sprint. Ebenfalls müssen die Datumsangaben vom frühesten Startdatum, spätesten Enddatum und eine Fortschrittseinschätzung angegeben werden können. Bei abzusehenden Abweichungen muss eine Anpassung der Zahlen erfolgen können.

3.4.1 Tabellenstruktur

Eine große Herausforderung bei der Modellerstellung liegt in der Auswahl und Gewinnung der benötigten Daten. [vgl. Zimmermann 1997, 10 ff]

Aus Erfahrung sind die geführten Tabellen mit Formeln überfüllt und mit Komplexität übersät. Auch ist dessen Zusammenhang oft nicht mehr einfach erkennbar. Dies führt dazu, dass eine beispielsweise geänderte Sprintzusammensetzung in mehreren Tabellen zu pflegen ist und oftmals auch strukturelle Änderungen einzelner Tabellen mit sich führt. [vgl. Weichenhain, Fengler, und Streitferdt 2019; vgl. Weichenhain und Fengler 2019a]

Aus diesem Grund muss eine Formalisierung der Tabellenstruktur stattfinden. Die Formalisierung bezieht sich auf den Mechanismus zum Befüllen der Tabellen und die Dateneingabe. Grundsätzlich hat dies zur Auswirkung, dass keine Übertragung von nicht enthaltenen Informationen aus der Tabelle in ein PN erfolgen kann. Dies könnte als Nachteil interpretiert werden. Jedoch überwiegen durch den zur Anwendung kommenden Formalismus die Vorteile. Dazu zählt:

- Jede Informationsaufnahme beginnt in der Tabelle und ist durch die Tabellenstruktur auf einen Tabellenbereich begrenzt. Dafür wird von den Vorteilen einer normalisierten Tabellenstruktur profitiert. Unter einer Normalisierung einer Tabellenstruktur wird die Aufteilung von Informationen in mehrere Tabellen vorgenommen, so dass keine vermeidbaren Redundanzen mehr enthalten sind.
- Durchzuführende Änderungen müssen nur in einer Tabelle gepflegt werden. Die Überführung in ein PN wird dafür Sorge tragen, dass die Auswirkungen dieser Änderungen (beispielsweise die Änderung der Eignungswerte) immer an bestimmten Elementen im PN gebündelt und angewendet werden.

- Auch die Befüllung der Tabellen unterliegt dem Wandel des Projektes. Anfangs werden alle Plandaten festgelegt und in die Tabelle eingetragen, um sie im weiteren Verlauf mit Ist-Daten anzureichern. Dies kann aber auch Änderungen auf zugrundeliegende Annahmen bei der Erstellung der Plandaten mit sich führen. Eine sofortige Simulation von Änderungen geht damit einher, wie auch das Aufzeigen möglicher Gefahren zur Projektbeendigung.
- Die benötigten Formeln zur Simulierung, Berechnung und Überprüfung werden durch den Einsatz eines PN-Modells abgenommen und aus der Tabelle komplett verbannt.

Zusammenfassend ist die Aufnahme folgender Angaben in der Tabellenstruktur zu berücksichtigen. Dabei ist die Aufzählung nicht vollumfänglich, sondern beschränkt sich auf den Focus des später angewandten Beispiels.

- Aufnahme der Ressourcen mit den Informationen über Art, Verfügbarkeit, Angaben zu Regenerationen (Urlaub, Ausfälle, Teilzeitarbeit, etc.)
- Planung der Building Blocks mit ihren Arbeitspaketen und den Angaben: Dauer, Ressourcenzuordnung, Anfangs- und Endzeitpunkte, Fortschritte, Zuordnung der entstandenen Aufgaben zu den Arbeitspaketen
- Planung der Sprints mit der Zuordnung von Anfang, Ende, Prioritäten, Aufgaben, Design, Development und Test
- Aufnahme der Aufgaben in eine Backlog mit Prioritäten, Fortschritte der einzelnen Phasen, Aufwandsschätzungen, Design- und Implementierungszeiten, Ressourcenzuordnungen, Sprintzuordnungen, Vorgängern
- Allgemeine Aufnahme von Plandaten und Ist-Daten, sowie Prognosedaten

3.4.2 Darstellungsform

Eine Grundanforderung bei der Formalisierung ist, dass keine PN-Kenntnisse benötigt werden. Deswegen ist das Zurückschreiben von relevanten Informationen in die Tabelle als Vorteil herauszustellen. Von dem Grundgedanken wird nicht abgewichen. Eine Simulation kann auch in Einzelschritten optisch Erkenntnisse bringen und deswegen sollen marginale Grundanforderungen an die Darstellungsform Beachtung finden. Aufgrund der Komplexität dieser Thematik wird die Darstellungsform aus der vorliegenden Arbeit ausgegrenzt.

Unabhängig davon wird für die beispielhafte Modellierung im 6. Kapitel eine ganz triviale Darstellungsform gefordert. Die Anordnung der Konstrukte soll auf bestimmte Bereiche festgelegt werden, damit sie später leichter zu identifizieren sind. Ebenfalls ist eine zusammenhängende, nicht verteilte Anordnung aller Elemente eines Konstruktes gewünscht. Diese soll für den geeigneten Leser zur besseren Lesbarkeit später Anwendung finden. Dabei wird sich auf den später benutzten Umfang begrenzt.

3.4.3 Anwendung im Projektkreislauf

Das geforderte Projektcontrolling ist ein iterativer Prozess zur kontinuierlichen Verbesserung, welcher die agile und dynamische Vorgehensweise unterstützt. Dabei handelt es sich um den angestrebten Kreislauf von Plan Do Check Act (PDCA) im Projektcontrolling, wie in Abbildung 8 ersichtlich. Die Vorgehensweise basiert auf dem PDCA-Zyklus nach Deming und umfasst die vier Phasen: Planen (P), Ausführen (Do), Überprüfen (Check) und Verbessern (Act). Die Phasen werden schrittweise durchlaufen und im Gesamtkontext als nie endender Prozess verstanden. [vgl. Zollondz 2012, 85 f] Der folgende Kreislauf zeigt die verschiedenen Aufgabenbereiche des Controllings:

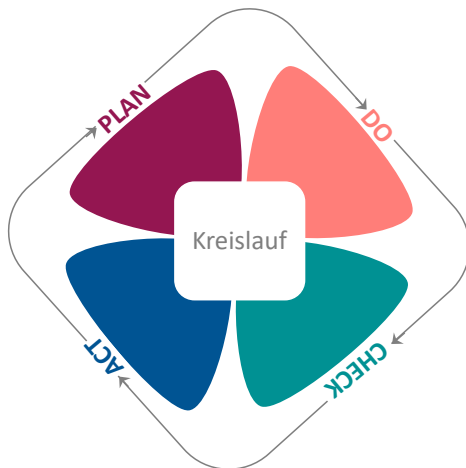


Abbildung 8: PDCA-Zyklus [vgl. Weichenhain, Fengler, und Streitferdt 2019]

Dieser Kreislauf soll beim angestrebten PPNM Anwendung finden. Dafür ist es wichtig, dass die benötigten Informationen zur Erstellung des PPNM später dem Projektmanager erneut greifbar gemacht werden müssen. Dies soll, wie in den vorherigen Kapiteln definiert, ohne PN-Kenntnisse möglich sein.

Eine Zusammenführung kann nur mittels einem gemeinsamen Detaillierungs- und Abstraktionsgrad erfolgen. Den Anfang bereitet die Identifizierung einer passenden Abbildung des PMs durch PN und damit einhergehend die Definition einer geeigneten gemeinsamen Abstraktionsebene. Den Modellierungsgegenstand bilden die Projekt-Objekte mit deren Operationen und Beziehungen zueinander. Zur Gewährleistung einer formalen und einheitlichen Darstellung ist es zudem notwendig, die Eigenschaften von Objekten zu klassifizieren und deren Anforderungen abzuleiten. Auf dieser Basis kann eine formale Tabellenstruktur und somit eine Methode zur Überführung und Adaption des PMs definiert werden. Aufgrund einer zugrundeliegenden dynamischen Betrachtung muss der Darstellung des Kreislaufs dieser Formalisierung ebenfalls Beachtung zukommen. Den Ausgangspunkt des Modells bilden die Angaben in den Tabellen. Prinzipiell ist zwischen den zwei Arten von Angaben, Plan und Ist, zu unterscheiden. Bezugnehmend auf dieser Grundlage soll das Modell generiert und die erforderlichen Analysen und Erkenntnisse gewonnen werden. Damit der Kreislauf von der Dateneingabe, über die Modellierung mittels PN und dessen Simulations- bzw. Prognosemöglichkeiten geschlossen wird, muss die Rückführung der Ergebnisse in die Tabelle gesichert sein. [vgl. Weichenhain, Fengler, und Streitferdt 2019]

Unabhängig davon stehen dem Projekt ebenfalls mächtige Kennzahlen und Analysen aus den zwei Welten, der des PMs und der PN, zur Verfügung.

3.4.4 Prospektives Projektcontrolling

Das prospektive Projektcontrolling ist ein Fachbegriff für das rechtzeitige Erkennen von Problemen in der Projektzielerreichung. Dabei dient es während der Realisierung einer aktiven Kontrolle von Kosten, Terminen und Fortschritten. Dies geschieht durch den Einsatz von Projektkennzahlen. Die vier Hauptaufgaben während des Projektcontrollings sind dabei die Budget- und Aufwandsschätzung, Ressourcenplanung, die planorientierte Steuerung des Projektes und die Nachkalkulation. [vgl. Aichele 2006, 168 ff] Es gibt eine große Menge an Kennzahlensystemen zur Steuerung, Analyse und Bewertung von Projekten. Dabei verfolgen sie das gleiche Ziel, die wichtigen und wesentlichen Information zu filtern und zu aggregieren, ohne dabei jedoch die Relationen und Zusammenhänge zu vernachlässigen. Eine schnelle und bessere Informationsgewinnung, kombiniert mit optimalen Führungsinstrumenten, kann auf einer fundierten und breiteren Basis eine komplexe Analyse für gezielte Disponierung und verbesserte Aktionsfähigkeit aufbauen. [vgl. Aichele 2006, 172 ff]

Im Rahmen dieser Arbeit wird beispielhaft auf die Kennzahlen oder Beurteilungskriterien zur **Eignung**, **Durchführung** und **Entscheidungsfindung** eingegangen. Während der **Durchführung** eines Projektes kann es jederzeit zu Abweichungen von den zuvor festgelegten Planungsdaten kommen. Mit dem Fortschritt des Projektes können die **Planungsdaten** durch die Angabe von **Ist-Daten** ersetzt werden. Mit diesen Informationen können im Rahmen der Eignung und Entscheidungsfindung weitere Prognosen resultieren. Bei der Eigenschaft der Durchführung wird sich auf den **Plan-Ist-Vergleich** beschränkt. [vgl. Schott 2019, 63]

Zur Gewährleistung eines Plan-Ist-Vergleiches muss eine Berücksichtigung des zu erstellenden Modells des Projektes mit Plandaten sowie mit aktuellen Ist-Daten erfolgen. Die notwendige Abgrenzung von Plandaten und Ist-Daten müssen im Tabellenformat Berücksichtigung finden. Für die Planung wurden im Kapitel 3.2.2 Zeitangaben identifiziert: Frühester Startzeitpunkt, spätester Endzeitpunkt und die Dauer der Tätigkeit. Diese drei Zeitangaben müssen für die Erfassung der Ist-Daten ebenfalls in die Tabelle

aufgenommen werden. [vgl. Schott 2019, 63] Es wird zwischen vier Modellen unterschieden: Das reine **Plan-Modell**, das **Plan-Ist-Modell**, das reine **Ist-Modell** und das **Prognose-Modell**. Das Thema Prognose wird im weiteren Verlauf in den Eigenschaften Eignung und Entscheidungsfindung thematisiert.

Das reine **Plan-Modell** wird anhand der gegebenen Plandaten erzeugt und ist meist vor dem Start des Projektes angedacht. Es dient vordergründig zur Überprüfung der getätigten Planung. Im Gegensatz dazu dient das reine Ist-Modell zur Analyse, nachdem ein Projekt beendet wurde. Hier liegt der Fokus auf das nachträgliche Simulieren der Vergangenheit bis zum jetzigen Zustand zur Historienaufbereitung. Ein **Plan-Ist-Modell** kombiniert die Angaben der jeweiligen Kategorien. Dies findet während des Projektes statt, um veränderte Parameter während des Projektverlaufes zu verdeutlichen. Somit können Auswirkungen anhand veränderter Angaben erkannt werden. Dies ist beispielsweise der Fall, wenn eine Tätigkeit länger dauert als geplant. Die neue Dauer wird in das neue Plan-Ist-Modell aufgenommen und die Auswirkung kann auf das gesamte Projekt simuliert und analysiert werden, beispielsweise durch die Entstehung einer Verschiebung. Dabei kommt es zu keinen neuen Annahmen oder Analysen der Ist-Daten. Es wird lediglich veranschaulicht, ob die Planwerte nicht mit den schon gegebenen Istwerten korrelieren.

Die **Eignung** und die **Entscheidungsfindung** beruhen auf ähnlichen Grundgedanken. Aufgrund von Erfahrungen oder Annahmen wird ein Ausgang vorhergesagt oder simuliert. Dies wird im Themenfeld des Prognose-Modells näher betrachtet.

Bei der Frage der **Eignung** wird näher untersucht, inwieweit die bestehenden Annahmen der Wirklichkeit gerecht werden. Dabei sind Erwartungen über die künftige Entwicklung des Projektes ebenso zu berücksichtigen, wie auch das Wissen um bestehende Kompetenzen, Stärken oder anderer möglicher Faktoren. Je komplexer und umfangreicher ein Modell ist, umso schwieriger wird die Überprüfung und Bewertung. Ein erster Ansatz stellt eine interaktive Simulation dar, welche aber als Ausnahme zu den vorher aufgestellten Bedingungen PN-Kenntnisse benötigt. [vgl. Zimmermann 1997, 119] Die Grundidee des Prognose-Modells beruht auf zwei Ansätzen. Auf der Analyse der schon vorhandenen Ist-Daten und auf deren anschließender Ableitung von Mustern. Aus diesen Mustern oder erkannten Regeln können Erwartungen für die Plandaten abgeleitet und für das Prognose-Modell berücksichtigt werden. [vgl. Benecke 1995, 6 ff]

Die Grundlage zur Bildung dieser Beurteilungskriterien beruht auf Annahmen, welche beispielsweise aus den Erfahrungen vom Projektleiter abzuleiten sind. Aufbauend auf festgelegten Eignungskriterien und Regeln kann ermittelt werden, ob eine Änderung in der Ressourcenzuordnung zu signifikanten Vorteilen im Projektverlauf führen könnte. Die Grundidee beruht darauf, dass mit dem aufgestellten Projektplan eine Untersuchung für die schon getroffenen Änderungen ohne mühsame Anpassung der angegebenen Dauer resultiert.

Mit der Anwendung des PPNM sollen Logiken beispielhaft aus der Tabelle entfernt und durch das Modell evaluiert werden. Im Vergleich zur Formelintegration in einer Tabelle geht dies zwar zu Lasten der schnellen Erweiterbarkeit, bringt aber die Vorteile des Formalismus mit sich. Beispielsweise wird es auf das gesamte Modell einheitlich angewendet und potentielle Fehlerquellen werden vermieden. Beim Nachweis qualitativer Eigenschaften des Modells geht es hauptsächlich um die Erkennung und somit der möglichen Korrekturen von Planungsfehlern.

Unter **Leistungsmaßen** werden Ergebnisdaten verstanden. Diese ergeben sich durch eine Simulation und stellen das Ergebnis dieser dar. Je komplexer und umfangreicher ein Modell ist, umso schwieriger ist die Überprüfung und Bewertung. Dabei handelt es sich oft um stochastische Variablen und der Auswertung der statistischen Ergebnisse. Ein erster Ansatz stellt die interaktive Simulation dar, bei der die Zustände schrittweise angezeigt und durchschaltbar sein können. Für den Nachweis bestimmter Eigenschaften und Leistungsmaße sind dagegen Methoden und Werkzeuge notwendig. [vgl. Zimmermann 1997, 119] Für den Nachweis qualitativer Eigenschaften des Modells geht es hauptsächlich um die Erkennung und somit mögliche Korrektur von Planungsfehlern. Diese Eigenschaften können durch das Modell ausgelesen und müssten in geeigneter Form wieder zurückgespielt werden. Hierzu gibt es verschiedenste wissenschaftliche Untersuchungen wie in [Kruse u. a. 2014] und die Auswertung der PN-Eigenschaften der Lebendigkeit, Beschränktheit und Sicherheit. Für systematische Vergleiche verschiedener Modelle kann die automatische Auswertung und grafische Darstellung der Ergebnisse nützlich sein. Durch den Vergleich verschiedener Varianten (Plan, Ist und Prognose) und ihrer Bewertung kann eine Optimierung aufgezeigt werden. Das hier für diesen Ansatz vorgestellte PN ist kein konfliktfreies PN. Im Rahmen dieser Arbeit wird dieses Themengebiet nicht weiter betrachtet.

4^{Kapitel} Entwicklung eines Petri-Netz-Modells für Projekte

Dieses Kapitel zielt darauf ab, die aus Kapitel 3 identifizierten Anforderungen auf das PN zu übertragen und als ein Modell umzusetzen. Die Idee der Konstrukte zur Abbildung dieser Anforderungen wird näher erläutert und an einzelnen dafür entwickelten Konstrukten aufgezeigt. Nach den Vorgaben des Autors erstellten und betreuten Masterarbeit von [Schott 2019] basieren die Ausführungen dieses Kapitels und das darin aufgenommene Konzept der Konstrukte, was durch den Autor erweitert wurde. Die Abbildungen der Konstrukte bestehen je nach Ausschnitt aus überarbeiteten, neu gestalteten oder erweiterten Bildern aus [Schott 2019]. In der Theorie wäre das PPNM und deren Logiken auch mit allgemeinen DE-Modellen oder Programmiersprachen umsetzbar. Die interne Logik des theoretischen Modells eines PN und dessen enthaltenen Transitionen müsste aber programmtechnisch speziell realisiert werden. Dies würde keiner allgemeinen Theorie unterliegen und der Ansatz der einheitlichen Logik für jeden Transitionstyp würde so verloren gehen.

Bei der Modellierung von Projekten mit Petri-Netz-Konzepten muss die Adaption von Zustandselementen und Ereignissen aus dem PN ins PM übertragen werden. Dafür wurde der Begriff Projekt-Objekt eingeführt und in Verbindung mit den diskreten Modellen näher betrachtet, welche sich in der Grundidee der modularen Konstrukte widerspiegeln. Im vorliegenden Kapitel wird darauf aufbauend die Tabellenstruktur exemplarisch dargestellt. Dabei liegt der Fokus nicht auf den inhaltlich zusammenhängenden Ausschnitt, sondern darauf, das Konzept zu verdeutlichen. Danach soll die aufgegriffene Problematik der automatischen Generierung anhand der Abweichung von Plandaten und Ist-Daten beispielhaft betrachtet werden.

4.1 Modellierungsgegenstand im Petri-Netz

Den Modellierungsgegenstand bildet die **gemeinsame Abstraktionsebene**. Diese basiert seitens PM auf den Projekt-Objekten. Seitens PN werden modulare Konstrukte zur Abbildung definiert.

4.1.1 Konzept der Konstrukte und Schnittstellen

Bei der vorliegenden komplexen Problemstellung ist eine Aufteilung des Problems in mehreren Komponenten anzuraten, welche untereinander mit definierten Schnittstellen kommunizieren. Im Rahmen dieser Arbeit wird auf die grundlegenden Konzepte eingegangen und nachfolgend identifiziert. [vgl. Reisig und Desel 2014, 173; vgl. Schott 2019, 28]

Unter einem **Konstrukt** wird der **Verbund von Petri-Netz-Elementen** verstanden, welche den identifizierten Anforderungen aus dem 3. Kapitel gerecht werden. Die Bildung der Konstrukte orientiert sich dabei teilweise an den definierten Projekt-Objekten Zeit, Ressourcen und Aktivitäten und deren Einteilung aus 3.2.4 vorgenommenen Vorgänger-Nachfolger-Abhängigkeiten. Dafür wird das Grundprinzip eines Konstruktes wie folgt erklärt:

Jedes Konstrukt muss eine **Schnittstelle** bzw. ein Interface zur Verfügung stellen. Diese Schnittstelle ermöglicht es anderen Konstrukten, auf Daten der einzelnen Komponenten zuzugreifen. Dies baut auf dem **Grundprinzip der Platzunternetze** bei PN auf. Für die bessere Verständlichkeit und zur farblichen Hervorhebung einzelner Verbindungen wird das folgende Farb- und Abbildungsschema eingeführt, welches in Abbildung 9 durch ein neutrales Beispiel-Konstrukt aufgezeigt wird. Dieses Beispiel dient dazu, die Bedeutung der eingeführten Farben zu erklären. In der Abbildung sind zwei Konstrukte mit jeweils einer Schnittstelle ersichtlich, wobei die Funktionsweise des Netzes nicht von Bedeutung ist. Fest belegte Farben im zugrundeliegenden Farbschema sind Dunkelgrau, Dunkelrot und Grün. Das Farbschema wird nur auf Elemente des PN angewendet (Transitionen, Plätze, Kanten). Jede andere farbliche Hervorhebung von Elementen oder Markierungen bestimmter Bereiche werden bei Erklärungen von bestimmten Sachverhalten eingesetzt. Im Anhang H ist die Legende für alle nachfolgenden Abbildung zu finden. Nachfolgend wird die Bedeutung des zugrunde liegenden Farbschemas erklärt: Dunkelgraue Petri-Netz-Elemente wie Kanten, Plätze und Transitionen stellen das abzubildende Konstrukt dar. Ein grüner Platz dient der Hervorhebung von Schnittstellen, die das Konstrukt zur Verfügung stellt. Wenn ein Zugriff auf andere Schnittstellen anderer Konstrukte erfolgt, wird dies durch einen grünen Platz und der Kenntlichmachung des Konstruktes durch ein Konstrukt-Viereck dargestellt. Eine dunkelrote Transition stellt eine zeitlose Transition

dar, welche inhaltlich später weiter erläutert wird. Bei den Marken kommen ebenfalls verschieden farbige Ausführungen zum Einsatz. So repräsentiert eine schwarze Marke eine Standardmarke. Neben den Standardmarken werden ebenso strukturierte und farbige Marken eingesetzt. Die jeweils herangezogenen Farben werden beim jeweiligen Beispiel zugewiesen.

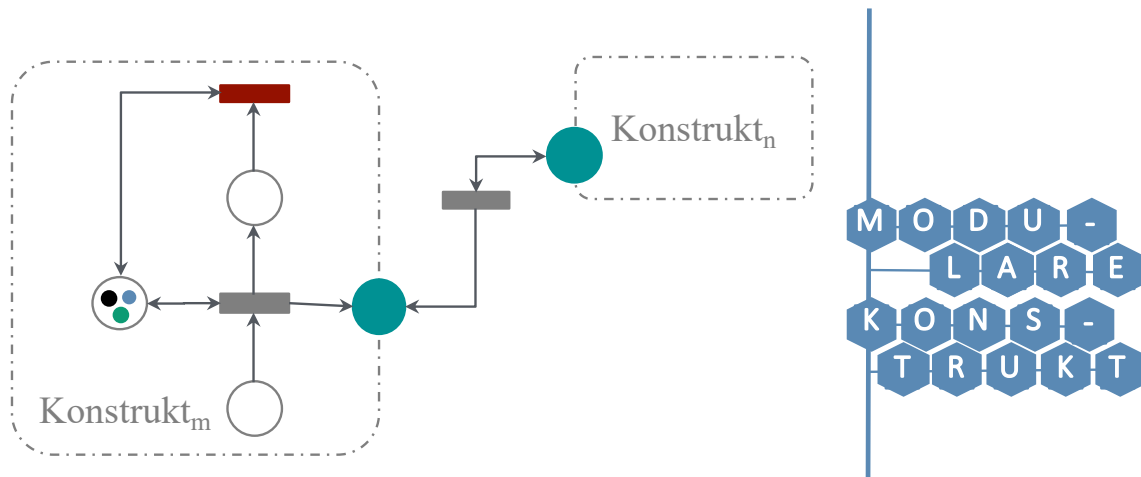


Abbildung 9: Konstrukt-Idee

4.1.2 Einbindung der Atomarität

Im Rahmen der Atomarität wurden folgende Begriffe definiert: atomare Einheit, atomare Aktivitätsphase und atomare Ressourcenzuordnung. Dabei wurde die Atomarität immer auf deren sinnvolle Eingriffsmöglichkeit untersucht. Doch wie spielen die verschiedenen Begriffe zusammen und ineinander?

Bei der **atomaren Einheit** handelt es sich um die am besten geeignete zu definierende Ebene des benötigten Modellierungsobjektes. Bei der Untersuchung hat sich herausgestellt, dass dies auf die Aktivität und Ressource anzuwenden ist.

Die **kleinste gemeinsame Einheit** ist durch das zugrunde gelegte Zeitmodell vorgegeben. In der Arbeit wird hier die Stunde als Einheit festgelegt. Die Angaben bei der Aktivitätsabarbeitung, als auch bei den Ressourcen beruhen auf dieser Zeiteinheit. Durch die Taktung des Netzes kann nach jeder Schaltfrequenz gestoppt und somit ein Eingreifen forciert werden. Für Analysen und Simulationen sind aber managementrelevante Eingriffe notwendig. Auf dieser atomaren Einheit können keine Eingriffsmöglichkeiten

definiert werden. Dies würde den Nutzen von Eingriffen und Steuerung durch zu hohen Aufwand relativieren.

Es muss eine geeignete Einteilung gefunden werden, um die Aktivitäten sinnvoll im Rahmen ihrer Abfolge durch Steuerung und Eingriffe zum Nutzen des Projekterfolges zu modellieren. Dabei wird auf die **Phasen der Aktivitäten** zurückgegriffen. Eine Aktivität an sich spiegelt eine Bündelung von Teilaktivitäten einer übergeordneten Gesamtaktivität wider. Dabei können im Rahmen dieser Aktivität verschiedene Phasen durchlaufen werden. Die Überwachung einer Aktivität hat vorrangig eine optimale Ressourcen- und Zeitauslastung zum Ziel. Aufgrund der Anforderungen, dass die Abarbeitung von Aktivitätsschritten in einem definierten Zeitrahmen erfolgen kann, begründet es die Verlegung von Eingriffsmöglichkeiten auf die Ebene der Phasen. Eine Aktivitätsphase ist innerhalb eines definierten Zeitraums abzuarbeiten. Zudem kann der Zeitraum größer als die zugewiesene Dauer ausfallen. Beispielsweise muss eine Aktivität mit einem Aufwand von drei Stunden innerhalb einer Woche abgearbeitet werden. Dabei muss per Definition nicht zu jeder Zeiteinheit ein Abarbeitungsschritt mit all seinen Vorbedingungen resultieren. Im Modell wird hierfür die Möglichkeit eingeräumt, eine Phase als Erfolg, Fehlschlag oder mit Überziehung zu signalisieren. Diese Parameter sind für Eingriffsentscheidungen notwendig.

Die **atomare Ressourcenzuordnung** ist darauf begründet, dass eine Aktivitätsphase nicht am Stück abgearbeitet werden muss. Daher wurde die Möglichkeit im Modell geschaffen, eine Ressourcenzuordnung nach jedem Abarbeitungsschritt vorzunehmen. Hier ist die Art der Ressource ausschlaggebend. Ressourcen, welche mit einer Mengeneinheit versehen sind – beispielsweise der Drucker oder das Papier – werden einmalig zugeordnet und sind bis zur Beendigung der Aktivitätsphase an diese gebunden. Der Einsatzzweck und die Wiederverwendung spricht gegen eine Zuordnung pro Abarbeitungsschritt. Deswegen werden die zwei unterschiedlichen Arten von Ressourcen anders behandelt und ihren Gegebenheiten angepasst. Die daraus resultierende Ressourcenzuordnung pro Abarbeitungsschritt begründet die atomare Ressourcenzuordnung. Im vorgeschlagenen PPNM ist dies widergespiegelt, indem eine Ressourcenanfrage nicht auf eine Ressource begrenzt, sondern auf dessen Eigenschaften ausgelegt ist. Somit kann bei jeder Ressourcenzuordnung anhand von hinterlegten Regeln eine Entscheidung getroffen werden. Ebenfalls wird bei einer nicht erfolgreichen

Ressourcenzuteilung und nach Ablauf der definierten Bearbeitungszeit einer Phase eine Konfliktsituation angezeigt.

4.2 Beschreibung des verwendeten Petri-Netz-Modells

Grundlegend wird das Verhalten des verwendeten PN so definiert, dass ein Platz einem Ereignis bzw. einem bestimmten Zustand im Umfeld einer Aktivität und der Transition die Aktivität (Aktivitätstransition) selbst mit einem Zustandsübergang zugeordnet wird. Ebenfalls wird für die Schaltregel vom Prinzip des maximalen Schaltens ausgegangen und es wird kein konfliktfreies PN für den hier vorgestellten Ansatz angestrebt. [vgl. Won, H.C. 1987]

Dabei gilt nach [vgl. Weichenhain und Fengler 2019a]:

Eine Aktivitätstransition ist schaltfähig, wenn sie aufgrund ihrer Vor- und Nachbedingungen schaltfähig ist. Dieses ist abhängig von der gewählten Interpretation und Erweiterung des PN und wird im Weiteren noch beschrieben. Die Schaltfähigkeit bzgl. der Nachbedingungen ist hier im Allgemeinen immer gegeben. Dieses folgt aus der Struktur der transformierten Netze. Mit dem Schalten der Vorbedingungen wird die zugeordnete Aktivität entsprechend dem gewählten Zeitmodell gestartet und somit verzögert beendet. Die Änderung der Markierung der Vorbedingungen erfolgt mit dem Start der Aktivität, die der Nachbedingungen mit deren Ende. Nachbedingungen von Aktivitätstransitionen können Vorbedingungen von weiteren Aktivitätstransitionen sein. Transitionen ohne Aktivität können ebenfalls vorkommen, beispielsweise zur Aussteuerung des Startens von Aufgaben.

Im Rahmen dieser Arbeit werden die Erweiterungen mit höheren zeitbewerteten PN, gefärbten PN mit strukturierten Marken, angelehnt an die Prädikat-Transitions-Petri-Netze Verwendung finden.

Diese strukturierten Marken, wie sie bereits für einen anderen Anwendungsfall definiert wurden [Al Ali 2010], ermöglichen, dass Marken z. B. Träger von Bearbeitungszuständen, Eigenschaften von Ressourcen oder projektbezogenen Informationen und Dokumenten sein können. Die in [Al Ali 2010] definierte Schaltregel ermöglicht durch geeignete strukturierte Vielfachheiten der Kanten, aus Operation auf

die Strukturen der Marken und logischen Verknüpfungen dieser, die Ausführung von Aktivitäten von der Existenz und dem Inhalt der Marken abhängig zu steuern. Strukturierte Marken beschreiben damit die Existenz und aktuelle Datenstruktur von Ereignissen und Zuständen im Umfeld der Aktivität.

Durch eine eindeutige Abbildung aller denkbar auftretenden Belegungen der Strukturen in einer Menge logischer Farben sind prinzipielle Untersuchungen zum Gegenstand des vorliegenden Beitrags auch mit Coloured Petri Nets des Tools PenecaChromos [Al Ali 2010] möglich (z. B. Transformation, Simulation). Für die vorliegenden Ergebnisse wird dies im Kapitel 6 genutzt. Für reale Anwendungen ist aufgrund der notwendigen Größe der Farbmenge und der Anzahl der Zuordnungen dieser Weg nicht praktikabel. Es wird dann ein speziell zugeschnittenes PN-Tool notwendig sein. Das wird entsprechend dem gewählten Prinzip, dass Dateneingaben und Ergebnisausgaben in den Tabellen erfolgen, vorwiegend die Simulation und Komposition der Konstrukte unterstützen, wobei dem graphischen Nutzerinterface dabei eine untergeordnete Bedeutung zukommt.

Ein Konstrukt im bisher schon benutzten Sinn, stellt in der hier verwendeten Art im PN ein hierarchisch untergeordnetes Platzunternetz dar. Ein Platzunternetz ist dabei ein Teilnetz, in dem Kanten aus dem umgebenden Netz nur an Plätzen enden und Kanten zum umgebenden Netz nur an Plätzen beginnen. Einschränkungen bzgl. der inneren Struktur des Teilnetzes existieren allgemein nicht, sondern sind vom Konstrukt abhängig. Dies gewährt eine immer gleichbleibende Kopplung von Konstrukten über die als Schnittstelle definierten Plätze, durch den Mechanismus des Übereinanderlegens von Plätzen.

Somit wird das im Rahmen dieser Arbeit entwickelte Petri-Netz-Modell wie folgt in Anlehnung an [Reisig 2010; Starke 1990a; Rokyta 1992; Fengler und Philippow 1991] und des Petri-Netz-Modellierungstools PenecaChromos des Fachgebietes Rechnerarchitektur und Eingebettete Systeme der Technischen Universität Ilmenau [Weichenhain und Fengler 2019b] definiert. Dieses Tool wird für eine beispielhafte Umsetzung später ausgewählt und deckt für den vorgesehenen Einsatzzweck alle erforderlichen Anforderungen ab. Für Arbeiten im Rahmen der Dissertation (z. B. [Schott 2019]) war es notwendig zur Dokumentation des Tools eine Reihe von Definitionen hinzuzufügen, die hierfür verwendet wurden und im Folgenden auszugsweise angegeben

werden [Weichenhain und Fengler 2019b]. Anschließende Definitionen sind [Weichenhain und Fengler 2019b] entnommen.

Struktur

$$PN = (P, T, F, V, K, m_0)$$

$$P \cap T = \emptyset$$

$$F \subseteq P \times T \cup T \times P$$

$$V: F \rightarrow \mathbb{N}$$

$$K: P \rightarrow \mathbb{N} + \infty$$

$$m_0: P \rightarrow 0 + \mathbb{N}$$

Weitere Erklärungen:

$$m: P \rightarrow 0 + \mathbb{N} + \infty$$

Die Schaltfähigkeit und das Schalten werden wie folgt definiert

Schaltfähigkeit (sf):

t_i ist sf, wenn

$$\forall (p, t_i) \in F \rightarrow m(p) \geq V(p, t_i)$$

$$\wedge (\forall (t_i, p) \in F \rightarrow m(p) + V(t_i, p) \leq K(p))$$

Schalten (s):

t_i schaltet wenn t_i sf ist

Ergebnis des Schaltens:

$$\forall (p, t_i) \in F \rightarrow m^{k+1}(p) = m^k(p) - V(p, t_i)$$

$$\wedge (\forall (t_i, p) \in F \rightarrow m^{k+1}(p) = m^k(p) + V(t_i, p))$$

Es kommen fünf verschiedene Schaltregeln zum Einsatz

$t_z(k)$ hier: Zeitpunkt

Δt_z Zeitbereich

$$t_z(k + 1) = t_z(k) + \Delta t_z$$

Free Choice für das beliebige Schalten:

$$0 \leq \Delta t_z \leq t_{zmax} \wedge t_i \text{ ist sf} \mid t_{zmax} \in \mathbb{R} \wedge \Delta t_z \in \mathbb{R}, \text{beliebig}$$

Maximal, wobei gleichzeitig schaltfähige Transitionen auch gleichzeitig schalten müssen:

$$\Delta t_z = (t_z(k+1) - t_z(k)) \text{ geht gegen } 0$$

Zeitverzögert bedeutet, dass die Transition im Zeitbereich vom Beginn der Schaltfähigkeit bis zum Schalten ununterbrochen schaltfähig sein muss. Das Schalten der Transition erfolgt mit einer Zeitverzögerung:

$$w_{tv}: T \rightarrow W_{tv}$$

$$t_z(k+1) = t_z(k) + w_{tv}(t_i) \mid (t_z(k) \leq t_z \leq (t_z(k) + w_{tv}(t_i))) \rightarrow t_i \text{ ist sf}$$

$$w_{tv} \in \mathbb{R}$$

Beim synchron getakteten Schalten muss eine ununterbrochene Schaltfähigkeit vorliegen:

Zeitperiode T:

$$\exists T \mid 0 < T \leq T_{max} \wedge T = \text{const}$$

Getaktete Transitionen sind Zeitdauertransitionen, die entweder mit dem Konzept der Zeit verbunden sind und somit an das Eintreffen von frühesten oder spätesten Zeitpunkten gebunden sind, sowie an das Einsetzen der Schaltfähigkeit durch alle Vorkanten. Oder sie stellen Zeitdauertransitionen dar, welche nicht an das Konzept der Zeit gebunden sind und somit lediglich an die Bedingungen aller Vorkanten knüpfen. Das Schalten einer getakteten Transition ohne zusätzliche Zeitdauer benötigt einen Takt.

$$t_z(k) = n_j \cdot T \mid n_j \in 0 + \mathbb{N}$$

$$t_z(k+1) = (n_j + 1) \cdot T$$

$$t_z(k) \text{ gilt für alle getakteten } t_i \in T$$

Synchron zeitverzögertes Schalten:

$$t_z(k) = n_j \cdot T \mid n_j \in \mathbb{N}$$

$$t_z(k+1) = (n_j + w_{tv}) \cdot T \mid w_{tv} \in \mathbb{N}$$

Die zeitlosen Transitionen sind im Netz später dunkelrot dargestellt und besitzen keine Verzögerungszeit. Daraus ergibt sich, dass eine zeitlose Transition bei Aktivierung sofort schaltet (Schaltregel Maximal). Es handelt sich um eine rein steuernde Transition. Es kann auch Gründe geben, eine Steuerung nicht mit einer zeitlosen Transition zu modellieren.

Die Definitionen zu Platzunternetzen sind folgende

UN: Umgebungsnetz

PU_i : Platzunternetz_i

PN besteht aus: UN, $PU_i \mid i = 1 \dots n$

Randplätze von PU_i sind P_{PU_iR}

$$UN = (P_{UN}, T_{UN}, F_{UN}, V_{UN}, K_{UN}, m_{0UN})$$

$$P_{UN} \cap T_{UN} = \emptyset$$

$$F_{UN} \subseteq P_{UN} \times T_{UN} \cup T_{UN} \times P_{UN}$$

$$\cup (P_{PU_iR} \times T_{UN} \mid i = 1 \dots n) \cup (T_{UN} \times P_{PU_iR} \mid i = 1 \dots n)$$

$$V_{UN}: F_{UN} \rightarrow \mathbb{N}$$

$$K_{UN}: P_{UN} \rightarrow \mathbb{N} + \infty$$

$$m_{0UN}: P_{UN} \rightarrow 0 + \mathbb{N}$$

Platzunternetze $PU_i(PU_i)$ ohne Hierarchie

$$PU_i = (P_{PU_i}, T_{PU_i}, F_{PU_i}, V_{PU_i}, K_{PU_i}, m_{0PU_i})$$

$$P_{PU_i} \cap T_{PU_i} = \emptyset$$

$$F_{PU_i} \subseteq P_{PU_i} \times T_{PU_i} \cup T_{PU_i} \times P_{PU_i}$$

$$V_{PU_i}: F_{PU_i} \rightarrow \mathbb{N}$$

$$K_{PU_i}: P_{PU_i} \rightarrow \mathbb{N} + \infty$$

$$m_{0_{PU_i}}: P_{PU_i} \rightarrow 0 + \mathbb{N}$$

$$P_{PU_i} \cap P_{UN} = \emptyset$$

$$P_{PU_i} \cap P_{PU_j} = \emptyset \forall i, j \mid i \neq j$$

$$T_{PU_i} \cap T_{UN} = \emptyset$$

$$T_{PU_i} \cap T_{PU_j} = \emptyset \forall i, j \mid i \neq j$$

$$F_{PU_i} \cap F_{UN} = \emptyset$$

$$F_{PU_i} \cap F_{PU_j} = \emptyset \forall i, j \mid i \neq j$$

Randplätze von PU_i

$$P_{PU_iR} \subseteq P_{PU_i}$$

$$\exists(p, t) \mid (p \in P_{PU_i} \wedge t \in T_{UN}) \vee \exists(t, p) \mid (p \in P_{PU_i} \wedge t \in T_{UN}) \leftrightarrow p \in P_{PU_iR}$$

$$p \in P_{PU_i} \cap P_{PU_iR} \leftrightarrow \exists(p, t) \in F_{UN} \wedge \exists(t, p) \in F_{UN}$$

Hierarchie

$$PU_i \text{ besteht aus } UN_{PU_i}, PU_{PU_{i,j}} \mid j = 1 \dots m$$

Definitionen analog zu UN, PU_i : Sollen Randplätze eines $PU_{i,j}$ gleichzeitig Randplätze von PU_i sein, gilt:

$$P_{PU_{i,j}R} \cap P_{PU_iR} \neq \emptyset$$

Sonderkanten

Testkante (p, t) :

$$\text{sf: } m(p) \geq V(p, t)$$

$$\text{s: } m^{k+1}(p) = m^k(p)$$

Inhibitorkante (p, t) :

$$\text{sf: } m(p) = 0$$

$$\text{s: } m^{k+1}(p) = m^k(p)$$

Unterlaufkante (p, t) :

sf: $m(p) < V(p, t)$

s: $m^{k+1}(p) = m^k(p)$

Setzkante (p, t) :

sf: $m(p)$ beliebig

s: $m^{k+1}(p) = V(p, t) \mid V(p, t) \leq K(p)$

Rücksetzkante (p, t) :

sf: $m(p)$ beliebig

s: $m^{k+1}(p) = 0$

Kapazitätsbeschränkter Platz

Ein kapazitätsbeschränkter Platz kann für die spätere Umsetzung in PenecaChromos oder in den Modellen ersetzt werden. Dieses Vorgehen gilt ebenfalls sinngemäß für strukturierte Marken. Die Kapazität wird dadurch ersetzt, dass zu jedem kapazitätsbeschränkten Platz (p) ein Ersatzplatz (p_E) generiert wird, definiert mit folgenden Konstruktionsregeln (je p mit $K(p) \neq \infty$):

1. zu p wird ein Ersatzplatz p_E zusätzlich eingeführt
2. $p, p_E: K(p) = K(p_E) = \infty$
3. $m_0(p_E) = K_{alt}(p) - m_0(p)$
4. $\forall t \mid (t, p)$ wird erzeugt: ein (p_E, t) mit $V(p_E, t) = V(t, p)$
5. $\forall t \mid (p, t)$ wird erzeugt: ein (t, p_E) mit $V(t, p_E) = V(p, t)$

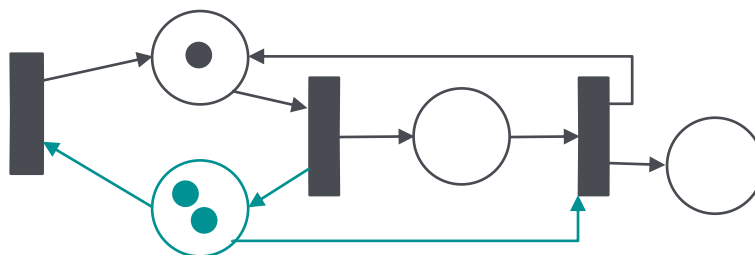


Abbildung 10: Auflösung kapazitätsbeschränkter Platz

Die im PPNM entwickelten modularen Konzepte verwenden Standard-Petri-Netz-Komponenten nach der Definition aus Kapitel 2.3.1 und die hier ausgeführten Definitionen. Ausnahmen stellen die Verwendung von gefärbten Marken bei der Abbildung einer Aktivität und der strukturierten Marke bei den Ressourcen dar. Ferner werden ausgewählte Plätze als Schnittstelle der modularen Konstrukte definiert.

