

Iterationsarten und deren Auslöser in der Frühen Phase der PGE – Produktgenerationsentwicklung

Miriam Wilmsen, Markus Spadinger, Albert Albers, Cong Minh Nguyen und Jonas Heimicke

1. Motivation

Insbesondere die frühen Phasen in Prozessen der Mechatroniksystementwicklung sind durch ein hohes Maß an Unsicherheit gekennzeichnet. Zu diesem Zeitpunkt des Produktentstehungsprozesses liegen lediglich vage und unscharfe Anforderungen an das Produkt vor, welche es zu konkretisieren gilt. Aktuelle Herausforderungen der Produktentwicklung haben einen verstärkenden Effekt auf die Unsicherheiten in frühen Entwicklungsphasen. Diesen begegnen Unternehmen aus dem Bereich der Mechatroniksystementwicklung zunehmend mit der Implementierung agiler Entwicklungsansätze in ihre etablierten Prozesse.

Neben der frühen und kontinuierlichen Kundeneinbindung, der klare Ausrichtung der Prozesse auf die Wertsteigerung der Produkte aus Kundensicht, flachen und offenen Hierarchien und dem stetigen Aufbau und Weiterentwicklung von Prototypen verleihen meist geplante Iterationen den jeweiligen Projekten das Adjektiv „agil“. Die Vielfalt der in der Literatur beschriebenen und meist generisch formulierten Arten von Iterationen (geplant oder ungeplant, korrekturbezogen oder progressiv) ist jedoch sehr groß. Zudem werden Iterationen in der Praxis meist intuitiv und unbewusst durchgeführt, was zum einen dazu führt, dass das jeweilige Entwicklungsvorgehen nicht situationsoptimal ausgeführt wird oder gar hinsichtlich der Ergebnissynthese und -Analyse redundante Tätigkeiten erfolgen. Aus diesem Grund verfolgt das vorliegende Forschungsvorhaben die Zielsetzung, einen Beitrag zur Unterstützung der Produktentwickler in frühen Entwicklungsphasen bei der Identifikation notwendiger Iterationen zu leisten. Durch eine Berücksichtigung

dieser in der kurz- und mittelfristigen Projektplanung kann somit die Prozessunsicherheit reduziert werden.

2. Stand der Forschung

2.1 Modellierung von Produktentstehungsprozessen

Produktentstehungsprozesse sind seit jeher von Unsicherheiten gekennzeichnet (Thomke & Reinertsen, 1998), die eine tages- und inhaltsgenaue Prozessplanung nicht zulassen (Smith & Morrow, 1999). Um die Orientierung von Entwicklerteams in dieser un stetigen und komplexen Umgebung zu unterstützen, wurde eine Vielzahl an Prozessmodellen mit unterschiedlichen Zwecken und Auflösungsbereichen (Detaillierungsgrade in der Prozessbeschreibung) entwickelt (Wynn & Clarkson, 2018).

Zudem lassen sich Prozesse unabhängig vom verwendeten Prozessmodell als plangesteuert, hybrid oder agil verstehen (Boehm & Turner, 2003). Während plangesteuerte Vorgehensweisen das klassische Projektmanagement in Anlehnung an den bewährten Stage-Gate-Prozess (Cooper, R. G., 1990) beschreiben, investieren agile Projektteams vergleichsweise weniger Aufwand in die Projektplanung und forcieren eine iterative und inkrementelle Entwicklung im Einklang mit dem agilen Manifest (Fowler & Highsmith, 2001). Praktiken wie Scrum (Schwaber & Sutherland, 2017) oder Design Thinking (Plattner, Meinel & Leifer, 2011) wurden bereits erfolgreich in der Softwareentwicklung angewandt und finden kontinuierlich ihren Weg in die Prozesse der Mechatroniksystementwicklung (Schmidt, Paetzold & Weiss, 2018). Da jedoch das Anforderungsprofil an die Ansätze, die Entwicklerteams im Bereich der physischen Entwicklung unterstützen, von dem in der Softwareentwicklung divergiert, stoßen die in der Software etablierten Ansätze im mechatronischen Umfeld schnell an ihre Grenzen (Heimicke et al., 2018; Schmidt et al., 2018). Aus diesem Grund fokussieren derzeit viele Forschungsvorhaben die Entwicklung hybrider Ansätze (Cooper, R. G., 2016; Spreiter, Böhmer & Lindemann, 2018), die jedoch zumeist eine einfache Adaption bestehender Ansätze an die neuen Umgebungsbedingungen darstellen.