Für das in dieser Arbeit entwickelte Petri-Netz-Modell werden die Farbsets wie folgt definiert: Das elementare Farbsset setzt sich aus schwarzen Marken (zur Abbildung des allgemeinen Verlaufes des Netzes) und farbigen Marken (zur Kennzeichnung, dass eine Aufgabe beendet ist) zusammen. Beim Aufbau des strukturierten Farbsets setzt sich die strukturierte Marke wie in Kapitel 4.5 zusammen.

In [Al Ali 2010] wird dargelegt, dass ein Netz mit strukturierten Marken auf ein farbiges PN transformiert werden kann. In [Schott 2019] wird ein ausgewählter vereinfachter Umfang des hier vorgestellten Konzeptes mit PenecaChromos umgesetzt. Somit kann das im Rahmen dieser Arbeit entwickelte Modell durch das PN Tool PenecaChromos simuliert und umgesetzt werden und stellt sicher, dass das Modell den Gegebenheiten eines PN entspricht. [vgl. Al Ali 2010, 85 ff; vgl. Schott 2019, 61 ff]

4.3 Tabellenstruktur

Die Darstellung und die Schnittstellen der Projektplanung zum Projektmanager erfolgen in Tabellenform. Im 5. Kapitel wird ein Beispielprojekt mit realen Tabellen aufgezeigt, welches die Grundlage für die hier entworfene Tabellenstruktur beeinflusste. Hier ist auch die enorme Komplexität dieser Tabellen ersichtlich. Damit wird auf eine Standardisierung der Daten abgezielt und erlaubt, automatisch Modelle zu erstellen. Eine Voraussetzung dafür ist, dass alle Daten, Fristen und Zusammenhänge in die Tabelle aufzunehmen sind.

Zur Überführung in ein PN werden folgende Regeln aufgestellt: Eine Zwischenmarkierung in einem PN entspricht einem Zwischenschritt bei der Erreichung der vorgegebenen Planwerte, unter Einbeziehung vorgegebener Istwerte. Eine Zwischenmarkierung ist durch n-maliges Schalten während der Simulation entstanden. Die Werte in den Tabellen können durch die vorgegebenen PN-Konstrukte und dessen Logik in Anfangsmarkierungen sowie Zwischenmarkierungen überführt werden. Durch den hier enthaltenen Formalismus ist auch die Rückführung einer Zwischenmarkierung

in Tabellenwerte gegeben. Dabei ist ausschlaggebend wie die Projekt-Objekte in der Projektplanung im Petri-Netz abgebildet werden, was in Tabelle 1 abstrahiert und als grober Überblick dargestellt ist:

Tabelle 1: Zusammenspiel Projektmanagement zu Petri-Netz

Projekt-Objekt	PM	PN	
Aktivität	Ein Schritt der Abarbeitung	Schalten einer Transition	
	Ressourcenzuordnung		
Ziel	Fertigstellung einer Aktivität	Marke im Nachplatz einer Transition	
Ressource	Daten oder Dinge	Marke in vordefiniertem Platz	
	Input / Output	Vorplatz / Nachplatz	
	Wiederverwendbarkeit	PN Konzept mit Rückführung oder nicht	
Zeit	Zeitangaben: Frühester Startzeitpunkt, spätestes Ende, Dauer	Globale und lokale Zeit-Konstrukt Anzahl Marken	
	Beziehungen	Vorgänger/Nachfolger	Richtung der Kanten
	Beurteilungskriterien	Bedingungen	Schaltregel an der Kante

Die Angaben in der Tabelle müssen den aufgestellten Anforderungen aus 3.2.2 entsprechen. Dabei müssen folgende Regeln berücksichtigt werden:

- Datum: Es muss ein einheitliches Datumsformat eingehalten werden.
- Dauer: Hier muss die Angabe dem Vielfachen der zugrunde gelegten Zeit aus dem Zeitkonzept entsprechen. Dies bedeutet bei der Zeiteinheit Stunden beispielsweise, dass die Angaben für einen Arbeitstag 8h entsprechen, bei einer 40 Stundenwoche.
- Menge: Entspricht die Angabe aus der Menge von natürlichen Zahlen.
- ID: Eindeutig, Anwendung eines Schemas, darf ungeordnet sein, darf nicht Null oder Unendlich sein.

- In den Tabellen müssen für relevante Plandaten auch die Möglichkeit der Angabe von Ist- und Prognosedaten gewährleistet werden. Gleichmaßen ist eine Fortschrittsangabe notwendig, um eine vorzeitige Planerfüllung zu berücksichtigen.

Die Datenaufnahme muss der Anforderung eines modular erweiterbaren Tabellenformats entsprechen. Dies ist für die automatische Generierung des PN und dessen zugrundeliegenden Konstrukten notwendig. [vgl. Weichenhain und Fengler 2019a]

Es kommt eine optimierte Tabellenform zum Einsatz, bei der die Tabellengröße reduziert und auf eine reine Dateneingabe beschränkt wird. Somit findet die Abbildung des Projektes und dessen Überwachung, Analyse und Simulation nicht durch hinterlegte Formeln oder Logiken statt, sondern während des Projektes durch den Einsatz von PN-Konstrukten und benötigten PN-Elementen zur Verbindung dieser. [vgl. Weichenhain und Fengler 2019a]

Die Tabellen zur Projektplanung werden auf mehrere Tabellen aufgeteilt. Die Backlog wird erweitert und zur Projektplanung genutzt. Dabei wird sich bei der Ausgestaltung der Spalten nur auf die schon zugeordneten Aufgaben beschränkt. Es wird die Annahme getroffen, dass sobald eine Sprintzuordnung erfolgt, diese Aufgabe umgesetzt wird. Die inhaltliche Verknüpfung wird dabei durch einheitliche IDs realisiert. Die ID unterliegt einem Schema was es erlaubt, beim Einsatz von strukturierten Marken auf Informationen zuzugreifen. Bei den festgelegten Generationsphasen ist mitzugeben, welches Zeit- bzw. Arbeitszeitmodell Anwendung finden soll. Dabei wird standardmäßig von einer 40-Stundenwoche ausgegangen, wobei Abweichungen hinterlegt werden müssen. [vgl. Weichenhain und Fengler 2019a]

Die nachfolgende Abbildung listet, eingeschränkt auf den Fokus der Sprints, alle benötigten Informationen im Tabellenformat auf.

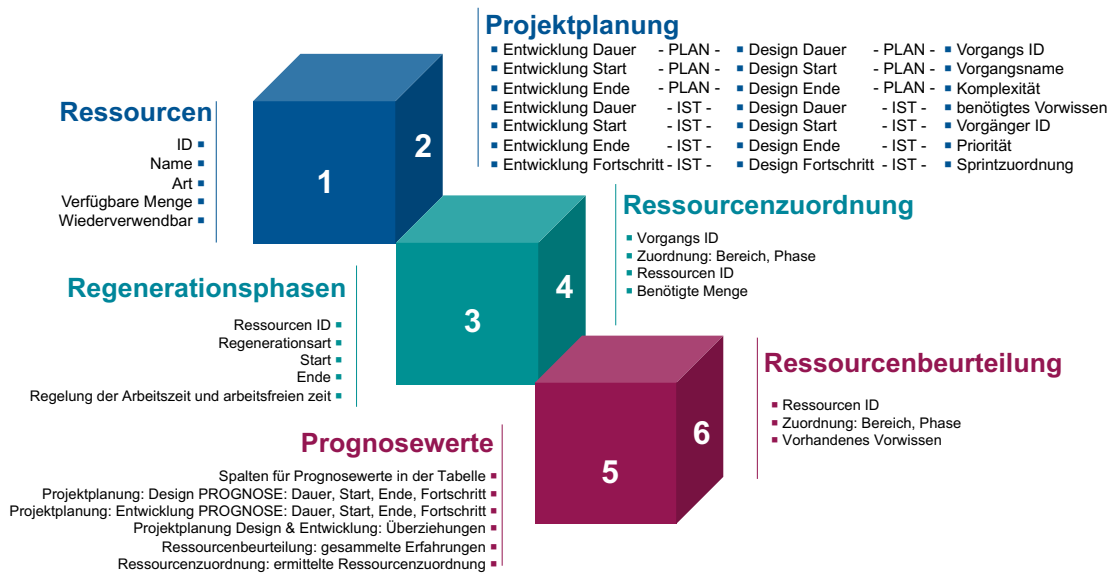


Abbildung 11: Benötigte Informationen und Aufbau der Tabellen

Besonderheit ID-Schema

Die ID muss nicht nur die eindeutige Identifikation einer Information in einer Tabelle bereitstellen, sondern auch deren Gruppierung.

Beispielsweise wird bei der wiederverwendbaren Ressource Mitarbeiter jede Aktivität (Teilinformationen der ID-Struktur) in der Marke gespeichert und durch das Konstrukt von wiederverwendbaren Ressourcen für spätere Aufgaben hinterlegt. Anhand der ID-Struktur der Aufgaben kann so beispielsweise ermittelt werden, ob die Ressource schon Erfahrung in dem Aufgabengebiet gesammelt hat. [vgl. Weichenhain und Fengler 2019a]

Das zu erstellende Schema ist dabei immer kontext- und projektabhängig. Im vorliegenden Beispielprojekt könnte das Schema aus <Bereich_Gebiet_fortlaufende Nummer> bestehen.

- Bereich: System, Daten, Report, fachliche Themengebiete
- Gebiet: Schnittstelle, Funktion, Design, Datendefinition
- Fortlaufende Nummer: 001, 002, 003

Eine ID für die Aktivität R_D_001 würde zum Beispiel aussagen, dass es sich um eine Aufgabe im Bereich der Reportingfunktionalitäten um das Design des Reports handelt. Eine Ressource, die schon viele Aufgaben im Bereich Report abgearbeitet hat, würde somit favorisierend zum Einsatz kommen.

Besonderheit der Priorität

Die Spalte Priorität ordnet die Aufgaben innerhalb eines Sprints nach ihrer Bedeutung an. Die Priorität wird als ganzzahlige Zahl größer null definiert. Dabei kann jeder Aufgabe die Standardpriorität von Eins mitgegeben werden. Je höher die Priorität, um so größer ist die gesetzte Zahl. Diese Prioritäten können dazu dienen, auftretende Konflikte im Petri-Netz zu steuern. Dabei wird die Priorität über die gesamte Aufgabe festgelegt, was zur Auswirkung hat, dass jegliche Art von Konflikten mit der gleichen Priorität angegangen werden. Dies ist eine vereinfachte Möglichkeit, Konflikte mit Regeln auszusteuern. Durch den Einsatz von strukturierten Marken kann dies noch optimierter erfolgen.

Besonderheit der Einbeziehung des Fortschritts

Im Tabellenformat ist die Aufnahme von Plandaten und Ist-Daten vorgesehen sowie ein Fortschrittsindikator. Die Annahme für das Plan-Modell, Plan-Ist-Modell und das reine Ist-Modell ist, dass solange der Fortschritt keine 100 % erreicht hat, die angegebenen Plan- und Istwerte für die Markierung der erledigten und unerledigten Arbeit herangezogen werden. Es werden keine proportionalen Annahmen getroffen. Im Falle eines Prognosemodells kann hiervon abgewichen werden.

Beispielhafter Ausschnitt einer Tabelle

Die nachfolgende Tabelle stellt einen Ausschnitt aus der Tabelle der Projektplanung mit den hier hervorgehobenen Aspekten dar. In der aufgezeigten Tabelle wurde beispielsweise die Zuordnung der Aufgaben zu den jeweiligen Sprints und dessen Start- und Endtermine festgehalten. Für das spätere Beispiel in Kapitel 6 werden im jeweiligen Kapitel alle benötigten Tabellen aufgezeigt.

Tabelle 2: Auszug aus der Tabelle Projektplanung zur Datenpflege von Sprints

ID	Vorgangsname	V ID	S	D	Start	Ende	D E	Start	Ende
				Ds	Ds	Ds		E	E
S_SS_001	Excel als SS	-	1	6	22.01	26.01	13	29.01	02.02
R_F_001	R Filterfunktionen	S_SS_001	1	14	22.01	26.01	29	29.01	02.02
R_Ds_001	R Standardlayout	S_SS_001	1	8	29.01	02.02	16	05.03	09.03
S_SS_002	Anbindung an S	-	2	16	23.02	02.03	15	04.03	15.03
PA_SS_002	Datenübertragung PZ Excel	S_SS_001, R_D_001	1	15	15.02	10.03	28	11.03	16.03

Legende zur Tabelle: Dauer (D), Design (Ds), Entwicklung (E), Funktionen (F), Planzahlen (PZ), Planzahlenabweichungen (PA), Report (R), Schnittstellen (SS), System (S), Vorgänger (V)

Besonderheit der Ressourcenzuordnung

Das Ressourcenmanagement gibt an, welche Aufgabe mit welcher Ressource besetzt werden soll. Dabei wird im Rahmen der Projektplanung meistens eine vorab definierte Zuteilung erfolgen. Diese basiert auf der Grundlage nach bestem Wissen und Gewissen der Verantwortlichen und den Kernkompetenzen der Ressourcen. Im Gegensatz dazu könnte eine Zuordnung und Priorisierungen durch angegebene Eigenschaften, Fähigkeiten und Wissensaneignung auch auf unvorhergesehene Eintritte reagieren und die Ressourcen immer optimal ausbalancieren. Dies bringt den Vorteil mit sich, dass eine Ressource optimal eingesetzt wird und so etwaige Engpässe durch andere Ressourcen ausgeglichen werden können. Bei einem kleinen Projekt ist dies noch selbst überschaubar und durch einen Dialog mit den jeweiligen Ressourcen heraus zu finden, aber auch hier würde immer die subjektive Beurteilung die Zuordnung dominieren.

Die Vorgehensweise der Ressourcenverwaltung wird im jeweiligen Ressourcen-Konstrukt näher erläutert. Im Rahmen dieses Abschnittes wird sich auf die zu benötigten Daten beschränkt. Die erforderlichen Informationen, um eine flexible Ressourcenzuteilung zu bewerkstelligen, sind die gesammelten Erfahrungen der Ressource mit deren Fähigkeiten und ob die Ressource zum Zeitpunkt der Aufgabentätigkeit zur Verfügung steht.

Gesammelte Erfahrungen: Dies wird durch das PN und den Einsatz strukturierter Marken umgesetzt und muss somit nicht in die Tabelle aufgenommen werden. Hierbei muss die Rückführung der Informationen durch das PN beachtet werden, was in Kapitel 4.5.1 und 4.5.2 aufgegriffen wird.

Fähigkeit: Zur Veranschaulichung der Grundidee wird auf einen vereinfachten Bewertungsmaßstab zurückgegriffen. Dabei werden Fähigkeiten definiert, die sich nach den Bereichen und Gebieten des ID-Schemas richten. Diese werden nach Beurteilungskriterien bewertet und beruhen auf den vom Projektleiter getroffenen Annahmen. Somit wird ein Bezug der Beurteilungskriterien zu der Ressource (Ausgangsdimension) und der Fähigkeit pro Bereich und Gebiet hergestellt. Dabei werden die vier Ausprägungen der Eignung für *ungeeignet (4)*, *mittel (3)*, *gut (2)* und *sehr gut (1)* vorgegeben.

Schnelligkeit: Ebenfalls wäre es denkbar, aufgrund von Erfahrungswerten (basierend ausschließlich auf Ist-Daten) einen Rückschluss auf die Abarbeitung der Schnelligkeit bei Aufgaben vorzunehmen. Auch diese Daten müssten pro Bereich und Gebiet – einheitlich zur Klassifizierung von Aufgaben – vorgenommen werden.

Aufbauend auf diesen Ausprägungen und hinterlegten Regeln bei der Ressourcenzuteilung kann eine optimale Ressourcenzuordnung ermittelt werden. Dabei liegt der Fokus nicht auf der Optimierung der Ressourcenauslastung über das gesamte Projekt, sondern anhand der verfügbaren Ressourcen aus dem Ressourcenpool auf der einzelnen Ressourcenzuordnung. Alles andere würde in den Bereich der Prognose und Netzoptimierung fallen, was in Kapitel 4.5 betrachtet wird. Die erwähnten Regeln könnten beispielsweise lauten, dass immer die freie Ressource zur Abarbeitung der Aufgabe heranzuziehen ist, welche am besten geeignet ist. Unabhängig davon kann dieser Algorithmus vereinfacht werden, indem eine Wunschbelegung getroffen wird und nur im Falle eines Konfliktes oder nicht fixierter Wunschbelegung dieser Algorithmus zum Einsatz kommt.

Verfügbarkeit der Ressource: Die Information, welche Ressource zum Zeitpunkt der anstehenden Aktivität zur Verfügung steht, wird durch das PN übernommen. Dafür wird es einen Ressourcenpool geben und diese Informationen müssen nicht in der Tabelle gepflegt werden.

Benötigte Erweiterungen in der Tabellenstruktur sind die in Abbildung 11 aufgelisteten Spalten für die Tabelle Prognosewerte.

4.4 Modellierung

Der Projektmanager hantiert im Vordergrund des PPNM nur mit Tabellen und es findet eine Überführung in sogenannte Konstrukte statt. Elementar ist, dass der Aufbau und die Schnittstellen der Konstrukte formalisiert und somit modular erweiterbar sein müssen. Die grafische Darstellung und Erzeugung der Simulation der Konstrukte geschieht im Hintergrund. Ein Mehrwert des modularen Aufbaus ergibt sich einerseits aus dem dynamischen Charakter des PPNM, welcher es ermöglicht alle möglichen Systemzustände darzustellen, sowohl andererseits durch die Möglichkeit der wiederholten Durchführung und Simulation.

Im Gegensatz zu den beleuchteten Publikationen wurde bei der Wahl der Abstraktionsebene der Ansatz erweitert. Die Identifikation der Eigenschaften und der Untersuchung der notwendigen Anforderungen an die Tabellenstruktur und benötigten Konstrukte befähigt zur Erstellung von systematisierten Konstrukten. Dies gewährleistet eine Formalisierung der zu verwendenden Abstraktionsebene im PM. Wie auch in der Literatur werden Varianten von Petri-Netzen um zusätzliche Konzepte erweitert. Im Rahmen dieser Arbeit wird speziell auf die Möglichkeiten der Erweiterung mit gefärbten, zeitbewerteten und Prädikat/Transitions-Netzen sowie dem Einsatz strukturierter Marken eingegangen. Weiterhin soll die aufgegriffene Problematik bei [Yen-Liang Chen, Ping-Yu Hsu, und Yuan-Bin Chang 2008] der Abweichung von Plandaten und Ist-Daten ebenfalls näher betrachtet werden. Die grafische Darstellungsmöglichkeit wird in den Publikationen zwar positiv betont, jedoch eher vernachlässigt. Gerade die gewählte Komplexität des vereinten Ansatzes aus klassischen und agilen PM-Methoden erfordert erhöhte Ansprüche an die Darstellung. Infolgedessen wird im Rahmen dieser Arbeit auch bei der grafischen Darstellung eine formalisierte Methode einzelner Projekt-Konstrukte

und dessen Schnittstellen vorgegeben. [vgl. Schott 2019, 19 ff] Die Modellierungsmethode wird anhand der acht Konstrukte Sprint, Aufgabe, Phase, Zeit, Ressourcenkonsum, Ressourcenverwaltung und zwei Ressourcenregenerationen näher erklärt. Bevor es zu der näheren Beschreibung der Konstrukte kommt, wird auf die Abbildung von Bedingungen, Vorgängern und Nachfolgern eingegangen. Die jeweiligen Schnittstellen für die Abbildung der Beziehungen der Konstrukte zueinander werden im jeweiligen Konstrukt thematisiert. Anschließend wird die Besonderheit des Einsatzes strukturierter Marken noch einmal herausgegriffen und näher erläutert. Bevor es dann zur Darstellungsform kommt, wird der Kreislauf des Modells noch einmal dargelegt.

4.4.1 Objektbeziehungen, Bedingungen, Vorgänger und Nachfolger

Ein PN ist ein mathematisches Modell zur Abbildung von Beziehungen und es beschreibt die wechselnde Abfolge von Zuständen und deren Zustandsumformungen. Daraus ergibt sich, dass die zueinander in Beziehung stehenden Objekte auf der gleichen Modellierungsebene notwendig sind. Das Zusammenspiel der Konstrukte findet demzufolge auf der atomaren Ebene, dem kleinstmöglichen Zustandselement, statt. Für dessen Kommunikation stehen Schnittstellen zur Verfügung, über die ebenso die Beziehungen der einzelnen Konstrukte geregelt werden. [vgl. Weichenhain und Fengler 2019a]

Die Modellierungsmethode darf nicht auf spezielle Projektstrukturen eingeschränkt und muss somit erweiterbar, also weitestgehend unabhängig, sein. Dies wird dadurch gewährleistet, dass sowohl klassische als auch agile Projektstrukturen durch die definierten Konstrukte und deren Schnittstellen beliebig kombinierbar und erweiterbar sind. Dabei kann ein Konstrukt für verschiedenste Bereiche eingesetzt werden und ist somit wiederverwendbar. Die Konstrukte sind nach der Belegung von Parametern instanziiert und sind zur Verwendung bereit. [vgl. Zimmermann 1997, 10 ff]

Im weiteren Verlauf der Arbeit wird sich auf die Betrachtung der Box des Sprints aus Kapitel 3.2.4 beschränkt. Der Sprint beinhaltet alle relevanten Beziehungen und benötigt Schnittstellen zu allen anderen Konstrukten. Deswegen wird er als Vertreter des Modells genauer betrachtet. Diese Logik muss entsprechend auf die anderen Bereiche angewendet

werden. Der Fokus wird auf die Elemente des Netzes gelegt, die für dieses Netz interessant sind.

Umsetzung der Beziehungen im PN mit Kanten und Transitionen

Wie schon beschrieben, stehen die Konstrukte zueinander in Beziehung. Zur Abbildung dieser Beziehungen werden einerseits Schnittstellenplätze zwischen den Konstrukten eingesetzt. Andererseits werden über das Element der Kante die Bedingungen, Vorgänger und Nachfolger abgebildet. Vorkanten bilden die Vorbedingungen ab. Im Gegensatz dazu stehen die Nachbedingungen, welche zur Abbildung der Ergebnisse benutzt werden. Dazu sind die Informationen zu den einzelnen Objekten in Abhängigkeit mit den auszuführenden Aktivitäten, welche im Verlauf des Projektes durchlaufen werden, zu speichern. Das wird mittels des Konzepts der strukturierten Marken bewerkstelligt. Diese werden mit den notwendigen Informationen angereichert, um die Bedingungen an den Kanten prüfen zu können. Nachbedingungen werden durch Nachkanten abgebildet. Auf den Einsatz von Sonderkanten wird in den entsprechenden Konstrukten im Kapitel 4.4.2 näher eingegangen.

Zum Einsatz kommen Zeitdauertransitionen. Diese sind dadurch definiert, dass sie eine Verzögerungszeit innehaben und somit die Schaltung für einen Takt des Netzes (in der Arbeit als eine Stunde definiert) mit Verspätung ausführen. Dies hat ebenfalls zur Folge, dass es keine gleichzeitige Schaltung von einer Vorbedingung und Nachbedingung gibt.

4.4.2 Konstrukte und deren Schnittstellen

Es werden acht Konstrukte definiert und in Abbildung 12 ist verdeutlicht, wie diese miteinander in Beziehung stehen. Dabei stellt jedes Konstrukt mindestens eine Schnittstelle bereit. Es ist zu erkennen, dass das Zeit-Konstrukt elementar ist und mit vielen Konstrukten die zeitliche Ebene synchronisiert. Die drei grau hinterlegten Konstrukte Zeit, Ressourcenkonsum und Ressourcenverwaltung sind nur einmal vorzufinden und werden nicht vervielfacht. [vgl. Weichenhain, Fengler, und Streitferdt 2019]

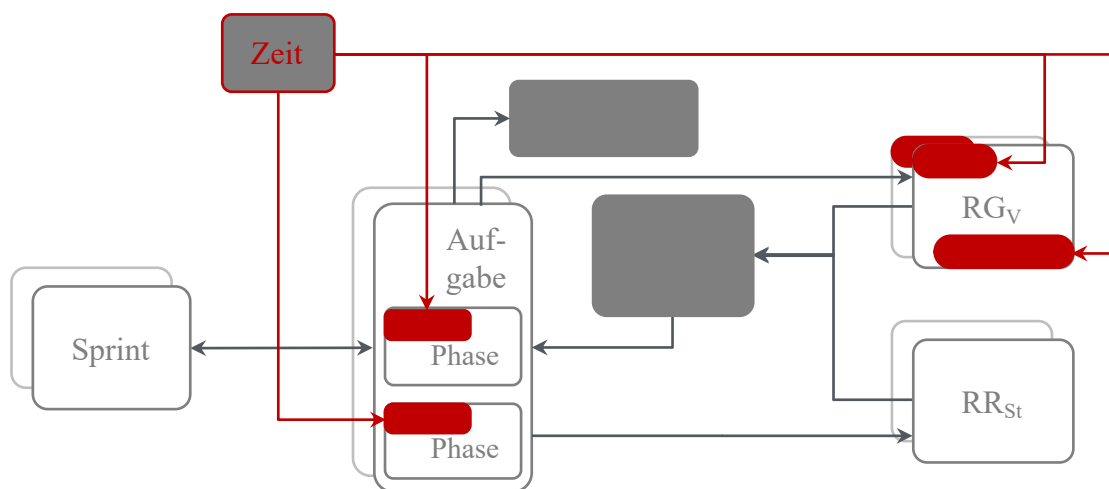


Abbildung 12: Konstrukte und deren Beziehungen

Zeit-Konstrukt

Die Zeit ist eine wichtige Komponente im PPNM. Dabei werden die zwei Zeitrestriktionen mittels zwei Datumsangaben und unter der Angabe einer Dauer zum Einsatz kommen. Ebenfalls kommt die Testkante in diesem zur Anwendung, worauf später noch eingegangen wird.

Als Alternative zu den zeitbewerteten PN in Kapitel 2.3.2 wird das globale Zeitkonzept angewendet. Dies hat zum Ziel, den aktuellen Zeitpunkt innerhalb des PN abzubilden. Damit wird eine vereinfachte Überprüfung von Fristen und zeitlichen Abschnitten sowie eine bessere Einordnung des jeweiligen Simulationsschritts ermöglicht. Auch vereinfacht es die Interpretation des PNs in Hinblick auf die eingetragenen Datumsangaben in den jeweiligen Tabellen. Die Handhabung von angegebenen Fristen im Sinne vom frühesten Startzeitpunkt und spätestem Endzeitpunkt kann mit diesem Ansatz ebenfalls abgebildet werden. [vgl. Schott 2019, 32 f; vgl. Grolleau und Choquet-Geniet 2000, 91 ff; vgl. Hoppe 2014, 34 ff]

Wegen der Einheitlichkeit und Reduzierung der Komplexität wird keine Vermischung von zeitbewerteten Komponenten mit dem hier vorgestellten globalen Zeitkonzept erfolgen. Dies wäre theoretisch möglich, bringt aber keinerlei Vorteile mit sich. Die Modellierung der Dauer zur Bearbeitung einer Aufgabe wird über die Markenanzahl vorgenommen und im Konstrukt der Aufgabe sowie der Phase integriert.

Die Zeit wird durch ein globales Zeitkonzept in Kombination mit einer lokalen Uhr umgesetzt. Die globale Uhr gibt den Zeittakt vor. Durch diesen Zeittakt können alle Transitionen zu synchronen Zeitpunkten betrachtet werden. Die Modellzeit definiert die zugrundeliegende Zeiteinheit (Tag, Stunde, Minute) pro Zeitmarke. Die gewählte Modellzeit gibt das kleinste Vielfache der Angabe einer Dauer vor und die Grundlage der Berechnung im Platz *Zeitpunkt* in der jeweiligen lokalen Zeit. Alle Angaben müssen demzufolge ein Vielfaches der gewählten Zeiteinheit entsprechen, es sind keine negativen Zahlen zulässig. Zur korrekten Abbildung von Datumsangaben müssen die Information über die Startzeit des Modells und einer Modellzeit angegeben werden.

Die lokale Uhr wird pro Konstrukt-Instanz, angestoßen von der globalen Uhr, synchronisiert. Das bringt den Vorteil mit sich, dass eine Variierbarkeit der Uhr auf der jeweiligen Instanz möglich wäre. Das globale Zeitkonzept umfasst einen Zeitgenerator (*Zeittransition*) und einen *Zeitplatz*. Das Zeit-Konstrukt wird wie in Abbildung 13 dargestellt. Zu jedem Takt wird durch die *Zeittransition* eine neue Zeitmarke erzeugt und für die lokale Zeit (*Zeitbestimmung* und *Arbeitszeitbestimmung*) im jeweiligen Konstrukt durch das Interface des *Zeitplatzes* in der lokalen Uhr berücksichtigt. Das maximale Schalten und die getaktete Transition stellen sicher, dass die *Zeittransition* pro Takt nur einmal schalten kann.

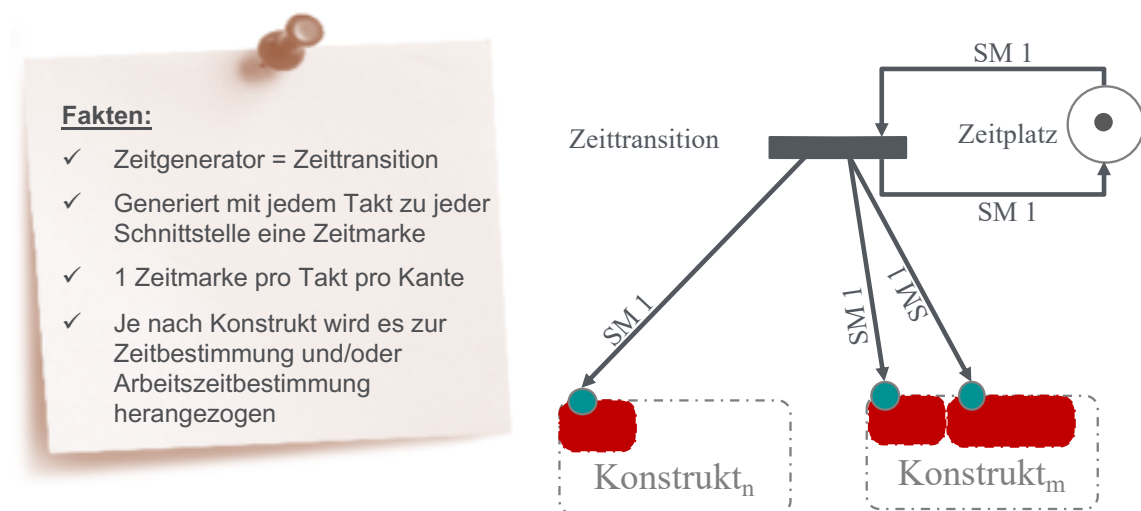
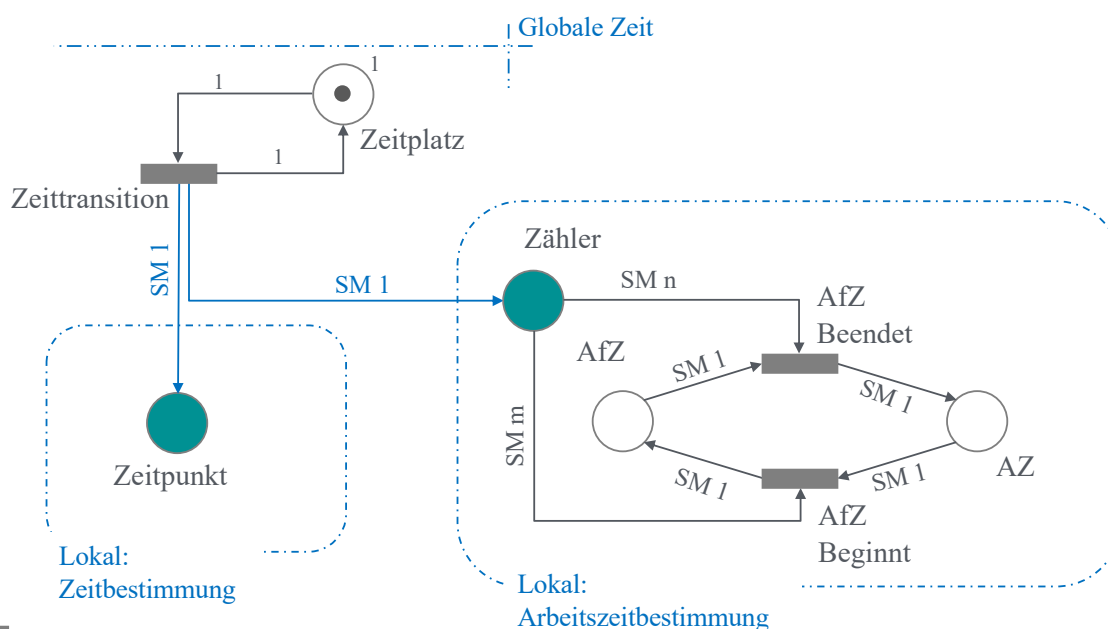


Abbildung 13: Modellierung globales Zeitkonzept - Zeit-Konstrukt

Um Fehler zu vermeiden, darf die Anzahl der Zeitmarken nur durch die *Zeittransition* beeinflusst werden. Deswegen darf ein Zugriff auf den *Zeitplatz* (zur Validierung von angegebenen Datumsangaben) nur mit einer Testkante erfolgen. Dies stellt sicher, dass die Markenanzahl nicht durch andere Transitionen beeinflusst werden kann. Die Anzahl an Zeitmarken kann somit im Laufe des Projektes nur streng monoton zunehmen und nie abnehmen. Bei der lokalen Arbeitszeitbestimmung erfolgt eine Unterteilung in Arbeitszeit (AZ) und arbeitsfreie Zeit (AfZ) und kann je nach Konstrukt durch die Belegung der Variablen unterschiedlich definiert werden. Die nachfolgende Abbildung zeigt das Prinzip der Modellierung der lokalen Zeitbestimmung mit allen benötigten Komponenten:

Bei der Bestimmung der Arbeitszeit sind die Kantenbeschriftungen der Kante SM_n (*Zähler, AfZ Beendet*) und Kante SM_m (*Zähler, AfZ Beginnt*) ausschlaggebend. Bei dieser Abbildung wird die Anzahl an Arbeitseinheiten pro Tag gesteuert. Nicht geleistete Arbeitseinheiten können auch während der AfZ abgearbeitet werden, sobald einer Ressource am Ende der AfZ noch Arbeitseinheiten zur Verfügung stehen. Durch eine geeignete Wahl von Vielfachheiten erfolgt die Unterteilung.



► Die Anbindung zur globalen Zeit ist in **Blau** hervorgehoben

Abbildung 14: Lokale Zeit-Konstrukt

Das Setzen der korrekten Anfangsmarkierung ist hierbei ausschlaggebend. Wie vorher erwähnt, muss die Anfangsmarkierung des Zeit-Konstruktes passend zum gewählten Startdatum erzeugt werden und wird am Beispiel der 40-Stundenwoche aufgezeigt. Ausschlaggebend hierfür ist der Wochentag des Startdatums und wird analog der Angaben aus Tabelle 3 erzeugt. Die Markierung für den Folgetag ergibt sich mit #Marken Vortag + 8.

Tabelle 3: Anfangsmarkierung Zeit-Konstrukt

	Zähler	AZ	AfZ
Montag	0	1	0
Dienstag	8	1	0
Mittwoch	16	1	0
Donnerstag	24	1	0
Freitag	32	1	0
Samstag	0	0	1
Sonntag	8	0	1

Sprint-Konstrukt

Das Sprint-Konstrukt berücksichtigt alle Anforderungen zur Abbildung eines agilen Sprints nach SCRUM und bündelt die ihm zugeordneten Aufgaben. Sie verlaufen immer nur sequentiell und nie parallel und besitzen deswegen nur vereinfachte Vorgänger-/Nachfolgeranforderungen. Der erste Sprint wird vom Projektstart her initiiert. Die Verkettung der Sprint-Konstrukte ist mit zwei Schnittstellen sehr einfach gehalten und bildet die Anforderungen ab.

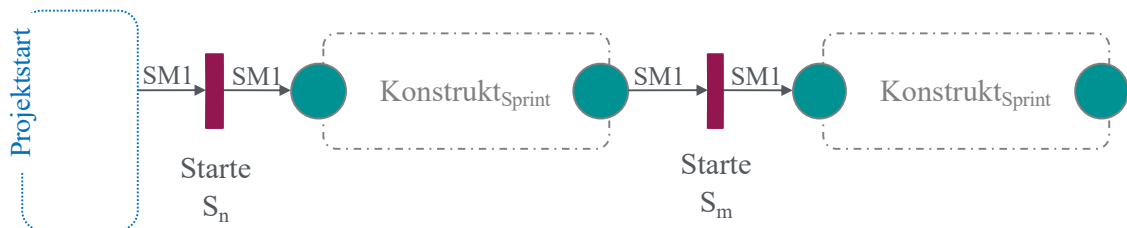


Abbildung 15: Verkettung von Sprint-Konstrukten

Im Sprint-Konstrukt selbst erfolgt ein Zugriff auf die Schnittstellen des Aufgaben-Konstruktes. Das Sprint-Konstrukt stellt nur den Rahmen der Aufgabengruppierung dar und somit ist keine Schnittstelle des lokalen Zeit-Konstruktes inkludiert. Die Zuordnung von Ressourcen und die eigentliche Abarbeitung von Aktivitäten findet nur auf der Ebene der Arbeitspakete und Aufgaben statt. In einem Sprint-Konstrukt können mehrere Aufgaben abgearbeitet werden. Zwei Interfaces stehen pro Sprint zur Verfügung: Platz *Start Sprint* und Platz *Sprint Beendet*. In Abbildung 16 ist ebenfalls ersichtlich, dass für die Koordination der einzelnen Aufgaben ein sogenannter *Koordinatorplatz* verwendet wird, dessen einziges Ziel die Initiierung der Aufgaben ist. Alle Transitionen zum Initiieren der Aufgaben sind somit schaltfähig und leiten deren Abarbeitung ein. Für eine korrekte Initiierung muss jede Aufgabe wiederum ein Interface bereitstellen, welches dieses Signal verarbeitet. Die Transitionen zur Initiierung der Aufgabe wird durch keine Vor- oder Nachbedingungen gehemmt und das maximale Schalten wird vorausgesetzt. Dies stellt sicher, dass keine Aufgabe mehrmals gestartet und somit keiner anderen Aufgabe die Startgrundlage entzogen wird. Reihenfolge und Abhängigkeiten der Aufgaben untereinander werden im Aufgaben-Konstrukt geregelt. Nachdem der Koordinator die einzelnen Aufgaben dazu angestoßen hat, die Bearbeitung zu beginnen, können diese parallel oder sequentiell entsprechend des entwickelten Konstruktes abgearbeitet werden. Aufgrund von unterschiedlichen Bearbeitungszeiten, Startdaten oder betrieblich bedingten Verzögerungen (Mitarbeiterzuordnung, Urlaub, etc.), werden die einzelnen Aufgaben zu unterschiedlichen Zeitpunkten beendet. Ein Sprint wiederum ist nur beendet, wenn alle seine Teilaufgaben fertiggestellt wurden. Daraus ergibt sich, dass die einzelnen Aufgaben zur Beendigung eines Sprints wieder zusammengeführt werden müssen. Die Zusammenführung der einzelnen Aufgaben wird durch die Transition *Beende Sprint* realisiert. Diese Zusammenführung mittels einer zusätzlichen Transition ist notwendig, weil für eine weitere Bearbeitung nach einem Sprint nur eine Marke benötigt wird. Die Transition *Beende Sprint* nimmt diese Reduktion auf eine Marke, unabhängig von der Anzahl zugeordneter Aufgaben, vor. [vgl. Weichenhain und Fengler 2019a]

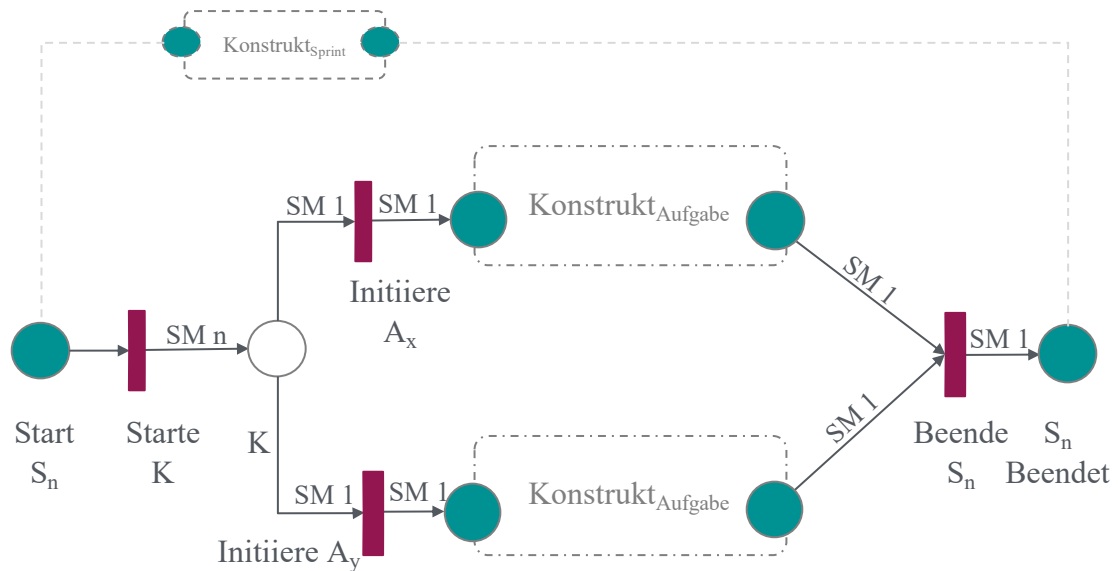


Abbildung 16: Das Sprint-Konstrukt

Alle Transitionen im Sprint-Konstrukt sind zeitlos. Eine Alternative wäre, sowohl das Starten als auch das Beenden eines Sprints mit einer getakteten Transition zu versehen. Diese würde je einen Schalttakt von einer Stunde benötigen und könnte somit den Organisationsaufwand zur Koordination des Verteilens der Aufgaben sowie mit dem Zusammenführen der Ergebnisse verglichen werden. Damit hier aber die Projektplanung nicht verfälscht wird, müsste ebenfalls die Ressourcenzuordnung dafür vorgenommen werden. Sollte dieser organisatorische Aufwand im Netz ebenfalls abgebildet werden, wird empfohlen, hierfür eigens eine Aufgabe mit nur einer Phase zu definieren. Ein Sprint ist durch die modulare Abbildung um beliebig viele Aufgaben erweiterbar. Diese Erweiterung umfasst die Erstellung der Initialisierungstransition für eine neue Aufgabe und dessen Anbindung an den Koordinator, sowohl auch die Erhöhung der Vielfachheit der Nachkante von *Start Sprint* und die Verknüpfung des Aufgabenendes mit der Transition *Sprint Beendet*. [vgl. Weichenhain und Fengler 2019a]

Aufgaben-Konstrukt

Aus den Arbeitspaketen der Building Blocks entstehen die Aufgaben und werden zur Abarbeitung einem Sprint zugeordnet. Eine Aufgabe besteht aus mehreren Phasen, welche sequentiell ablaufen. Es können beliebig viele Phasen in einer Aufgabe nacheinander erfolgen. Das zugrundeliegende Beispiel dieser Arbeit unterscheidet zwei Phasen. Dabei folgt die **Phase der Entwicklung** immer auf die **Phase des Designs**.