Der Ansatz des ASD – Agile Systems Design ermöglicht es Entwicklerteams, ein situations- und bedarfsgerechtes Maß an Agilität in ihrem Entwicklungs-

vorgehen zu integrieren und auf unterschiedlichen Prozessebenen anzuwenden (Albers, Heimicke et al., 2019). Dies erfolgt durch die zielgerichtete Kombination strukturierender und flexibler Prozesselemente (siehe Abbildung 1).

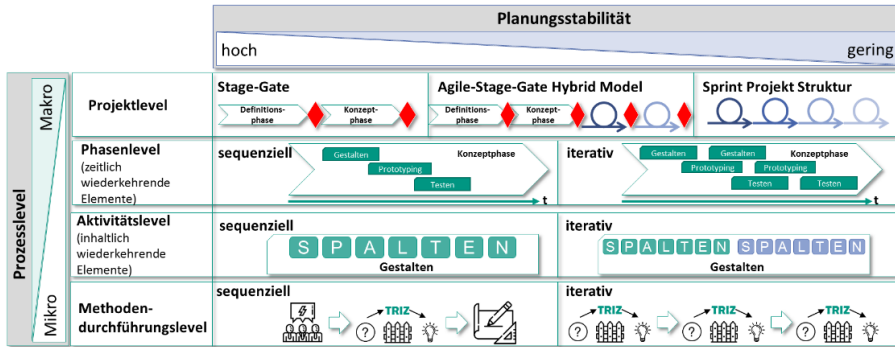


Abbildung 1: Situations- und bedarfsgerechte Kombination strukturierender und flexibler Elemente (Albers, Heimicke et al., 2019)

Dabei ist der Ansatz des ASD – Agile Systems Design auf eine über 20 Jahren zurückliegende Anwendung und Weiterentwicklung in Innovations- und Forschungsprojekten begründet (Albers, Bursac, Heimicke, Walter & Reiß, 2017). ASD – Agile Systems Design ist ein ganzheitlicher strukturierender Ansatz zur agilen Entwicklung mechatronischer Systeme, der zugehörigen Produktions- und Validierungssysteme sowie der Strategie basierend auf Denkweisen, Methoden und Prozessen der PGE – Produktgenerationsentwicklung (Albers et al., 2018). Er basiert auf 9 Grundprinzipien (Albers, Heimicke et al., 2019), die situations- und bedarfsgerecht zur Unterstützung von Entwicklerteams im Produktentstehungsprozess operationalisiert werden.

Das in Abbildung 1 dargestellte Modell zur Implementierung folgt den Prinzipien der Skalierung von Methoden und Prozessen, Berücksichtigung der Einzigartigkeit von Projekten, der Kombination strukturierender und flexibler Elemente sowie der Modellierung von Produktentstehungsaktivitäten als Problemlösungsprozess. In Abhängigkeit der einer Situation zugrundeliegenden, erwarteten Planungsstabilität (Breitschuh et al., 2018) ausgewählter Prozesselemente lässt sich auf unterschiedlichen Prozessebenen zwischen sequenziellen oder iterativen Vorgehensweisen unterscheiden (Albers, Heimicke et al., 2019).

Dabei unterstützt der Ansatz des ASD – Agile Systems Design Entwickler in der Prozessgestaltung und legt als Prozessmodell das iPeM – integrierte Produktentstehungsmodell (siehe Abbildung 2) zugrunde (Albers, Reiß, Bursac & Richter, 2016).

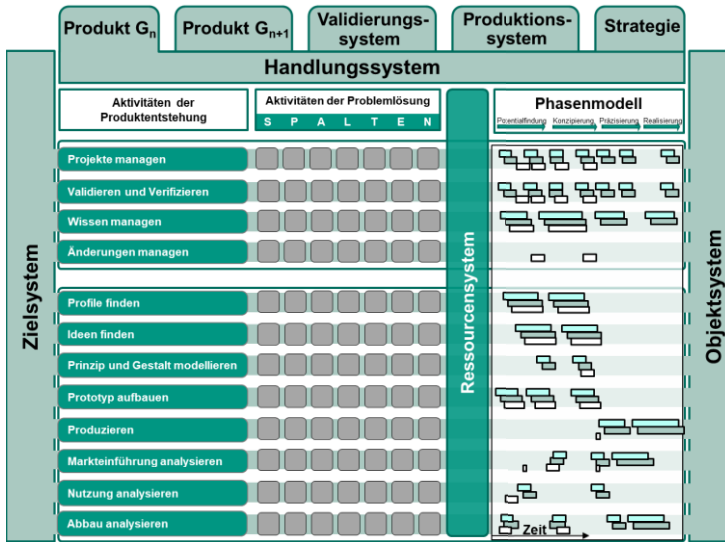


Abbildung 2: iPeM – integriertes Produktentstehungsmodell (2D Darstellung) (Albers et al. 2016b)

Das iPeM basiert auf der Systemtheorie nach Ropohl (Ropohl, 1979) und modelliert die Produktentstehung in Einklang mit dem ASD – Agile Systems Design als Überführung eines Zielsystems in ein Objektsystem durch ein Handlungssystem (Albers, Lohmeyer & Ebel, 2011). Das iPeM besteht aus einem statischen Teil und einem dynamischen Teil. Im statischen Teil sind alle Aktivitäten dargestellt, die zur Erschaffung eines Produkts potentiell durchgeführt werden. Diese sind wiederum jeweils als Problemlösungsprozess unter Nutzung der Problemlösungsmethodik SPALTEN (Albers, Reiß, Bursac & Breitschuh, 2016) modelliert. Hierdurch wird die sogenannte Aktivitätsmatrix aufgespannt, die 84 Felder enthält und als eine Art Methodenbaukasten dient, welcher Methoden und weitere Prozesselemente enthält, die zur Unterstützung der Aktivitäten dienen (Albers, Reiß, Bursac & Richter, 2016).

Im dynamischen Teil erfolgt die Prozessmodellierung im sogenannten Phasenmodell durch die zeitliche und inhaltliche Eintragung der Aktivitäten in ihren Ausprägungen. Aktivitäten lassen sich sequenziell, iterativ und parallel modellieren und durchführen (Albers, Heimicke et al., 2019). In diesem lassen sich Referenz-, SOLL- und IST-Prozess darstellen (Albers, Wilmsen, Duehr & Heimicke, 2019). Dabei stellt der SOLL-Prozess die vor dem spezifischen Projekt im Projektplan definierte Abfolge der Aktivitäten dar. Im IST-Prozess wird der tatsächlich im Projekt durchgeführte Verlauf dokumentiert. Im Referenzprozess werden Best-Practices, Prozessmuster und Erfahrungen aus der IST-SOLL-Konstellation (Abweichung, Eintreffen, etc.) mehrerer Projekte gesammelt, kondensiert und ein projektunspezifischer Prozess formuliert. Dieser Referenzprozess kann wiederum Projektleiter bei der Ableitung eines projektspezifischen SOLL-Prozesses unterstützen, da dieser für ähnliche Produkte oder auch für Produkte aufeinanderfolgender Generationen gültig ist. (Albers, Wilmsen et al., 2019)