Die Zuordnung von Ressourcen und die eigentliche Abarbeitung von Aktivitäten findet innerhalb der Phasen statt. Das Aufgaben-Konstrukt fungiert wie eine Hülle und koordiniert die notwendigen Informationen zum Starten und Beenden einer Phase, was auch die zeitliche Kontrolle anhand der zugewiesenen Datumsangaben pro Phase beinhaltet. Das Aufgaben-Konstrukt stellt Interfaces pro Aufgabe zur Verfügung: Platz *Initiiere Aufgabe* und Platz *Beende Aufgabe* der letzten Phase.

Alle Aufgaben werden immer parallel pro Sprint initiiert. Das Ende einer Aufgabe ist die Startbedingung einer anderen Aufgabe. Für die Realisierung der Vorgänger-/Nachfolgerbeziehung muss das Konzept zusätzlich zu der Testkante noch um eine farbige Marke erweitert werden, wie in Abbildung 17 ersichtlich.

Die schwarze Marke wird zur Beendigung der Phase oder des Sprints weiter verschaltet und ist deshalb nicht geeignet, um als Grundlage für die Vorgänger-/Nachfolgerbeziehungen herangezogen zu werden. Damit es hier nicht zu einer doppelten Schaltfähigkeit dieser Bedingung kommen kann, wird zur schwarzen Marke auch eine zusätzliche farbige Marke zur Beendigung einer Phase erzeugt. Hier wird das Konzept von farbigen Marken und nicht von strukturierten Marken herangezogen. Der Grund dafür ist, dass keine Informationen verbunden werden müssen und das Konstrukt einfach gehalten wird. Durch diese farbige Marke wird potentiellen Nachfolgern angezeigt, dass die Phase oder Aufgabe beendet ist. Damit eine Aufgabe beliebig viele Nachfolger haben kann, wird eine Testkante anstelle einer Vorkante verwendet.

Die Ressourcenzuordnung geschieht im Konstrukt der Phase. Durch das Konstrukt der Aufgabe wird die benötigte Ressourcenanforderung pro Ressourcenart (Stück oder Vielfaches) vor dem Start der Phase durch den Einsatz strukturierter Marken mitgegeben ($StM_{RA-R-XX}$). Auf die strukturierte Marke wird später als Ganzes eingegangen.

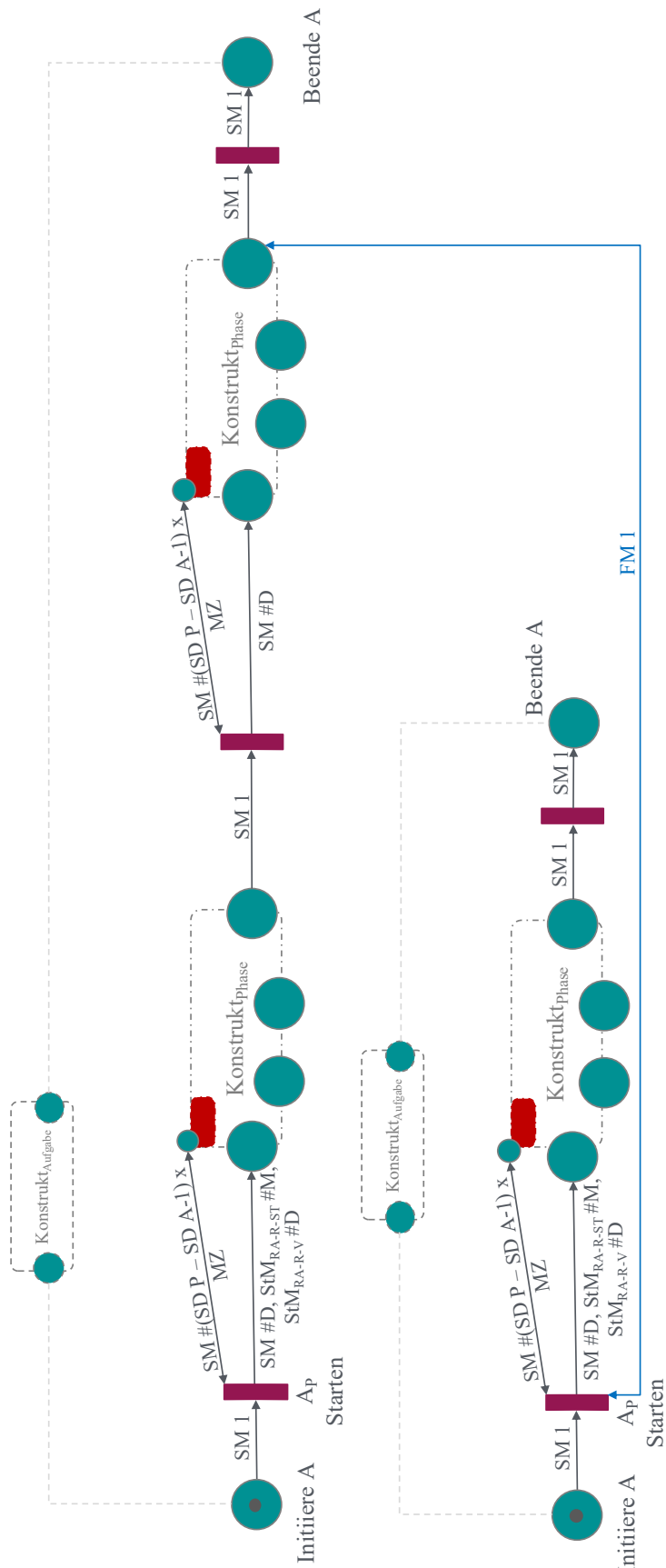


Abbildung 17: Aufbau Aufgaben-Konstrukt

Phasen-Konstrukt

Wie schon in Kapitel 3.2.4 dargestellt, ist der generelle Ablauf dieser zwei Phasen identisch und erfolgt strikt hintereinander und die aus ihnen gestellten Anforderungen sind ebenfalls identisch. Zur Vereinfachung wird auf diese weitere Unterteilung verzichtet und nur eine Phase betrachtet.

Die Abarbeitung der Phasen einer Aufgabe gliedert sich in die Ressourcenzuordnung, Angabe der zeitlichen Dauer und die eigentliche Bearbeitung. Die Ressourcenzuordnung wird im Ressourcen-Konstrukt thematisiert. Wie in Abbildung 18 ersichtlich, stehen fünf Schnittstellen pro Aufgabenphase zur Verfügung: Platz *Unerledigt*, *Abarbeitung*, *Fortschritt*, *Zeitpunkt* und Platz *Beendet*. Bevor die Abarbeitung der Aufgabe startet muss sichergestellt werden, dass alle Ressourcen zur Abarbeitung vorhanden sind. Dabei wird zwischen den Ressourcen als Stückzahl und der der Vielfachheit unterschieden. [vgl. Weichenhain und Fengler 2019a]

Die Schnittstelle zum lokalen Konstrukt *Zeitpunkt* stellt sicher, dass die Aufgabe frühestens zum angegebenen Startdatum beginnen kann. Dabei ist wichtig, dass drei Zeitangaben (Dauer, frühester Start und spätestes Ende) berücksichtigt werden. Somit ist ein späterer Start ebenfalls möglich, da es unter Einbeziehung der angegebenen Dauer eine Überprüfung der Frist zur Fertigstellung gibt. Ist dies nicht der Fall, wird die Überziehungsdauer gemessen. Auch ist keine stetige Abarbeitung verpflichtend, dazu wird der Fortschritt dediziert festgehalten.

In Abbildung 18 ist auch der Einsatz von Test- und Inhibitorkanten beispielhaft ersichtlich. Die Transition *Überzogen* zur Dokumentation der Überziehung ist durch den Einsatz einer Inhibitorkante an den Platz *Beendet* nicht mehr schaltfähig, sobald die Phase beendet ist. Der Erfolg oder Fehlschlag einer Phase ist an die zeitliche Dimension gebunden. Die Fortschrittsangabe in den Tabellen beeinflusst, ob eine Aufgabe beendet ist. Sobald die Fortschrittsangabe bei 100 % ist, wird die benötigte Ist-Dauer für die Anzahl der Marken herangezogen und eine gegebenenfalls höhere Plandauer nicht mehr berücksichtigt.

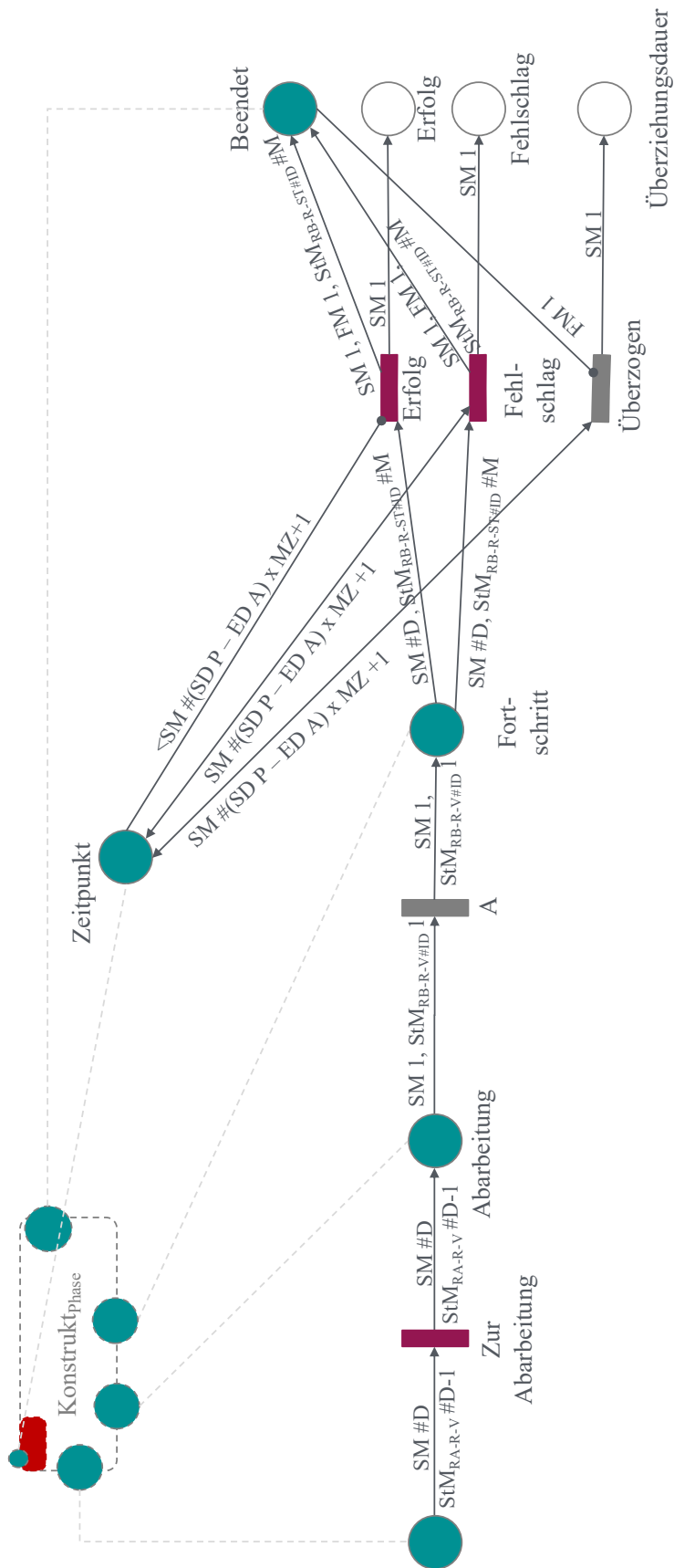


Abbildung 18: Aufbau Phasen-Konstrukt

Ressourcen-Konstrukte

Mit dem hier vorgestellten PN können einer Aufgabe beliebig viele Ressourcen durch die zur Verfügung gestellten Schnittstellen zugeteilt werden. Nach der Zuteilung kann sich eine Ressource regenerieren, wieder in den Ressourcenpool aufgenommen oder zur späteren Dokumentation konsumiert werden.

Zur Darstellung dieses Sachverhaltes kommen vier Konstrukte zum Einsatz: Die **Ressourcenverwaltung**, der **Ressourcenkonsum**, die **Ressourcenregeneration Stück** und die **Ressourcengeneration Vielfachheit**.

Das Zusammenspiel ist in Abbildung 19 dargestellt und wird einzeln näher erläutert. Abgesehen vom Ressourcenkonsum ist ein konstanter Austausch zwischen den Konstrukten der Phasen und der Ressourcen gut ersichtlich. Die Kapazität wird im Modell ersetzt durch die Vorgehensweise der Ersetzung von kapazitätsbeschränkten Plätzen, wie beschrieben in Kapitel 4.2.

Das **Ressourcenverwaltungs-Konstrukt** stellt fünf Schnittstellen bereit. Eine zur Aufnahme der Ressourcenanforderungen, zwei für die Ressourcenbereitstellung, eine zur Aktualisierung der Ressourcenerfahrungen und die letzte für den Ressourcenpool. Die anderen drei Ressourcen-Konstrukte (**Ressourcenkonsum**, **Ressourcenregeneration Stück** und **Ressourcengeneration Vielfachheit**) stellen jeweils nur eine oder vier Schnittstellen zur Verfügung.

Die **Ressourcenanfrage (RA)** erfolgt für alle Ressourceneinheiten in Stück vor der eigentlichen Abarbeitung der Aufgabe. Zudem sind die Ressourcen mit der Aufgabe bis zu deren Beendigung gebunden. Bei der Ressourceneinheit der Vielfachheit muss nur die erste Einheit beim Start der Phase zur Verfügung stehen. Alle weiteren Einheiten werden mit dem jeweiligen Abarbeitungsschritt angefragt. Deswegen ist die Ressourcenanfrage auf drei Transitionen aufgeteilt. Ein Unterschied dabei ist, dass bei der ersten Anfrage alle Transitionen zeitlos sind. Dies gewährleistet, dass für die Anfrage kein weiterer Takt erfolgen muss, bevor die Abarbeitung beginnen kann. Zeitgleich mit einem Abarbeitungsschritt wird auch die nächste Ressourceneinheit des Vielfachen angefragt. Dies muss getaktet erfolgen, damit nicht alle Ressourcen auf einmal angefragt und so bis zum Beenden der Phase blockiert sind. Aus dem gleichen Grund erfolgt die **Ressourcenbereitstellung (RB)** auf zwei Plätze (*Abarbeitung, Fortschritt*) und wird je Ressourceneinheit aufgeteilt. Hier erfolgt die Annahme, dass unabhängig vom Abarbeitungsschritt alle Ressourcen mit der Einheit Stück immer benutzt werden können.

Bei der Wiederverwendung spielt die Ressourceneinheit ebenfalls eine Rolle. So erfolgt bei der Ressourceneinheit Stück die **Ressourcenregenerierung** erst nach der Beendigung der Phase, was ebenfalls für den **Ressourcenkonsum** gilt. Die Einheit Vielfachheit hingegen ist von keiner Ressourcenregeneration betroffen. Hier ist die Annahme, dass eine Ressource mit der Einheit Vielfachheit abhängig vom zugrundeliegenden Zeitkonzept durch die **Ressourcengeneration** neu erstellt und eine einmal beanspruchte Einheit lediglich konsumiert wird.

Der zentrale **Ressourcenpool** bündelt alle Ressourcenanfragen und Ressourcenverfügbarkeiten. Falls bei einer Ressourcenanfrage keine Ressourcenzuteilung möglich ist, wird nach Ablauf der definierten Bearbeitungszeit einer Phase eine **Konfliktsituation** angezeigt. Hier wird zur Vereinfachung angenommen, dass diese einen manuellen Eingriff benötigt und eine Managemententscheidung gefällt werden muss. Dafür muss ein Mechanismus gewährleistet werden, welcher die Kontrolle der Simulation zurück an den Simulierenden gibt und eine benötigte Entscheidung anzeigt. Alternativ könnten Debuggingplätze, die eine wartende Ressourcenanforderung als Ausnahmeüberwachung kontrollieren, dienen. Hierbei würde wie in Abbildung 20 ein *Debuggingplatz* als Aufnahmestelle für eingetroffene Ressourcenanforderungen über eine zeitverzögerte Transition, welche das Abbruchkriterium definiert, eine fehlerhafte

oder verzögerte Ressourcenzuordnung signalisieren. Das Abbruchkriterium wird durch eine zeitverzögerte Schaltung definiert, welche erst nach einer ununterbrochenen Schaltfähigkeit von n schaltet. Beispielsweise würde dies bei $n=10$ bedeuten, dass eine nicht bediente Ressourcenanfrage nach zehn ununterbrochenen Zeiteinheiten (Takten) signalisiert wird. In Abbildung 20 ist dies am Beispiel der Ressource des Typs Stück aufgezeigt.

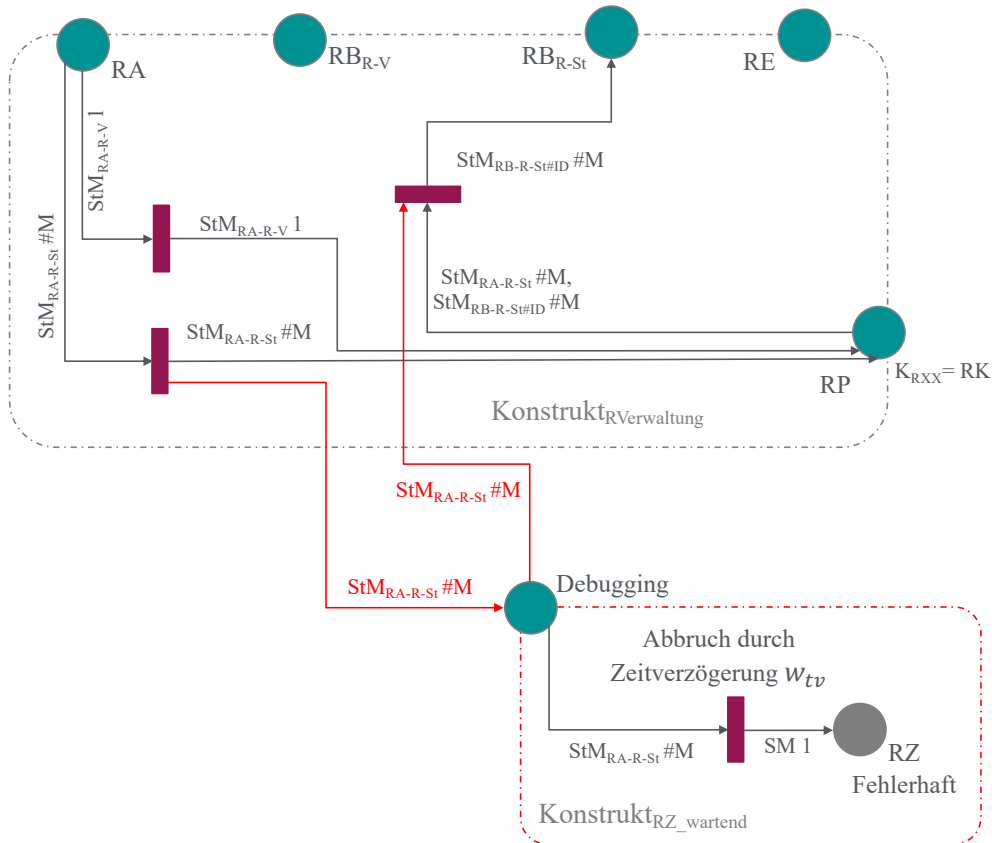


Abbildung 20: Fehlerhafte Ressourcenzuordnung

Beim Ressourcenpool ist die Anfangsmarkierung hervorzuheben. Alle Ressourceneinheiten von Stück werden als Marken für die Anfangsmarkierung generiert. Alle Ressourcen der Einheit Vielfachheit werden durch das Ressourcengenerations-Konstrukt der Vielfachheit und dessen Anfangsmarkierung bestimmt. Bei der **Ressourcenmodifikation** steht im Vordergrund, dass die gemachten Erfahrungen in der strukturierten Marke mitgeführt werden. Dabei spielen zwei Schritte im Zusammenspiel der einzelnen Ressourcen-Konstrukte eine große Rolle. Zum einen das Modifizieren von Ressourceneinheiten, nachdem diese einen Abarbeitungsschritt erfolgreich beendet haben. Der aktuelle Erfahrungswertschatz wird im Platz *Lernen/Modifizieren erhöht*

gespeichert, wobei zur Vereinfachung Misserfolge und Verringerungen von Erfahrungen außer Acht gelassen werden. Der zweite wichtige Schritt ist der Platz der *Ressourcenerfahrung*, welcher immer jeweils den aktuellen Erfahrungsstand pro Ressource aufnimmt und im Ressourcenpool alle vorhandenen strukturierten Marken aktualisiert. Dabei kommt eine Setzkante zum Einsatz, damit der Platz von jeder Ressource nur eine strukturierte Marke beherbergen kann. Bei beiden Plätzen muss die Anfangsmarkierung mit den gegebenen Erfahrungswerten gewährleistet werden.

Nachfolgend soll noch auf zwei Besonderheiten eingegangen werden. Die Art, wie Ressourcenanfragen in strukturierten Marken abgebildet werden und die Ressourcengeneration der Vielfachheit.

Ressourcengenerations-Konstrukt

Ausgangsbasis ist die Modellzeit in Stunden und die Annahme, dass ein Arbeitstag acht Stunden umfasst. Sollte diese Annahme geändert werden, müssen die entsprechenden Variablen im lokalen Zeit-Konstrukt zur Arbeitszeitbestimmung sowie die Interpretation der Marken im Platz *Zeitpunkt* angepasst werden. Dieser Abschnitt bezieht sich auf das Zusammenspiel mit der Ressourcengeneration, wie in Abbildung 21 ersichtlich. Das lokale Konstrukt der Zeit und die Bestimmung der Arbeitszeit wurde im Rahmen dieses Kapitels unter dem Punkt Zeit-Konstrukt bereits erläutert.

Das Konstrukt der Ressourcengeneration muss mindestens einmal im PN pro Ressource mit einer Vielfachheit abgebildet sein, unabhängig davon, ob eine Generationsphase für diese Ressource definiert wurde oder nicht. Dabei wäre der Platz *Regenerationsphase geplant* mit keiner Marke belegt. Hier gibt es einen Unterschied in der Begrifflichkeit. Die Ressourcen mit der Einheit Vielfachheit können Regenerationsphasen zugeordnet werden, um arbeitsfreie Zeit zu modellieren. Demzufolge wird in dieser Art von Regeneration die Generation dieser Einheiten gestoppt. Wobei bei den Ressourcen mit der Einheit Stück eine Regeneration damit verbunden wird, dass sie zum erneuten Einsatz bereitgestellt wird.

Eine Unterbrechung der Generation wird durch einen Zeitrahmen angegeben und muss deswegen eine Abfrage zum Zeitpunkt enthalten. Beim Start der Regenerationsphase kommt eine Setzkante zum Einsatz, welche in der Abbildung blau hervorgehoben wurde. Diese bezweckt, dass noch vorhandene Marken auf null gesetzt werden, da ein Einsatz dieser Ressource ab dem Start der Regenerationsphase nicht mehr möglich ist. Ebenfalls wird während der Regenerationsphase die Transition zur Generierung neuer Marken durch eine Inhibitorikante gehemmt.

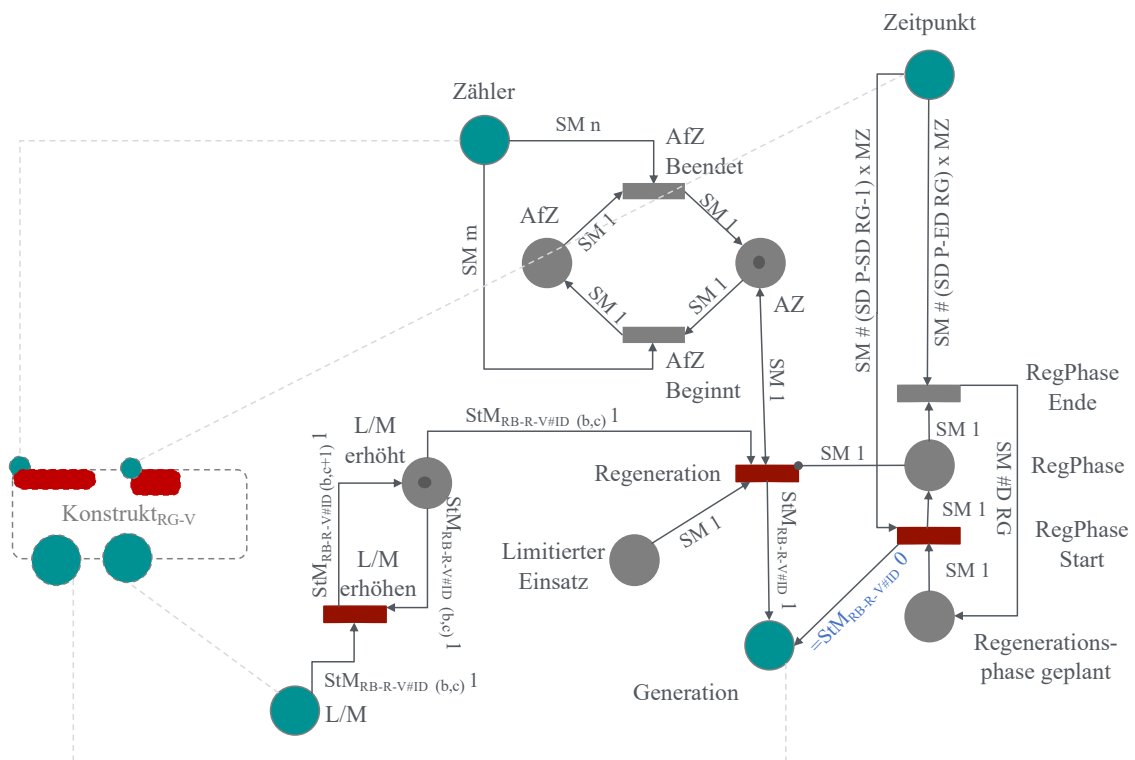


Abbildung 21: Aufbau Ressourcengeneration mit Vielfachheit

Die Modellierung zur Dauer eines limitierten Einsatzes wird über die Markenanzahl in dem Platz *Limitierter Einsatz* vorgenommen (z. B. Budget für externe Berater ist auf 50h limitiert). Bestehen keine Beschränkungen, wird die Markenanzahl auf die Laufzeit des Gesamtprojektes gesetzt.

Bei der Abbildung der Kapazität einer Ressource im Ressourcenpool ist zu erkennen, dass eine individuelle Begrenzung der Arbeitszeit durch die Kapazitätsangabe beim Platz *Ressourcenpool* realisiert ist. Zur Vereinfachung wird das Beispiel der menschlichen Ressource mit einer Begrenzung der Arbeitszeit auf sechs Stunden herangezogen. Hier

würde die Kapazitätsangabe $K=6$ lauten. Der Grund warum diese Kapazitätsbegrenzung nicht durch die Steuerung der arbeitsfreien Zeit realisiert wird ist, dass auch bei Teilzeit die reduzierte Arbeitszeit während eines Tages verteilt werden kann und eine Gleitzeitregelung vorausgesetzt wird. Durch die Kapazitätsbeschränkung des Platzes wird sichergestellt, dass bei einer Modellzeit von Stunden und der Annahme, dass ein Tag durch acht Marken im Netz symbolisiert wird, auch Teilzeitarbeitsmodelle abgebildet werden können. Zusätzlich wird durch die lokale Uhr gewährleistet, dass eine Ressource seine definierten Arbeitsstunden (z. B. 40) nicht in drei Tagen abarbeitet hat, falls die Definition der Stunden pro Tag geändert wird. Nicht geleistete Arbeitseinheiten können auch am Wochenende abgearbeitet werden. Sobald eine Ressource am Ende der Woche noch unverbrauchte Arbeitseinheiten besitzt, stehen diese auch am Wochenende zur Abarbeitung von Aufgaben zur Verfügung. Dies beruht auf der Anforderung, die Arbeitszeit flexibel einzuteilen. Deswegen wird nur die Generierung der Marken durch eine Regenerationsphase gestoppt und nicht schon generierte Marken im Ressourcenpool. Eine Mehrarbeit vom definierten Arbeitstag wird als Überstunde angesehen. Dabei wird ebenfalls unterbunden, dass diese für die nächste Arbeitswoche aufsummiert werden, weil ein Arbeitstag acht Marken nicht überschreiten darf. Die Modellierung von Überstunden wurde dahingehend berücksichtigt, dass eine zweite Ressource für den gleichen Mitarbeiter angelegt werden müsste. Diese Ressource müsste zudem als Kapazitätsangabe die bewilligten Überstunden pro Arbeitstag mitgeliefert bekommen. Die Regenerationsphase steuert den Zeitraum der bewilligten Überstunden, verbunden mit einer Möglichkeit durch den limitierten Einsatz auch die maximale Anzahl an Überstunden zu begrenzen. Dies muss bei der Ressourcenanforderung ebenfalls berücksichtigt werden und resultiert in einer zweiten Anfrage der Überstunden durch die zweite generierte Ressource.

Durch die lokale Einbindung der Arbeitszeit als lokales Zeit-Konstrukt könnte für jede Ressource ein individuelles Arbeitszeitmodell hinsichtlich der Einteilung der Zeit festgelegt werden. Die Modifizierung und das Lernen von Ressourcen mit Vielfachheiten, ist ebenfalls in Abbildung 22 durch die Transition (*Platz Fortschritt*, *Platz Generation*) dargestellt.

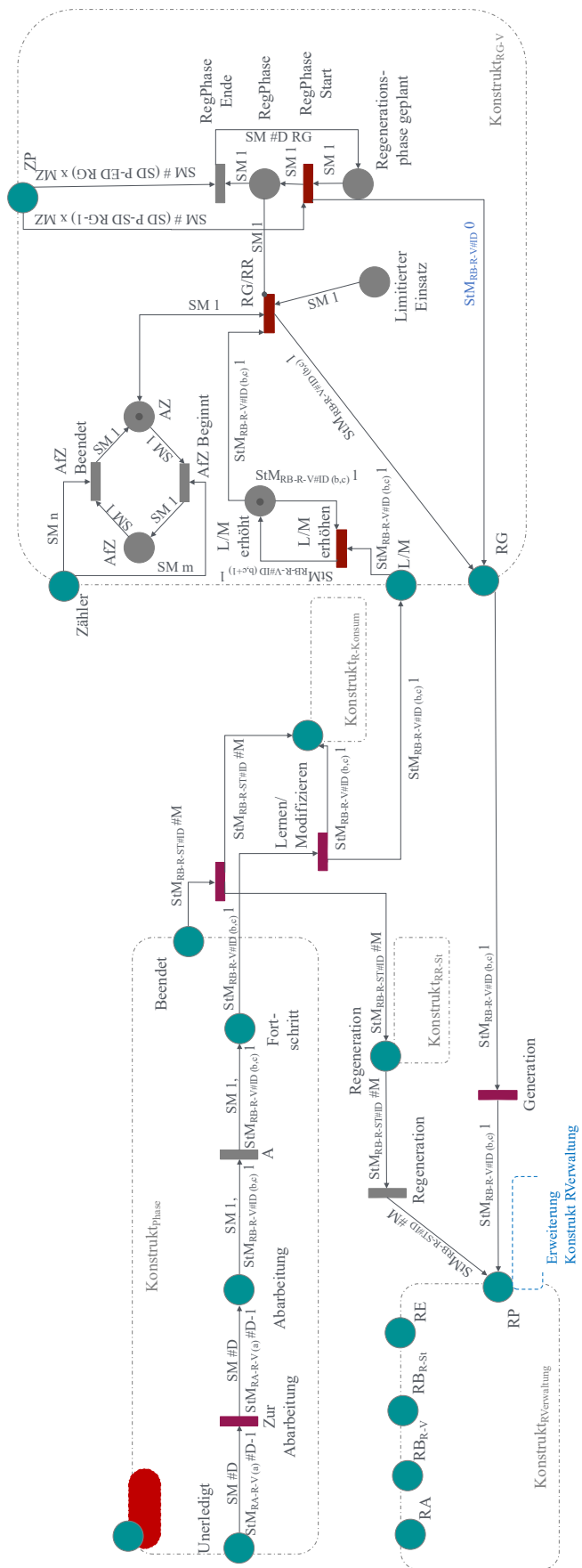


Abbildung 22: Aufbau Ressourcengeneration mit Schnittstellen

4.4.3 Anpassbarkeit, Erweiterbarkeit und Nachweis

Dieser Abschnitt soll als kurzer Überblick auf die Anpassbarkeit des Petri-Netzes an sich verändernde Projektdaten eingehen sowie an die Erweiterbarkeit des Modells der modularen Konstrukte an neue Anforderungen. Die folgenden Erkenntnisse basieren auf der nach den Vorgaben des Autors erstellten und betreuten Masterarbeit von [Schott 2019], welche ein vereinfachtes Beispiel auf der Grundlage der Erweiterbarkeit und Objektorientierung umsetzt.

Durch die Aufnahme von Ist-Daten wurden feste Eingriffspunkte definiert, was eine Berücksichtigung von Abweichungen der Plandaten ermöglicht. Diese neuen Ist-Daten werden durch geeignete Netz-Modelle entsprechend berücksichtigt. Die modulare Struktur der Konstrukte, welche zur Modellierung der Anforderungen gewählt wurde, ermöglicht es dem Modell, gut auf neue und geänderte Anforderungen zu reagieren. Bei der Erstellung wurden sowohl klassische als auch agile PM-Methoden berücksichtigt und somit sind eine Vielzahl von Situationen abbildbar. Die Konstrukte sind so aufgebaut, dass sie in ihrer inneren Struktur jederzeit modifizierbar sind. Die Anbindung zu anderen Konstrukten über fest definierte Schnittstellen ermöglicht jederzeit eine Erweiterung oder Abänderung, ohne im Konstrukt selbst Anpassungen vornehmen zu müssen. Bei der Notwendigkeit, das Modell um neue Komponenten zu erweitern ist es erforderlich, die Schnittstelle des neuen Konstrukts und die Anbindung an das bestehende Modell zu definieren.

4.5 Strukturierte Objekte (Marken)

Für die Ressourcenzuordnung wird das Konzept der strukturierten Marke zum Einsatz kommen. Eine optimierte **Ressourcenzuordnung** wird in drei Bereiche eingeteilt: Die **Ressourcenanfrage**, die **Ressourcenverwaltung** mit einem gesammelten **Ressourcenpool** und die **Ressourcenbereitstellung**. Es geschieht unter der Annahme, dass die Ressourcenzuteilung auf der gleichen Granularität wie die zugrunde gelegte Zeiteinheit erfolgt und erst beim Eintreffen der abzuarbeitenden Aufgabe stattfindet. Damit können auch Verzögerungen aus vorangegangenen Aufgaben, welche Ressourcen binden und Wissensaneignungen zum jetzigen Stand Berücksichtigung finden. Dafür muss die Ressourcenanforderung auch in der Lage sein, neben direkten Ressourcenzuweisungen auch auf der Grundlage von Eigenschaften und Anforderungen

eine Zuordnung zu ermöglichen. Daraus ergeben sich folgende Vorteile: **(1)** Einsatz von Ressourcenalternativen statt einer Ressource; **(2)** heterogene Ressourcenalternativen; **(3)** Einsatz von mehreren Ressourcen parallel an einer Aufgabe; **(4)** Möglichkeit von Ressourcenallokation mit mehreren Bewertungskriterien.

Danach können festgelegte Ressourcenallokationsstrategien (RAS) als Bedingung bei der Ressourcenbereitstellung angewandt werden. Die nachfolgenden RAS repräsentieren eine beispielhafte Auflistung:

- (1) RAS: Immer die billigste Ressource wählen, welche zum frühestmöglichen Zeitpunkt frei ist.
- (2) RAS: Immer die erfahrenste Ressource in der Abarbeitung dieser Aufgabenkategorie wählen, die zum frühestmöglichen Zeitpunkt frei ist.
- (3) RAS: Anhand der Schwierigkeit einer Aufgabe die passenden Ressourcen auswählen. Ebenfalls könnte als Lernhilfe immer ein vorgegebener Anteil der benötigten Dauer als Mix aus Anfängern und beispielsweise Experten eingesetzt werden.
- (4) RAS: Hierbei können auch RAS gemischt werden.

Danach kann eine Ressourcenbereitstellung gemäß der Präferenz der angewendeten RAS für ein gesetztes Ziel simuliert und bewertet werden. Dies wird durch strukturierte Marken ermöglicht. Die Grundidee beruht darauf, dass mit dem aufgestellten Projektplan die schon getroffenen Änderungen ohne mühsame Anpassung der angegebenen Informationen untersucht werden können. Dabei können verschiedenste Variationen von Parametern durchlaufen und dessen Auswirkungen beobachtet werden. Mit mehreren Durchläufen kann so eine Variation des Ergebnisses erfolgen. Ebenfalls ist es dafür notwendig, dass die Information über alle zur Verfügung stehenden Ressourcen in einem Platz gebündelt werden und für die Abfrage von einem alternativen Ressourceneinsatz (bei Auftreten eines Konfliktes) zum Einsatz kommen. Der Startpunkt dieser Simulation ist von den gegebenen Ist-Daten abzuleiten (neue Anfangsmarkierung des jetzigen Projektstandes) und kann nicht in der Vergangenheit durchlaufen werden.

Wie aus Kapitel 4.3 ersichtlich, gibt es folgende Kriterien, nach denen eine Ressourcenzuteilung im zugrunde gelegten Modell erfolgt: **(1)** Fest zugeordnete Ressource. Dies ist die einfachste Variante und eine RAS kommt nur zum Einsatz, wenn diese nicht verfügbar oder nicht angegeben ist. **(2)** Nach deren Vorwissen, welches sich pro Bereich und Gebiet im Laufe des Projektes steigert. Dabei wird nach jeder Abarbeitung von einer Aufgabe, ausgehend von der ID, das Vorwissen um eins hochgezählt. **(3)** Komplexität der Aufgabe. Die Komplexität der Aufgabe dient als Grundlage, um Regeln basierend auf Vorwissen für die Ressourcenzuordnung zu definieren.

Am Zusammenspiel des Vorwissens und der Komplexität wird eine mögliche Regelstruktur aufgezeigt. Zur Umsetzung dieser werden folgende Bedingungen benötigt, welche in den nachfolgenden Unterkapiteln näher beleuchtet werden: **(1)** Die Datengrundlage in einer Tabelle. **(2)** Genereller Aufbau und Verarbeitung strukturierter Marken. **(3)** Regeln zur Anwendung bei der Verarbeitung und Zurückführen der Erkenntnisse in die Tabelle.

4.5.1 Tabellarische Datengrundlage

Alle benötigten Informationen müssen in einer Tabelle mit folgendem Format aufgenommen werden: Die Ressourcenzuordnung wird zur Aufgabe und dessen Phase in der Tabelle Ressourcenzuordnung vorgenommen. Hier wird pro Aufgabe die anzufragende Ressource, die erwartete Komplexität und das benötigte Vorwissen eingetragen. Die Vorgangs ID beinhaltet die Phase und den Bereich zur späteren Zuordnung.

Tabelle 4: Tabelle für Ressourcenzuordnung

Vorgang ID	Zuordnung		RZ ID	Kom- plexität	Benötigtes Vorwissen	Ermittelte RZ
	Phase	Bereich				
R_D_001	Design	Report	R25	3	>10	
R_E_001	Entwicklung	Report	R25	2	(2)	

Ausstehend ist noch die Information, wie das Vorwissen der jeweiligen Ressource im angefragten Bereich im Bezug zur Phase ist. Dafür wird die nachfolgende Tabelle 5 definiert. Die Ressourcenzuordnung wird dabei anhand von zwei Faktoren beeinflusst. In den Tabellen kann hinterlegt werden, ob eine vordefinierte Ressourcenzuordnung erfolgen soll und welches Vorwissen diese Aktivitätsphase beansprucht. Bei der dann zu tätigen Ressourcenzuordnung kann je nach hinterlegten Regeln oder Szenarien, eine Zuordnung anhand der vordefinierten Ressourcenzuordnung, benötigtes Vorwissen plus gesammelte Erfahrung oder einem Mix aus diesen beiden Angaben erfolgen. Als Alternative könnte auch die Komplexität einer Vorwissensbandbreite für die Zuteilung definiert werden. Als Arbeitshypothese wird für die nächsten Abbildungen angenommen, dass eine Zuteilung anhand des benötigten Vorwissens getroffen wird. Dies hat zur Konsequenz, dass keine vordefinierte Ressourcenzuteilung bekannt oder definiert ist und sich im Ressourcenpool anhand des Vorwissens eine Zuteilung ereignet. Dabei wird die Komplexität der Zuteilung einfach gehalten. Auch hier könnten verschiedenste Regeln definiert werden. Die Ausprägungen der einzelnen Merkmale sind wie folgt definiert und können einfach erweitert werden:

- Bereich: System, Daten, Report, fachliche Themengebiete
- Gebiet: Schnittstelle, Funktionen, Design, Datendefinition
- Komplexität: (1) sehr hoch, (2) hoch, (3) mittel, (4) gering
- Vorwissen: (1) Anfänger < 10, (2) Erfahrung 11 - 50, (3) Experte 51 - 100, (4) Profi >100

Tabelle 5: Tabelle für Ressourcenbeurteilung

Ressourcen ID	Zuordnung		Vorhandenes Vorwissen	Gesammelte Erfahrung
	Phase	Bereich		
R23	Design	Report	12	
R23	Entwicklung	Report	6	
R25	Design	Report	2	
R25	Entwicklung	Report	8	

4.5.2 Genereller Aufbau und Verarbeitung strukturierter Marken

Detailinformationen können in den strukturierten Marken mitgeführt und als Kriterien für definierte Regeln/Bedingungen herangezogen werden. Ebenfalls kann eine Abspeicherung der Bearbeitungsstände des Objektes auch als Informationen in der strukturierten Marke stattfinden. Mit dieser Art von Information kann der Fortschritt durch eine Simulation als Ergebnis zurückgeschrieben werden. Auch dafür sind Regeln zu definieren. Beispielsweise bildet sich der Fortschritt aus der angegebenen Dauer im Verhältnis zu den schon abgearbeiteten Zeittakten. Hierbei muss aber berücksichtigt werden, dass sich der gefühlte abgefragte Ist-Fortschritt nicht an der noch verbleibenden Dauer orientiert.

In diesem Abschnitt wird sich darauf konzentriert, den Aufbau und die Weiterverarbeitung der strukturierten Marke, inklusive der Rückführung möglicher Simulationsergebnisse anhand der Ressourcenzuordnung auf der Grundlage von Vorwissen zu betrachten. Dabei wird die Ressourcenart mit der Vielfachheit am Beispiel eines Mitarbeiters herausgegriffen. Diese Logik ist für alle anderen Ressourcen analog anzuwenden. Der generelle Aufbau der strukturierten Marke ist wie folgt festgelegt: $StM_{<RA/RB>}_R_{<St/V>}$ (mitgelieferte Informationen) #<Dauer/Anzahl>

Die strukturierte Marke wird dabei immer drei Schritte durchlaufen: Die Ressourcenanfrage mit allen notwendigen Informationen, den Vergleich im Ressourcenpool und die Ressourcenbereitstellung (RB).