2.2 Iterationsarten in der Produktentstehung

Nach Wynn und Eckert (2017) lassen sich innerhalb der Produktentwicklung drei übergreifende Kategorien von Iterationen unterscheiden. Diese Kategorien sind progressive Iterationen, korrektive Iterationen und koordinative Iterationen. Der Kategorie progressive Iterationen sind wiederum fünf Stereotypen untergeordnet (siehe Abbildung 3). Die fünf progressiven Iterationsarten sind wie folgt definiert:

1. **Exploration** beschreibt das gleichzeitige und iterative Erkunden des Problem- und Lösungsraumes, was die Basis für jeden kreativen Problemlösungsprozess darstellt.
2. **Concretisation** beschreibt einen iterativen Prozess in dem die Elemente eines Entwurfs durch eine stetige Steigerung des Detaillierungsgrades weiterentwickelt werden.
3. **Convergence** beschreibt Iterationen, die durch das Festlegen unterschiedlicher Parameter oder Details dazu beitragen ein definiertes Leistungsziel zu erreichen. Hierbei wird oftmals eine Schritt-für-Schritt Anpassung der Parameterwerte zum Erreichen einer zufriedenstellenden Lösung verfolgt.

4. **Refinement** beschreibt eine Vorgehensweise bei der Lösungen, die bereits primäre Ziele (z.B. Erfüllung einer technischen Funktion) weiterhin verbessert werden, um sekundäre Ziele (z.B. Design, Kosten) zu verbessern.
5. **Incremental Completion** beschreibt die geplante Wiederholung von Aufgaben oder Prozessen, um schrittweise ein anvisiertes Ziel zu erreichen.

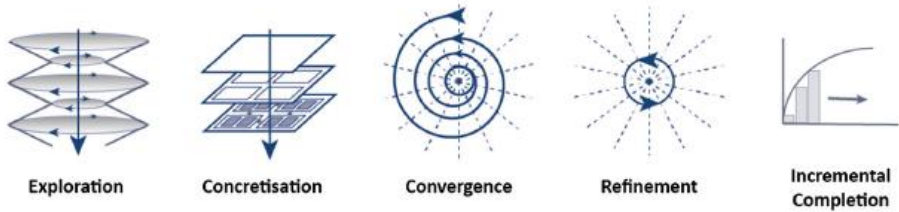


Abbildung 3: Stereotypen der Kategorie progressive Iterationen (Wynn & Eckert 2017)

Die Kategorie korrektive Iterationen umfasst die drei Iterationsarten New work, Rework und Churn (siehe Abbildung 4). Diese drei Iterationsarten lassen sich wie folgt beschreiben:

1. **New work** beschreibt die Korrektur oder Überarbeitung einer Lösung, um die vorherrschenden Anforderungen auf eine andere Weise zu erfüllen (z.B. alternative Lösungsprinzipien). Hierfür werden neue Vorgehensweisen, Prozesse oder Methoden genutzt.
2. **Rework** beschreibt eine Iteration, bei der Aufgaben auf eine ähnliche Weise erneut durchgeführt werden, da sich Zielsetzungen, Anforderungen oder Annahmen geändert haben.
3. **Churn** beschreibt Iterationen bei denen durch die Lösung von Teilproblemen neue Probleme hervorgerufen werden und es somit nicht zu einer Konvergenz der Lösung kommt.

Weiterführend werden in der Kategorie der koordinativen Iterationen von Wynn und Eckert (2017) die fünf Iterationsarten Governance, Negotiation, Parallelisation, Comparison und Concentration definiert.

Die unterschiedlichen Iterationsarten stellen einen elementaren Baustein der agilen Produktentwicklung dar. Dennoch umfassen die beschriebenen Betrachtungen in der Literatur vorwiegend Iterationsarten auf Projektebene.

Diese Sichtweise kann dem Entwicklungsteam dabei helfen die unterschiedlichen Iterationsarten zu verstehen. Jedoch wurde bisher noch nicht untersucht, wie sich diese Iterationsarten auf einer Projektplanungs- und Prozessebene veranschaulichen lassen und welche Implikationen sich hierdurch für den gezielten Einsatz der Iterationsarten ergeben. Hierdurch soll vor allem ein Beitrag zur Minderung der Planungsunsicherheit in Entwicklungsprojekten geleistet werden.

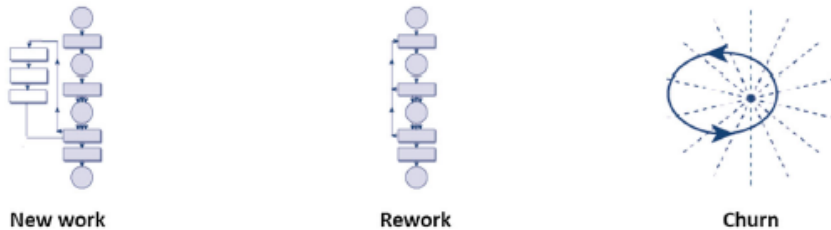


Abbildung 4: Stereotypen der Kategorie korrektive Iterationen (Wynn & Eckert 2017)

3. Beschreibung des Forschungsvorhabens

3.1 Forschungsbedarf

Agile Ansätze beinhalten Iterationen von Aktivitäten und zeichnen sich durch ihre hohe Reaktionsfähigkeit gegenüber ungeplanten Veränderungen des Kontexts aus. In der Literatur sind verschiedene Iterationsarten bekannt, die als in Projekten auftretend postuliert werden. Über Ursachen und Folgen der verschiedenen Iterationsarten ist jedoch in der Literatur wenig bekannt. In Projekten der Mechatroniksystementwicklung macht dies einen proaktiven Umgang von Entwicklerteams mit geplanten oder ungeplanten sowie gewollten oder ungewollten Iterationen schwer möglich. Auch ist nicht eindeutig, welche der Iterationsarten tatsächlich für das co-evolutionäre Fortschreiten des Reifegrades von Ziel- und Objektsystem förderlich sind. Um Entwicklerteams in ihrem Umgang mit Iterationen im Prozess der Produktentstehung zu unterstützen und potentiell auftretende Iterationen vorhersagen und identifizieren zu können, werden im vorliegenden Forschungsvorhaben Ursachen-Wirkungszusammenhänge unterschiedlicher Iterationsarten modelliert und zu einer Klassifizierungssystematik zusammengefasst. Um dies zu erreichen, werden die folgenden Forschungsfragen beantwortet:

- Welche in der Literatur beschriebenen Iterationsarten treten in frühen Phasen der Mechatroniksystementwicklung mit einem erhöhten Neuentwicklungsanteil auf?
- Wie stellen sich diese Iterationsarten bezüglich der vier Planungsebenen des Ansatzes des ASD – Agile Systems Design dar?

3.2 Forschungsmethodik

Basierend auf einer initialen Literaturrecherche konnten die bereits beschriebenen Iterationsarten aus der Literatur identifiziert und analysiert werden. Zur Beantwortung der ersten Forschungsfrage, wurde untersucht inwiefern diese Iterationsarten in Produktentwicklungsprojekten mit einem erhöhten Neuentwicklungsanteil auftreten und wie diese ausgestaltet sind. Hierfür wurden zwei Entwicklungsprojekte analysiert, die im Zuge des Live-Labs AIL – Agile Innovation Lab am KIT – Karlsruher Institut für Technologie durchgeführt wurden. Die Aufgabenstellungen dieser Entwicklungsprojekte waren „Entwicklung handgehaltener Geräte zur Unkrautregulierung auf befestigten Flächen“ und „Entwicklung von Konzepten zur ölfreien Kompression in stationären Anwendungen“. Live-Labs werden als Validierungsumgebungen für Entwicklungsprozesse, -Methoden und -Tools verstanden und ermöglichen die Kontrolle von relevanten Rahmenbedingungen bei gleichzeitiger Wahrung einer erhöhten Realitätsnähe (Albers et al. 2017).