- $StM_{RA}_R_V$ (Vorgangs ID; Phase; Bereich, RZ ID, Komplexität, benötigtes Wissen) #Anzahl
- $StM_{RB}_R_V_{RID}$ (Ressourcen ID - RID, Phase, Bereich, vorhandenes Wissen, gesammelte Erfahrung) #Anzahl

In Abbildung 23 ist ein Ausschnitt der schon bekannten Ressourcen-Konstrukte im Zusammenspiel mit dem Phasen-Konstrukt herausgegriffen, an welchem der Kreislauf einer strukturierten Marke verdeutlicht wird.

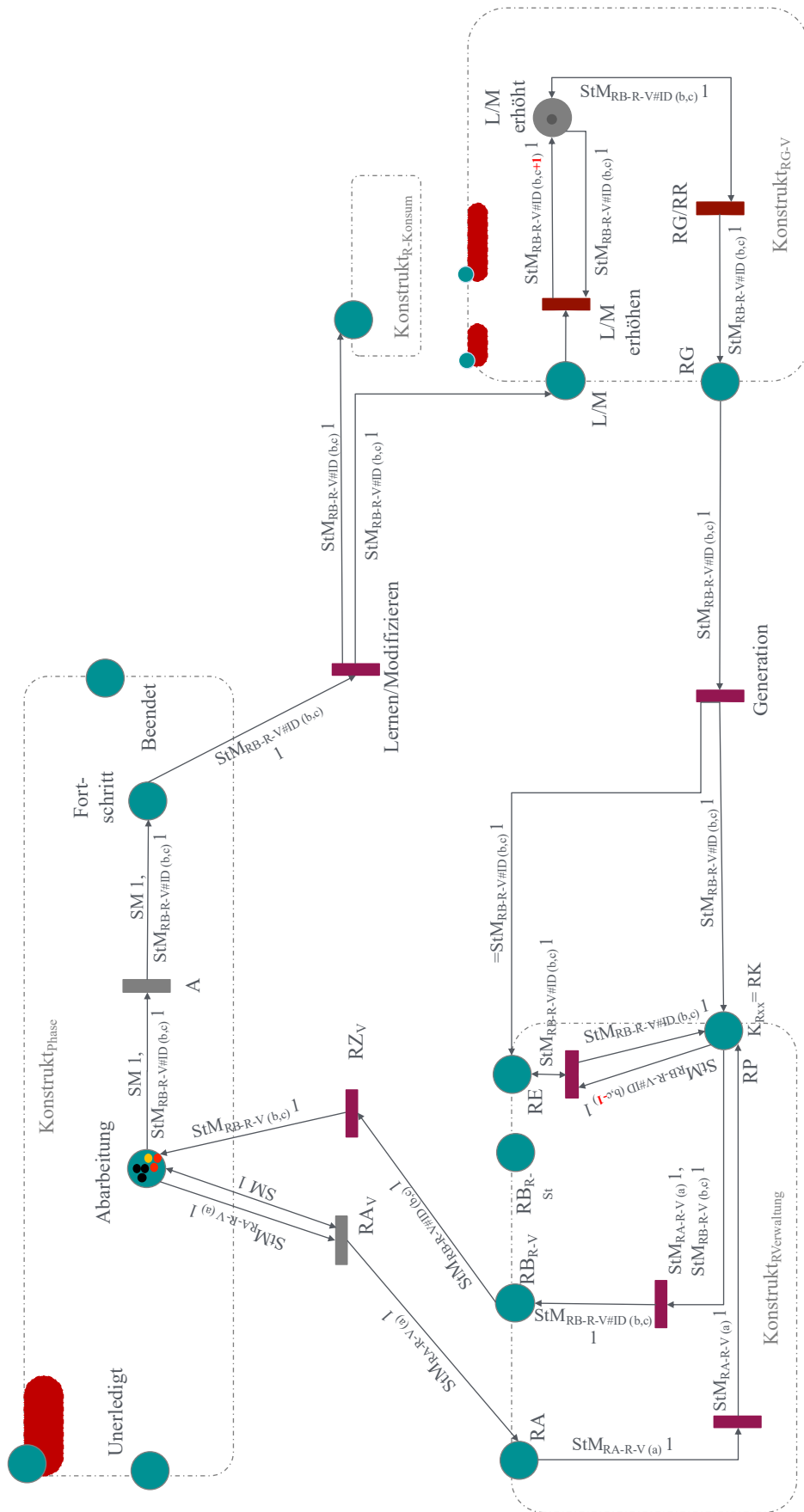


Abbildung 23: Ausschnitt & Beispiel - Verwendung einer strukturierten Marke

Zur Vereinfachung wird eine Zuordnung der Ressource nur anhand des benötigten Wissens vorgenommen und die Struktur der strukturierten Marke wie folgt reduziert:

- StM_{RA-R-V} (benötigtes Wissen) #Anzahl
- StM_{RB-R-V_RID} (vorhandenes Wissen, gesammelte Erfahrung) #Anzahl

Zur vereinfachten Erklärung wird das Beispiel aus Abbildung 23 stark vereinfacht in Abbildung 24 dargestellt. Dabei ist der Fokus in Abbildung 24 allein auf der Erklärung des Kreislaufes und der Funktionsweise einer strukturierten Marke gerichtet und ist nicht mehr konform mit der Abbildung eines PN. Die Abbildung 24 wird durch das dahinterliegende Netz ersetzt. Abarbeitungsschritte sind durch schwarze Marken, Ressourcenanforderungen durch rote und Ressourcenbereitstellungen durch gelbe Marken im Platz *Abarbeitung* gekennzeichnet. Den Anfang macht die Ressourcenzuordnung **(1)**. Diese strukturierte Marke ist wie folgt vereinfacht aufgebaut: StM_{RA-R-V} (a=benötigtes Wissen) #Anzahl. Im Ressourcenpool kommt diese Ressourcenanfrage an. Anschließend werden definierte Regeln angewandt, um eine Ressourcenzuteilung – StM_{RB-R-V_RID} (b= vorhandenes Wissen, c=gesammelte Erfahrungen) #Anzahl – zu gewährleisten. Darauf wird im nächsten Abschnitt näher eingegangen. Das vorliegende Beispiel wird mit der Annahme weitergeführt, dass eine optimierte Ressourcenzuordnung stattgefunden hat **(2)**. Die Schritte **(1 und 2)** wurde bereits vorgenommen und im Platz *Abarbeitung* ist dies als Zwischenmarkierung ersichtlich. Der nächste Takt löst zwei parallele Abfolgen aus. Nach der getroffenen Ressourcenbereitstellung kann ein Abarbeitungsschritt **(3)** erfolgen. Für spätere Auswertungen über den schon verbrauchten Ressourceneinsatz wird im Schritt **(4)** der Ressourcenverbrauch protokolliert. Der Schritt **(5)** gibt die strukturierte Marke für die Erhöhung zum Erfahrungszuwachs weiter. Im Schritt **(6)** wird der Erfahrungszuwachs in der strukturierten Marke um Eins erhöht (StM_{RB-R-V_RID} (b=11, c=1) und zur Regeneration weitergegeben. Im Regenerations-Konstrukt wird je nach definierten Regenerationsphasen die strukturierte Marke an den Ressourcenpool zurückgegeben **(7)**. Die schon generierten Marken müssen noch um den aktuellen Erfahrungszuwachs aktualisiert **(8)** werden. Anschließend findet eine neue Ressourcenanfrage und -bereitstellung **(1, 2)** statt.

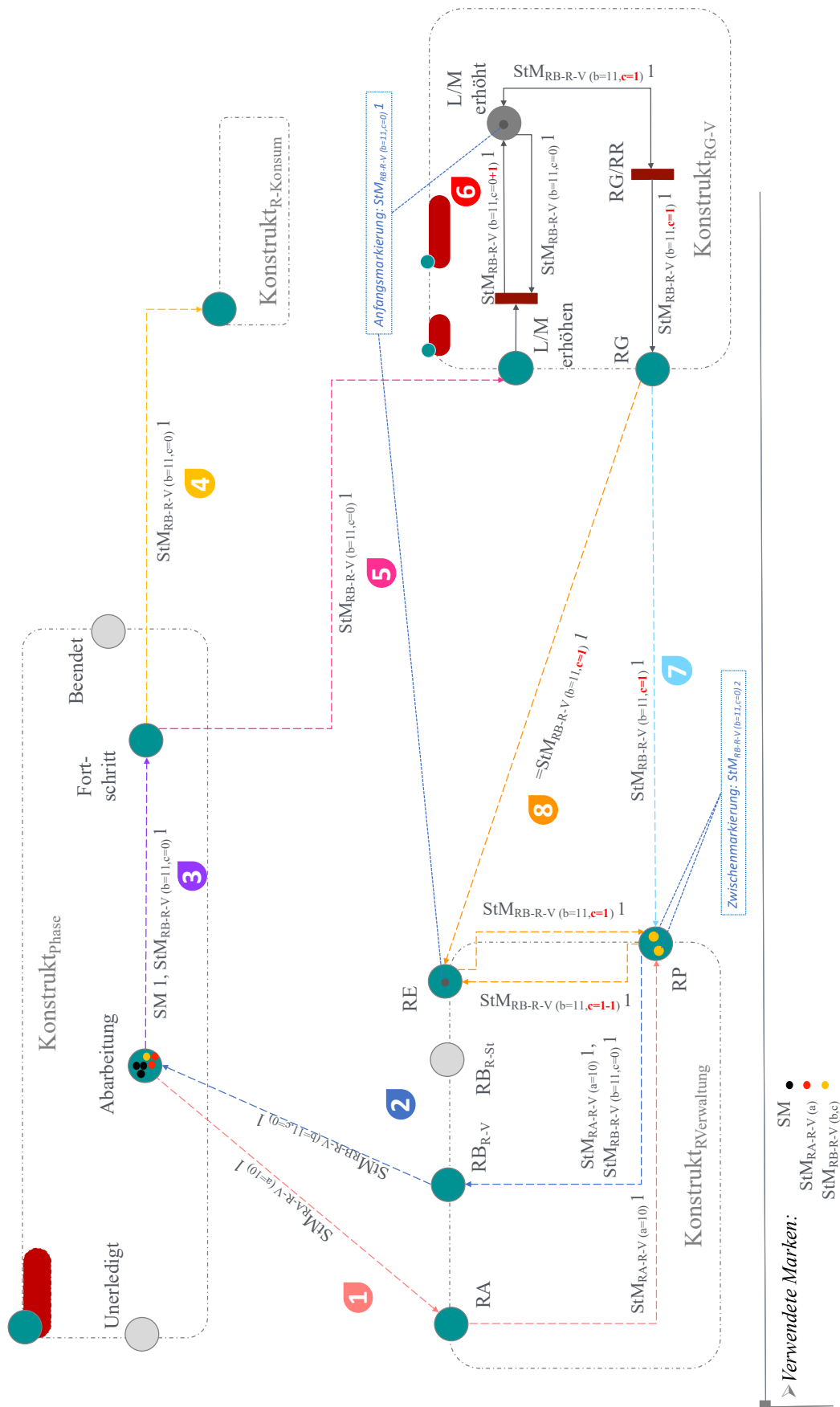


Abbildung 24: Beispiel Verwendung strukturierter Marke – stark vereinfacht

4.5.3 Regeln zur Anwendung bei der Verarbeitung und Zurückführen der Erkenntnisse in die Tabelle

Aufbauend auf den mitgelieferten Kriterien in der strukturierten Marke kann mittels einer Simulation und definierten Regeln betrachtet werden, ob eine Änderung in der Ressourcenzuordnung zu signifikanten Vorteilen im Projektverlauf führen könnte. Im durchgespielten Beispiel kommt die RAS 2 zum Einsatz. Es gibt bestimmte Auslegungen des Verfahrens und hier wird nur ein Beispiel aufgezeigt.

Der Platz *Ressourcenpool* bündelt alle Ressourceninformationen in Form von Marken und bedient die Ressourcenanfragen. Die Information an der strukturierten Marke wird durch den Mechanismus mit dem Platz der *Ressourcenerneuerung* mit jedem Schalten aktualisiert. Die einzelnen Konstrukte verarbeiten, je nach Fokus, die enthaltenen Informationen und geben die strukturierte Marke nach jedem abgearbeiteten Schritt wieder zurück in den Ressourcenkreislauf.

Eine Art der zu betrachtenden Ressourcen sind Mitarbeiter eines Projekts. Wie schon eingangs beschrieben, wird ein Mitarbeiter selbst nicht als atomare Einheit modelliert, sondern seine zur Verfügung stehenden Arbeitseinheiten. Im Rahmen des gewählten Zeitkonzeptes (Takt) erfolgt eine Abbildung dieser Arbeitseinheiten stundenweise. Beim zugrunde gelegten Beispiel wurde für die Ressourcenanfrage zuerst geprüft, welche Ressourcen noch zur Verfügung stehen (Anzahl Marken im Platz). Diese werden anhand des Vorwissens und der gesammelten Erfahrung bewertet. Wie schon erwähnt, sind hier verschiedenste RAS mit unterschiedlichsten Regeln denkbar. Dabei kann an jeder Kante des PN eine Bedingung hinterlegt werden, welche mit strukturierten Marken verknüpft sind. Dies symbolisiert Objekte mit Strukturen für Operationen auf einzelne Informationen – die Schaltregeln.

Nach jedem Takt können die Zwischenmarkierungen in der Tabelle festgehalten werden. Die Tabellen sind dahingehend mit notwendigen Spalten angereichert. Ebenfalls wäre eine Erfassung der Historie machbar, weil bei jeder Simulierung alle durchlaufenden Markierungen bekannt sind und festgehalten werden können. Dafür ist jede geänderte Markierung mit einem Zeitstempel vom globalen Zeit-Konstrukt abzuspeichern.

4.6 Entwurf und Zugriff auf die Modelle im Projektkreislauf

Das entwickelte Modell, mit den modularen Konstrukten zur Abbildung von Projektmanagementaufgaben auf Petri-Netze, wird automatisch generiert und anschließend simuliert. Die Grundlage des PPNM bildet eine formale Tabellenstruktur und somit eine Methode zur Überführung und Adaption des PMs in ein PN. Da es sich um eine dynamische Betrachtung handelt, ist ebenfalls der Kreislauf dieser formalen Modellierung als Grundlage der entwickelten Methode darzustellen. [vgl. Weichenhain und Fengler 2019a] Erkenntnisse über die aufgestellte Planung und möglicherweise existierenden Konflikte werden durch die Simulation erkennbar und aufgezeigt. Die Abbildung der Dynamik und der damit einhergehenden Veränderung des Netzes über die Zeit wird zur Erlangung der gewünschten Informationen ausgenutzt. Damit der Kreislauf von der Dateneingabe, über die Modellierung mittels PN und dessen Simulations- bzw. Prognosemöglichkeiten geschlossen wird, werden die Ergebnisse dessen wieder in die Tabelle zurückgeschrieben. Sollen aus den Erkenntnissen der Prognose bestimmte Projektparameter angepasst werden, muss eine Anpassung der Daten in der Tabelle erfolgen und der Ablauf beginnt erneut. [vgl. Weichenhain und Fengler 2019a]

Wie in Kapitel 3.4.4 beschrieben, wird zwischen vier Modellen unterschieden: Das reine **Plan-Modell**, das **Plan-Ist-Modell**, das reine **Ist-Modell** und das **Prognose-Modell**. Dabei haben alle Modelle den gleichen Ursprung und durchleben den gleichen Kreislauf, wie in Abbildung 25 grafisch dargestellt.

Der Ausgangspunkt eines Modells sind die Angaben in den Tabellen. Es sind zwei grundsätzliche Arten von Angaben zu unterscheiden, die als Input für das Modell herangezogen werden: Plan und Ist. Das PPNM an sich generiert Prognosewerte. Die Darstellung der Modellierung des PN erfolgt nur vor dem Projektstart als bekannter Gesamtzustand mit der Anfangsmarkierung am Projektstart. Sobald das Projekt gestartet ist, liegt der Focus auf der Abbildung von Zwischenzuständen. Dabei wird nicht die Struktur an sich geändert, sondern vorrangig nur die Markierungen verändert.

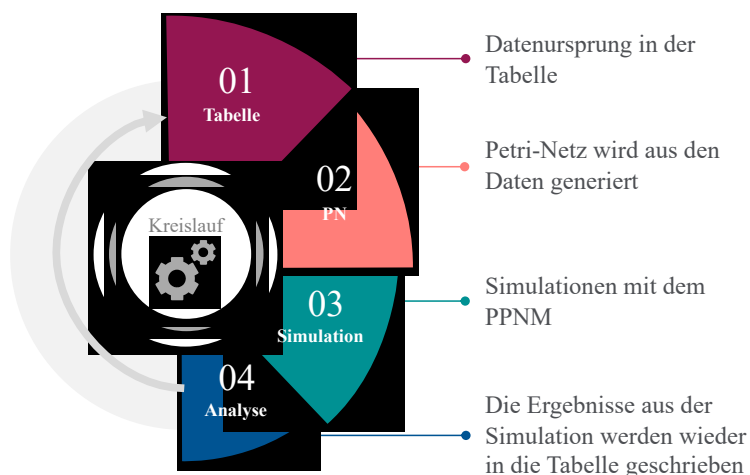


Abbildung 25: Kreislauf des Projekt-Petri-Netz-Modells

Der Umgang mit den Ist-Daten und deren Auswirkungen auf das Plan-Ist-Modell bzw. Ist-Modell, wird nachfolgend kurz erläutert. Ist-Daten sind in den dafür vorgesehenen Spalten in den Tabellen zu pflegen und signalisieren einen schon erfolgten Projektstart. Hierbei können die drei Zeitangaben sowie der prozentuale Fortschritt für die Design- und Entwicklungsphase einer Aufgabe angegeben werden. Der prozentuale Fortschritt ist somit eine reine Ist-bezogene Angabe. Sobald in diesen Spalten ein Datum bzw. Wert eingetragen ist, sind im Plan-Ist-Modell die Ist-Daten für die Modellgenerierung heranzuziehen. Liegt keine Ist-Angabe vor, findet die Berücksichtigung von Planwerten statt. Anschließend kann durch die Ersetzung der Plandaten mit den vorhandenen Ist-Daten ein neues Netz generiert werden. Bei der Generierung ist ein Zugriff auf definierte Leistungsmaße gegeben und somit eine Beurteilung des Plan-Ist-Vergleichs machbar. Der Eingriffspunkt für die Steuerung des Plan-Ist-Modells ist für den Nutzer durch die Spalten gegeben. Das Modell muss an den aktuellen Zustand angepasst und die neue Anfangsmarkierung auf das aktuelle Datum des Projektes gesetzt werden.

Im Themengebiet der Prognose geht es um den Umgang mit Plandaten, Ist-Daten und den zu tätigen Annahmen. Dabei wird von einem definierten Zeitpunkt ab simuliert, was ebenfalls die Auswirkungen der Ist-Daten aufzeigt. Es sollen schon vorhandene weitere Ansätze zu PN-Erweiterungen bei der aufgestellten Formalisierungsmethode nicht ausgeschlossen werden. Deswegen wird im Rahmen dieser Arbeit auf potenzielle vorhandene Ansätze der mathematischen Analyse und Validierung verwiesen und illustriert, wie diese das PM noch weiter bereichern könnten. Ebenfalls ergibt sich die Frage, welches Potential das vorgestellte Modell mitbringt und für diese Zwecke schon

nutzbar ist. Wie in Kapitel 2.3.2 dargelegt, kann zusammenfassend für die Analyse durch PN und den Einsatz stochastischer PN festgehalten werden, dass mit Petri-Netzen eine einfache Form für die Entscheidungsfindung zur Verfügung steht, welche ebenfalls relativ leicht modellierbar und nachzuvollziehen sind. Die Kombination mit Erweiterungen aus der Stochastik oder Fuzzy-Logik bietet ein großes Spektrum für die Prognose. Ebenfalls bringen sie folgende Vorteile für die Ausführung einer Prognose mit sich: **(1)** Eindeutige Semantik, welche das Verhalten des jeweiligen Netzes festsetzen. Dies ermöglicht Implementierungen und formale Analysen. **(2)** Explizite Beschreibungen für Zustände und Aktionen sind vorhanden. Wohingegen andere Beschreibungssprachen oft nur Zustände oder Aktionen modellieren. **(3)** Die Nebenläufigkeit von gleichzeitigen Aktionen einbinden zu können. **(4)** Eine interaktive Simulation im Einzelschrittmodus, falls gewünscht. **(5)** Vielzahl von Tools, mit denen ein PN erstellt, simuliert und formal ausgewertet werden kann. [vgl. Benecke 1995, 6 ff]

Bei diesen Ansätzen sind Aspekte aus der realistischen Projektplanung mit der Einbeziehung von Erfahrungen und Annahmen durch die Modellierung von Unschärfen oder zufälligen Annahmen/Ergebnissen angereichert. Für die Auswertungen in dem hier vorgesehen Rahmen, wird sich daher auf das Zurückgreifen von Erfahrungen und daraus abzuleitenden Annahmen konzentriert. Eine Abbildung kann mit dem Konzept der strukturierten Marken erfolgen. Dafür müssen alle erforderlichen Informationen vorher aufgenommen und im PN verarbeitet werden.

4.6.1 Die Netzgenerierung

Die in Tabellenform vorliegenden Rohdaten des Projektes bilden die Grundlage. Darauf aufbauend ist eine Transformation dieser Daten nach den vorgestellten Regeln der Konstrukte auf ein PN zu übertragen. Wie die einzelnen Schritte zu programmieren sind, hängt von der Wahl des zugrundeliegenden Tools ab. In Kapitel 6 wird eine automatische Generierung mit PenecaChromos aufgezeigt. In dem Fall ist das Tabellenformat in ein cne-Format zu übertragen. Dafür sind alle wichtigen Informationen aus den Tabellen pro Konstrukt zu entnehmen. Für einen Sprint muss beispielsweise die Information gewonnen werden, wie viele Aufgaben der Koordinator zu initiieren hat, was einen Einfluss auf die Kantenmarkierung der Kante (*Starte Koordinator, Koordinator*) im Sprint-Konstrukt herbeiführt.

Dabei muss das dafür entwickelte Programm alle definierten Konstrukte abbilden. Die Übernahme der Planungsdaten soll nicht an ein bestimmtes Tool gebunden sein, was die standardisierte Tabellenform mit sich bringt. Es müssen Plandaten und Ist-Daten verarbeitet und eine grafische Darstellung des PN gewährleistet sein. Die modularen Konstrukte sollen durch das Programm, nicht durch ihre Anpassbarkeit und Erweiterbarkeit eingeschränkt werden. Sobald das Netz durch die Angaben in den Tabellen generiert wurde, sind auch die notwendigen Anfangsmarkierungen im Netz gesetzt. Demzufolge kann eine Zuordnung erreichbarer Markierungen zu den Tabellenwerten erfolgen und die Reproduzierbarkeit ist gegeben. Ebenfalls bedeutet dies für alle Modelle und vor allem für die Prognose, dass für jede erreichbare mögliche Markierung die Rückführung in die Tabelle stattfinden kann. Dafür müssen alle dynamischen Werte mit Prognosespalten erweitert werden. Eine Speicherung des Netzes mit allen Zwischenmarkierungen kann so durch die Tabelle erfolgen und eine Rückdarstellung ist gewährleistet.

4.6.2 Besonderheit der Ermittlung der korrekten Anfangsmarkierung

Zusammengefasst existieren drei verschiedene Arten von Daten: **Planungsdaten**, **Ist-Daten** und **Prognosedaten**. Die Kombination dieser Datentypen ermöglicht die Erstellung unterschiedlicher Modelle und Simulationsmöglichkeiten. Als eine Besonderheit bei dem Plan-Ist-Modell soll auf die korrekte Ermittlung der Anfangsmarkierung bei einem Sprint eingegangen werden. Am Beispiel einer Phase wird hier die Vorgehensweise näher verdeutlicht. Die Reihenfolge der Berechnung muss dabei so aufeinander abgestimmt sein, dass die Ergebnisse aufeinander aufbauen. Eine Besonderheit stellt dabei dar, dass Kantenbeschriftungen zur Abbildung der Zeitrestriktionen von den Ist-Daten nicht zu beeinflussen sind. Nur so ist zu gewährleisten, dass die Ermittlung der Überziehungsdauer auf die Auswertung und die Relation zu den geplanten Daten bestehen bleibt und Überziehungen weiterhin korrekt dargestellt werden. Die Berechnung der neuen Anfangsmarkierung ereignet sich für alle Konstrukte, sobald der gewählte Zeitpunkt nach dem Projektstart liegt. Mit der Annahme, dass bis zu diesem Datum alle Ist-Daten vorliegen.

Bei dem Sprint-Konstrukt ist nur eine Stelle für die Ermittlung der korrekten neuen Anfangsmarkierung relevant. Diese ist in dem Platz *Sprint Beendet* des aktuell

betrachteten Sprints. In diesem Platz muss eine Marke liegen, wenn nach aktuellen Projektstand alle Aufgaben zu 100 Prozent bearbeitet sind und noch keine Aufgabe des nachfolgenden Sprints gestartet wurde. Die zu ermittelnden Anfangsmarkierungen der einzelnen Aufgaben und ihrer Phasen eines Sprints sind in Abbildung 26 mit Zahlen farblich hervorgehoben.

Der Platz *Initiiere Aufgabe* im Aufgaben-Konstrukt ist nur während der Designphase relevant. Bei der Entwicklungsphase entfällt dieser Schritt, er ist Bestandteil der Aufgabe und folgt direkt auf die schon initiierte Aufgabe in der Designphase. Für diesen Platz erfolgt die Berechnung so, dass sobald das Startdatum der Aufgabe dem aktuellen Datum entspricht oder größer ist und der Fortschritt 0 % beträgt, wird eine schwarze Marke in den Platz gelegt. Wenn diese Bedingungen nicht zutreffen, wird der Platz ohne Marke initiiert. Die nachfolgende Tabelle zeigt die Berechnung der anderen Plätze in einer stark zusammengefassten Erklärung. [vgl. Weichenhain und Fengler 2019a]

Tabelle 6: Bedingungen/Berechnungen Anfangsmarkierung einer Aufgabe

	Bedingung	Markierung
(1)	Startdatum entspricht aktuellem Datum und Bearbeitung hat noch nicht begonnen	1 schwarze Marke
(2)	Aufgabe wird gerade nicht initiiert und Fortschritt \neq 100 %	(Dauer * Fortschritt) schwarze Marke
(3)	Aufgabe wird gerade nicht initiiert und Fortschritt \neq 100 %	Dauer - #(Marken in Fortschritt) schwarze Marke
(4)	Aufgabe zu 100 % erledigt Nächste Phase noch nicht gestartet	1 farbige Marke 1 schwarze Marke
(5)	Ist-Enddatum vor bzw. am Plan-Enddatum und Fortschritt = 100 %	1 schwarze Marke
(6)	Überschreitung des Plan-Enddatums	1 schwarze Marke
(7)	Überschreitung des Plan-Enddatums	(Ist-Enddatum - Plan-Enddatum) schwarze Marke
(8)	Nächste Phase noch nicht gestartet	1 schwarze Marke

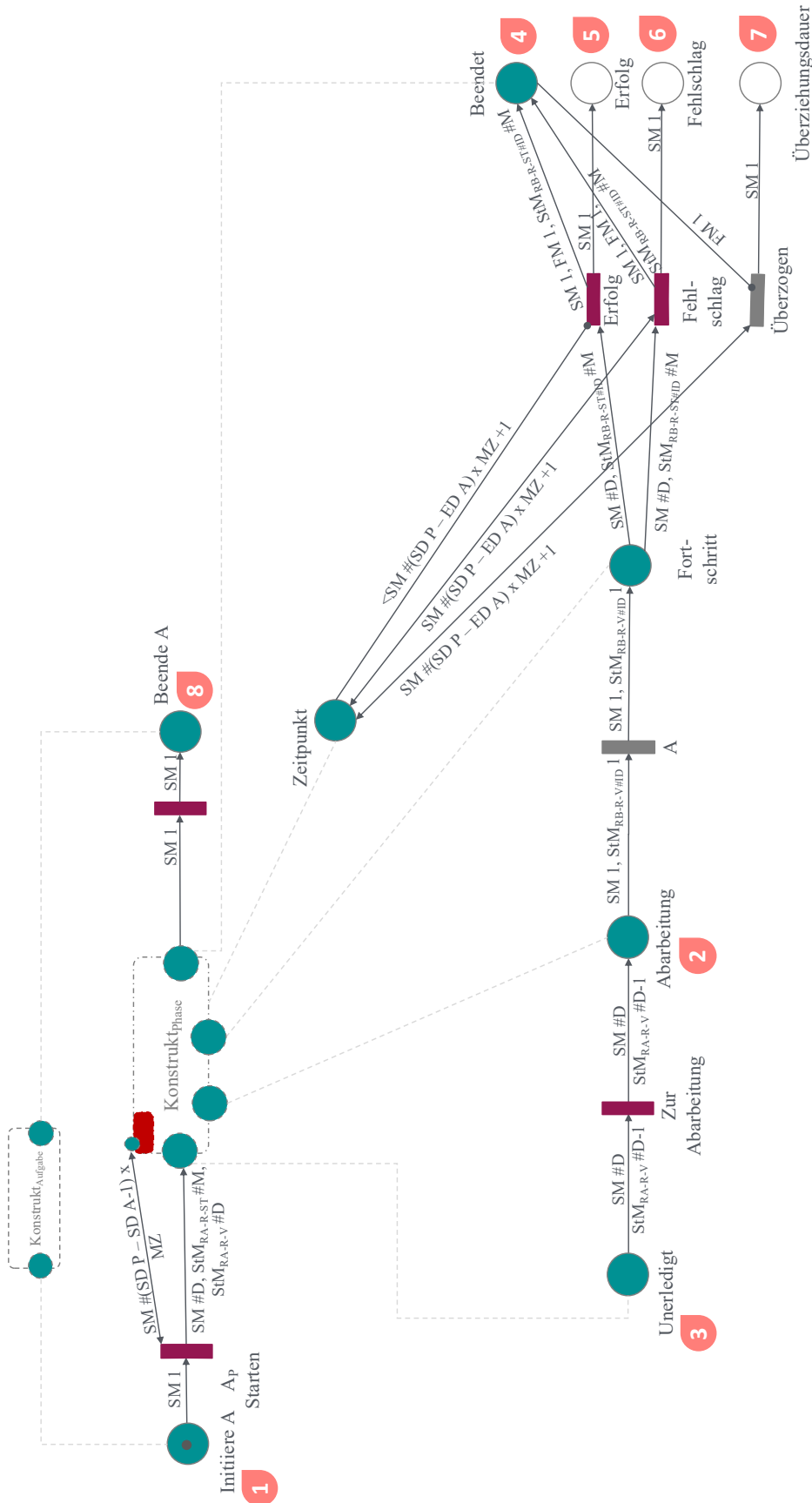


Abbildung 26: Anfangsmarkierung

4.6.3 Besonderheit der verschiedenen Zeitrestriktionen

Die Abbildung der definierten Zeitrestriktionen ist in den zwei Konstrukten der Aufgabe und der Phase integriert. Abbildung 27 zeigt die Ausschnitte aus der Aktivität und der Phase zusammenhängend, wobei eine farbliche Aufteilung in vier Bereiche für die einzelnen Zeitrestriktionen zu erkennen ist. Zur Vereinfachung wurden alle anderen Farben aus der Abbildung entfernt.

Mit **(1)** wird der Anforderung der Zeitvorgabe vom Typ 1, der festen Datumsangaben (Zeitraum mittel frühestes Startdatum und spätestes Enddatum), nachgekommen. Dies geschieht durch die Verknüpfung zum Zeitpunkt mit Testkanten oder Inhibitorkanten sowie der Prüfung durch Kantenbedingungen zu den definierten Zeitangaben. Bei der Berechnung der Kantenbedingungen ist nur beim frühesten Start zu berücksichtigen, dass mit der Abarbeitung der Aufgabe erst nach dem Startdatum begonnen werden kann. Dies führt dazu, dass ein Tag in der Berechnung der Kante abgezogen wird. Dagegen wird die Angabe vom Typ 2, der Zeitdauer, über die Anzahl der Marken im Bereich **(2)** verwirklicht.

Eine Aufgabe kann erfolgreich, fehlerhaft oder mit einer Überziehung abgeschlossen werden. Das Erreichen eines dieser Ausgänge ist unter anderem von der Einhaltung der zeitlichen Restriktionen abhängig. Sobald die angegebenen Zeitangaben überschritten sind, ist eine Überziehung **(3)** zu signalisieren. Die Reaktion bei einer Überziehung war nicht Gegenstand der Betrachtung. Es gibt verschiedenste Möglichkeiten, es in diesem Modell zu integrieren. Beispielsweise kann eine Überziehung toleriert werden oder eine Unterbrechung nach sich ziehen. Dabei ist die Frage der Ressourcenfreigabe zu beachten. In dem PPNM wird die Überziehung signalisiert und die Überziehungsdauer durch die Markenanzahl gemessen. Die Überziehung wird gemessen, sobald das späteste Enddatum überschritten, aber die Phase noch nicht beendet ist. Im Bereich **(4)** wird neben der Beendigung, auch die Abbildung eines erfolgreichen Abschlusses der Phase, als auch der Fehlschlag, angezeigt. Wie schon erwähnt, ist die Markenanzahl im *Zeitplatz* immer monoton wachsend und verhindert eine erneute Schaltfähigkeit der Bedingungen.

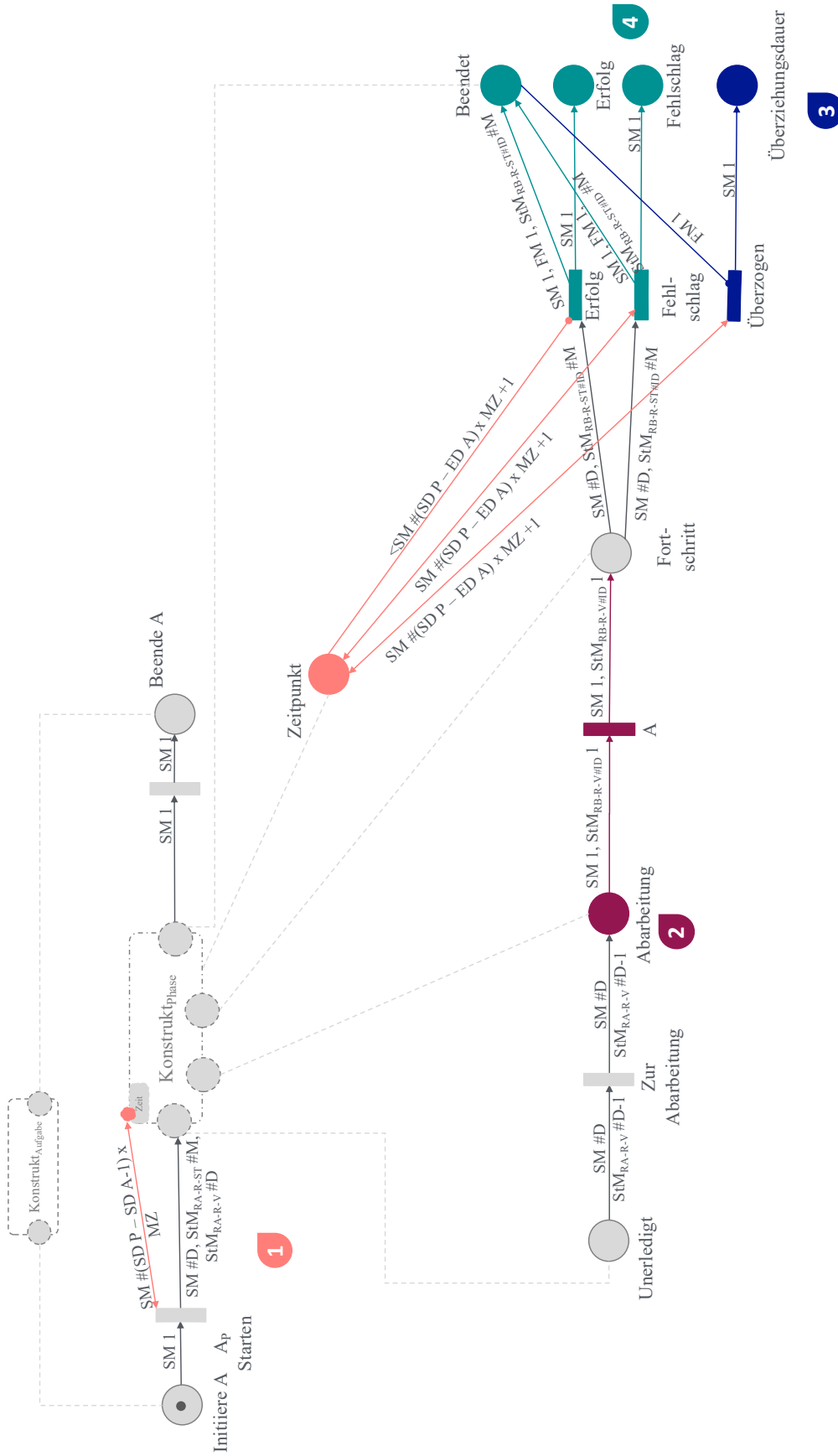


Abbildung 27: Einbindung Zeitrestriktionen und dessen Aussagen

4.7 Darstellungsform

Das zu erstellende Petri-Netz ist eine Zusammensetzung aus den vorher vorgestellten Konstrukten. Das Ziel soll sein, die Struktur des Projektes im Netz entsprechend anzuordnen. Dies verbessert in erster Line insbesondere die Übersichtlichkeit und evtl. führt es zu einer Verbesserung der Verständlichkeit des Modells. Eine so grafische Anordnung spiegelt die Modularität des entwickelten Modells wider. Wie schon bei der Anforderungserhebung festgestellt, wird die Thematik der Darstellungsform aufgrund ihrer Komplexität ausgegrenzt. Den marginalen Grundanforderungen aus Kapitel 3.4.2 soll nachgekommen werden. Abbildung 28 zeigt die festgelegte grafische Anordnung der Konstrukte.

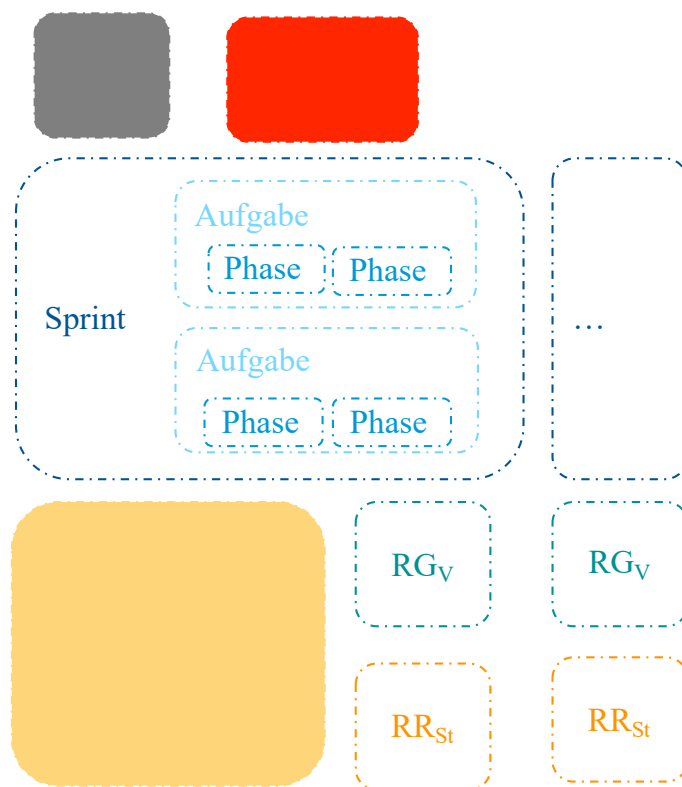


Abbildung 28: Grafische Anordnung der Konstrukte

Das Zeit-Konstrukt, auf das sowohl vom Konstrukt der Generationsphasen der Ressourcen als auch von den Aufgaben zur Einhaltung ihrer Zeitrestriktionen zugegriffen werden muss, ist ganz oben platziert. Hier lässt sich jederzeit leicht die aktuelle Zeit innerhalb des Modells ablesen. Zwei Arten von Ressourcen sind zu unterscheiden. Das Ressourcenkonsum-Konstrukt (**rot**) hat keinerlei Interaktionen mit den anderen Konstrukten und eine Platzierung erfolgt deswegen neben dem Zeit-Konstrukt. Die nacheinander ablaufenden Sprints (**blau**) werden nebeneinander angeordnet. Der erste Sprint ist hierbei ganz links platziert und die nachfolgenden Sprints jeweils rechts daneben, sodass die zeitliche Reihenfolge der Sprints im Projekt ebenfalls im Petri-Netz-Modell erkennbar ist. Jeder Sprint gliedert sich, wie später noch dargestellt, in die ihm zugeordneten Aufgaben. Die Aufgaben (**hellblau**) sind pro Sprint untereinander angeordnet. Die Ressourcenverwaltung (**gelb**) ist ein umfangreiches Konstrukt und folgt auf der nächsten Ebene. Daneben erfolgt eine Anordnung aller Generationen (**grün**) und Regenerationen (**orange**) von Ressourcen.

5^{Kapitel} Einführung und Aufbau eines Beispielprojektes

Für den weiteren Verlauf der Arbeit bedarf es eines zugrundeliegenden Beispiels, welches in diesem Kapitel vorgestellt wird. Einführend ist hervorzuheben, dass berichtend aus Erfahrungen aus dem täglichen Berufsalltag viele dieser Beispiele vorzufinden sind und zumindest auf der Ebene einzelner Projekte, fast ausnahmslos ein großer Teil des PM, immer auf Excel bzw. PowerPoint basierte. Das Kapitel beginnt mit einer allgemeinen Einführung und Vorstellung des Projektes. Weiterführend wird ein Einblick aus der Praxis gegeben, bevor im zweiten Teil das ausgewählte Projekt bezüglich des konkreten Anwendungsbeispiels für den späteren Gebrauch vorbereitet wird.

Bei dem nachfolgenden Beispiel wurden alle Daten anonymisiert und eine branchenunabhängige Darstellung explizit gewählt. Das ausgewählte Beispiel entstand aus realen Praxiserfahrungen aus mehrjährigen unterschiedlichen Projektstätigkeiten. Für die spätere Untersuchung liegt die Konzentration auf einen Teilbereich des vorgestellten Projektbeispiels. Bei diesem Projektbeispiel handelt es sich um ein Informationssystemprojekt, konkret um die Erstellung eines Data-Warehouses mit einem darauf aufsetzenden Reportingsystem. Im Rahmen der Projektmanagementtätigkeiten wurde sich dafür entschieden einige Teilbereiche agil umzusetzen, worauf im Anschluss genauer eingegangen wird. Dieses Projekt beruht auf realen Praxiserfahrungen der Autorin, wo sie ebenfalls im Projektmanagement involviert und die später aufgeführten Excel-Beispiele selbst entworfen hat oder unter Ihrer Anleitung entstanden sind.