Für die Identifikation der in den Projekten vollzogenen Iterationsarten wurden die jeweiligen IST-Prozesse der Produktentwicklungsprojekte analysiert und hinsichtlich der Definitionen und Eigenschaften der verschiedenen Iterationsarten aus der Literatur klassifiziert. Die IST-Prozesse wurden von den Entwicklungsteams mit Hilfe von physischen Scrum Boards dokumentiert. Hier wurde jeder Prozessschritt mit Hilfe eines vorgefertigten Template beschrieben (siehe Abbildung 5). Diese Prozessschritte wurden entsprechend ihrem Bearbeitungsstatus in die jeweilige Spalte des Scrum Boards (Backlog, To Do, In Progress, Review, Done) gehängt und täglich fotografiert und in eine Excel Tabelle übertragen, um den IST-Prozess der Projekte zu dokumentieren. Diese IST-Prozesse wurden anschließend mit Hilfe der Aktivitäten der Produktentstehung aus dem iPeM (Albers 2018b) strukturiert, um die unterschiedlichen Iterationsarten besser identifizieren und analysieren zu können.

Aufgabe:	Dauer:
	Start:
Beschreibung: - Ziel - Leitfrage - Erwartetes Ergebnis	Wer:
	Ende:

Abbildung 5: *Template zur Beschreibung von Prozessschritten bzw. Aufgaben während der Entwicklungsprojekte in AIL – Agile Innovation Lab*

Weiterführend wurden die jeweiligen Entwicklungssituationen hinsichtlich der Auslöser für die aufgetretenen Iterationsarten analysiert. Hier konnte beobachtet werden, dass viele Iterationen bereits zum Beginn einer Phase bzw. sogar zum Beginn des Projektes basierend auf Erfahrungen mit vergangenen Projekten eingeplant wurden.

4. Iterationsarten im Live-Lab AIL – Agile Innovation Lab

In AIL – Agile Innovation Lab gab es auf dem Projektlevel (vgl. Abbildung 1) eine übergreifende Meilensteinplanung, welche vergleichbar mit einem Agile-Stage-Gate Hybrid Modell ist. Wie in Abbildung 6 dargestellt wurden die ersten zwei Phasen, Analyse und Potentialfindung, klassisch nach dem Stage-Gate Modell mit einer spezifischen Ergebniserwartung durchgeführt. Anschließend wurden vier Sprints durchgeführt, welche je das Ziel hatten den Reifegrad der technischen Lösung zu steigern. So war das Ergebnis von Sprint 1 eine technische Lösungsidee, welche in einem ersten Grobkonzept skizziert wurde. In dieser Phase wurden erste physische Funktionsprototypen zur Evaluation von ein bis zwei Funktionen realisiert. Diese Ergebnisse wurden in Sprint 2 weiter präzisiert, um den Prototypen mit weiteren, kunden-relevanten Funktionen zu erweitern. In Sprint 3 wurde kontinuierlich der Reifegrad des Prototyps und der zugehörigen Funktionen gesteigert und es wurden ebenfalls Geschäftsmodelle entwickelt. Im letzten Sprint 4, wurde die

Reife des Prototyps so weit gesteigert, dass er als Ausgangspunkt für die Serienproduktion genutzt werden kann. Der Prototype beinhaltet einen vollumfänglichen Funktionsumfang und erfüllt somit das Zielsystem.