5.1 Allgemeine Vorstellung des zugrundeliegenden Projektes

Die Ausgangssituation bildet eine heterogene Datenbasis und eine auf Eigenentwicklung oder Excel aufgebaute Berichterstattung bzw. Steuerung. Die Abbildung der operativen Prozesse ist schon einheitlich über eine Softwarelösung umgesetzt. Das Projektziel ist eine zentrale Reporting- und Steuerungsplattform mit zeitgemäßer IT-Architektur und flexibler technologischer Plattform für eine effiziente Bewältigung, auch mit zukünftigen Herausforderungen. [vgl. BearingPoint GmbH 2013; vgl. BearingPoint GmbH und Weichenhain 2013]

Für die Realisierung des Projektes ist es notwendig, ein Data-Warehouse aufzubauen, worauf mit einem Reporting Front End die erforderlichen Reports erweiternd abgebildet werden können. Um dies zu gewährleisten, ist eine ausführliche Beschreibung der Anforderungen (genannt Fachkonzepte) von Nöten. Zur Realisierung wurde ein Phasenmodell erarbeitet. Das Ziel dabei ist, Methoden des klassischen Projektmanagements mit agilen Methoden des SCRUM-Frameworks zu verbinden. Somit ergab sich ein paralleler Ansatz von klassischen und agilen Methoden. Die Grobplanung und fachliche Ausarbeitung des Projektes wurden klassisch durchgeführt. Die Umsetzung erfolgte agil.

Dazu wurde eine Unterteilung in Building Blocks (im weiteren Verlauf BB) vorgenommen. Das Projekt wurde in Building Blocks aufgeteilt, welche größere technische und fachliche Themenbereiche zusammenlegen. Diese Building Blocks werden wiederum in Arbeitspakete strukturiert. Arbeitspakete können rein technisch, rein fachlich oder gemischt definiert werden. Sie repräsentieren ein Fachkonzept oder falls notwendig, einzelne Themenbereiche eines zusammenhängenden Fachkonzeptes. Das Phasenmodell ist unterlegt mit einer detaillierten Projektplanung auf Basis von einzelnen Arbeitspaketen zu den jeweiligen Building Blocks, versehen mit Verantwortlichkeiten und Terminen. Ergebnisse aus dem Arbeitspaket sind einzelne Aufgaben, die in sich gebunden implementiert werden können. Die eigentliche Umsetzung der Aufgaben erfolgt in Sprints. Der Begriff Sprint wurde vom SCRUM Framework adaptiert und repräsentiert regelmäßige und wiederholbare Iterationen. Diese werden Sprint genannt und sind zeitlich begrenzt. Ziel eines jeden Sprints ist es, ein funktionsfähiges Inkrement, ein Zwischenprodukt, auszuliefern. Ein weiteres übernommenes Element des SCRUM Frameworks ist die Backlog-Liste. Die Backlog-Liste (nicht nach Produkt bzw. Sprint unterteilt) enthält alle umgesetzten, umzusetzenden oder geparkten Aufgaben an das zu liefernde Endprodukt. [vgl. BearingPoint GmbH und Weichenhain 2017]

Der Bereich rund um das Implementierungsteam arbeitet somit immer agil. Wohingegen der Fachbereich bei der Erstellung der Fachkonzepte einen gemischten Ansatz verfolgt. Grund dafür ist die fachliche Komplexität und Überschneidung der einzelnen Themen. Dies bedeutet, dass die Fertigstellung eines Fachkonzeptes auf einen längeren Zeitraum geplant wird. Was zur Folge hat, dass häufige und nutzbare Zwischenergebnisse (in Form von Aufgaben) geliefert werden müssen und somit stets eine enge Abstimmung mit dem

Entwicklungsverantwortlichen gesichert sein muss. Dazu ist es notwendig, ein besonderes Augenmerk auf die Einhaltung des Scopes zu legen. Nur so kann gewährleistet werden, dass die benötigten Zwischenergebnisse für die agile Implementierung auch Bestand haben, um später darauf aufzubauen. Der Fachbereich muss dahingehend überwacht werden, dass so früh wie möglich einzelne Anforderungen abgekapselt und in sich ausdefinierte Aufgaben übergabefähig sind. Der Anspruch auf Vollständigkeit muss nicht immer gegeben sein, sondern eine Funktion/Anforderung kann auch später erweitert werden. [vgl. Strasser o. J.]

Für die Darstellung im Rahmen dieser Arbeit wurden alle Praxisbeispiele anonymisiert oder entfremdet. In Abbildung 29 ist die vorgenommene Aufteilung der Building Blocks angedeutet. Der Aufbau der BB wurde so gewählt, dass Funktionalitäten in späteren BBs erweitert und wenn gefordert auch in schon vorhandenen Berichten nachgezogen werden können. [vgl. BearingPoint GmbH 2013]

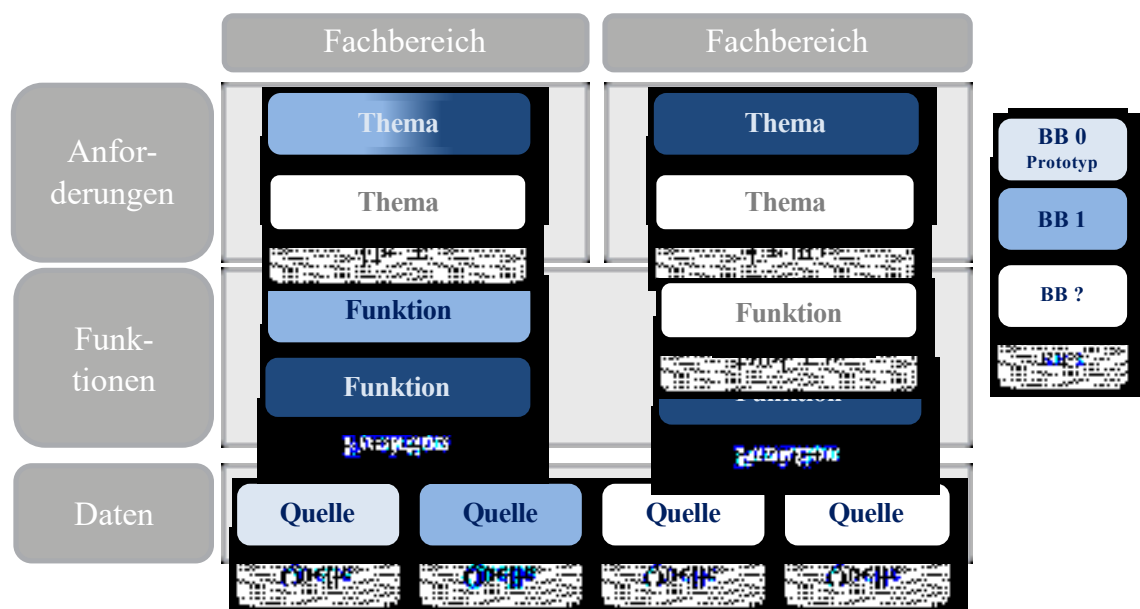


Abbildung 29: Schematische Darstellung der Building Blocks [vgl. BearingPoint GmbH 2013]

Den Start bildet der Building Block BB0 – eine Art Prototyp, welcher das Hauptaugenmerk auf die Bereitstellung des Reporting- und Data-Warehouse-Systems legt; inklusive der ersten Anbindung von Datenquellen. Aufbauend werden im ausgedehnten Projektverlauf weitere Funktionalitäten, Kennzahlen sowie fachliche und technische Erweiterungen dazukommen. Zur Vereinfachung wurde der Scope verschlankt. Im realen Projekt war die Parallelität mehrerer Fachbereiche gegeben.

Der BB0, welcher den Prototyp beinhaltete, wurde im ersten Sprint umgesetzt. Das zugrundeliegende Beispiel wird im Rahmen des BB1 angesiedelt. Es wird die Annahme getroffen, dass die Aktivitäten resultierend aus BB1 ab Sprint zwei integriert werden. [vgl. BearingPoint GmbH 2013]

Für das Projektsteering/-management während des Projektes wurden insgesamt drei Excel Dateien mit über 25 verschiedenen Reitern gepflegt. Zusätzlich wurden die Erkenntnisse daraus mit unterstützenden Diagrammen in einer PowerPoint Datei wöchentlich vorgestellt, was ebenfalls in unzähligen Slides resultierte. Wie dies im täglichen Projektalltag Anwendung fand, wird im nächsten Abschnitt aufgegriffen.

Die erste Datei war eine umfassende Planungsexcel von mehreren Projekten auf einem hohen Abstraktionsniveau mit Budget, groben Schätzungen und Priorisierungen von Kosten und Personentagen. In der zweiten Datei wurde das hier rausgezogene Projekt aufbereitet. Dies beinhaltete beispielsweise eine Gesamtprojektübersicht aller Building Blocks, inkl. deren Illustrierung als Gantt-Diagramm, Budgetverbrauch als Gesamtübersicht, Budgetverbrauch im Abgleich zum wöchentlichen Sprintstatus und Scope, die Ausplanung einzelner Arbeitspakete mit Gegenüberstellung Ist/Plan in Bezug auf die Zeit und den Ressourcenverbrauch oder der Gegenüberstellung von Plan Backlog, dem Ist und dem proportionalen Zeitverbrauch. Für die Darstellung der Diagramme war es oftmals notwendig, mehrere Diagramme übereinander zu legen, um die gewünschten Aussagen grafisch aufzubereiten. Im Anhang C ist ein Auszug aus einem Tabellenblatt dargestellt. Hier ist ersichtlich, dass auch eine große Anzahl an verschiedenen Formeln Anwendung fand. Hinzu kommen noch bedingte Formatierungen. Ebenfalls ist es notwendig zu wissen, dass die Pflege dieser Tabelle nicht nur darin bestand Werte nachzupflegen. Diese gesamte Konstruktion wurde bei Erweiterungen an den verschiedensten Stellen angepasst. Beispielsweise ist das Hinzufügen eines Sprints auch bei anderen Tabellenblättern zu berücksichtigen und auch alle Formeln in den Tabellen sind nachzuziehen. Diese Tätigkeit war wiederkehrend und gehörte ebenfalls zu den wöchentlichen Tätigkeiten. Auch eine Aufnahme von Aufgaben in der Backlog hat sich nicht automatisch in die anderen Listen integriert. Um auch die neu definierte Aufgabe in die Berechnung mit einzubeziehen, musste dies in den entsprechenden Tabellen angelegt werden. Ebenfalls wurde ein Ausblick errechnet, welche Ressourcenanforderungen in den nächsten vier Wochen laut Planung zur Verfügung stehen müssen und wie viele

angefordert sind. Dies ist vor allem bei einem agilen Projektansatz wichtig, da eine agile Ressourcenbereitstellung viel komplexere Anforderungen stellt. Eine tägliche Schwankung wäre administrativ nicht händelbar und würde auch den Arbeitsfluss einzelner Ressourcen erheblich beeinträchtigen. Nur um dies sicherzustellen, dürfen nicht nur die schon ausgeplanten Sprints Berücksichtigung finden. Der Ausblick wurde auf alle schon bekannten Themenblöcke angewendet und die proportionale Abarbeitung mit einbezogen. Dies wurde durch verschiedenste Berechnungen und Formeln, verteilt auf 63 Spalten, umgesetzt.

5.2 Reales anonymisiertes Praxisbeispiel

Für den weiteren Verlauf dieses Beispiels wird folgende Projektsituation angenommen und darauf aufgebaut: Im Rahmen des ersten Building Blocks (BB0) wurde ein Prototyp erfolgreich umgesetzt und alle notwendigen Tätigkeiten im Rahmen des Projektmanagements (Ressourcenallokation, Budgetplanung, grobe Projektplanung, Einführung von SCRUM, Verantwortlichkeitsmatrix, Abnahmekriterien, etc.) abgeschlossen. Der Prototyp beinhaltet die Bereitstellung des Reporting- und Data-Warehouse-Systems sowie die Implementierung eines Standardreports ohne Erweiterung der gegebenen Funktionalitäten. Das Projekt befindet sich somit vor dem Start des BB1. [vgl. BearingPoint GmbH 2013; vgl. BearingPoint GmbH und Weichenhain 2013]

Der allgemeine Aufbau des Projektes wurde, wie oben erwähnt, durch Building Blocks, Arbeitspakete (Identifikation von einzelnen Aufgaben) und Sprints (Bündelung von zu implementierenden Aufgaben) dominiert.

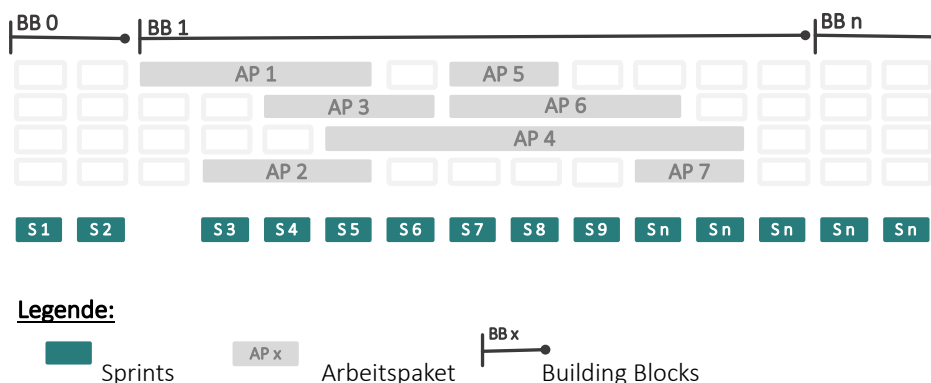


Abbildung 30: Zusammenhang Building Blocks, Arbeitspakete und Sprints

Das Projekt wurde in zwei unterschiedliche Dimensionen aufgeteilt. Das ist dem Mix der agilen und klassischen Vorgehensweise geschuldet. Die horizontale Dimension soll die fachliche Perspektive verdeutlichen, hier benannt als Fachkonzepte. Die vertikale Dimension wird als Implementationssprints bezeichnet. Es ist ersichtlich, dass ein Building Block n-Arbeitspakete parallel beinhaltet. Das führt dazu, dass ein Sprint Anforderungen aus n-Arbeitspaketen enthalten kann. Ein Arbeitspaket ist, soweit möglich, ein in sich geschlossenes Thema und enthält n-Aufgaben. In der Realität ist dies nicht machbar. Die Querschnittsthemen oder noch offenen Punkte wurden identifiziert und ebenfalls in eigenständige Arbeitspakete eingeteilt. Ein Arbeitspaket wurde in folgende Phasen unterteilt: Vorklärung – Abklärung – Design – Realisierung – Test – Einführung. Im Anhang F sind noch weitere Informationen zum anonymisierten Beispiel zu finden.

Die Dokumentation aller Anforderungen wurde durch den Fachbereich in einem Fachkonzept zusammengeführt. Einzelne fertige Abschnitte (Aufgaben) wurden so bald als möglich bestätigt und für die Sprints freigegeben. Die anschließende Dokumentation der Umsetzung wurde in der technischen Spezifikation festgehalten.

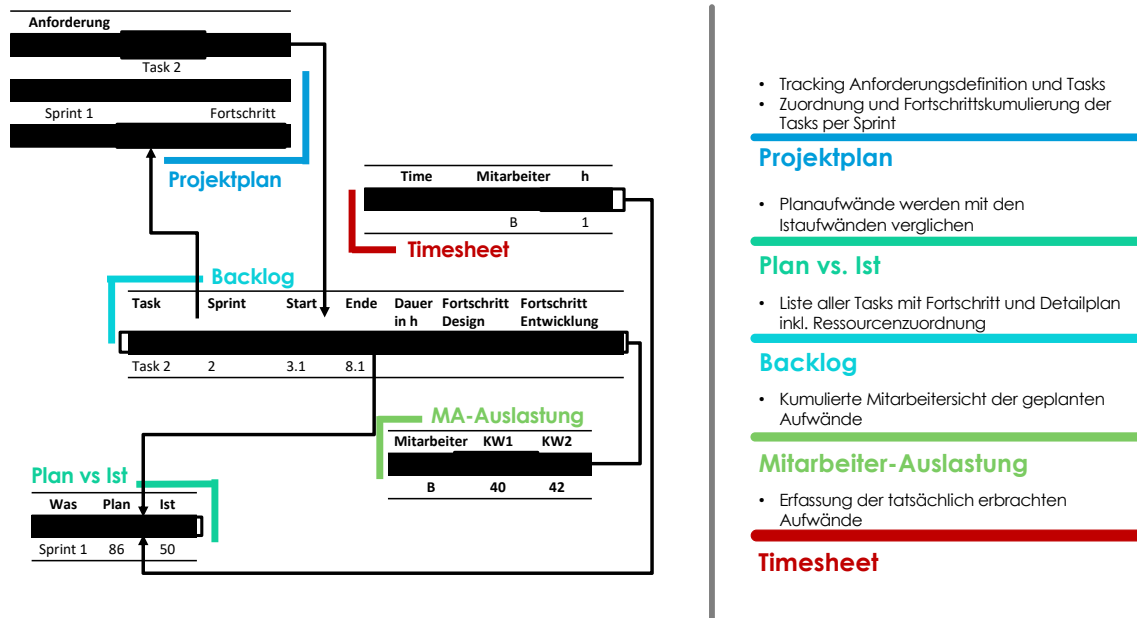
Nachfolgend eine kurze Beschreibung des inhaltlichen Zieles des betrachteten Building Blocks. Dieser hat sich zum Ziel gesetzt, Excel als Datenquelle anzubinden. Weiterhin soll die Definition eines Standardreports ohne Zusatzfunktionen resultieren. Dies wird beispielhaft am Thema Planzahlenabweichungsanalyse durchgeführt. Dieses fachliche Thema wird mit späteren Building Blocks weiter ausgebaut. Bei der Definition eines einheitlichen Standardreports müssen alle Fachbereiche mit einbezogen werden. Als Gesprächsgrundlage wird im Rahmen der Planzahlenabweichungsanalyse der erste Entwurf entwickelt und durch einen Abstimmungsprozess finalisiert. [vgl. BearingPoint GmbH und Weichenhain 2013]

Die Abweichungsanalyse bezieht sich auf Planzahlen und soll mit Hilfe eines Plan/Plan- bzw. Plan/Ist-Vergleichs Planzahlenänderungen systembasiert quantifizieren. Planzahlenänderungen umfassen in diesem Zusammenhang auch Abweichungen eingetretener Ist-Zahlen im Vergleich zu vorher bestehenden Planzahlen. Dazu werden dem ursprünglichen Planzahlenstand das tatsächlich eingetretene Ist, bzw. die geänderten Planzahlenstände gegenübergestellt und Differenzen berechnet. Darüber hinaus sollen die

einzelnen Differenzen in Relation zu der ursprünglichen Ausgangsbasis und in Relation zueinander gesetzt werden, um sie der relativen Höhe nach zu bewerten und kategorisieren zu können. [vgl. BearingPoint GmbH und Weichenhain 2013]

Um diese Komplexität zu managen, auch projektintern, bedarf es einer genauen Ausplanung und Abstimmung aller Aktivitäten. Grundlage für diese Abstimmung bildet eine gute Ressourcenallokation. Diese Schwachstelle war immer bemerkbar, wenn während des Projektes neue Abhängigkeiten entdeckt wurden oder Verzögerungen einzelner Anforderungen in einem Sprint entstanden sind. Somit waren von langer Hand ausgeplante Projektpläne anzupassen; verbunden mit der Aussage einer Machbarkeit und deren Auswirkungen. Aus realer Erfahrung ist festzuhalten, dass es oft an einer Simulation der geplanten Ereignisse scheiterte. Dies wurde, mittels komplexer Excel Tabellen und zur Unterstützung bei kurzfristig eintretenden Schwierigkeiten oder Verzögerungen, versucht auszugleichen. [vgl. BearingPoint GmbH und Weichenhain 2013] Das Zusammenspiel einiger Dokumente wird in der nachfolgenden Abbildung 31 verdeutlicht.

Die Komplexität entsteht durch das Zusammenspiel von mehreren Dateien und die Aussagen, welche ableitbar sind: Überplanung, Budgetüberschreitung, geplante Auslastung für die nächsten vier Wochen. Den Anfang bildet der Projektplan. Er dient hauptsächlich dazu, die fachlichen Aktivitäten zu steuern. Ein Auszug ist im Anhang C schemenhaft dargestellt. Aus dem Projektplan ist ersichtlich, dass die einzelnen Themengebiete und deren fachliche Anforderungen unterteilt wurden. Somit kann die Fertigstellung einzelner Anforderungsthemen festgehalten und überwacht werden. Die Anforderungen zu den fachlichen Themen werden teilweise versetzt fertig gestellt, um der Entwicklung so früh wie möglich Aufgaben übergeben zu können. Natürlich ist auch schon während der Definition der Anforderungen die Technik involviert. Ebenfalls wird überwacht, dass alle aufgekommene Aktivitäten (Aufgaben) aus den Anforderungen in die Backlog zu übertragen sind. Auch die optische Darstellung der Aufgabenzuordnung zu den Sprints und deren Fortschrittsstatus (gebildet aus einer Summe von Fortschrittsangaben aus der Backlog) waren im Projektplan ersichtlich.



- Tracking Anforderungsdefinition und Tasks
- Zuordnung und Fortschrittskumulierung der Tasks per Sprint

Projektplan

- Planaufwände werden mit den Istaufwänden verglichen

Plan vs. Ist

- Liste aller Tasks mit Fortschritt und Detailplan inkl. Ressourcenzuordnung

Backlog

- Kumulierte Mitarbeitersicht der geplanten Aufwände

Mitarbeiter-Auslastung

- Erfassung der tatsächlich erbrachten Aufwände

Timesheet

Abbildung 31: Grobes Zusammenspiel der Tabellen

Als ein weiteres Instrument dient die Backlog-Liste. Im Anhang F ist ein Auszug aus der Backlog-Liste zu finden. Hier wird nur ein Auszug dargestellt, da sie sehr groß und zu umfangreich ist. Aus der Backlog-Liste ist ersichtlich, in welchem Sprint die Aufgabe eingeplant ist; von wem sie hauptverantwortlich bearbeitet wird und wie der aktuelle Status in den einzelnen Phasen (Design, Entwicklung, Test, Dokumentation) ist. Hier werden die identifizierten und beschriebenen Anforderungen der einzelnen Aufgaben aufgenommen. Nach der Zuordnung zu den einzelnen Sprints und deren Tracking der Phasen wird der Gesamtstatus wieder zurück in den Projektplan übermittelt. Ebenfalls kann hier anhand der festgelegten Ressourcen und dessen Bearbeitungszeit pro Aufgabe ein Rückschluss auf die Planung pro Ressource im Zeitverlauf gezogen werden. Dies wird in einem weiteren Dokument (Plan vs. Ist) mit den Ist-Zahlen aus den Timesheets gegenübergestellt. Ebenfalls kann mit den so gewonnenen Plandaten eine Prognose auf die bevorstehende Mitarbeiterauslastung in den nächsten Wochen erfolgen.

Im Timesheet werden die wirklich erbrachten Stunden auf die jeweiligen Aufgaben aufgeteilt und einmal monatlich berichtet. Der Status des Projektes wurde auf Rot gesetzt, da eine Verspätung schon absehbar war. Es wurde damit gerechnet, dass bei anhaltendem Fortschritt die Verspätung noch weiter ausgebaut wird. Ebenfalls gab es ein schon bekanntes Ressourcenproblem.

6 Kapitel Die Projekt-Petri-Netz-Modell-Erweiterung

Das in Kapitel 4 entwickelte Modell wird mit Hilfe des aufgezeigten Beispiels aus Kapitel 5 umgesetzt. Dafür werden einzelne Aspekte aus dem Beispiel rausgenommen, um das Zusammenspiel der Konstrukte zu verdeutlichen.

Es wird aus dem Beispiel ein in seinem Umfang reduzierter Teilumfang ausgewählt und in die definierte Tabellenform transformiert. Darauf aufbauend werden einzelne Zusammenhänge des Beispiels im Kontext des PPNM veranschaulicht. Abschließend wird auf die automatische Generierung des PPNM eingegangen und seine Machbarkeit aufgezeigt.

6.1 Beispiel zur Umsetzung für den weiteren Verlauf der Arbeit

Die inhaltlich grobe Vorstellung ist bekannt und dieser Teilbereich konzentriert sich auf die Erstellung einer Projektplanung. Die nachfolgende Planung gibt einen Einblick in die Ausplanung eines Fachkonzeptes und zwei Sprints. Es besteht kein Anspruch auf Vollständigkeit, sondern soll lediglich als weitere Grundlage dienen. Die Planung des Projektes spielt sich im Zeitraum des ersten Quartals 2018 ab. Der allgemeine Status des Projektes ist als verspätet und über Budget gekennzeichnet. Die Perspektive wird vom 10.03.2018 eingenommen.



Abbildung 32: Projektbeispiel Fakten

In einem der nächsten Unterkapitel ist eine Skizze des PPNM zu finden, welche auf der folgenden Ausgangslage beruht. Die Daten dazu stammen aus dem Beispielprojekt aus Kapitel 5. Dargestellt werden die zwei Sprints im Rahmen des fachlichen Building Blocks der Planzahlenabweichungsanalyse innerhalb des Projektes. Der erste Sprint besteht hierbei aus drei Aufgaben, die sich jeweils in ihren Design- und Entwicklungsabschnitt aufteilen. Dem zweiten Sprint sind zwei Aufgaben zugeordnet. Für die Bearbeitung der Aufgaben stehen zwei Mitarbeiter, Julia Maier und Max Bauer sowie zwei Betriebsmittel zur Verfügung. Der Mitarbeiterin Julia Maier sind in diesem Beispiel zwei Regenerationsphasen in Form von Urlaub zugeordnet. Dem Mitarbeiter Max Bauer ist ein Urlaub zugeordnet. Eine Übersicht von der Projektplanung ist in den nachfolgenden Tabellen dargestellt.

6.2 Transformation in die Tabellenstruktur

Die Angaben aus dem Beispiel werden in das Zieltabellenformat transformiert. Die Dauer der einzelnen Phasen war jeweils in Tagen angegeben und es musste eine Umrechnung in das zugrunde gelegte Zeitkonzept erfolgen. Nachfolgend werden die fünf Tabellen mit den benötigten Informationen aufgezeigt. Die Prognosewerte waren nicht im Scope und die Spalten zur Aufnahme wurden aus der Tabelle entfernt.

Tabelle 7: Projektplanung – Beispiel

Projektplanung									
Vorgangs ID	Vorgangsname	Komplexität	Benötigtes Vorwissen min.	Vorgänger ID	Priorität	Sprintzuordnung	Design Dauer PLAN	Design Start PLAN	Design Ende PLAN
D_SS_001	Excel als Schnittstelle	4	0	-	2	2	6 h	22.01.18	26.01.18
R_D_001	Report Standardlayout	4	5	-	1	2	14 h	22.01.18	26.01.18
R_F_001	Report Filterfunktion	3	10	R_D_001	3	2	8 h	29.01.18	02.02.18
S_SS_001	Anbindung Schnittstelle	2	50	-	3	3	16 h	23.02.18	02.03.18
D_SS_001	Datenübertragung	4	10	S_SS_001	2	3	15 h	15.02.18	10.03.18

Vorgangs ID	Design Dauer IST	Design Start IST	Design Ende IST	Design Fortschritt IST	Entwicklung Dauer PLAN	Entwicklung Start PLAN	Entwicklung Ende PLAN	Entwicklung Dauer IST	Entwicklung Start IST	Entwicklung Ende IST	Entwicklung Fortschritt IST
D_SS_001	-	-	-	-	11 h	29.01.18	02.02.18	-	-	-	-
R_D_001	-	-	-	-	27 h	29.01.18	02.02.18	-	-	-	-
R_F_001	-	-	-	-	16 h	05.03.18	09.03.18	-	-	-	-
S_SS_001	-	-	-	-	15 h	04.03.18	15.03.18	-	-	-	-
D_SS_001	-	-	-	-	25 h	11.03.18	19.03.18	-	-	-	-

Tabelle 8: Ressourcen und Regenerationsphase – Beispiel

Ressourcen				
ID	Name	Art	Verfügbare Menge	Wiederverwendbar
R_V_01	Julia Maier	Mitarbeiter	-	Ja
R_V_02	Max Bauer	Mitarbeiter	-	Ja
R_ST_01	Drucker 001	Material	2	Ja
R_St_02	Papier	Verbrauchsmaterial	1000	Nein

Ressourcen ID	Regenerationsart	Start	Ende	Regelung AZ & AFZ
R_V_01	-	-	-	N: 16; m:40
R_V_01	Urlaub	02.01.18	26.01.2018	-
R_V_01	Urlaub	04.02.2018	04.04.2018	-
R_V_02	-	-	-	N: 16; m:40
R_V_02	Urlaub	15.01.18	18.01.18	-

Regenerationsphasen

Tabelle 9: Ressourcenzuordnung und Ressourcenbeurteilung – Beispiel

Ressourcenzuordnung				
Vorgangs ID	Zuordnung		Ressourcen ID	Benötigte Menge
	Bereich	Phase		
D_SS_001	Daten	Design	R_V_01	-
D_SS_001	Daten	Design	R_V_02	-
D_SS_001	Daten	Entwicklung	R_V_02	-
D_SS_001	-	-	R_ST_01	1
D_SS_001	-	-	R_ST_02	1
R_D_001	Report	Design	R_V_02	-
R_D_001	Report	Entwicklung	R_V_01	-
R_D_001	Report	Entwicklung	R_V_02	-
R_D_001	-	-	R_ST_01	1
R_F_001	Report	Design	R_V_02	-
R_F_001	Report	Entwicklung	R_V_02	-
R_F_001	-	-	R_ST_02	2
S_SS_001	System	Design	R_V_01	-
S_SS_001	System	Design	R_V_02	-
S_SS_001	System	Entwicklung	R_V_01	-
S_SS_001	-	-	R_ST_01	1
D_SS_001	Daten	Design	R_V_01	-
D_SS_001	Daten	Entwicklung	R_V_01	-
D_SS_001	Daten	Entwicklung	R_V_02	-
D_SS_001	-	-	R_ST_01	1

Ressourcen ID	Zuordnung		Vorhandenes Vorwissen
	Bereich	Phase	
R_V_01	System	Design	0
R_V_01	System	Entwicklung	0
R_V_01	Daten	Design	20
R_V_01	Daten	Entwicklung	0
R_V_01	Report	Design	20
R_V_01	Report	Entwicklung	10
R_V_01	Fachlich 1	Design	20
R_V_01	Fachlich 1	Entwicklung	10
R_V_02	System	Design	50
R_V_02	System	Entwicklung	50
R_V_02	Daten	Design	10
R_V_02	Daten	Entwicklung	10
R_V_02	Report	Design	0
R_V_02	Report	Entwicklung	5
R_V_02	Fachlich 1	Design	0
R_V_02	Fachlich 1	Entwicklung	0

Ressourcenbeurteilung

6.3 Erstellung und automatische Generierung des Beispiels

In der basierend auf der nach den Vorgaben des Autors erstellten und betreuten Masterarbeit von [Schott 2019] wurde das in diesem Kapitel eingeführte Beispiel mit Konstrukten umgesetzt. Die folgenden Ausführungen basieren auf dieser Masterarbeit. Dabei wurde der Umfang der Funktionalität der Konstrukte verringert sowie der Aufbau der Konstrukte vereinfacht. Das Grundprinzip ist aber identisch und beruht auf dem hier entwickelten PPNM. Dieses Kapitel soll die aufgegriffene Problematik der Erstellung und automatischen Generierung anhand der Abweichung von Plandaten und Ist-Daten beispielhaft betrachten. Zur Vereinfachung wird dabei auf eine automatische Neuplanung explizit verzichtet, sondern die Möglichkeit einer Rekonfiguration durch den Projektplaner offengelassen.

In Abbildung 33 ist das angewandte Vorgehen der experimentellen Toolunterstützung zur automatischen Generierung des PN veranschaulicht. Die Rohdaten des konkreten Projektes werden in Tabellen im Excel erfasst. Auf der Grundlage der unabhängigen Tabellenformate erfolgt mit Hilfe eines entwickelten Programms (hier in CSharp) das Einlesen der Daten für das experimentelle verwendete Modellierungstool PenecaChromos. Dies beinhaltet die korrekte Formatierung in das benötigte cne-Format und darauf aufbauend die Generierung des PNs. [vgl. Fengler 1990; vgl. Rokyta 1992]

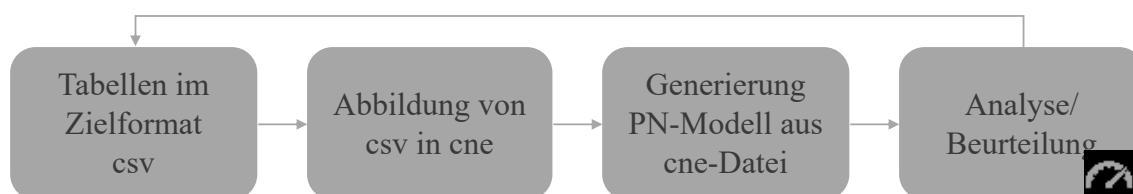


Abbildung 33: Schritte automatischer Generierung

Die Pfade der relevanten Tabellen werden dem Programm für das Parsen der Planungsdaten aus den Tabellen übergeben. Das Parsen, implementiert als eigene Klasse, erzeugt Listen zur Abspeicherung, welche vom Programm weiterverarbeitet werden können. Zur Abspeicherung der Listen existieren eigene Strukturen. Eine Zeile aus einer Tabelle entspricht einer Struktur und ein Feld dieser Zeile entspricht einem Feld in der Struktur. Durch diese gewählte Zuordnung kann das Programm einen einfachen Rückschluss auf die Planungsdaten bewerkstelligen. Zeilenweise werden die

Ausgangstabellen abgearbeitet und der Inhalt einer Tabelle wird in eine Liste von Strukturen des passenden Typs erzeugt. Der Projektstart sowie der Startzeitpunkt der Simulation müssen abgefragt und für die korrekte Abbildung berücksichtigt werden. Dabei wird objektorientiert jedes Konstrukt durch eine Klasse und dessen Schnittstellen vertreten. Die Struktur legt nur die Form der Abspeicherung fest und die Befüllung der Strukturen wird durch Funktionen vorgenommen. Nach dem erfolgreichen Parsen der Projektplanungsdaten aus den formalisierten Tabellen folgt die Erstellung des PPNM. Das wurde erreicht, indem die Konstrukte instanziiert wurden und jedes Konstrukt innerhalb des entwickelten Programms durch eine eigene Klasse repräsentiert wurde. Überwiegend findet eine eins zu eins Zuordnung der Petri-Netz-Komponenten zu ihrer Repräsentation innerhalb des Programms statt. Eine Besonderheit des Programms war, dass das Konstrukt der Aufgabe in zwei Klassen unterteilt wurde, damit eine Kapselung des Vorgänger-Nachfolger-Konzepts von der eigentlichen Aufgabe abgebildet werden konnte.

Nicht nur die schon erwähnten Anfangsmarkierungen sind wichtig bei der Erstellung des PN, sondern ebenfalls die Reihenfolge der Elemente im Konstrukt. Hier soll beispielhaft auf die Reihenfolge bei der Sprinterstellung eingegangen werden, wie in Abbildung 34 ersichtlich.

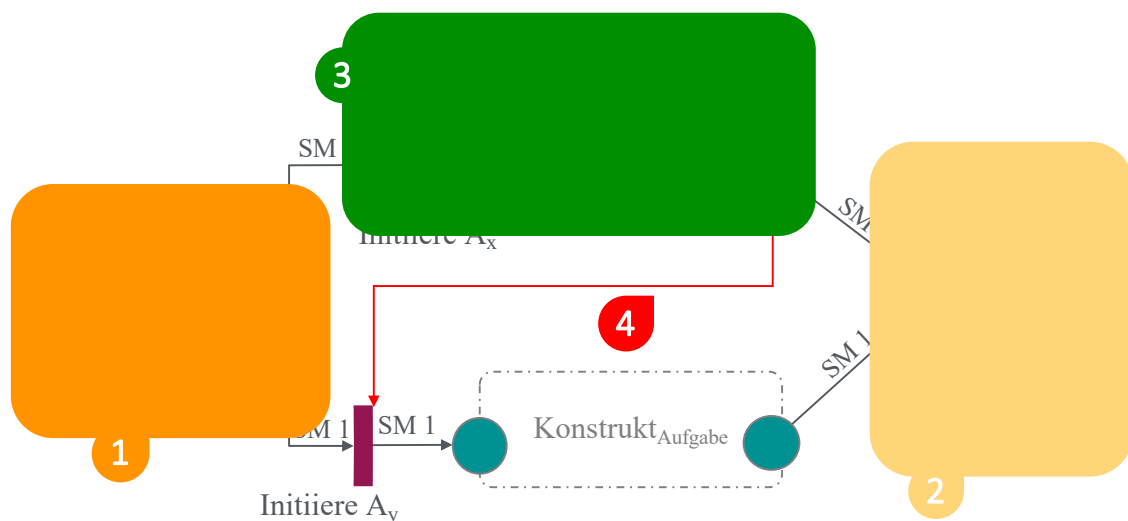


Abbildung 34: Reihenfolge Sprinterstellung – abgeänderte Grundlage

Den Anfang macht die Ermittlung der Anzahl an Tasks, es folgt daraus die Erstellung der Startelemente. Als Nächstes folgt das Ende des Konstrukts. Mit den zwei Schritten sind alle Ansatzpunkte für eine Aufgabe gegeben und die einzelne Aufgabe kann nun instanziiert werden. Dies geschieht für alle Aufgaben, welche diesem Sprint zugeordnet sind. Zuletzt wird nach der Instanziierung aller Aufgaben überprüft, ob Vorgängerbeziehung zwischen den Aufgaben bestehen. Die notwendigen Kanten, falls Vorgängerbeziehungen vorhanden, müssen eingefügt werden.

Während der Durchführung eines Projekts kann es jederzeit zu Abweichungen von Planungsdaten kommen. Hier wird das Plan-Ist-Modell (Abbildung 35) betrachtet, welches sowohl auf Planungsdaten und aktuellen Ist-Daten beruht. Die Abgrenzung von Planungsdaten und Ist-Daten wurde bei der Entwicklung des Tabellenformats berücksichtigt.

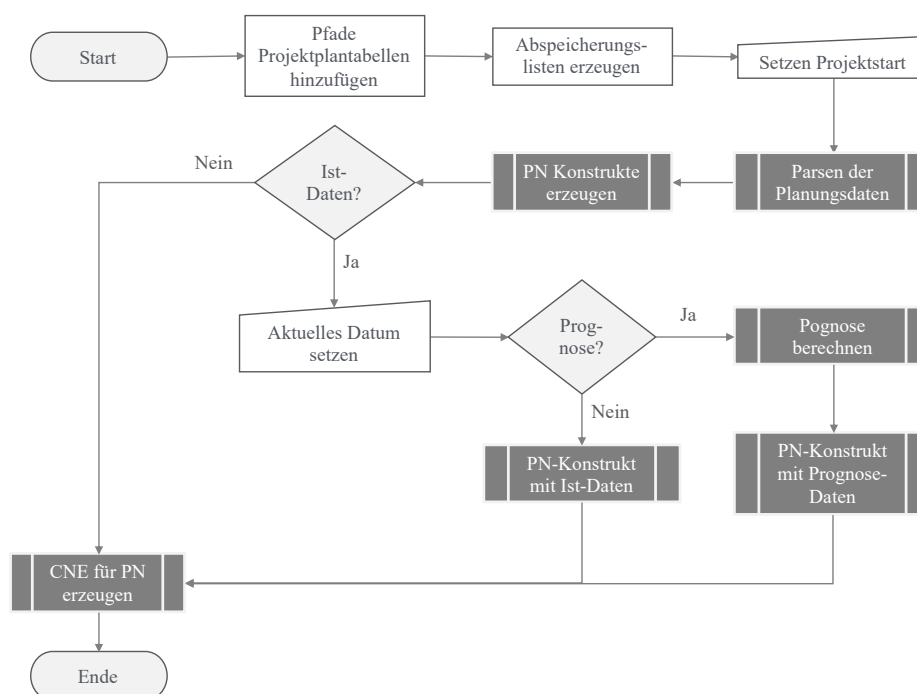


Abbildung 35: Überblick Programm zur Erstellung des Plan-Ist-Modells

Zu Beginn des Programms werden die Pfade der Projektplanungstabellen hinzugefügt und Listen zur Abspeicherung der Daten aus diesen Tabellen erzeugt. Auf der Grundlage des Datums für den Projektstart wird das Parsen der Planungsdaten aus den Tabellen vorgenommen. Nachdem die Ist-Daten bis zum Stichtag beachtet worden sind, werden zur Vereinfachung die Plandaten ohne weitere Logik berücksichtigt. Für das Plan-Ist-

Modell ist die Berechnung der korrekten Anfangsmarkierung ein ausschlaggebender Berechnungsschritt, welcher für die jeweiligen Konstrukte zu durchlaufen ist.

In der Umsetzung wurde ebenfalls eine stark vereinfachte Variante der Prognose für die Berechnung von auftretenden Verzögerungen mit einbezogen. Die Genauigkeit der Simulation und dessen Aussage wurde somit erhöht. Dazu wurde zeilenweise geprüft, ob Ist-Daten vorhanden sind oder auf Grundlage von angegebenen Plandaten ein Prognosewert zu berechnen ist. Dies wurde für jede Aufgabe durchgespielt und eine proportionale Berechnung der Verzögerung erfolgte.

6.4 Beispieldarstellung als Projekt-Petri-Netz-Modell stark zusammengefasst

Zur Umsetzung des PPNM mit den Beispieldaten aus Kapitel 6.2 müssen nachstehende Konstrukte generiert werden: Zwei Sprints, fünf Aufgaben mit Design- und Entwicklungsphase, zwei Ressourcengenerationen Typ Zeiteinheit (Mitarbeiter), eine Ressourcenregeneration Typ Mengeneinheit (Betriebsmittel), ein globales Zeit-Konstrukt, ein Ressourcenkonsum und eine Ressourcenverwaltung.

In der Abbildung 36 ist das zu entwickelnde PPNM mit allen benötigten Konstrukten skizziert, wobei die Anforderungen an die Darstellungsform berücksichtigt worden sind.

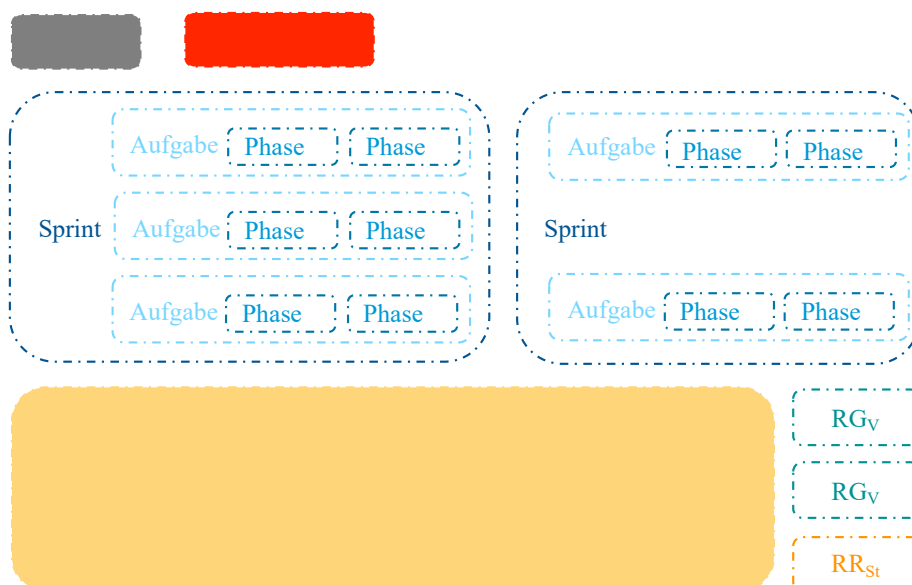


Abbildung 36: Projekt-Petri-Netz-Modell Skizze – Beispiel

6.5 Ausschnitte des generierten Projekt-Petri-Netz-Modells

Im Fokus steht das Zusammenspiel der einzelnen Konstrukte im PPNM. Die folgenden Ausführungen basieren auf der nach den Vorgaben des Autors erstellten und betreuten Masterarbeit von [Schott 2019]. Die Masterarbeit beinhaltet eine ausführliche Beschreibung des generierten Beispiels eines vereinfachten Modells, was das Zusammenspiel gut aufzeigt und die Machbarkeit beweist. In Kapitel 4.4.2 sind alle Beschreibungen und Konstrukte sowie dessen Kommunikation mit anderen Schnittstellen zu finden. Deswegen wird sich in diesem Abschnitt nur auf die Kommunikation von zwei ausgewählten Bereichen aus dem PPNM begrenzt.

Anfangen mit dem lokalen Zeit-Konstrukt ist in Abbildung 37 zu erkennen, welche Konstrukte mit der lokalen Zeit gleich getaktet werden. Die Ressourcenregeneration vom Typ Stückzahl regeneriert die wiederverwendbaren Ressourcen beispielsweise unabhängig vom Zeitpunkt, sondern allein, wenn die Ressource durch die Aufgabe wieder freigegeben wurde.

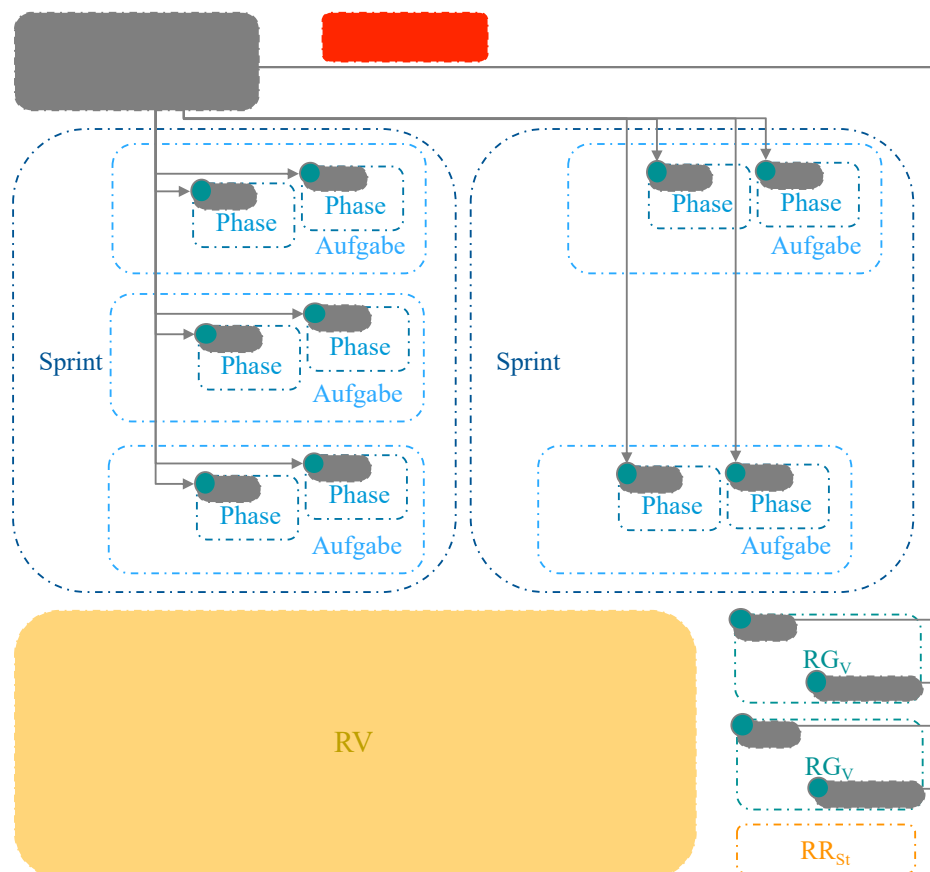


Abbildung 37: Beispiel Projekt-Petri-Netz-Modell - Zeit-Konstrukt Gesamtblick

Zur besseren Darstellung wird vorgeschlagen, die Kanten vom lokalen Zeit-Konstrukt mit der Farbe Weiß darzustellen, was erheblich zur späteren Lesbarkeit beiträgt. Dies kann ebenfalls für den Ressourcenkonsum adaptiert werden, da aus diesem Konstrukt keine weiteren Aktionen mehr hervorgehen.

Um das Zusammenspiel der Aufgaben untereinander zu verdeutlichen, ist in der Abbildung 38 ein Ausschnitt mit ausgewählten Elementen zu finden. Die relevanten Elemente für die Vorgänger sind Rot hervorgehoben.

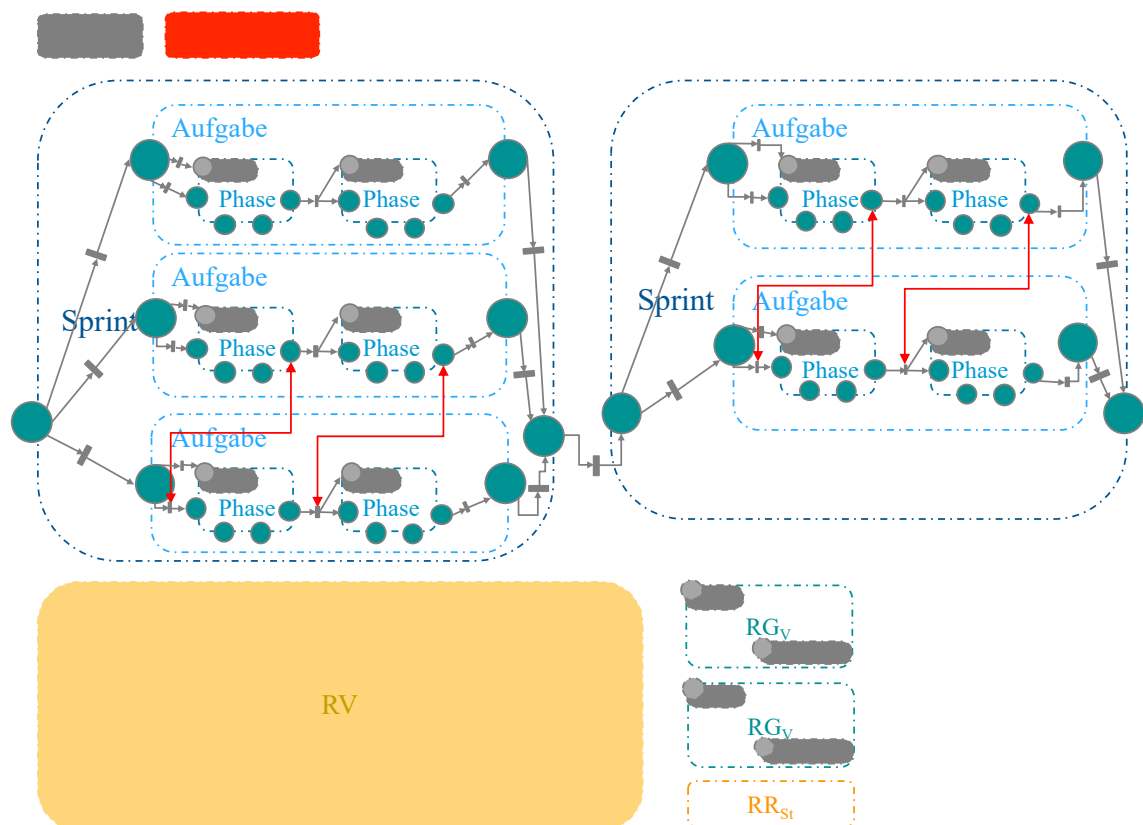


Abbildung 38: Beispiel Projekt-Petri-Netz-Modell - Aufgabe und Vorgänger

Beim Ablauf einer Phase wird die zur Phase benötigte Dauer in Form von Marken im Platz *Unerledigt* modelliert. Wie in Abbildung 39 zu sehen, werden auch hier die relevanten Elemente Rot hervorgehoben. Nachdem der Platz *Unerledigt* mit Marken befüllt ist, ist der frühestmögliche Zeitpunkt zur Abarbeitung der Phase eingetreten und die erforderliche Ressourcenanforderungen zur Abarbeitung mit der Dauer mitgegeben. Sobald auch die Ressourcenbereitstellung erfolgreich war, und durch die entsprechende Marke im Platz *Abarbeitung* signalisiert ist, wird mit dem nächsten Takt die Dauer reduziert, bis die Transition *Aufgabe* ihre Schaltfähigkeit verliert, da alle Marken aus

diesem Platz verschaltet sind. Wie schon vorher erwähnt, muss dafür die Angabe der Dauer in der Tabelle ein Vielfaches des zugrunde gelegten Zeitkonzeptes entsprechen. Die Erledigung einer Phase benötigt genauso viele Zeittakte, wie als Dauer angegeben worden ist und somit wird der Fortschritt in einem anderen Platz festgehalten. Dies gewährleistet, dass aus der Addition der vorhandenen Markenanzahl der beiden Plätze *Abarbeitung* und *Fortschritt* die angegebene Dauer berechnet werden kann. Die Angabe der Dauer muss mit der Angabe des frühestmöglichen Startzeitpunktes gekoppelt werden. Dies wird, wie in diesem Kapitel schon beschrieben, über eine Testkante zum Platz *Zeitpunkt* realisiert.

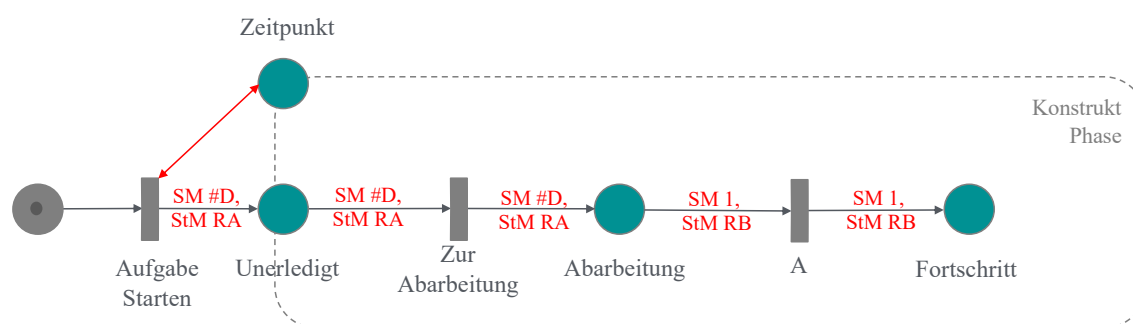


Abbildung 39: Zeit-Konstrukt – Zugriff auf Dauer im Aufgaben-Konstrukt

7 ^{Kapitel} Zusammenfassung und Ausblick

In erster Betrachtung kommen das Projektmanagement und die Petri-Netze aus zwei unterschiedlichen Welten. Versucht man die zwei Welten trotzdem zusammen zu bringen, kommen viele Gemeinsamkeiten zum Vorschein und es besteht die Möglichkeit, vorteilhafte Synergieeffekte entstehen zu lassen. Die vorliegende Arbeit hatte sich zum Ziel gesetzt, ein leistungsfähiges PN-Modell zur Abbildung und Simulation von Projektmodellierungen zu entwerfen.

Die Arbeit widmet sich vor allem dem methodischen Ansatz und der damit geschaffenen Möglichkeit der Umsetzung. Letzteres erfolgt exemplarisch, das vorgeschlagene PPNM soll mit Einschränkungen formalisiert werden. Das Modell ist ein Kreislauf und wurde abschließend experimentell mit dem Petri-Netz-Tool PenecaChromos umgesetzt.

7.1 *Inhalt der Arbeit*

Anfangs wurde ein grundlegendes Verständnis für Methoden und Definitionen in Bezug auf das Projektmanagement (agil und klassisch) und der Petri-Netze hergestellt. Unter Berücksichtigung des dynamischen und komplexen Charakters heutiger Projekte wurden die eher typischen Einsatzfelder von PN, wie das Geschäftsprozess- und Workflowmanagement, vom PM abgegrenzt. Dabei haben die gefundenen Gemeinsamkeiten den Ansatz bekräftigt, PN zur Modellierung von Projektmanagement heranzuziehen. Dies ist neben einem formal-mathematischen Formalismus auch einem breiten, anpassbaren und erweiterbaren Funktionsumfang geschuldet. Zusätzlich kann ein Einsatz in Echtzeit und eine Skalierung zum Projektumfang stattfinden, was dem agilen Projektvorgehen sehr zu Gute kommt. Auch etwaige Nachteile (beispielsweise hoher Aufwand, zwingende SW-Unterstützung, schwer lesbare Darstellung ohne spezifische Kenntnisse) wurden betrachtet und erfordern, wenn gewollt, eigene Erweiterungen in der Zukunft.

Das Ziel dieser Arbeit ist nicht die Ersetzung eines vollumfänglichen PM-Tools, sondern die Unterstützung des PM, bei dem ein derartiger Tooleinsatz aus inhaltlichen und wirtschaftlichen Gründen nicht bevorzugt wurde. Der Bedarf einer modularen, in Echtzeit

simulierbaren Methode, beruht auf den Unzulänglichkeiten von herkömmlichen PM-Tools (beispielsweise keine Möglichkeit einer automatischen Neuplanung oder angemessenen Konfliktlösung, ebenso keine Simulationsmöglichkeiten) und der Feststellung, dass in vielen Projekten Excel als Ersatz oder zumindest begleitend zur Projektplanung im Einsatz ist. Die Schwäche dieses Vorhabens ist die fehlende Formalisierung, was mit einem ständigen Neuentwurf von Vorlagen, Komplexität von Tabellen und nicht mehr überschaubarer integrierter Logik als Einzellösung zugeschnitten für das jeweilige Projekt einhergeht. Gerade bei einem agilen Projekteinsatz gehören Änderungen zum Konzept und eine variierbare Sprintzusammensetzung ist unumgänglich. Mit dem hier vorgestellten Modell kann ein einheitlicher Zugang der zugrunde gelegten Logik im PM durch das PN gewährleistet werden. Dies ermöglicht eine effiziente und zentrale Abbildung von Änderungen, gänzlich unabhängig vom Tooleinsatz. In bisher unternommenen Untersuchungen wurde noch nicht hinreichend erforscht, wie ein konsistentes Framework zur Modellierung und Analyse von Projekten gestaltet werden kann, aufbauend auf einer formalisierten Tabellenstruktur.

Dafür müssen für die Modellierung des Projektmanagements deren komplexe Aufgaben abstrahiert und wie bei Petri-Netzen eine Zerlegung des gesamten Modells in Teilsysteme inkl. deren Wechselwirkungen und deren Komponenten erfolgen. Zur Integration von agilen Projektmethoden wurden Bestandteile von SCRUM, als eines der bekanntesten Vertreter, mit im Modell berücksichtigt. Um eine geeignete gemeinsame Abstraktionsebene zu finden, wurden anfangs mit Hilfe einer Definition der atomaren Einheit die benötigten Anforderungen an einer Abbildung des PM durch PN näher betrachtet.

Die Untersuchungen zur Identifizierung der Anforderungen beruhen auf realen Projektbeispielen aus der Praxis. Die Erfahrungen sowie die enthaltene Analyse von PM hat zu dem hier vorgestellten systematischen Modell geführt. Dabei wurde eine einfach handhab- und erweiterbare Tabellenstruktur gefordert, welche eine in Echtzeit simulierbare Projektplanung ohne notwendige PN-Kenntnisse ermöglicht. Weiterhin sollen sich aus den Tabellen voll automatische PN-Konstrukte generieren lassen, welche verschiedenste Vorgänger-Nachfolger-Beziehungen und somit auch verschiedenste Dynamiken abbilden können. Die Auswahl der zu nutzenden PN-Erweiterungen muss so

erfolgen, dass brauchbare und funktionsfähige Projektmodelle abbildbar sind. Dies beinhaltet auch die vielfachen Anforderungen von realen Projekten, gekennzeichnet durch ihre hohe Dynamik, teils großen Umfängen, Unsicherheiten und hochgradiger Komplexität.

Als Modellierungsgegenstand wurden Projekt-Objekte mit deren Operationen und Beziehungen zueinander identifiziert. Auf dieser Basis konnten die Eigenschaften von Objekten klassifiziert und deren Anforderungen abgeleitet werden, was eine formale und einheitliche Darstellung gewährleistet. Als nächstes wurde eine formale Tabellenstruktur und somit eine Methode zur Überführung und Adaption des PMs in ein PN definiert. Die Dynamik der Projekte erfordert einen Kreislauf der formalen Modellierung von der Dateneingabe bis zur Erkenntnisgewinnung.

Die Grundidee beruht darauf, die Dateneingabe von der Planung und den aktuellen Zwischenständen in einer formalisierten und toolunabhängigen Tabellenstruktur vorzunehmen und im Hintergrund automatisch das PN zu generieren bzw. zustandsabhängig zu verändern. Das PN wird mit geeigneten leistungsfähigen Erweiterungen und mit erforderlichen Interpretationen aus dem Gebiet der höheren, gefärbten und zeitbewerteten Petri-Netze mit strukturierten Marken zur Abbildung und Simulation verwendet. Das so generierte PN mit seinen Strukturen und Simulationsmöglichkeiten, mit seinen Anfangs- und Zwischenmarkierungen kann das PM unterstützen, indem die Ergebnisse wieder in die Tabelle zurückgeführt werden und mit in die Projektentscheidungen einfließen. Dies wird durch die Pflege der Tabellen mit aktuellen Informationen im Lebenszyklus der Planung realisiert. Die Tabelle bildet die Grundlage zur Überführung in die korrespondierenden Zustände des definierten PNs (Markierungen und Parameter). Dafür werden Konstrukte aufgezeigt, welche über Schnittstellen kommunizieren und so modular und beliebig einsetzbar sind.

Die Bildung der Konstrukte hat sich an den Projekt-Objekten orientiert. Die Vorteile des PN-Konzeptes können so vollumfänglich ausgenutzt und für den Projektmanager auch ohne PN-Kenntnisse verwendbar gemacht werden. Dabei ist die Methode dynamisch und für verschiedenste Projekte universell einsetzbar. Hiermit wurde für das PM eine effiziente und flexible Methodik realisiert, um geeignete Simulationen und Auswertungen für die Steuerung und Entscheidungsfindung während des Projektes in Echtzeit

durchzuführen. Bei der experimentellen Umsetzung mit dem Petri-Netz-Tool PenecaChromos wurde der Fokus nicht auf eine ausgereifte Nutzerfreundlichkeit in der grafischen Darstellung gelegt, sondern vielmehr auf die Machbarkeit und das Aufzeigen der theoretischen Möglichkeiten.

Es wurde auf wichtige Schritte und zu beachtende Faktoren bei der Generierung des PPNM hingewiesen. Bei der experimentellen Toolumsetzung wurde ein ausgewähltes und eingeschränktes Projektbeispiel vereinfacht realisiert. Dabei wurde das Konzept der strukturierten Marken nur vereinfacht berücksichtigt und hat erhebliches Entfaltungspotenzial. Dagegen wurden Transformationsregeln aus der formalisierten Tabelle einschließlich der zusätzlichen Hierarchieinformationen durch geeignete erweiterte PN aufgestellt und in die Umsetzung mit einbezogen. Die automatische Generierung und die Modularität der zu erzeugenden PN-Konstrukte wurde ebenfalls berücksichtigt.

7.2 Möglichkeiten, Grenzen und zukünftige Arbeiten

Für die Modellierung und Simulation von Projektmanagementaufgaben mit dem Einsatz von Petri-Netzen bedarf es einer praktischen Umsetzung und flexiblen Gestaltung von Projekten. Es hat sich gezeigt, dass sowohl klassische als auch agile Projektmethoden umgesetzt werden können. Ein großer Vorteil bei diesem entwickelten Modell ist, dass vom Projektmanager keine PN-Kenntnisse erforderlich sind. Dabei gilt es vorweg zu erwähnen, dass die Grenzen und Handhabung solch einer Modellierung auf der Einteilung und der benötigten Detailierungsebene liegen. Was oft davon abhängig ist, in wie weit die Stakeholder des Projektes ins Geschehen eingreifen möchten oder Managemententscheidungen obliegen – was wiederum eine Auswirkung auf die Granularität der atomaren Einheit ausübt. Die atomare Einheit hingegen definiert, welche Aspekte in welcher Granularität bei der Modellierung berücksichtigt werden und stellt somit die Grenze modellierbarer Sachverhalte dar. Auch ist bei der Wahl des Tools zur praktischen Umsetzung erhebliches Potenzial vorhanden. Ein Nachteil der in dieser Arbeit entworfenen Abbildung der Projektplanung liegt in der fehlenden Differenzierung von Überziehungen.

Die genannten Schwachstellen, wie die Ursachenanalyse für Überziehungen, könnten in weiterführenden Untersuchungen verifiziert und validiert werden. Zum jetzigen Stand wird lediglich aufgezeigt, dass eine Überziehung vorliegt. Eine Unterscheidung der Gründe für die Überziehung könnte mit der Anreicherung und dem Einsatz von strukturierten Marken ermöglicht werden. Damit wären auch geeignete Anpassungen der Planung im Netz heraus modellierbar, mit der einer Überziehung entgegengewirkt werden kann.

Die gesamte Bandbreite an PN-Erweiterungen kann aus unterschiedlichsten Gebieten (Workflow, Stochastik, Fuzzy-Logik, etc.) ihren Einsatz im PM finden. Ferner könnten die Ansätze aus weiteren benachbarten Forschungsfeldern, z. B. kritischer Pfad, im Kontext der zeitbewerteten PN mit aufgenommen und integriert werden. Diese Arbeit besitzt den Charakter einer methodischen Untersuchung mit einem prinzipiellen Nachweis der Umsetzbarkeit, die gegebenenfalls im Rahmen weiterführender Arbeiten mit Experimenten fortgeführt werden könnte.

Die aufgeführten Folgeaufgabestellungen beziehen sich auf weniger prinzipielle, aber sinnvolle Erweiterungen im Rahmen der entwickelten Methode. In dem hier entworfenen Petri-Netz-Modell werden beispielsweise der Ressource Mitarbeiter wenige Kriterien bei der Modellierung mitgegeben. Der Zugang zur Ressourcenoptimierung ist gegeben und muss ausgearbeitet werden. Hier wäre eine Erweiterung gerade für Simulationen und Prognosen interessant, wie z. B. die Eignung, Schnelligkeit oder gar der Einsatz von Fuzzy-Logik. Eine Schwierigkeit besteht hier allerdings bei der Messbarkeit der Eigenschaften. Auch die Regenerationsphasen von Ressourcen könnten um viele allgemeine Informationen, wie Urlaub und Feiertage, erweitert werden. Momentan müsste dies pro Ressource in dessen Regenerations-Konstrukt erfolgen – hier wäre eine globale Steuerung eine Verbesserung. Ein weiterer Punkt wäre die Simulation an sich. Diese könnte manuell gesteuert, zeitlich bestimmt oder gar event-basiert geschehen (beispielsweise durch einen Konflikt oder Ressourcenverbrauch). Überdies ist die grafische Darstellung im Generellen ein wichtiger Punkt und bietet großes Potential. Bekanntlich ist, um so komplexer das abzubildende Projekt wird, um so komplexer ist die grafische Darstellung. Zusätzlich könnten für eine übersichtlichere Darstellung alle Kanten, die für eine Beurteilung irrelevant sind, durch die Farbe Weiß dargestellt werden.

Literaturverzeichnis

Aalst, W. M. P. 1998. „Three Good Reasons for Using a Petri-Net-Based Workflow Management System“. In *Information and Process Integration in Enterprises*, herausgegeben von Toshiro Wakayama, Srikanth Kannapan, Chan Meng Khoong, Shamkant Navathe, und JoAnne Yates, 161–82. Boston, MA: Springer US. https://doi.org/10.1007/978-1-4615-5499-8_10.

Aalst, Wil M. P. van der. 2011. *Process Mining*. Berlin, Heidelberg: Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-19345-3>.

———. 2013. „Business Process Management: A Comprehensive Survey“. *ISRN Software Engineering* 2013: 1–37. <https://doi.org/10.1155/2013/507984>.

Aalst, W.M.P. van der. 1994. „Putting High-Level Petri Nets to Work in Industry“. *Computers in Industry* 25 (1): 45–54. [https://doi.org/10.1016/0166-3615\(94\)90031-0](https://doi.org/10.1016/0166-3615(94)90031-0).

Aalst, W.M.P. van der, und K.M. van Hee. 1996. „Business Process Redesign: A Petri-Net-Based Approach“. *Computers in Industry* 29 (1–2): 15–26. [https://doi.org/10.1016/0166-3615\(95\)00051-8](https://doi.org/10.1016/0166-3615(95)00051-8).

Abel, Dirk. 2014. *Regelungstechnik und Ergänzungen (Höhere Regelungstechnik) Umdruck zur Vorlesung [Hauptbd.] [Hauptbd.* Herausgegeben von RWTH Aachen und Institut für Regelungstechnik. Aachen: Mainz.

Ahlemann, Frederik. 2002. „Das M-Modell Eine konzeptionelle Informationssystemarchitektur für die Planung, Kontrolle und Koordination von Projekten (Projekt-Controlling)“. Fachgebiets Betriebswirtschaftslehre/Organisation und Wirtschaftsinformatik: Universität Osnabrück Uwe Hoppe.

Aichele, Christian. 2006. *Intelligentes Projektmanagement*. Stuttgart: Kohlhammer.

Al Ali, Khaoula. 2010. „Modellierung und Verifikation von verteilten/parallelen Informationssystemen“. Dissertation, Ilmenau: TU Ilmenau, Fakultät für Informatik und Automatisierung.

Ali, Khaoula Al, Wolfgang Fengler, und Bernd Däne. 03.07.2011. „Extended coloured Petri nets with structured tokens formal method for distributed systems.“ In *Proceedings of the 2011 Symposium on Theory of Modeling & Simulation: DEVS Integrative M&S Symposium (TMS-DEVS)*. Bd. 4. Boston, MA, USA.

Allen, Rob. 2000. *Workflow Handbook 2001: Published in Association with the Workflow Management Coalition WfMC*. Herausgegeben von Layna Fischer. Lighthouse Point, Fla: Future Strategies Inc.

Allisy-Roberts, P., P. Ambrosi, D. T. Bartlett, B. M. Coursey, L.A. DeWerd, E. Fantuzzi, und J.C. McDonald. 2006. „EXECUTIVE SUMMARY“. *Journal of the ICRU* 6 (2): 7–8. <https://doi.org/10.1093/jicru/ndl025>.

Aloini, Davide, Riccardo Dulmin, und Valeria Mininno. 2012. „Modelling and Assessing ERP Project Risks: A Petri Net Approach“. *European Journal of Operational Research* 220 (2): 484–95. <https://doi.org/10.1016/j.ejor.2012.01.062>.

Ashok Kumar, V. K., und L. S. Ganesh. 1998. „Use of petri nets for resource allocation in projects“. *IEEE Transactions on Engineering Management* 45 (1): 49–56. <https://doi.org/10.1109/17.658660>.

Badiru, Adedeji B. 1995. „Incorporating learning curve effects into critical resource diagramming“. *Project Management Journal* 26 (2): 38–45.

Baumgarten, Bernd. 1990. *Petri-Netze: Grundlagen und Anwendungen*. Bibliographisches Inst. Mannh.

Bause, Falko, und Pieter S. Kritzinger. 1998. „Stochastic Petri Nets: An Introduction to the Theory“. *ACM SIGMETRICS Performance Evaluation Review* 26 (2): 2–3. <https://doi.org/10.1145/288197.581194>.

Bauske, Joachim. 1999. *Ein objektorientiertes Verfahren zur Optimierung von Geschäftsprozessen unter Verwendung eines genetischen Algorithmus*. Berlin, Heidelberg: Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-47977-9>.

BearingPoint. 2013. „Client Projekthandbuch“. internes Material / unveröffentlichtes Dokument; Mitautor. Wolfsburg.

BearingPoint GmbH. 2013. „Client Kick-Off Folien“. Client Projekt.

———. 2015. „Messbare Verbesserung der Leistungsfähigkeit durch Prozessmanagement“. *Business Process Management - Studie*, 2015. <http://www.bearingpoint.com/de-de/adaptive-thinking/insights/business-process-management-studie-2015>.

BearingPoint GmbH, und Maxi Weichenhain. 2013. „Client Projekthandbuch“. Client Projekt.

———. 2017. „Client Projektbeschreibung“. Client Projekt.

Becker, Jörg, und Dieter Kahn. 2012. „Der Prozess im Fokus“. In *Prozessmanagement*, herausgegeben von Jörg Becker, Martin Kugeler, und Michael Rosemann, 3–16. Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-33844-1_1.

Becker, Jörg, Martin Kugeler, und Michael Rosemann, Hrsg. 2012. *Prozessmanagement*. Berlin, Heidelberg: Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-33844-1>.

Becker, Jörg, Christoph Mathas, und Axel Winkelmann. 2009. *Geschäftsprozessmanagement*. Informatik im Fokus. Berlin, Heidelberg: Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-540-85155-4>.

Belhe, Upendra, und Andrew Kusiak. 1993. „Performance Analysis of Design Process Using Timed Petri Nets“. *Concurrent Engineering* 1 (3): 147–52. <https://doi.org/10.1177/1063293X9300100302>.

Benecke, Carsten. 1995. „Modellierung sprachlicher Handlungsmuster mit Petri Netzen“. Die vorliegende Arbeit wurde im Rahmen des Verbundvorhabens Verbmobil vom Bundesministerium für Bildung, Wissenschaft, Forschung und Technologie (BMBF) unter dem Förderkennzeichen 01 IV 101 A/O gefördert. Gehört zum Antragsabschnitt 13.4 Materialaufnahme. Uni Hamburg. Hamburg. <https://www.rz.uni-hamburg.de/ueber-uns/personen/6-net/benecke/daten/verbmobil-memo81-sig-pdf.pdf>.

Bevilacqua, Maurizio, Filippo Emanuele Ciarapica, und Mazzuto Giovanni. 2018. „Timed Coloured Petri Nets for Modelling and Managing Processes and Projects“. *Procedia CIRP* 67: 58–62. <https://doi.org/10.1016/j.procir.2017.12.176>.

Bobbio, Andrea. 1990. „System Modelling with Petri Nets“. In *Systems Reliability Assessment*, herausgegeben von A. G. Colombo und A. Saiz de Bustamante, 6:103–43. Dordrecht: Springer Netherlands. https://doi.org/10.1007/978-94-009-0649-5_6.

Brecht-Hadraschek, Barbara, und Rainer Feldbrügge. 2013. *Prozessmanagement: Geschäftsprozesse analysieren und gestalten*. 3., aktualisierte und erw. Aufl. Alles, was sie wissen müssen. München: Redline-Verl.

Burghardt, Manfred. 2012. *Projektmanagement Leitfaden für die Planung, Überwachung und Steuerung von Projekten*. Erlangen: PUBLICIS. <http://nbn-resolving.de/urn:nbn:de:101:1-201408082591>.

Busch, Carsten. 1998. *Metaphern in der Informatik Modellbildung -- Formalisierung -- Anwendung*. Wiesbaden: Deutscher Universitätsverlag. <http://www.springerlink.com/content/978-3-663-08761-8>.

Chang, C.K., C. Chao, T.T. Nguyen, und M. Christensen. 1998. „Software Project Management Net: a new methodology on software management“. In *Proceedings. The Twenty-Second Annual International Computer Software and Applications Conference (Compsac '98) (Cat. No.98CB 36241)*, 534–39. Vienna, Austria: IEEE Comput. Soc. <https://doi.org/10.1109/CMPSAC.1998.716715>.

Chang, C.K., und M. Christensen. 1999. „A net practice for software project management“. *IEEE Software* 16 (6): 80–89. <https://doi.org/10.1109/52.805479>.

Cheng, Feifei, Heng Li, Y.-W. Wang, Martin Skitmore, und Perry Forsythe. 2013. „Modeling Resource Management in the Building Design Process by Information Constraint Petri Nets“. *Automation in Construction* 29 (Januar): 92–99. <https://doi.org/10.1016/j.autcon.2012.08.005>.

Chun-suk Park, Gang-soo Lee, und Jung-mo Yoon. 1993. „An executable software project management model by beta-distributed stochastic Petri nets“. In *Proceedings of TENCON '93. IEEE Region 10 International Conference on Computers, Communications and Automation*, 430–34. Beijing, China: IEEE. <https://doi.org/10.1109/TENCON.1993.320019>.

Cohen, Yuval, und Ofer Zwikael. 2008. „Modelling and Scheduling Projects Using Petri Nets“. *International Journal of Project Organisation and Management* 1 (2): 221. <https://doi.org/10.1504/IJPOM.2008.022193>.

Corcoran, Simmi Singh. 2019. „Are You Still Using Spreadsheets for Project Management?“ *VIEWPOINT CONSTRUCTION BLOG* (blog). 14. Mai 2019. <https://blog.viewpoint.com/viewpoint-team-project-management-software/>.

Corsten, Hans, Hilde Corsten, und Ralf Gössinger. 2008. *Projektmanagement: Einführung ; [mit Aufgaben und Lösungen]*. 2., vollst. überarb. und wesentlich erw. Aufl. Lehr- und Handbücher der Betriebswirtschaftslehre. München: Oldenbourg.

Daeumlich, Benjamin. 2006. „Einführung Petrinetze“. 2006. https://www2.informatik.hu-berlin.de/top/lehre/WS06-07/se_se/fohlen/petrinetze.pdf.

David, René, und Hassane Alla. 2010. *Discrete, Continuous, and Hybrid Petri Nets*. Berlin, Heidelberg: Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-10669-9>.

Davidrajuh, Reggie. 2012. „Modeling Resource Management Problems with Activity-Oriented Petri Nets“. In *2012 Sixth UKSim/AMSS European Symposium on Computer Modeling and Simulation*, 179–84. Malta, Malta: IEEE. <https://doi.org/10.1109/EMS.2012.62>.

Deng, Wei, Almuth Wendt, und Asbjørn Rolstadås. 1998. „A Set of Modeling Methods for Process-Oriented Project Modeling“. *Project Management* 4, 1998.

Diaz, Michel. 2009. „Application of Petri Nets to Communication Protocols“. In *Petri Nets*, herausgegeben von Michel Diaz, 27–39. London, UK: ISTE. <https://doi.org/10.1002/9780470611647.ch2>.

Dimitrov, Vladimir, und Victor Korotkich. 2010. *Fuzzy Logic: A Framework for the New Millennium*. 1., st Edition. Softcover version of original hardcover edition 2002. Studies in Fuzziness and Soft Computing 81. Heidelberg, Neckar: Physica-Verlag.

Fengler, Wolfgang. 1990. *Handbook for Petri Nets Development System PENECA*. 1.0. TU Ilmenau.

Fengler, Wolfgang, und Ilka Philippow. 1991. *Entwurf industrieller Microcomputersysteme*. München: C. Hanser.

Fleischmann, Albert, Stefan Oppl, Werner Schmidt, und Christian Sary. 2018. „Modelle“. In *Ganzheitliche Digitalisierung von Prozessen*, von Albert Fleischmann, Stefan Oppl, Werner Schmidt, und Christian Sary, 19–69. Wiesbaden: Springer Fachmedien. https://doi.org/10.1007/978-3-658-22648-0_2.

Francke, Matthias. 2017. „Vergleich von Projektmanagement- und PN-Werkzeugen“. Whitepaper, Ilmenau: TU Ilmenau, Fakultät für Informatik und Automatisierung, Betreuerin: Maxi Weichenhain.

———. 2018. „Evaluierung von Methoden zur Extraktion von Steuerstrukturen als Petri-Netze aus einer tabellarischen Notation“. Hauptseminar, Ilmenau: TU Ilmenau, Fakultät für Informatik und Automatisierung, Betreuerin: Maxi Weichenhain.

Gadatsch, Andreas. 2013. *Grundkurs Geschäftsprozess-Management*. Wiesbaden: Vieweg+Teubner Verlag. <https://doi.org/10.1007/978-3-8348-2428-8>.

Genrich, Hartmann J., und Kurt Lautenbach. 1979. „The Analysis of Distributed Systems by Means of Predicate/Transition-Nets“. In *Semantics of Concurrent Computation*, herausgegeben von Gilles Kahn, 70:123–46. Berlin/Heidelberg: Springer-Verlag. <https://doi.org/10.1007/BFb0022467>.

Giua, Alessandro, und Carla Seatzu. 2014. „Petrietze und die Steuerung Ereignisdiskreter Systeme“. *Informatik-Spektrum* 37 (3): 199–210. <https://doi.org/10.1007/s00287-014-0766-8>.

Gobert, Anika. 2008. „Anforderungen und Auswahlkriterien für Projektmanagement-Software am Beispiel von Chipkartenprojekten“. gehalten auf der Giesecke & Devrient, Projektmanagement Zahlungsverkehr.

Gold, Benjamin. 2017. „Untersuchung von Methoden zur Extraktion von Steuerstrukturen als Petri-Netze aus einer tabellarischen Notation“. Hauptseminar, Ilmenau: TU Ilmenau, Fakultät für Informatik und Automatisierung, Betreuerin: Maxi Weichenhain.

Grolleau, Emmanuel, und Annie Choquet-Geniet. 2000. „SCHEDULING REAL-TIME SYSTEMS BY MEANS OF PETRI NETS“. In *FAC Real-Time Programming*,. Palma, Spain. https://ac.els-cdn.com/S147466701739938X/1-s2.0-S147466701739938X-main.pdf?_tid=a7de50d5-6c3f-492b-965c-580199a479be&acdnat=1551477034_6089b2cd0609d187498ec2700ea93547.

Groß, Hendrik. 2014. „Und immer wieder hilft Excel aus“. In *www.mittelstandswiki.de*. Market Paper Projektmanagement im Mittelstand 2014, Teil 1. https://www.mittelstandswiki.de/wissen/Studien:Market_Paper_Projektmanagement_im_Mittelstand_2014,_Teil_1.

Gudd, Dominik. 2017. „Automatisierung von Prozessmanagementfunktionen“. Hauptseminar, Ilmenau: TU Ilmenau, Fakultät für Informatik und Automatisierung, Betreuerin: Maxi Weichenhain.

Haji, Maryam, und Houshang Darabi. 2007. „Petri Net based Supervisory Control Reconfiguration of Project Management Systems“. In *2007 IEEE International Conference on Automation Science and Engineering*, 460–65. Scottsdale, AZ, USA: IEEE. <https://doi.org/10.1109/COASE.2007.4341743>.

Hansmann, Holger, Michael Laske, und Redmer Luxem. 2005. „Einführung der Prozesse — Prozess-Roll-out“. In *Prozessmanagement*, herausgegeben von Jörg Becker, Martin Kugeler, und Michael Rosemann, 269–98. Berlin/Heidelberg: Springer-Verlag. https://doi.org/10.1007/3-540-27153-8_8.

Hastie, Shane, und Stéphane Wojewoda. 2015. „Standish Group 2015 Chaos Report - Q&A with Jennifer Lynch“. <https://www.infoq.com/articles/standish-chaos-2015>.

Heilmann, Heidi, Hrsg. 2003. *IT-Projektmanagement - Fallstricke und Erfolgsfaktoren: Erfahrungsberichte aus der Praxis*. 2., überarb. und erw. Aufl. Heidelberg: dpunkt-Verl.

Hertlein, Manuel. o. J. „Petri Netz / Eine Einführung (Teil 2)“. gehalten auf der Seminar Systementwurf, Lehrstuhl Theorie der Programmierung.

Hoppe, Benjamin. 2014. „Entwicklung einer Modellierungsmethode zur Abbildung von Produktionsprozessen mithilfe objektorientierter Petrinetze“. Bachelorarbeit, Fakultät Maschinenbau: TU Dortmund. http://www.itpl.mb.tu-dortmund.de/publikationen/files/BA_2014_Hoppe.pdf.

Hrúz, Branislav, und MengChu Zhou. 2007. *Modeling and Control of Discrete-Event Dynamic Systems*. Advanced Textbooks in Control and Signal Processing. London: Springer London. <https://doi.org/10.1007/978-1-84628-877-7>.

Hu, W.-B, Y Yue, L Xu, K.-K Wang, C Xia, B Du, und W Song. 2012. „Resource constrained multi-project scheduling modeling under certainty and uncertainty duration“. In *Jisuanji Jicheng Zhizao Xitong/Computer Integrated Manufacturing Systems, CIMS*. 18. 2398-2408.

Hu, Wenbin, und Huan Wang. 2014. „A Novel Petri-Net Based Resource Constrained Multi-project Scheduling Method“. In *Algorithms and Architectures for Parallel Processing*, herausgegeben von Xian-he Sun, Wenyu Qu, Ivan Stojmenovic, Wanlei Zhou, Zhiyang Li, Hua Guo, Geyong Min, Tingting Yang, Yulei Wu, und Lei Liu, 8630:270–84. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-11197-1_21.

Jakoby, Walter. 2015. *Intensivtraining Projektmanagement*. Wiesbaden: Springer Fachmedien Wiesbaden. <https://doi.org/10.1007/978-3-658-08284-0>.

Jeetendra, V. A., O. V. Krishnaiah Chetty, und J. Prashanth Reddy. 2000. „Petri Nets for Project Management and Resource Levelling“. *The International Journal of Advanced Manufacturing Technology* 16 (7): 516–20. <https://doi.org/10.1007/s001700070059>.

Jensen, Kurt. 1992. *Coloured Petri Nets*. Berlin, Heidelberg: Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-662-06289-0>.

Jensen, Kurt, und Lars M. Kristensen. 2009. *Coloured Petri Nets*. Berlin, Heidelberg: Springer Berlin Heidelberg. <https://doi.org/10.1007/b95112>.

Jensen, Kurt, Lars Michael Kristensen, und Lisa Wells. 2007. „Coloured Petri Nets and CPN Tools for Modelling and Validation of Concurrent Systems“. *International Journal on Software Tools for Technology Transfer* 9 (3–4): 213–54. <https://doi.org/10.1007/s10009-007-0038-x>.

Jongwook Kim, A.A. Desrochers, und A.C. Sanderson. 1995. „Task planning and project management using Petri nets“. In *Proceedings. IEEE International Symposium on Assembly and Task Planning*, 265–71. Pittsburgh, PA, USA: IEEE Comput. Soc. Press. <https://doi.org/10.1109/ISATP.1995.518781>.

Kaack, Alexander. 2017. „Eine empirische Studie zum Stand von agiler Software-Entwicklung in der Praxis“. Masterarbeit, Department Informatik der Fakultät Technik und Informatik: Hochschule für Angewandte Wissenschaften Hamburg. http://edoc.sub.uni-hamburg.de/haw/volltexte/2017/4092/pdf/MINF_MA_KAACK_ALEXANDER.pdf.

Kaltwasser, Silvia. 2017. „Entwicklung und Evaluierung eines Kriterienkataloges zur Auswahl von Projektmanagement-(PM-)Werkzeugen“. Bachelorarbeit, Ilmenau: TU Ilmenau, Fakultät für Informatik und Automatisierung, Betreuerin: Maxi Weichenhain.

Kao, Hsing-Pei, Ben Hsieh, und Yingchieh Yeh. 2006. „A PETRI-NET BASED APPROACH FOR SCHEDULING AND RESCHEDULING RESOURCE-CONSTRAINED MULTIPLE PROJECTS“. *Journal of the Chinese Institute of Industrial Engineers* 23 (6): 468–77. <https://doi.org/10.1080/10170660609509343>.