Basierend auf diesen Beschreibungen kann rückgeschlossen werden, dass es sich auf dem Projektlevel bei Sprint 1 und Sprint 2 um die Iterationsart *Incremental Completion* handelt. In diesen Sprints wurden nämlich unterschiedliche Funktionen (geplante Inkremente) prototypisch umgesetzt. Bei Sprint 2, Sprint 3 und Sprint 4 handelt es sich wiederum um die Iterationsart *Concretisation* auf dem Projektlevel. Dies ist dadurch begründet, dass die Zielsetzung der jeweiligen Sprints eine weiterführende Konkretisierung und Verbesserung der bestehenden Funktionen des Prototyps waren. Diese Iterationen wurden vor Beginn des Projektes durch die Projektleitung, basierend auf Erfahrungswerten festgelegt.



Abbildung 6: Projektlevel in AIL – Agile Innovation Lab: Agile-Stage-Gate Hybrid Modell

Durch eine Analyse der Aktivitäten der Produktentstehung während der Entwicklungsprojekte konnte herausgefunden werden, dass innerhalb der vier Sprints die gleichen Aktivitäten der Produktentstehung (vgl. Abbildung 2) wiederholt durchgeführt wurden. Hierzu wurden in jedem Sprint die Aktivitäten Prinzip und Gestalt modellieren, Prototyp aufbauen, sowie Validieren und Verifizieren durchgeführt. Diese Vorgehensweise lässt sich auf dem Projektlevel durch die Iterationsart *Rework* beschreiben. Das lässt sich dadurch begründen, dass sich die Ziele und Anforderungen nach jedem Sprint verändert haben und auf dem Projektlevel die gleichen Aktivitäten der Produktentstehung durchgeführt wurden. An dieser Stelle ist zu berücksichtigen, dass bei der Durchführung dieser Produktentstehungsaktivitäten auf Phasenlevel, Aktivitätslevel oder Methodendurchführungslevel andere Iterationsarten auftreten.

Durch eine Analyse des IST-Prozesses der Potentialfindungsphase in AIL (siehe Abbildung 7 und Abbildung 8) konnten unterschiedliche Iterationsarten identifiziert werden. Bei einer Betrachtung des Phasenlevels konnte ent-

sprechend dem ASD-Ansatzes (siehe Abbildung 1) eine iterative Vorgehensweise identifiziert werden. Durch die wiederholte Durchführung der Produktentstehungsaktivitäten Profile finden, sowie Nutzung analysieren und Wissen managen ist auf dem Phasenlevel ebenfalls die Iterationsart *Rework* zu beobachten.

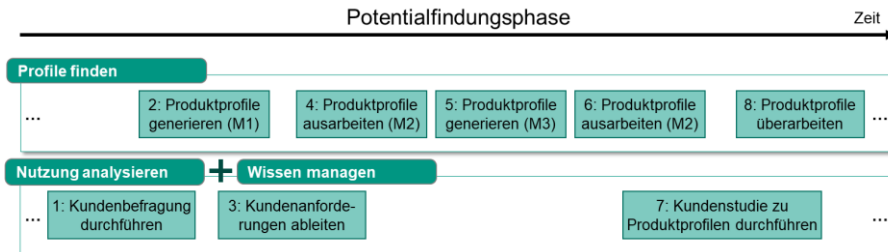


Abbildung 7: Auszug von Prozessschritten der Potentialfindungsphase in AIL – Agile Innovation Lab