Kao, Hsing-Pei, Brian Wang, James Dong, und Kuo-Cheng Ku. 2006. „An Event-Driven Approach with Makespan/Cost Tradeoff Analysis for Project Portfolio Scheduling“. *Computers in Industry* 57 (5): 379–97. <https://doi.org/10.1016/j.compind.2005.11.004>.

Kaya, Ömer. 2016. „Anforderungen an Prozessmanagementmethoden und Tools“. Hauptseminar, Ilmenau: TU Ilmenau, Fakultät für Informatik und Automatisierung, Betreuerin: Maxi Weichenhain.

———. 2017a. „Werkzeuge zur Unterstützung des Projektmanagements“. Whitepaper, Ilmenau: TU Ilmenau, Fakultät für Informatik und Automatisierung, Betreuerin: Maxi Weichenhain.

———. 2017b. „Methoden, Möglichkeiten und Grenzen von Petri-Netzen zur Automatisierung von Workflows im Rahmen der Internet der Dinge“. Masterarbeit, Ilmenau: TU Ilmenau, Fakultät für Informatik und Automatisierung, Betreuerin: Maxi Weichenhain.

Kiencke, Uwe. 2009. *Ereignisdiskrete Systeme Modellierung und Steuerung verteilter Systeme*. München; Wien: Oldenbourg. <http://dx.doi.org/10.1524/9783486593426>.

Kleine-Döpke, Timo. 2013. „Konfliktlösungen für den nebenläufigen Feuerprozess bei gefärbten Petri-Netzen und ihre Anwendungen“. Bachelorarbeit, Studiengang Angewandte Mathematik: Fachhochschule Bielefeld.

Komus, Ayelt. 2015. „Agiles Prozessmanagement - Wie Scrum nachhaltiges und erfolgreiches Prozessmanagement ermöglicht - 3 Dinge ...“.

Komus, Ayelt, und Moritz Kuberg. 2015. „Status Quo Agile“. http://www.gpm-ipma.de/fileadmin/user_upload/Know-How/studien/Studie_Agiles-PM_web.pdf.

Königsberger, Thomas, und Bernhard Schloß. 2014. „<https://www.openpm.info/display/openPM/Projektplanung+-+Ablauf+und+Termine>“. In *openpm*. Projektplanung. <https://www.openpm.info/display/openPM/Projektplanung+-+Ablauf+und+Termine>.

Koschmider, Agnes, und Daniel Ried. 2005. „Semantische Annotation von Petri-Netzen“. *Proceedings des 12. Workshops Algorithmen und Werkzeuge für Petrinetze (AWPN'05)* (Humboldt-Universität zu Berlin): 66–71.

Krähmer, Johannes. 2016. „Adaptivity of Petrinets – Geschäftsprozesse und Projektmanagement“. Hauptseminar, Ilmenau: TU Ilmenau, Fakultät für Informatik und Automatisierung, Betreuerin: Maxi Weichenhain.

Kruse, Hermann-Josef, Bernhard Bachmann, Timo Lask, Lennart Ochel, und Sabrina Pross. 2014. *Petri-Netz-Formalismen und Lösungsansätze für allgemeine Konfliktsituationen bei Feuerprozessen in Petri-Netz-Modellen*. Bd. 2. FH Bielefeld University of Applied Sciences Bde. Forschungsreihe des Fachbereichs Ingenieurwissenschaften und Mathematik.

Kuhlmann, Karl Kristian. 2013. *Methodik zur Gestaltung von Geschäftsprozessen im industriellen Werkzeugbau*. Aachen: Apprimus Wissenschaftsverlag. <http://nbn-resolving.de/urn:nbn:de:101:1-2013062516113>.

Kumanan, S., und K. Raja. 2008. „Modeling and Simulation of Projects with Petri Nets“. *American Journal of Applied Sciences* 5 (12): 1742–49. <https://doi.org/10.3844/ajassp.2008.1742.1749>.

Kumar, A., und L.S. Ganesh. 1998. „Use of Petri nets for resource allocation in projects“. *IEEE Transactions on Engineering Management* 45 (1): 49–56. <https://doi.org/10.1109/17.658660>.

Kuster, Jürg, Eugen Huber, Robert Lippmann, Alphons Schmid, Emil Schneider, Urs Witschi, und Roger Wüst. 2011. *Handbuch Projektmanagement*. Berlin, Heidelberg: Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-21243-7>.

Kuzenkov, Vadim, Alexey Zebzeev, und Evgeniy Gromakov. 2014. „Resource Leveling in the Project Design Process by Petri Net Using“. *Advanced Materials Research* 905 (April): 752–56. <https://doi.org/10.4028/www.scientific.net/AMR.905.752>.

Lee, Kuen-Ching, Iuan-Yuan Lu, und Hsin-Hui Lin. 1994. „PM-Net: A Software Project Management Representation Model“. *Information and Software Technology* 36 (5): 295–308. [https://doi.org/10.1016/0950-5849\(94\)90085-X](https://doi.org/10.1016/0950-5849(94)90085-X).

Lin, Chang-Pin, und Hung-Lin Dai. 2014. „Applying Petri Nets on Project Management“. *Universal Journal of Mechanical Engineering* 2 (8): 249–55. <https://doi.org/10.13189/ujme.2014.020801>.

Lin, Chang-Pin, Hung-Lin Tai, und Shih-yan Hu. 2013. „An Enhanced Critical Path Method for Multiple Resource Constraints“. In *unknown*. [http://ir.lib.ksu.edu.tw/bitstream/987654321/20263/2/101-2221-E-19-15\(Poster\).pdf](http://ir.lib.ksu.edu.tw/bitstream/987654321/20263/2/101-2221-E-19-15(Poster).pdf).

Litke, Hans-Dieter. 2005a. *Projektmanagement - Handbuch für die Praxis: Konzepte - Instrumente - Umsetzung*. München: Carl Hanser Verlag GmbH & Co. KG.

———, Hrsg. 2005b. *Projektmanagement: Handbuch für die Praxis; Konzepte - Instrumente - Umsetzung*. München: Hanser.

Litke, Hans-Dieter, und Ilonka Kunow. 2006. *Projektmanagement*. 5., überarb. Aufl. TaschenGuide 14. Planegg/München: Haufe.

Liu, Cong, Jiujun Cheng, Yirui Wang, und Shangce Gao. 2016. „Time Performance Optimization and Resource Conflicts Resolution for Multiple Project Management“. *IEICE Transactions on Information and Systems* E99.D (3): 650–60. <https://doi.org/10.1587/transinf.2015EDP7397>.

Lung-chun Liu, und E. Horowitz. 1989. „A Formal Model for Software Project Management“. *IEEE Transactions on Software Engineering* 15 (10): 1280–93. <https://doi.org/10.1109/TSE.1989.559781>.

Lunze, Jan. 2006. *Ereignisdiskrete Systeme: Modellierung und Analyse dynamischer Systeme mit Automaten, Markovketten und Petrinetzen*. München: Oldenbourg Wissenschaftsverlag GmbH. <https://doi.org/10.1524/9783486593440>.

Maktabi, M. Ali. 2019. „Analyse zu wesentlichen PM-Erweiterungen von Petri-Netzen“. Hauptseminar, TU Ilmenau, Fakultät für Informatik und Automatisierung, Betreuerin: Maxi Weichenhain.

Mangold, Pascal. 2011. *IT-Projektmanagement kompakt*. 3., erw. Aufl., Nachdr. kompakt-Reihe. Heidelberg: Spektrum Akad. Verl.

Margea, Romeo, Camelia Margea, Hiroshi Yamamoto, Eiki Kikuchi, und Akihiko Tanaka. 2011. „Open Source Approach to Project Management Tools“. *Informatica Economica* 15 (1): 196–206.

Margherita, Alessandro. 2014. „Business Process Management System and Activities: Two Integrative Definitions to Build an Operational Body of Knowledge“. *Business Process Management Journal* 20 (5): 642–62. <https://doi.org/10.1108/BPMJ-04-2013-0050>.

Mejía, Gonzalo, und Carlos Montoya. 2010. „Applications of Resource Assignment and Scheduling with Petri Nets and Heuristic Search“. *Annals of Operations Research* 181 (1): 795–812. <https://doi.org/10.1007/s10479-010-0686-1>.

Mejía, Gonzalo, Karen Niño, Carlos Montoya, María Angélica Sánchez, Jorge Palacios, und Lionel Amodeo. 2016. „A Petri Net-Based Framework for Realistic Project Management and Scheduling: An Application in Animation and Videogames“. *Computers & Operations Research* 66 (Februar): 190–98. <https://doi.org/10.1016/j.cor.2015.08.011>.

Moldt, Daniel, Thomas Wagner, und Michael Haustermann. o. J. „Petri Nets Tool Database“. Uni-Hamburg. Informatik Uni-Hamburg. Zugegriffen 26. Februar 2019. <http://www.informatik.uni-hamburg.de/TGI/PetriNets/tools/db.html>.

Niño Mora, Karen Yineth, Gonzalo Mejia Delgadillo, Maria Angelica Sanchez Olaya, Jorge Palacios Bonilla, und Hassub Vargas Flores. 2015. „UNA PLATAFORMA PARA LA GESTIÓN Y PROGRAMACIÓN DE PROYECTOS UTILIZANDO REDES DE PETRI“. *DYNA MANAGEMENT* 3 (3): [12 p.]-[12 p.]. <https://doi.org/10.6036/MN7588>.

Parida, Sasmita, Suwendu Chandan Nayak, Piyush Priyadarshi, Prasant Kumar Pattnaik, und Gaurav Ray. 2018. „Petri Net: Design and Analysis of Parallel Task Scheduling Algorithm“. In *Advances in Electronics, Communication and Computing*, herausgegeben von Akhtar Kalam, Swagatam Das, und Kalpana Sharma, 443:765–76. Singapore: Springer Singapore. https://doi.org/10.1007/978-981-10-4765-7_79.

Perković, T., D. Šturlić, und M. Meden. 2011. „Efficient project management by quality improvement of project processes“. *IEEE*, 2011.

Perwiss. 2018. „Wo liegt der Unterschied zwischen BPM und Workflow Management?“
<https://www.perwiss.de/business-process-management.html>.

Prashant Reddy, J., S. Kumanan, und O.V. Krishnaiah Chetty. 2001. „Application of Petri Nets and a Genetic Algorithm to Multi-Mode Multi-Resource Constrained Project Scheduling“. *The International Journal of Advanced Manufacturing Technology* 17 (4): 305–14. <https://doi.org/10.1007/s001700170184>.

Qu, Ying, und Min Wang. 2015. „A risk emergency management model for software project based on stochastic Petri Nets“. In *2015 7th International Conference on Modelling, Identification and Control (ICMIC)*, 1–6. Sousse, Tunisia: IEEE. <https://doi.org/10.1109/ICMIC.2015.7409457>.

Raabe, Oliver, Richard Wacker, Daniel Oberle, Christian Baumann, und Christian Funk, Hrsg. 2012. *Recht ex machina: Formalisierung des Rechts im Internet der Dienste*. Berlin Heidelberg: Springer Vieweg.

Raja, K., und S. Kumanan. 2007. „Resource Leveling Using Petrinet and Memetic Approach“. *American Journal of Applied Sciences* 4 (5): 317–22. <https://doi.org/10.3844/ajassp.2007.317.322>.

Rau, Rebecca. 2016. „Gegenüberstellung und Abgrenzung von Methoden und Beschreibungsmitteln im Umfeld des Projektmanagements bei System- und Softwareentwicklung“. Hauptseminar, Ilmenau: TU Ilmenau, Fakultät für Informatik und Automatisierung, Betreuerin: Maxi Weichenhain.

Reinheimer, Stefan. 2014. „Prozessmanagement für Experten – Impulse für aktuelle und wiederkehrende Themen“. *HMD Praxis der Wirtschaftsinformatik* 51 (5): 721–23. <https://doi.org/10.1365/s40702-014-0068-2>.

Reisig, Wolfgang. 2010. *Petrinetze*. Wiesbaden: Vieweg+Teubner Verlag. <https://doi.org/10.1007/978-3-8348-9708-4>.

Reisig, Wolfgang, und Jörg Desel. 2014. „Konzepte der Petrinetze“. *Informatik-Spektrum* 37 (3): 172–90. <https://doi.org/10.1007/s00287-013-0758-0>.

- Rokyta, PAtrik. 1992. „Theoretische Grundlagen für Tools für gefärbte Petri-Netze“. Diplomarbeit, Ilmenau: TU Ilmenau, Fakultät für Informatik und Automatisierung.
- Sapper, Ralph. 2007. *Kriterien und Elemente zum spezifischen Projektmanagement von Investitionsprojekten im chemischen und pharmazeutischen Anlagenbau*. Schriftenreihe Projektmanagement 3. Kassel: Kassel Univ. Press.
- Sawhney, Anil, André Mund, und Tavatchai Chaitavatputtiporn. 2003. „Petri Net-Based Scheduling of Construction Projects“. *Civil Engineering and Environmental Systems* 20 (4): 255–71. <https://doi.org/10.1080/10286600310001633848>.
- Scharfe, Christian. 2013. „Modellierung von adaptiven Geschäftsprozessen“. TU Ilmenau, Technische Informatik und Ingenieurinformatik.
- Schenk, Sebastian. 2017. „Anforderungen an Projektmanagement- (PM-) Zertifikate und Tools“. Hauptseminar, Ilmenau: TU Ilmenau, Fakultät für Informatik und Automatisierung, Betreuerin: Maxi Weichenhain.
- Schmelzer, Hermann J., und Wolfgang Sesselmann. 2013. *Geschäftsprozessmanagement in der Praxis: Kunden zufriedenstellen, Produktivität steigern, Wert erhöhen: [das Standardwerk]*. 8., überarbeitete und erweiterte Auflage. München: Hanser.
- Schott, Hanna. 2019. „Projektressourcenmanagement mit Petri-Netzen“. Masterarbeit, Ilmenau: TU Ilmenau, Fakultät für Informatik und Automatisierung, Betreuerin: Maxi Weichenhain.
- Schröder, Joachim. 2009. *Adaptive Verhaltensentscheidung und Bahnplanung für kognitive Automobile*. Karlsruhe: Univ.-Verl.
- Seidlmeier, Heinrich. 2015. *Prozessmodellierung mit ARIS®*. Wiesbaden: Springer Fachmedien Wiesbaden. <https://doi.org/10.1007/978-3-658-03905-9>.
- Shi, G.-Q, Bo Li, und X.-D. Chai. 2007. „Modeling of multi-project scheduling for complex product development based on CPN“, September 2007.

Silberberg, Julian, Timo Lask, und Bernhard Bachmann. 2016. „Formalismen für gefärbte Petri-Netze und Verfahren zur effizienten Bestimmung von aktiven Modus-Mengen“. Unpublished. <https://doi.org/10.13140/RG.2.1.3432.8566>.

Silva, Manuel. 2013. „Half a Century after Carl Adam Petri’s Ph.D. Thesis: A Perspective on the Field“. *Annual Reviews in Control* 37 (2): 191–219. <https://doi.org/10.1016/j.arcontrol.2013.09.001>.

Starke, Peter H. 1990a. *Analyse von Petri-Netz-Modellen*. Wiesbaden: Vieweg+Teubner Verlag. <https://doi.org/10.1007/978-3-663-09262-9>.

———. 1990b. „Prädikat/Transitions-Netze“. In *Analyse von Petri-Netz-Modellen*, von Peter H. Starke, 236–45. Wiesbaden: Vieweg+Teubner Verlag. https://doi.org/10.1007/978-3-663-09262-9_21.

———. 1995. *A Memo on Time Constraints in Petri Nets*. Informatik-Berichte, No. 46. Humboldt-Univ. zu Berlin.

Stoimenova, Borislava. 2008. „Evaluation von Workflow-Sprachen zur Prozessmodellierung in Multi-Domain Umgebungen am Beispiel von Géant2 E2E Links“. Diplomarbeit, Fakultät für Informatik: Technischen Universität München.

Strasser. o. J. „Hybrides Projektmanagement – Wie Sie agile und klassische Methoden verbinden“. Zugegriffen 20. Juli 2018. <http://www.theprojectgroup.com/blog/hybrides-projektmanagement-so-gehts/>.

Sun, Xing-cun, Hao Wu, Qiang Wang, und Su-ling Jia. 2013. „A New Framework on Monitoring Equipment Technology Reformation Project Management Based on Petri Net“. In *Proceedings of 20th International Conference on Industrial Engineering and Engineering Management*, herausgegeben von Ershi Qi, Jiang Shen, und Runliang Dou, 511–19. Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-40072-8_50.

Thein, Christian. 2016. „Anforderungen an operative Prozessmanagement-Methoden und -Tools“. Hauptseminar, Ilmenau: TU Ilmenau, Fakultät für Informatik und Automatisierung, Betreuerin: Maxi Weichenhain.

———. 2017. „Untersuchung der Möglichkeiten der Projektmodellierung auf Basis eines theoretischen Petri-Netz-Konzeptes“. Masterarbeit, Ilmenau: TU Ilmenau, Fakultät für Informatik und Automatisierung, Betreuerin: Maxi Weichenhain.

Topic, Gordan, und Dragan Jevtic. 2019. „Modelling, Simulation and Resource Optimisation of Complex Development Project by Fusion of Multiple-Domain Matrix and Coloured Petri Nets Methods“. *International Journal of Simulation and Process Modelling* 14 (1): 51. <https://doi.org/10.1504/IJSPM.2019.097713>.

Traeger, Dirk H. 1994. *Einführung in die Fuzzy-Logik*. Wiesbaden: Vieweg+Teubner Verlag. <https://doi.org/10.1007/978-3-322-99590-2>.

V. A. Jeetendra, Kumar, S. Kumanan, und Chetty O.V. Krishnaiah. 1998. „Application of Petrinets to Project Management“. *Manufacturing Engineering Section, Indian Institute of Technology Madras*, Nr. Chennai-600 036.

Varga, A. 2001. „Discrete Event Simulation System“. In .

Warmuth, Daniel. 2009. „Zeit in Petrinetzen (korrigierte Fassung)“. Blockseminar, HU Berlin, Januar 19.

Weichenhain, Maxi, und Svatopluk Alexander. 2018. „The new way of project management – Agility on the rise“. <https://www.linkedin.com/pulse/new-way-project-management-agility-rise-svatopluk-alexander/>.

Weichenhain, Maxi, und Wolfgang Fengler. 2019a. „A Petri Net Table Model Applied to Classic and Agile Project Management“. In *Proceedings of the International Workshop on Petri Nets and Software Engineering »PNSE'19«*, 125–40. Hamburg.

———. 2019b. „Zusätzliche Definitionen zur Dokumentation des PN-Tools Peneca Chromos“. TU Ilmenau, Fachgebiet Rechnerarchitektur und Eingebettete Systeme. <https://www.tu-ilmenau.de/ra/r-stu/r-lv/r-tapn/>.

Weichenhain, Maxi, Wolfgang Fengler, und Detlef Streitferdt. 2019. „How to Bring Project Management of Embedded Systems to an Appropriate Level of Abstraction for a Discrete Event Model“. In *2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)*.

Weske, Mathias. 2012. *Business Process Management*. Berlin, Heidelberg: Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-28616-2>.

Windhager, Lukas. 2013. „Modeling of dynamic systems with Petri nets and fuzzy logic.“ Dissertation, LMU München. <https://edoc.ub.uni-muenchen.de/15655/>.

Wirtschaftslexikon Gabler. 2017. „Geschäftsprozessmanagement“. In *Wirtschaftslexikon Gabler*.

<http://wirtschaftslexikon.gabler.de/Definition/geschaeftsprozessmanagement.html>.

Wolf-Ulrich, Raffel. 2005. „Agentenbasierte Simulation als Verfeinerung der Diskreten-Ereignis-Simulation unter besonderer Berücksichtigung des Beispiels Fahrerloser Transportsysteme“. Dissertation, Mathematik und Informatik: FU Berlin. <https://refubium.fu-berlin.de/handle/fub188/10677>.

Won, H.C. 1987. „State Classification Method under Maximum Firing Rule in Petri Net Models.“ *Journal Korea Inf. Sci. Soc. (South Korea)* 14 (No. 2): 131–39.

Wu, Yu, Xin-cun Zhuang, Guo-hui Song, Xiao-dong Xu, und Cong-xin Li. 2009. „Solving Resource-Constrained Multiple Project Scheduling Problem Using Timed Colored Petri Nets“. *Journal of Shanghai Jiaotong University (Science)* 14 (6): 713–19. <https://doi.org/10.1007/s12204-009-0713-z>.

Yen-Liang Chen, Ping-Yu Hsu, und Yuan-Bin Chang. 2008. „A Petri Net Approach to Support Resource Assignment in Project Management“. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 38 (3): 564–74. <https://doi.org/10.1109/TSMCA.2008.918632>.

Yu, Ruiqiang, Zhiqiu Huang, und Lin Wang. 2009. „Modeling and Analyzing Project Performance with Timing Constraint Petri Net“. In *2009 International Conference on Computer Engineering and Technology*, 243–46. Singapore, Singapore: IEEE. <https://doi.org/10.1109/ICCET.2009.32>.

Zapf, Michael, und Armin Heinzl. 2000. „Ansätze zur Integration von Petri-Netzen und objektorientierten Konzepten“. *Wirtschaftsinformatik* 42 (1): 36–46. <https://doi.org/10.1007/BF03250715>.

Zhou, Kai-Qing, und Azlan Mohd Zain. 2016. „Fuzzy Petri Nets and Industrial Applications: A Review“. *Artificial Intelligence Review* 45 (4): 405–46. <https://doi.org/10.1007/s10462-015-9451-9>.

Zimmermann, Armin. 1997. „Modellierung und Bewertung von Fertigungssystemen mit Petri-Netzen“. Dissertation, TU Berlin.

Zollondz, Hans-Dieter. 2012. *Grundlagen Qualitätsmanagement: Einführung in Geschichte, Begriffe, Systeme und Konzepte*. <https://doi.org/10.1524/9783486712025>.

Übersicht der einzelnen Anhänge

Anhangsübersicht

- Anhang A** Tooleinsatz im Projektmanagement
- Anhang B** Weitere Phasen des Projektmanagements
- Anhang C** Tabellengrundlage des Beispielprojektes
- Anhang D** Vergleich von Geschäftsprozess-, Workflowmanagement und diskrete Modelle mit dem Projektmanagement
- Anhang E** Petri-Netze und das Projektmanagement
- Anhang F** Detailliertere Information zum anonymisierten Beispiel
- Anhang G** Vergleich der Publikationen zu Petri-Netz-Ansätzen im Projektmanagement
- Anhang H** Legende für PPNM-Abbildungen

A Anhang A **Tooleinsatz im Projektmanagement**






In Bezug auf die Toolauswahl stellt sich die Frage, wann Projektmanagementmethoden und die Unterstützung durch Software überhaupt einsetzbar sind. Die Anwendung muss dabei projektbezogen abgestimmt und für jedes Projekt erneut angepasst werden. Damit ist die pauschale Verallgemeinerung für eine Vielzahl von Projekten nicht möglich. Dies lässt sich mit der erläuterten Begriffsdefinition des Projektes unterstreichen. Ein Vorhaben wird in den Mittelpunkt gestellt, welches sich durch seine Einzigartigkeit und Einmaligkeit der Bedingungen auszeichnet. Schlussfolgernd müssen somit die Projektmanagementmethoden in Kombination einzigartig sein. [vgl. Kaltwasser 2017, 9 f]

Als nur ein Referenzprogramm von mehreren in Bezug auf Projektmanagement, ist MS Project® zu nennen. Es beinhaltet im Kern lediglich die Ressourcenliste und die Vorgangsliste. Diese beiden Tabellen ermöglichen alle jeweils relevanten Daten, wie Aufwand, Dauer, Termine, Namen, Arbeitszeiten und Stundensatz zu erfassen, auszuwerten und darzustellen. [vgl. Jakoby 2015, 171 ff; vgl. Rau 2016, 7] In der Realität wird der Einsatz von Microsoft Excel oder anderen Applikationen oft hauptsächlich in allen kleinen und mittleren Projekten vorgezogen. Gerade Excel ist ein Tool mit einem enormen Umfang und ist in der Lage, viele Sachverhalte abzubilden. Aber es ist mit Aufwand und Umwegen verbunden, welche nicht immer zum Vorteil führen.

Sapper [vgl. Sapper 2007, 100] unterscheidet fünf Unterteilungen die zeigen, wie viele und welche Methoden zum Projektmanagement eingesetzt werden können. Die Unterteilung beruht auf der Einteilung Projektgröße und Projekttyp. Dies ist in folgender Tabelle dargestellt:

Tabelle 10: Eignung PM-Methoden mit Bezug zur Projektgröße [Sapper 2007, 100]

Projekttyp / Projektgröße	sehr klein (<3 MA)	klein (3-10 MA)	mittel (10-50 MA)	groß (50-150 MA)	sehr groß (>150 MA)
Grundlagenentwicklung					
Entwicklung von Mustern					
Entwicklung von Produkten					
Entwicklung von Systemen					
Betreuung von Verfahren					
Modifikationsentwicklung					

-  Projektmanagement-Methoden kaum anwendbar
-  elementare Projektmanagement-Methoden anwendbar
-  gängige Projektmanagement-Methoden anwendbar
-  fast vollständige Anwendung von Projektmanagement-Methoden
-  vollständige Anwendung von Projektmanagement-Methoden

Es ist gut erkennbar, wie sich in Relation zur Projektgröße auch die Vollständigkeit der anwendbaren Methoden zum Projektmanagement verändert. Dies liegt darin begründet, dass mit wachsenden Projekten ebenfalls die Planungs-, Verwaltungs- und Kontrollaufgaben ansteigen. [vgl. Kaltwasser 2017, 9 ff]

Das Managen von mittleren und großen Projekten, bei denen eine Vielzahl von Ressourcen involviert sind, stellt eine Herausforderung dar. Eine Möglichkeit, um diese Herausforderung bewältigen zu können, ist zum einen die Verwendung von Handlungsempfehlungen und Richtlinien der gültigen PM-Standards (PMI, Prince2, SCRUM, usw.) und zum anderen die Nutzung von Software. Software kann dahingehend unterstützen, da viele der Aufgaben und Prozesse innerhalb von Projekten standardisiert sind und sich zumeist in den einzelnen Projektphasen wiederholen. [vgl. Schenk 2017, 19 f]

Mit dem Einsatz von Programmen wird versucht, einen wesentlichen Beitrag zur Unterstützung von Führungsaufgaben und -techniken sowie für die Initiierung, Definition, Planung, Steuerung und den Abschluss von Projekten beizutragen. PM-Software ermöglicht es, die kritischen und komplexen Schritte im Rahmen eines effizienten Managements zu gestalten. [vgl. Margea u. a. 2011, 197 f]

Auf dem Markt gibt es eine Vielzahl an Tools für das Projektmanagement. Jedes bietet einen anderen Funktionalitätsumfang sowie einen anderen Fokus auf die zu erledigende Aufgabenstellung und können in unterschiedlichste Kategorien eingeteilt werden. Eine Unterteilung kann nach [vgl. Ahlemann 2002] in die fünf Systemkategorien Einzelprojektmanagement Systeme, Multiprojektmanagement Systeme, Enterprise Projektmanagement Systeme, Dienstleistungsorientierte Multiprojektmanagement Systeme und Projekt Collaboration-Plattform vorgenommen werden. Bei dieser Einteilung liegt der Fokus auf der Unterscheidung von Projekten in großen, teils globalen Teams und dessen Verwaltung. Zusätzlich werden Features wie Video-Chats oder die Verwaltung eines Wikis unterstützt. Diese Systeme dienen vorrangig dem Daten- und Informationsaustausch zwischen Teammitgliedern und bedienen somit die zentralen Aspekte wie Teamarbeit und Dokumentenmanagement.

Es existieren zudem weitere Arten der Einteilung. Es kann z. B. eine Einteilung nach dem Nutzungsmodell vorgenommen (On-Premises- und Cloud-Lösungen) oder zwischen betriebsfertig und umfassend konfigurierbar unterschieden werden. Weiterhin kann eine Einteilung nach der verwendeten Technologie (Java, HTML5, .Net) oder Programmart (Widgets, Applets und der klassischen Fenstertechnik) erfolgen. [vgl. Kaltwasser 2017, 12 ff]

Somit lässt sich festhalten, dass unterschiedlichste Einteilungen existieren und es abhängig vom Einsatz, Projektvorhaben, Sichtweise oder anderen Gegebenheiten ist. Zum heutigen Stand geben PM-Standards keine Vorgaben in Bezug auf die Toolwahl. [vgl. Schenk 2017, 19 ff]

B Anhang B **Weitere Phasen des Projektmanagements**

Neben den vier Phasen **Initialisierung**, **Planung**, **Steuerung** und **Abschluss** ist die Durchführung von Risiko-, Qualitätsmanagement und der Steuerung der Informationsflüsse ebenfalls als wichtiger Bestandteil im Projektmanagement integriert.

Risikomanagement agiert im optimalen Fall schon von vorn herein und kann somit mögliche Risiken in der Entstehung verhindern. Hierzu zählen die rechtzeitige Risikoidentifizierung und deren Analyse und Quantifizierung, mit anschließenden Maßnahmen zur Risikobekämpfung, -reduzierung oder im Falle von nicht beeinflussbaren Risiken, die Erarbeitung von Alternativen. [vgl. Kuster u. a. 2011, 167 ff; vgl. Rau 2016, 5]

Die Messung der Anforderungs- und Zielerreichung ist die wesentliche Grundlage für das **Qualitätsmanagement**. Während der Qualitätsplanung werden die zur Messung herangezogenen Anforderungen und Ziele festgelegt und die Maßnahmen zur Zielerreichung bestimmt. So kann die Qualitätserhebung gemessen werden. Die Qualitätslenkung überwacht die zur richtigen Umsetzung geplanten Qualitätsmaßnahmen in Bezug auf den Projektfortschritt aus Sicht der Qualität. [vgl. Kuster u. a. 2011, 167; vgl. Rau 2016, 5; vgl. Jakoby 2015, 131]

Eine nicht zu unterschätzende Tätigkeit ist hierbei die Organisation der **Informationsflüsse**. Informationen haben unterschiedliche Bedeutungen für die verschiedensten Beteiligten in einem Projekt. Der Umgang mit anfallenden und auftretenden Informationen muss festgelegt werden. Hierzu zählen auch der Kommunikationsweg und die Dokumentationsart, einschließlich der im Projekt entstehenden Dokumente. [vgl. Rau 2016, 7; vgl. Jakoby 2015, 51 f]

C Anhang C

Tabellengrundlage des Beispielprojektes

21 Reiter mit unzähligen Spalten, Zeilen und Formeln
 Auszug von dem hier vorgestellten Reiter:
 ✓ 152 Spalten
 ✓ 850 Zeilen
 ✓ Formeln: nicht zählbar
 ✓ Scope: 2 Sprints und 10 Themen

Projektplan

ID	Task	Backlog-Link	Fachkonzept-Link	Start	Ende	Milestone	Fertigstellungsgrad in %	Abweichung?	Fertigstellungsgrad Gruppe	Plan	Zeitliche Darstellung
FK_1	Fachkonzept		Name	SVERWEIS	SVERWEIS	SVERWEIS	Formel				
FK_T1	Thema 1		Kapitel 2.2.3	SVERWEIS	SVERWEIS	SVERWEIS	SVERWEIS				
FK_T2	Thema 2		Kapitel x	SVERWEIS	SVERWEIS	SVERWEIS	SVERWEIS				
...
SI	Sprint 1										
SI_A1	Aktivität 1	Name	Kapitel 2.1	SVERWEIS	SVERWEIS	SVERWEIS	Formel				
SI_A1_D	Design			SVERWEIS	SVERWEIS	SVERWEIS	SVERWEIS				
SI_A1_I	Implementierung			SVERWEIS	SVERWEIS	SVERWEIS	SVERWEIS				
SI_A1_Te	Test			SVERWEIS	SVERWEIS	SVERWEIS	SVERWEIS				
SI_A1_Tr	Training			SVERWEIS	SVERWEIS	SVERWEIS	SVERWEIS				
SI_A1_Dc	Dokumentation			SVERWEIS	SVERWEIS	SVERWEIS	SVERWEIS				
SI_A1_Ms	Milestone			SVERWEIS	SVERWEIS	SVERWEIS	SVERWEIS				
...

Zwischenformeln zur Berechnung vom Fertigstellungsgrad pro Paket (Gruppierungen und Berechnungen über 20 Spalten die ineinandergreifen - über SVERWEIS, MITTELWERTWENN und SUMMEWENN(S))

Hier wird der Trend der Abweichung angezeigt. Dies wird über mehrere WENN-Formeln über 3 Spalten realisiert.

Formeln zur Berechnung des Fertigstellungsgrades - abhängig von den Angaben in der Backlog

Auch hier wird über SVERWEIS und SUMMEWENN(S) der gewünschte Wert berechnet.

Angaben als Grundlage von 0, 25, 50, 75 oder 100

Planungswerte von Backlog und dem Reiter für die übergeordnete Planung

Auf einen Bereich von 73 Spalten wird durch mehrere WENN-Formeln die jeweilige Zeile - als Teil eines Zeitstrahles - farblich hervorgehoben. Dabei gibt es nicht nur Abhängigkeiten zu den in Bezug stehenden Datumsangaben, sondern auch zur Art der Aufgabe. Sowie unterschiedliche farbliche Gestaltungen.

Abbildung 40: Ausschnitt der Komplexität aus dem Projektplan

D Anhang D

Vergleich von Geschäftsprozess-, Workflowmanagement und diskrete Modelle mit dem Projektmanagement

Der Anhang zeigt einen Vergleich der in Kapitel 2 vorgestellten integrativen und flankierenden Bereiche in Bezug auf Informationsmanagement im Projektmanagement.

Darüber hinaus unterscheiden sich Prozesse und Projekte in einigen Punkten, welche in der nachstehenden Tabelle überblickartig dargestellt werden.

Tabelle 11: Unterschiede zwischen Prozess und Projekt [vgl. Becker und Kahn 2012, 6 f; vgl. Becker, Kugeler, und Rosemann 2012, 17 f; vgl. Margherita 2014, 652; vgl. Thein 2016, 8]

Prozess	Projekt
<ul style="list-style-type: none"> • <i>Zeitlich begrenzt, aber sich wiederholend</i> 	<ul style="list-style-type: none"> • Zeitlich begrenzt und einmalig <ul style="list-style-type: none"> ➤ Gilt nur für das gesamte Projekt bezogen. Einzelne Aktivitäten können dem widersprechen.
<ul style="list-style-type: none"> • <i>Start ausgelöst durch bestimmtes Ereignis</i> 	<ul style="list-style-type: none"> • Start im Projektplan festgelegt <ul style="list-style-type: none"> ➤ Gilt nur für das gesamte Projekt. Einzelne Aktivitäten können dem widersprechen.
<ul style="list-style-type: none"> • <i>Kontinuierlicher Ressourcenverbrauch</i> 	<ul style="list-style-type: none"> • Temporärer Ressourcenverbrauch mit festem Budget
<ul style="list-style-type: none"> • <i>Wissensbasis der Teammitglieder homogen (aus dem gleichen Bereich)</i> 	<ul style="list-style-type: none"> • Wissensbasis der Teammitglieder heterogen (interdisziplinär)
<ul style="list-style-type: none"> • <i>Nimmt Routineaufgaben des Tagesgeschäftes wahr</i> <ul style="list-style-type: none"> ➤ <i>Auch hier muss auf unerwartete Störungen reagiert werden</i> 	<ul style="list-style-type: none"> • Nimmt seltene und komplexe Aufgaben wahr

Der Begriff Geschäftsprozessmanagement ist der übergeordnete Begriff für Methoden der strategischen Ausrichtung, Entwicklung, Analyse, Optimierung von Prozessen und deren strategische Einbindung im Ganzen. Darunter fallen die Erhebung, Gestaltung, Dokumentation und Durchführung von Prozessen. Workflowmanagement hingegen legt den Fokus auf die Modellierung, Simulation, Steuerung, Protokollierung und Analyse von Workflows bzw. von Geschäftsprozessen. Der Fokus ist kleiner und konzentrierter, da einzelne Prozesse im Detail und in ihrer Abfolge analysiert und optimiert werden.

Workflowmanagement kann daher als ein Teilbereich oder die Ausführung von einzelnen diskreten Prozessen angesehen werden. [vgl. Perwiss 2018]

Vorweg wird darauf hingewiesen, dass im weiteren Verlauf die Begriffe wie Prozessmanagement, Geschäftsprozessmanagement (GPM) oder Business Process Management (BPM) als Synonyme und somit gleichwertig anzusehen sind. [vgl. Schmelzer und Sesselmann 2013, 5; vgl. Kaya 2016, 4]

Prozess

Was ist ein Prozess? „Ein Prozess ist eine Kette von zusammenhängenden Aktivitäten, die gemeinsam einen Kundennutzen schaffen.“ [Brecht-Hadraschek und Feldbrügge 2013, 13] Diese Definition beruht auf der Mehrzahl von Aktivitäten (was im Zusammenspiel mit einer Mehrzahl von Personen einhergeht) und der Erreichung eines Nutzens. [vgl. Brecht-Hadraschek und Feldbrügge 2013, 13 f]

Geschäftsprozessmanagement

Eine Studie von BearingPoint GmbH untermauert die Bedeutung des Geschäftsprozessmanagements. Unter Geschäftsprozessmanagement werden, ebenso wie beim Projektmanagement, die planbaren, organisatorischen, kontrollierenden und steuernden Maßnahmen im Hinblick auf Qualität, Zeit oder Kosten verstanden. [vgl. BearingPoint GmbH 2015]

Geschäftsprozessmanagement wird als ein integriertes System aus Führung, Organisation und Controlling verstanden. Mit der Absicht, eine zielgerichtete und zeitlogische Steuerung der Prozesse zu ermöglichen sowie mit dem Vorsatz, gestellte strategische und operative Ziele zu erreichen. [vgl. Kuhlmann 2013, 17; vgl. Gadatsch 2013, 41] In der wissenschaftlichen Literatur sind unterschiedlichste Definition für den Prozessbegriff zu finden. Diese teilen sich die Gemeinsamkeit, dass ein Prozess als Ablauffolge von Vorgängen verstanden wird, mit dem Ziel und unter Einbeziehung von Inputfaktoren (Ressourcen, Inputs), die erwarteten Leistungen (Outputs) zu erbringen. [vgl. Reinheimer 2014, 5]

Geschäftsprozess- und Projektmanagement besitzen klare Unterscheidungsmerkmale. Allerdings werden sie in Unternehmen oft ineinandergreifend eingesetzt, da die Vorteile

einer Verzahnung dieser Disziplinen die Aufgabenerfüllung effektiv und effizient unterstützt. [vgl. Litke 2005b, 8; vgl. Kaya 2016, 4 ff]

Es sollte stets eine differenzierte Sicht auf die Disziplinen des Projektmanagements und des Geschäftsprozessmanagements eingenommen werden. Bei einem einmaligen Vorhaben mit einem festgelegten Start- und Endtermin, das geplant, gesteuert und kontrolliert werden muss, wird von Projektmanagement gesprochen. [vgl. Litke 2005b, 8; vgl. Kaya 2016, 4 ff]

Geschäftsprozessmanagement-Tools können in die verschiedenen Kategorien wie Visualisierung, Modellierung oder Simulation eingeteilt werden. Die Programmunterstützung dieser Funktionen ist auch hier sehr verschieden und wird ebenfalls in einer großen Programmvietfalt widerspiegelt. Seit neuester Zeit wird SCRUM auch als Geschäftsprozessmanagementmethode eingesetzt. Was durchaus erfolgsversprechend, jedoch in diesem Zusammenhang noch wenig bekannt ist. Eine Analyse hat allerdings ergeben, dass die Entwicklung neuer Geschäftsprozessmanagementmethoden in den letzten fünf Jahren nachgelassen hat. [vgl. Komus 2015, 40 f; vgl. Kaya 2016, 8 ff; vgl. Gadatsch 2013, 113]

In der Geschäftsprozessmodellierung erfreuen sich die dynamischen Modelle einer großen Bedeutung. Die Unified Modelling Language (UML) bietet zudem die Möglichkeit, spezialisierende Erweiterungen zu den bestehenden Modellen zu definieren. Es ist „ein Sammelbegriff für grafische Methoden zur objektorientierten Entwicklung und Dokumentation von Software.“[Becker, Mathas, und Winkelmann 2009, 58] UML hat nicht nur Diagramme zum Entwurf von Software, sondern auch Methoden zur Darstellung von Geschäftsprozessen im Fundus. Eine weitere genutzte Modellierungssprache ist die BPMN. Diese dient der Abbildung von Prozessabläufen und wird, genau wie die UML, von der Object Management Group (OMG) betreut. [vgl. Seidlmeier 2015, 160; vgl. Rau 2016, 10 ff; vgl. Gadatsch 2013, 2016; vgl. Becker, Mathas, und Winkelmann 2009, 58–70]

Workflowmanagement

Nach der Organisation Workflow Management Coalition (WfMC) wird unter einem automatisierten Prozess – auch Workflow genannt – die Automatisierung eines

Geschäftsprozesses als Ganzes oder eines Teils davon definiert. Unter Einhaltung von prozessualen Regeln werden Dokumente, Informationen oder Aufgaben von einem Prozessbeteiligten (z. B. Personen, Computersysteme oder Maschinen) zu einem anderem weitergereicht. [vgl. Allen 2000, 15; vgl. Kaya 2017b, 20]

Dies auf die Definition von Geschäftsprozessmanagement angewendet ergibt, dass ein Workflow ein vollständiger oder teilweise automatisierter Geschäftsprozess ist. In diesem werden Dokumente, Informationen und Aufgaben zwischen Prozessbeteiligten, entsprechend bestimmter Regeln zur Ausführung übergeben. Daraus lässt sich ableiten, dass Geschäftsprozesse und Workflows ähnliche Merkmale besitzen. Wohingegen ein Geschäftsprozess die Umsetzung der Businessstrategie beschreibt, legt der Workflow rechnergestützt fest, womit dies geschieht. Das bringt im Rahmen des Workflowmanagements eine Verfeinerung der Geschäftsprozesse durch Spezifikationen mit sich, welche für eine automatische Prozessausführung notwendig sind. [vgl. Scharfe 2013, 13 ff; vgl. Gadatsch 2013, 36 f]

Im Rahmen des Workflowmanagements werden die Prozesse auf atomare Tasks runtergebrochen. Neben den Notationselementen in Form von Bedingungen, gibt es verschiedenste Tasks. Der atomare Task repräsentiert z. B. eine Aktivität, welche nicht weiter heruntergebrochen werden kann. Diese können in sogenannten Unternetzen zusammengeführt werden. Die Tools zum Workflowmanagement heißen Workflow-Management-Systeme. [vgl. Scharfe 2013, 31 ff; vgl. Kaltwasser 2017, 2]

Als mögliche Sprache einer solchen Workflowmodellierung soll in den kommenden Zeilen die Variante von YAWL etwas näher erläutert werden. YAWL ist eine komplett neue Sprache mit eigener Semantik, basiert aber auf dem theoretischen Ansatz von Petri-Netzen. Die Modellierung mit YAWL wird durch eine grafische Notation zur Verfügung gestellt und mittels Extensible Markup Language (XML) Syntax als ausführbarer Code beschrieben. Die Sprache ähnelt BPMN und der Notation für UML-Aktivitätsdiagrammen, lehnt sich jedoch eigentlich an die Petri-Netz-Notation an. YAWL wird noch nicht in sehr vielen Projektmanagementsystemen unterstützt. Der Einsatz aber zeigt eine sehr hilfreiche Darstellungsmethode, welche Fehler oder Optimierungen in Prozessen erkennbar machen kann. [vgl. Kaltwasser 2017, 2; vgl. Stoimenova 2008, 22 f]

E^{Anhang E}

Petri-Netze und das Projektmanagement

Schwächen klassischer PM-Methoden

Klassische PM-Methoden können nur bedingt bzw. gar nicht folgende Aufgaben bewältigen: [vgl. Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000, 517 ff; vgl. Thein 2017, 37 f; vgl. Xing-cun Sun u. a. 2013, 515; vgl. Yen-Liang Chen, Ping-Yu Hsu, und Yuan-Bin Chang 2008, 151; vgl. Ashok Kumar und Ganesh 1998, 50 ff; vgl. Jongwook Kim, Desrochers, und Sanderson 1995, 266 ff; vgl. Lung-chun Liu und Horowitz 1989, 1282]

- Modellierung der verschiedenen Arten des Ressourcenverbrauchs (z. B. konkurrierend, gemeinsam, teilweise, und substituiert; siehe [Thein 2017, 101]) durch Aktivitäten.
- Durchführung der Ressourcennivellierung und Ermittlung der Ressourcenauslastung.
- Anpassungen der Projektplanung und des geplanten Ablaufes.
- Automatisierte Durchführung des Projekt-Monitorings und -Controllings, aber in Echtzeit und mittels Simulation.
- Modellierung von mehreren parallelen Vorgängen (gleichzeitig und asynchron), Nebenläufigkeiten, Konflikte und Blockierungen.
- Kann Informationen bei der Ursachensuche bezüglich der Trägheit eines definierten Prozesses oder Aktivitäten bereitstellen.
- Regenerierung und Neeterminierung von Aktivitäten, Einbindung von Dummy-Aktivitäten oder priorisierten Beziehungen in PN.
- PN repräsentiert ein dynamisches System und eignet sich deswegen zum Ad-Hoc-Monitoring.

- Abbildung der Logik in einer geeigneten grafischen Darstellung sowie einer mathematischen Form für eine angemessene Analyse.
- Analytische Modelle bieten maximale Flexibilität in der Modellierung im Gegensatz zu Simulierenden, wo die Flexibilität reduziert ist. PN positionieren sich zwischen diesen beiden Ansätzen.
- Grafische Visualisierungsmöglichkeit der dynamischen Simulation. Mit der Einteilung verschiedener Bereiche/Teile eines Projektes in sogenannte Subnetze ist eine Simulation einzelner Teilbereiche, ohne die Gegebenheiten des gesamten Projektes zu verlieren, möglich.
- Verhaltenseigenschaften wie Erreichbarkeit und Beschränktheit in Bezug auf Ressourcen können die Projektplanung positiv beeinflussen.

Vorteile von Petri-Netz-Konzepten

Die Vorteile des Einsatzes von PN werden durch folgende Aspekte begründet: [vgl. Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000, 519; vgl. Kumar und Ganesh 1998, 55]

(1) Die Basis von PN begründet sich auf einen mächtigen und gut verständlichen Formalismus. Dies ermöglicht eine mehrmalige und parallele Durchführung von Aktivitäten (Nichtdeterminismus und Nebenläufigkeit). Ein weiterer großer Vorteil ist der Umstand, dass nicht nur ein einziger Systemzustand (statisch), sondern alle möglichen Systemzustände (dynamisch) darstellbar sind. Es ermöglicht eine Modellierung des Fortschritts eines Projektes sowie das Regenerieren und Neuplanen von Aktivitäten zum Zeitpunkt von Ausfällen und Ressourcenbeschränkungen. Durch die formal-mathematische Basis kann die automatisierte Bearbeitung von Aufgaben des PM sichergestellt werden und die in Worten generierte Logik nicht nur grafisch, sondern in mathematischer Form zur Analyse aufbereitet werden. Die Möglichkeiten zur grafischen Darstellung verbessern und erleichtern die Kommunikation zwischen den Beteiligten und schaffen ein einheitliches Projektverständnis. [vgl. W.M.P. van der Aalst und van Hee 1996, 25; vgl. Mejía u. a. 2016, 191; vgl. Aichele 2006, 113; Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000]

(2) Ein breiter Funktionsumfang lässt vielfältige Einsatzzwecke zu. Sie können überdies zur Simulation eingesetzt werden und erweitern den Aufgabenkreis des PMs zur Planung, Modellierung, Durchführung, Monitoring und Controlling erheblich. [vgl. Mejía u. a. 2016, 191]

(3) Projekte können von der Vielzahl vorhandener PN-Erweiterungen profitieren, was folgende Eigenschaften positiv beeinflussen kann:

- Ein dynamisches Projektverhalten ist in der Lage, ungeplant häufig und zufällig eintretende Störungen und Ausnahmen (Ressourcen, Zeit und Umfang) sowie geplante Zustandsänderungen (Erreichen von Meilensteinen oder Ressourcenverbrauch) abzubilden. [vgl. Deng, Wendt, und Rolstadås 1998, 84]
- Ressourcen können durch einen konkurrierenden, gemeinsamen, teilweisen und substituierten Ressourcenverbrauch durch Aktivitäten gekennzeichnet sein. [vgl. Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000, 517; vgl. Thein 2017, 101]
- Positiven Einfluss bei Projektbeschränkungen und –aspekten wie bei Mehrmodus-Aktivitäten, Multi-Projekt-Umfeldern oder bei der Einbindung von Mitarbeitern mit unterschiedlichen Qualifikationsstufen. [vgl. Mejía u. a. 2016, 191]
- Konventionelle PM-Tools sind in ihrer Anwendung im Bereich der Ressourcenallokation eingeschränkt. Eine Alternative bietet PN zur Ressourcenallokation mit zusätzlichen Erweiterungsmöglichkeiten. Die klassischen PM-Methoden weisen die folgenden Defizite auf: Sie eignen sich nur eingeschränkt zur Automatisierung, keine Anwendung in Echtzeit möglich und der dynamische Charakter von Projekten findet keine Beachtung. Zusätzlich ebenso keine Beachtung von konkurrierenden, gemeinsamen und teilweisen Ressourcenverbrauch durch Aktivitäten. [vgl. Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000, 517]
- Ein System modellieren, welches viele Aktivitäten zeitgleich und zeitversetzt synchron modellieren kann und somit Konflikte und Engpässe abbildbar macht.

Es befähigt dazu, Aktivitäten zu erneuern und zu planen. Dies führt zur Unterstützung von anteiliger Auslastung/Zuteilung, gegenseitiger Exklusivität und Ressourcenersatz. [vgl. Kumanan und Raja 2008, 1742 ff; vgl. V. A. Jeetendra, Kumanan, und O.V. Krishnaiah 1998]

- Dynamische Simulierung und online Monitoring sowie maximale modellierbare Flexibilität und trotzdem analysierbare Ergebnisse. [vgl. Kumanan und Raja 2008, 1742 ff]

(4) Modellierung von sehr umfangreichen und komplexen Projekten unter Zuhilfenahme von Vereinfachungen ohne Informationsverlust mit Hilfe von Teilnetzen oder Zusammenfassungen. [vgl. Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000, 517]

(5) Unterstützung des Projekt-Monitorings durch Echtzeitüberwachung und -analyse des Projektfortschrittes. Dies beinhaltet ebenfalls die Identifikation von Gründen bezüglich trägen oder raschen Vorankommens des Projektes. [vgl. Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000, 517; vgl. Mejía u. a. 2016, 191]

(6) Unterstützung des Projekt-Controllings durch Echtzeitanpassung und Echtzeitüberarbeitungen der Projektplanung. Dies wird insbesondere bei zufällig eintretenden Störungen und Ausnahmen relevant. [vgl. Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000, 517]

(7) Existierende Projektmodellierungsmethoden und Techniken wie Gantt-Chart, Critical Path Method, PERT, WBS und PNTs weisen die folgenden typischen Nachteile auf: Keine umfangreiche Beschreibung (i. d. R. lediglich teilweise und mit unzureichenden funktionalem Aspekt) der Prozesse und deren Beziehungen, Mangel an einer formalen Beschreibung/Integration des Projektsopes, Ressourcen und Informationen, limitierte Simulationsmöglichkeiten und Unzulänglichkeiten im Umgang mit unvorhersehbaren Daten. [vgl. Deng, Wendt, und Rolstadås 1998]

(8) Dies ermöglicht die Analyse von Wechselwirkungen, Schäden, Konflikten, Ersatzmöglichkeiten, Verfügbarkeiten und Priorisierung von Ressourcen im Einsatz bei einer Ressourcenallokation in Echtzeit in einem ressourcenkritischen Projekt. Eine

dynamische Darstellung kann den Projektmanagern auch dahingehend helfen, die Projektdynamik zur Modellierung von übergreifenden Beziehungen zu verwirklichen. In diesem Kontext können Time Coloured Petri Net (TCPN) ihre Stärken im Bereich Modellierung eines Systems mit vielen gleichzeitig abhängigen, in Konflikt stehenden Aktivitäten erkennen sowie Blockaden identifizieren. [vgl. Bevilacqua, Ciarapica, und Giovanni 2018, 58 ff]

(9) PM-Tools erlauben bei der Umverteilung von Ressourcen keine Zuordnung nach Rollen. Ebenfalls ist die Implementierung von Optimierungsmethoden für Projektabschätzungen nicht möglich. [vgl. Niño Mora u. a. 2015, 3]

(10) Der professionelle Tooleinsatz ist auf seinen ausgelegten Funktionsumfang beschränkt, auch wenn dieser sehr umfangreich ist. Sie sind nicht von außen veränderbar. Wohingegen ein formaler Ansatz, beruhend auf formalisierten Tabellen, erweitert und auf das Problem angepasst werden kann.

(11) Beim Vergleich mit der bekannten PERT-Technik wird die Auswertung der Ergebnisse erweitert. Erweiterungen waren unter anderem die Nutzung von Wenn-Dann-Logik oder booleschen Bedingungen sowie einer Simulation von Zeitunsicherheiten. Weiterhin konnten Ressourceneinschränkungen, stochastischen Dauern, Wahrscheinlichkeitsaussagen und Modellierungsmöglichkeiten zur Verfügung gestellt werden. [vgl. Cohen und Zwikael 2008, 230 f]

(12) Analytische Modelle bieten maximale Flexibilität bei der Modellierung, während die Simulation die Flexibilität verringert. Petri-Netze liegen zwischen diesen beiden Ansätzen und liefern analytische Ergebnisse mit einem Großteil der Modellierungsflexibilität der Simulation. [Prashant Reddy, Kumanan, und Krishnaiah Chetty 2001]

Nachteile von Petri-Netz-Konzepten

(1) PN-Modelle von komplexen Projekten können zu großen Umfängen und hoher Komplexität neigen. Wenn auf eine Vereinfachung verzichtet wurde, wird die grafische Darstellung schnell unübersichtlich und ohne tieferegehende PN-Kenntnisse nicht mehr erfassbar. Klassische PM-Methoden eignen sich im Zweifelsfall besser zur reinen

Visualisierung von Projekten. [vgl. Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000, 516; vgl. Zapf und Heinzl 2000, 36]

(2) Die Modellierung und Bearbeitung mit Hilfe von PN kann selbst bei kleinen und einfachen Projekten manuell schwer durchgeführt werden. Es wird festgehalten, dass dies beim Ausschöpfen aller beschriebenen Vorteile eine Aufwandserhöhung mit sich bringt. Diese Aussage geht einher mit der Annahme, dass der Einsatz von PN die aufgeführten Vorteile nur mit einem formalen mathematischen Ansatz voll ausschöpfen kann und dies auf der Zuhilfenahme von Software beruht. [vgl. Kumar und Ganesh 1998, 55]

(3) Erst im Jahr 2000 wurde hinreichend versucht, PN auch mit anderen Methoden zu kombinieren. Dies schränkte den Funktionsumfang immer ein. Einige Autoren versuchen seitdem, den Funktionsumfang durch eine Kombination mit anderen Modellierungskonzepten zu erweitern. Beispielsweise soll folgende Auflistung einen Einblick geben: [vgl. Zapf und Heinzl 2000, 36; vgl. Deng, Wendt, und Rolstadås 1998, 82 ff]

- Mit objektorientierten Modellierungsmethoden, welche den Fokus darauf legen, Systeme aus verschiedenen Perspektiven zu betrachten und verschiedene Systemobjekte zu identifizieren (z. B. Funktionen, Daten, Vorgänge).
- Mit der Hilfe von Projektstrukturplänen und Balkendiagrammen die statische Dekomposition von Projekten in Projektprozesse vorantreiben und atomare Einheiten bedienen.
- Mit der Modellierungssprache IDEF0 zur Beschreibung von Prozessen und Informationen aus der funktionalen Sichtweise.
- Zur Berechnung von Zeitdauern und zur Modellierung der Projektstruktur wird die Netzplantechnik herangezogen.
- Mit dem Konzept der Fuzzy-Logik, welche zur Modellierung nicht präziser Daten dient.

F Anhang F

Detailiertere Information zum anonymisierten Beispiel

Einblick in das Projektsteering

Die nachfolgende Auflistung soll einen kleinen Einblick geben, was alles während eines Projektes für das Projektsteering/-management aufbereitet werden musste und welche Aussagen von Bedeutung waren. Einige Beispiele werden mit Ausschnitten von realen Projektdaten untermauert. Die Bilder sollen nur der Verdeutlichung dienen. [vgl. BearingPoint GmbH 2013; vgl. BearingPoint GmbH und Weichenhain 2013]

- Eine umfassende Planungsexcel mit Budget, groben Schätzungen und Priorisierungen von Kosten und Personentagen. Dies wurde in Excel erstellt und umfasste 30 Spalten mit etlichen Funktionen und Berechnungen.
- Gesamtprojektübersicht aller Building Blocks in zusammengefassten Zahlen: Status Gesamtprojekt in einem Gantt-Diagramm zur Verfolgung der High-Level-Planung aller Building Blocks. Das Gantt-Diagramm wurde in Excel umgesetzt. Für dessen Erstellung mussten mehrere Diagramme übereinandergelegt werden.
- Gesamtübersicht Budgetverbrauch
- Budgetverbrauch im Abgleich zum wöchentlichen Sprintstatus und Scope. In dieser Excel wurden alle Backloginträge und dessen Status in den einzelnen Phasen festgehalten. Es diente ebenfalls zur Überwachung des Implementierungsfortschritts. Dafür waren 40 Spalten notwendig und diese wurden mit anderen Tabellen verlinkt.
- Ausplanung einzelner Arbeitspakete mit Gegenüberstellung Ist/Plan in Bezug auf Zeit und Ressourcenverbrauch: Die abgerechneten Ist-Stunden (externe) werden beispielsweise dem Plan gegenübergestellt – wieder in einer eigenen Excel Tabelle.
- Projektsteering-Folien für den Fortschritt von Arbeitspaketen bzw. Sprints, Externes Budget, Verfügbarkeiten Plan/Ist auf einen 4-wöchigen Ausblick.

- **Sprint Überblick:** Dieser wurde durch ein Gantt-Diagramm im Excel dargestellt. Auch dafür mussten mehrere Diagramme übereinandergelegt werden. Somit war es machbar, eine Überschreitung mit einer anderen Farbe sowie die unterschiedlichen Phasen farblich zu unterscheiden.
- **Aufgabenverteilung:** Die detaillierte Aufgabenverteilung einzelner Schritte wurde in Excel in Form einer Liste vorgenommen.
- **Budgetüberwachung – Planung Backlog vs. Ist:** Budgetüberblick mit der Gegenüberstellung von Plan Backlog, Ist und dem proportionalen Zeitverbrauch. Auch hierfür wurde sich wieder im Excel einer komplexen Tabelle mit grafischer Darstellung bedient. Die Budgetüberwachung ist eine Zusammenführung mehrerer Tabellenblätter, um eine schon eingetretene Überschreitung zu erkennen. Ebenfalls wurde in Excel integriert, dass noch nicht beendete bzw. nicht angefangene Aufgaben zu den Ist-Zahlen addiert werden. Dies ermöglichte ein Frühwarnsystem zu integrieren, welches wöchentlich auf der Grundlage von der geschätzten Aufgabenerfüllung beruhte.
- **Ressourcenallokation für die nächsten vier Wochen:** Die Einbringung von SCRUM als agile Vorgehensweise hat vor allem in der Ressourcenallokation seine Herausforderungen. Ganz klar zu erkennen ist die Abhängigkeit der Umsetzung. Die Zuordnung von fertig definierten Anforderungen zu einem Sprint kann aus unterschiedlichen Fachthemen resultieren und somit nicht immer alle zur Verfügung gestellten Entwickler beanspruchen. Aber auch im Fachbereich ist durch den damit verbundenen Test-Ansatz immer wieder der pünktliche Abruf von Ressourcen ein wesentlicher Faktor. Somit ist die Aussage über die Ressourcenallokation, für mindestens die nächsten vier Wochen, von Bedeutung. Dazu wurde ebenfalls aus unterschiedlichen Excel Quellen eine Art Matrix entworfen, welche auf der Basis der geplanten Bearbeitungszeiträume, gefühlten Fertigstellungsgraden und schon geleisteten Tagen eine Hochrechnung für die nächsten vier Wochen bereit gestellt.
- **Für das Projektsteering** wurde der Status der laufenden Arbeitspakete wöchentlich in einer Präsentation zusammengestellt. Dabei wurde der Durchschnitt des

gefühlten Fertigstellungsgrades proportional auf die Zeitachse gelegt. In Excel wurde die Grundlage der ersten Gruppe – Vorklärung, Abklärung und Design – mit einem gefühlten Fertigstellungsgrad wöchentlich abgefragt und eingetragen. Ebenfalls wird hier die Eingliederung aller Aufgaben in die Backlog durch eine Statusabfrage festgehalten. Diese Summe ergibt den kumulierten Fertigstellungsgrad per Arbeitspaket. Für die Betrachtung eines Sprints wurde anschließend der Durchschnitt in einer Gleichverteilung über alle bekannten Aufgaben (inkl. noch nicht gestarteter) gebildet.

Rollierend wird eine monatliche Arbeitspaketplanung geführt, welche immer mindestens wöchentlich aktualisiert wird. Auch wenn die Abklärung pro Aufgabe erfolgte war das Ziel gesetzt, alle Aufgaben in einer vorgegebenen Zeit abzuklären. [BearingPoint 2013]

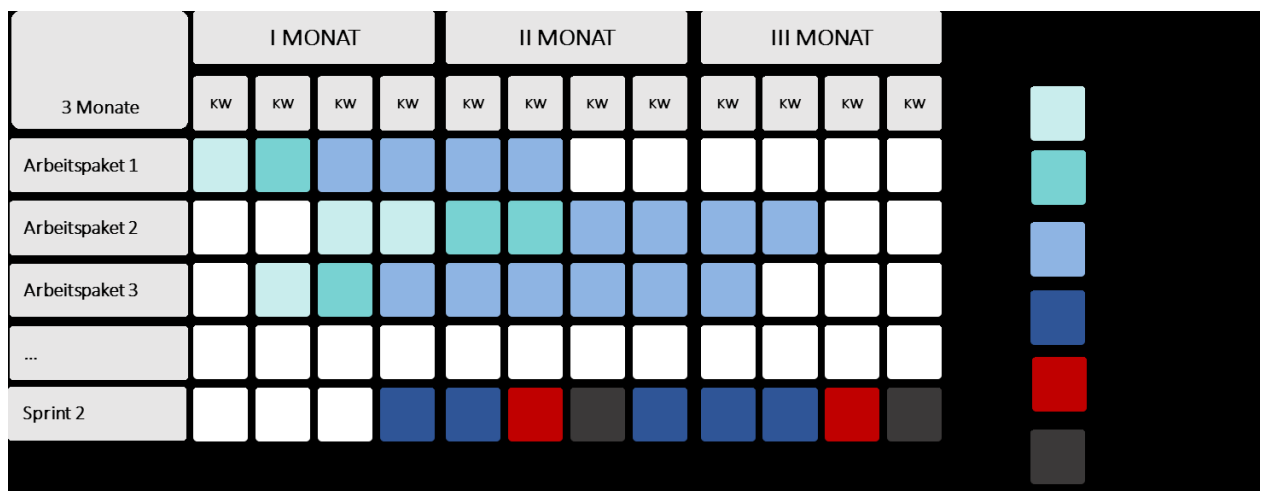


Abbildung 41: Dreimonatsplanung Arbeitspakete

In Summe kann festgehalten werden, dass ein großer Aufwand von Nöten war, um die Simulation der niedergeschriebenen Plandaten (Starttermine, Endtermine, Ressourcenallokation, Fortschritt, Verschiebung, Abhängigkeiten) mit Excel zu prüfen und Aussagen über die Machbarkeit und den Projektstatus zu liefern. In jedem Projekt gibt es einen gefühlten Projektstatus des momentanen Arbeitsstandes. Aber es zu belegen, dass beispielsweise der Fachbereich es in drei Monaten nicht testen kann oder dass in der Vergangenheit in der Fachspezifikation 50 % mehr Zeit benötigt wurde, ist mit großem Aufwand verbunden.

Nähere Informationen zum Arbeitspaket

Ein Arbeitspaket wurde in die Phasen Vorklärung, Abklärung, Design, Realisierung, Test und Einführung unterteilt.

Auszug Projektplan:

Tabelle 12: Projektplan Beispiel

ID	Bezeichnung	Start	Ende	Status
F1	Fachliches Thema 1	08.01.2018	14.01.2018	100 %
PAA	Fachliches Thema PAA	14.02.2018	30.03.2018	40 %
...				
S2	Sprint 2	08.01.2018	23.02.2018	
PAA_001	Task 1 aus F1	-	-	90 %
S3	Sprint 3	26.02.2018	13.04.2018	
PAA_003	Task 3 aus PAA	-	-	10 %

Die Phase Vorklärung ist Bestandteil des Arbeitspaketes und dient der Absteckung des Scopes des gesamten Arbeitspaketes. Die Abklärung beinhaltet die Identifikation der Querschnittsthemen und die grobe Besprechung des Scopes mit allen Verantwortlichen. Das Design beinhaltet Definitionen der fachlichen Anforderungen (Fachkonzept) pro identifizierte Aufgaben, und den technischen Teil zur groben Abklärung, Schätzung und Design der Anforderung. Die Phasen ab Abklärung wiederholen sich somit pro absteckbarer Anforderung (Aufgaben) so oft in einem Arbeitspaket, bis der abgesteckte Scope fertiggestellt wurde. Eine Unterscheidung ist nur darin zu finden, dass die Phasen ab Realisierung im Rahmen des Sprints – zeitlich vorgegeben – erledigt werden müssen. Wobei der Zeitrahmen i. d. R. sechs Wochen nicht überschreitet. Dagegen unterliegen die ersten drei Phasen je nach fachlicher Anforderung keiner strikten Zeitvorgabe. Hervorzuheben ist wie schon erwähnt, die Einhaltung der kontinuierlichen Lieferung von fachlichen Anforderungen. [vgl. BearingPoint GmbH und Weichenhain 2017]

Backlog Beispiel

Tabelle 13: Backlog Beispiel

Task ID	Task	Priority	Sprint	Estimation Total	Design days	Design Start	Design End	Mitarbeiter Design
PAA_003	Excel als Schnittstelle einrichten	Medium	2	2	0,7	22.01.18	26.01.18	Person 5
PAA_004	Report - Filterfunktion	Medium	2	5	1,7	22.01.18	26.01.18	Person 5
PAA_005	Report - Standardlayout festlegen	High	3	3	1,0	15.01.18	19.01.18	Person 5
PAA_006	...	Low	2	1	1,0	15.01.18	19.01.18	Person 6
PAA_008	...	Low	2	10	3,3	26.02.18	09.03.18	Person 7

Task ID	Dev days	Dev Start	Dev End	Developer	Expected final due date	Status Design	Status Development (Build + pretest)	Status Test	Technical Doku	On Hold/Risk
PAA_003	1,3	29.01.18	02.02.18	Person 11	09.03.18	100% Done	100% Done	75% Work in Progress	100% Done	
PAA_004	3,3	29.01.18	02.02.18	Person 11	09.03.18	100% Done	100% Done	75% Work in Progress		
PAA_005	2,0	22.01.18	26.01.18	Person 11	09.03.18	100% Done	100% Done	75% Work in Progress	100% Done	
PAA_006	0,0					100% Done	100% Done	100% Done	100% Done	
PAA_008	6,7	12.03.18	23.03.18	Person 12	30.04.18	50% Work in Progress		5% Started		

GAnhang G

Vergleich Publikationen zu Petri-Netz-Ansätzen im Projektmanagement

Es wurden verschiedenste Forschungen über den Einsatz von Petri-Netzen im Projektmanagement oder angrenzenden Themengebieten betrachtet. Die interessantesten Forschungen mit einem ganzheitlichen Bezug zu PM werden zur Übersicht in der nachfolgenden Tabelle vorgestellt.

Tabelle 14: Vergleichstabelle Publikationen Petri-Netze in Projektmanagement

Thema	Jahr	Titel	Ziel	Vorgehen
Zeit- & Kostenplan SW-Projekten [Chun-suk Park, Gang-soo Lee, und Jung-mo Yoon 1993]	1993	An executable software project management model by beta-distributed stochastic Petri nets	Entwicklung eines ausführbaren Modells zur Unterstützung der Zeit- und Kostenplanung.	<ul style="list-style-type: none"> ▪ Stochastisches PN mit wahrscheinlichkeitsverteilten Durchführungskosten und -zeiten von Aktivitäten
PM zur Erstellung von Ablaufplänen [Jongwook Kim, Desrochers, und Sanderson 1995]	1995	Task planning and project management using Petri nets	Entwicklung eines interaktiven PM-Ansatzes, welcher die initiale Erstellung des Ablaufplans sowie die Überwachung und die Anpassung des Ablaufplans automatisiert.	<ul style="list-style-type: none"> ▪ Grundidee: Historische Planungsmethoden (Netzplantechnik) und markierte/zeitbewertete PN-Erweiterung mit drei unterschiedlichen PN-Modellen, welche das Projekt in unterschiedlichen Zuständen abbilden (Referenz-PN, Monitor-PN, Voraussage-PN) ▪ Vorgang automatisiert ▪ Beziehung zwischen den drei PN-Modellen = Iteration mit ständiger Rückkopplung von Informationen
Prozessorientierte Projektmodellierung [Deng, Wendt, und Rolstadås 1998]	1998	A Set of Modeling Methods for Process-Oriented Project Modeling	Die Komplexität des Projektmodells beschreibbar und hantierbar machen. An Echtzeit angelehntes Modell zur Analyse von dynamischen Verhalten eines Systems/Prozesses.	<ul style="list-style-type: none"> ▪ Nicht relevante Informationen rausfiltern ▪ Unvorhergesehene Ereignisse mittels einer Mustererkennung integrieren ▪ Mehrere Techniken aufeinander aufbauend zu benutzen <ul style="list-style-type: none"> ○ Beispielsweise Prozessduration mit PERT und CPM ○ Set aus Methoden von IDEF0, Petri-Netze und Fuzzy-Logik ○ Die Prozessdauer wird mit Hilfe von PERT oder CPM berechnet ▪ Coloured Petri-Netze zur endlichen Beschreibung der Projektsicht ▪ Fuzzy-Logik zur Verarbeitung von unvorhergesehenen Ereignissen

Thema	Jahr	Titel	Ziel	Vorgehen
Ressourcenallokation [Kumar und Ganesh 1998]	1998	Use of petri nets for resource allocation in projects	Dieses Dokument beschreibt die Verwendung von Petri-Netzen zur Erleichterung der Ressourcenzuteilung in Projekten unter bestimmten Bedingungen, die in der Praxis häufig anzutreffen sind.	<ul style="list-style-type: none"> Integration eines Formalismus zur Darstellung und Analyse gleichzeitig ablaufender Systeme Demonstration der Nützlichkeit des vorgeschlagenen Modells für die Echtzeitplanung von Aktivitäten in einer Projektumgebung mit begrenzten Ressourcen Das Modell wurde unter Verwendung der Sprache C ausgebaut. In diesem Dokument werden auch die Implikationen des Modells und die von ihm unterstützte Analyse erörtert
PN auf Projekt-management [V. A. Jeetendra, Kumanan, und O.V. Krishnaiah 1998]	1998	Application of Petrinets to Project Management	Eine Vielzahl konkurrierender Methoden mit unterschiedlichen Zeit-, Kosten- und Ressourcenverbrauch sollen zur Erfüllung von Aufgaben in die Planung mit aufgenommen werden und so einen besseren Vergleich gewährleisten, bzw. den optimalen Weg identifizieren.	<ul style="list-style-type: none"> Drei Komponenten (Project-Net, RESL-Net und SIM-Net) PERT mit den Vorteilen von PN anreichern Quelldatei enthält Informationen von optimistischen, realistisch und pessimistischen Schätzung aller Aktivitäten und deren benötigten Ressourcen Simulation
PM-Net [Chang und Christensen 1999]	1999	A net practice for software project management	Die technische und verwaltungstechnische Komplexität überfordert zunehmend die Projektmanager. Um diese Komplexität in den Griff zu bekommen, wird PM-Net vorgeschlagen.	<ul style="list-style-type: none"> Fokus: Koordinieren und Kontrollieren von Ressourcen Es verwendet die grundlegenden Konzepte von Petri-Netzen und grafischen Modellen des Informationsflusses mit Erweiterungen, um sowohl Entscheidungen als auch Artefakte darzustellen Ein sogenanntes PMN Projekt Management Netz wird als Idee vorgestellt. Die Ausgangsbasis für das PN bilden ein TPG (Task Precedence Diagramm) für die spätere Überführung

Thema	Jahr	Titel	Ziel	Vorgehen
Ressourcennivellierung [Jeetendra, Krishnaiah Chetty, und Prashanth Reddy 2000]	2000	Petri nets for project management and resource levelling	Eine grafische und analytische Methode zur Modellierung, Analyse und Unterstützung der Ressourcennivellierung.	<ul style="list-style-type: none"> ▪ Focus auf Ressourcen ▪ Der Umgang mit Ressourcen wird in zwei Gruppen eingeteilt: <ul style="list-style-type: none"> ○ Ressourcen-Beschränkung (Ziel: Projektlaufzeit zu minimieren) ○ Ressourcen-Nivellierung (gleichmäßige Ressourcenauslastung unter Berücksichtigung der Verfügbarkeiten) ▪ Mit Hilfe einer Heuristik (Einsatz von Entscheidungsregeln und einer Matrix) ▪ Erweiterungen für das PN und deren softwareseitige Umsetzung ▪ Anwendung einer P-Matrix zur Token-Weitergabe im PN
	2001	Application of Petri Nets and a Genetic Algorithm to Multi-Mode Multi-Resource Constrained Project Scheduling	Ziel ist die Verwendung eines Petri-Netzes als Modellierungs- und Zeitplanungstool für die Planung eines Projektes mit mehreren Betriebsarten und mehreren Ressourcen.	<ul style="list-style-type: none"> ▪ Bei einer auf Ressourcen beschränkter Projektplanung im Fertigungsbereich spielt die Unsicherheit der Vorhersagegenauigkeit eine große Rolle. Die Vorteile von Petri-Netzen bei der Projektplanung werden diskutiert. Erweiterungen für Petri-Netze werden vorgeschlagen, um die Planung von Aktivitäten in einem Entscheidungs-CPM anzupassen. Das in Borland C entwickelte MMRCS-Netz besteht aus drei Modulen. Ebenfalls kommt eine P-Matrix für Token-Bewegungen in Petri-Netzen zum Einsatz. In Anbetracht von Informationen wie z. B. Vorrangbeziehungen ▪ Entscheidungsknoten, Anzahl Alternativen werden in alternativen Pfaden generiert ▪ Ein genetischer Algorithmus wird verwendet, um eine bessere Lösung zu finden ▪ Petri-Netz gestützte Software, die auf genetischen Algorithmen basierende Suche und Heuristiken umfasst, wird beschrieben, um ein Multi-Mode-Planungsproblem mit mehreren Einschränkungen und vorhergehender Aktivität zu behandeln

Thema	Jahr	Titel	Ziel	Vorgehen
Ressourcenplanung Multi-Projektumfeld [Kao, Hsieh, und Yeh 2006]	2006	A Petri-Net based approach for scheduling and rescheduling resource-constrained multiple projects	Bildet einen eventbasierten Ansatz ab, der versucht, alle in sich gebundenen Projekten wie in einem dynamischen Netzwerk zu gruppieren.	<ul style="list-style-type: none"> ▪ Multi-Projekt-Netzwerk mit Höheren PN aufbauen ▪ Bei Eintritt eines signifikanten Events wird die modellierte Planung bis zum nächsten bevorstehenden Ereignis durchlaufen ▪ Vorhandene Blockierungen ermitteln und mögliche Varianten generieren ▪ Lösung identifizieren und anwenden ▪ Darstellung eines reduzierten Umfangs durch PN und mit anschließenden Vergleich gegenüber CPM
Ressourcenplanung [Raja und Kumanan 2007]	2007	Resource Leveling Using Petrinet and Memetic Approach	Ziel ist es einen gleichbleibenden optimalen Ressourcenverbrauch sicher zu stellen. Gleichzeitige Planung und Nivellierung der Ressourcen (dynamische Komponenten der Aktivitäten mit starren Priorisierungen/Ordnungen).	<ul style="list-style-type: none"> ▪ Fokus nur auf Ressourcenplanung ▪ Petri-Netz basiertes Software PERTIPM in Kombination mit einem Memetik Algorithmus und der Kombination von Evolutionären Algorithmen (stochastische Optimierungsverfahren) und lokalen Suchverfahren ▪ Optimierungsverfahren mit der Berücksichtigung verschiedener Ressourcenallokationen
Überwachung / Rekonfiguration Projektplanung [Haji und Darabi 2007]	2007	Petri Net based Supervisory Control Reconfiguration of Project Management Systems	Überwachung und Rekonfiguration der Projektplanung zur Vermeidung von Unsicherheiten bei der Ressourcenverfügbarkeit oder Planungsabweichungen bei Durchführungsdauern.	<ul style="list-style-type: none"> ▪ Eine einfache PN-Erweiterung und heuristische Methode der Ablaufplanung bei Ressourcenbeschränkungen ▪ Modellierung des Projektes als PN ▪ Projektplanung unter Beachtung alternativer Zeiten und Ressourcen ▪ Einplanung von Ressourcenbeschränkungen ▪ Monitoring und Controlling des Projektfortschritts ▪ Bei Unterbrechung: Rekonfiguration bzw. Anpassung in Echtzeit. ▪ Überwachungscontroller beendet die Schritte

Thema	Jahr	Titel	Ziel	Vorgehen
Modellierung/ Simulation von Projekten mit PN [Kumaran und Raja 2008]	2008	Modeling and Simulation of Projects with Petri Nets	Gebrauch des PN als grafisches und mathematisches Tool im Rahmen von PM.	<ul style="list-style-type: none"> ▪ Fokus Ressourcenzuweisung und die Zurückführung von wiederverwendbaren Ressourcen durch einen Puffer-Platz und Transitionsbedingungen ▪ Prozess der Tokenentnahme oder -ablage wird mithilfe einer PPC-Matrix umgesetzt ▪ Grafische Simulation als Hilfestellung zur Projektplanung ▪ Projektanalyse, Inzidenzmatrix (Beziehungen der Knoten und Kanten) ▪ Erreichbarkeitsanalyse oder Analysen zu gewichteten Marken- bzw. Schaltzahlen
Unterstützung der Ressourcenauslastung [Yen-Liang Chen, Ping-Yu Hsu, und Yuan-Bin Chang 2008]	2008	A Petri Net Approach to Support Resource Assignment in Project Management	PN-Erweiterung, welche die Modellierung und Simulation von Strategien der Ressourcenauslastung ermöglicht (Object Composition Petri Net).	<ul style="list-style-type: none"> ▪ Plätze werden mit einem Zeitattribut versehen und fungieren somit als zusätzliche Schaltbedingung einer nachfolgenden Transition ▪ Plätze zur Modellierung unterschiedlicher Zustände im System ▪ Transitionen mit unterschiedlichen Aufgaben und PN-Muster zur Modellierung der Ressourcenauslastung zwischen den Aktivitäten
Projektmodellierung & -planung [Cohen und Zwickael 2008]	2008	Modelling and scheduling projects using Petri nets	Ziel ist, die Modellierbarkeit von Petri-Netzen für die Projektplanung und -steuerung zu betrachten. In diesem Kontext gibt es bisher kaum Erwähnungen. Die Modellierungsvorteile sollen aufgezeigt und die Eignung somit untersucht werden.	<ul style="list-style-type: none"> ▪ Abbildung von Projektplanungsproblemen (Planung und Ressourcenplanung) ▪ Anfangs wird das gewählte Beispiel mit einem klassischen PN abgebildet. Dies wird schrittweise um die Ressourcen und stochastische Angaben bei den Zeitangaben sowie AND/OR-Operationen in der Entscheidungsfindung erweitert ▪ Ebenfalls wird eine Zeitanzeige integriert: Dabei wird beispielsweise eine Zeitleiste über das PN gelegt und die Anordnung der Transitionen und Plätze mit Regeln vorgenommen werden ▪ Abschließend wird auf die Möglichkeit einer Simulation eingegangen

Thema	Jahr	Titel	Ziel	Vorgehen
Erweiterung des kritischen Pfades [Lin, Tai, und Hu 2013]	2013	An Enhanced Critical Path Method for Multiple Resource Constraints	Die Berücksichtigung von Aktivitäten mit Ressourceneinschränkungen und dessen Ressourcenanzahl zu identifizieren.	<ul style="list-style-type: none"> ▪ Kritischen Pfad mit CPM ermitteln ▪ Konvertierung von CPM zu PN ▪ PN verifizieren ▪ Ressourceneinschränkungen hinzufügen ▪ Erneute Rückführung ins CPM zur Untersuchung des Ergebnisses bezüglich des kritischen Pfades
	2013	A New Framework on Monitoring Equipment Technology Reformation Project Management Based on Petri Net [Xing-cun Sun u. a. 2013]	Eine Überwachung des Projektfortschritts und Budgeteinhaltung in Echtzeit.	<ul style="list-style-type: none"> ▪ Zeiten und Kosten simuliert ▪ Überwachung des laufenden Projektes ▪ Ist-Plan Vergleich und somit Ermittlung von Abweichungen zum aufgestellten Plan ▪ Basis zur Umsetzung dieses Frameworks bildet eine nicht näher spezifizierte PN-Erweiterung, das Konzept der serviceorientierten Architekturen und des bestehenden PM des Unternehmens
PN im PM [Lin und Dai 2014]	2014	Applying Petri Nets on Project Management	Erweiterung des CPM mit logischen Abhängigkeiten zwischen Aktivitäten, welche die gleichen Ressourcen beanspruchen können. Analyse des Ressourceneinsatzes zur Performancesteigerung.	<ul style="list-style-type: none"> ▪ Klassische CPM erstellen ▪ Konvertierung von CPM zu PN ▪ PN um Ressourcenbeschränkungen erweitert ▪ Konvertierung zurück zu einem klassischen CPM (kritischen Pfad ermitteln)

Thema	Jahr	Titel	Ziel	Vorgehen
Risiko- & Notfall-Management von SW-Projekten [Qu und Wang 2015]	2015	A risk emergency management model for software project based on stochastic Petri Nets	Entwicklung einer PN-Erweiterung, welche den Projektfortschritt in Echtzeit analysiert und Risiken erkennt.	<ul style="list-style-type: none"> ▪ Stochastisches PN mit Durchführungshäufigkeiten der Transitionen pro Zeiteinheit ▪ Zeitkontinuierliche Markow-Ketten ▪ Startpunkt: Definition Notfall-Management-Prozess ▪ Modellierung des Prozesses als PN ▪ Konstruktion einer Markow-Kette an ▪ Qualitative Analyse des PN-Modells, sowie die quantitative Analyse der Markow-Kette ▪ Ableitung von Aussagen über Verfügbarkeit und Effektivität des Notfall-Management-Prozesses
Framework zur Planung & Steuerung [Niño Mora u. a. 2015]	2015	A Framework for Project Management and Scheduling Using Petri Nets	Ziel ist ein kollaboratives Framework für die Planung und Steuerung von Projekten mithilfe von Business Process Model und Notationen (BPMN), Extended Petri Nets und kommerzieller Projektmanagementsoftware.	<ul style="list-style-type: none"> ▪ Scope: Ressourcenzuweisung, -beschränkungen und Vorrang ▪ Das als Entscheidungshilfesystem entwickelte Framework kann mehrere Alternativen (Szenarien) für die Ressourcenzuteilung und Zeitpläne generieren ▪ Die Funktionalität des Frameworks wurde in mehreren realen Fällen in der Animations- und Videospiegelindustrie in Kolumbien bewertet ▪ Dabei wurde im ersten Schritt (Modellierung) das Projekt als BPMN abgebildet, Anhand einer Festlegung von Überführungsregeln (Symbol BPMN = PN Komponente) ▪ In der Planungsphase werden die durch die Konvertierung in ein PN gewonnenen Aussagen (wann kann eine Transition gefeuert werden) weiter definiert und Prioritäten mit Regeln (längste oder kürzeste Dauer, usw.) festgelegt

Thema	Jahr	Titel	Ziel	Vorgehen
Modellierung, Planung & Simulation von Projekten <i>[Mejta u. a. 2016]</i>	2016	A Petri Net-based framework for realistic project management and scheduling: An application in animation and videogames	Darstellung eines allumfassenden Frameworks zur Modellierung, Ablaufplanung und Simulation von Projekten mit (1) eine neu entwickelte PN- in animation and Erweiterung (2) zwei PN-Muster und (3) ein Graphen-Such-Algorithmus.	<ul style="list-style-type: none"> ▪ Erstellung eines Business-Process-Model-and-Notation-Projekt-Modells ▪ Überführung dieses Modells in ein PN-Modell mit anschließender Durchführung der Projektplanung mittels eines Algorithmus ▪ Erneute Projektplanung unter Berücksichtigung von zufällig auftretendem Überarbeitungsbedarf
	2018	Timed Coloured Petri Nets for Modelling and Managing Processes and Projects	Ziel ist die Ressourcenallokation mit einer Kombination aus TPN, SPN und CPN.	<ul style="list-style-type: none"> ▪ WBS: bricht Projektaktivitäten in diskrete einzelne Arbeitselemente runter, beschreibt und gruppiert sie ▪ Projektaktivitätenanalyse ▪ TCPN erstellt und simuliert

H^{Anhang H}

Legende für Abbildungen des Projekt-Petri-Netz-Modells

Folgende Legende ist für alle PPNM-Abbildung relevant:

- Arbeitsfreie Zeit (AfZ)
- Arbeitszeit (AZ)
- Aufgabe (A)
- Dauer (D)
- Enddatum (ED)
- Farbige Marke (FM)
- Lernen/Modifizieren (L/M)
- Menge (M)
- Modellzeit (MZ)
- Natürliche Zahl (n)
- Koordinator (K)
- Projekt (P)
- Ressource (R)
- Ressourcenanforderung (RA)
- Ressourcenbereitstellung (RB)
- Ressourcenerneuerung (RE)
- Ressourcengeneration (RG)
- Ressourcenpool (RP)
- Ressourcenregeneration (RR)
- Ressource Stück (R-St)
- Ressource Vielfachheit (R-V)
- Ressourcenverwaltung (RV)
- Ressourcenzuordnung (RZ)
- Schwarze Marke (SM)
- Sprint (S)
- Startdatum (SD)
- Strukturierte Marke (StM)