Auf dem Aktivitätslevel (siehe Abbildung 1) des ASD Ansatzes konnten unterschiedliche Iterationsarten identifiziert werden. Wie in Abbildung 7 dargestellt, wurde in der Potentialfindungsphase die Iterationsart *Exploration* durchgeführt. Dies ist daran zu erkennen, dass einerseits Produktprofile generiert wurden (2 in Abbildung 7), um den Lösungsraum zu erschließen. Andererseits wurde ebenfalls der Problemraum durch die Durchführung einer Kundenbefragung (1 in Abbildung 7) konkretisiert. In der Potentialfindungsphase ist ebenfalls die Iterationsart *Concretisation* aufgetreten. In Abbildung 7 wird diese Iterationsart beispielsweise durch die Prozessschritte 5, 6 und 8 dargestellt, da diese Prozessschritte eine Konkretisierung des Objektes „Produktprofil“ forcieren. In Abbildung 8 adressiert Prozessschritt 1 die Präzisierung der technischen Lösung für Funktion A, was beispielsweise die Iterationsart *Refinement* darstellt. Dies ist darauf zurückzuführen, dass im vorherigen Sprint 1 bereits ein Prototyp für Funktion A zur Erreichung primärer Ziele erfolgreich erarbeitet wurde und das Ziel der dargestellten Präzisierung der Funktion A in Sprint 2 eine Reduzierung der Herstellungskosten adressiert. Die Iterationsart *Incremental Completion* kann ebenfalls in Abbildung 8 festgestellt werden. Durch die Prozessschritte 5-8 bzw. 9-12 in Abbildung 8 wird ersichtlich, dass das Entwicklungsteam gezielt Funktion B und Funktion C als unterschiedliche Inkremente getrennt voneinander entwickelt und prototypisch umgesetzt hat.

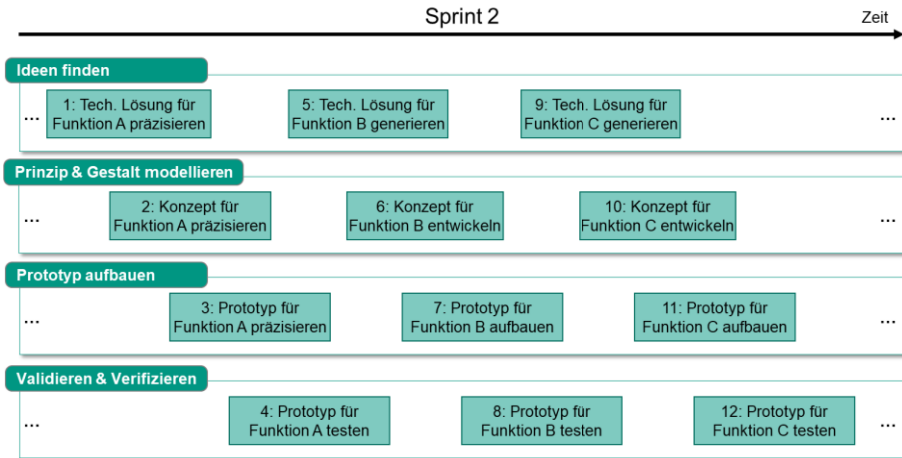


Abbildung 8: Auszug von Prozessschritten des Sprint 2 in ALL – Agile Innovation Lab

Auf dem Methodendurchführungslevel konnte die Iterationsart *Convergence* identifiziert werden. Diese wurde beispielsweise bei der Durchführung von Prozessschritt 8 in Abbildung 8 forciert. Hier testete das Entwicklungsteam den Prototyp der Funktion B und passte Leistungsparameter wie Geschwindigkeit, Geometrie oder Neigungswinkel zur Erreichung eines Leistungsoptimums an. In Abbildung 7 ist ebenfalls die Iterationsart *New work* zu erkennen, da der Prozessschritt „Produktprofile generieren“ (2 und 5 in Abbildung 7) mit unterschiedlichen Methoden (M1 und M3) durchgeführt wurden. Außerdem wurde ebenfalls die Iterationsart *Rework* auf dem Methodendurchführungslevel in der Potentialfindungsphase durchgeführt. Hierfür wurde der Prozessschritt „Produktprofile ausarbeiten“ (4 und 6 in Abbildung 7) beide male mit der gleichen Vorgehensweise und Methode (M2) durchgeführt, da sich der Input und die Anforderungen durch andere Prozessschritte verändert haben.

Basierend auf diesen Untersuchungen konnten mögliche Auslöser der Iterationsarten identifiziert werden. Eine treibende Größe für den Einsatz von geplanten Iterationen ist die Planungsunsicherheit. Diese Spezifikation der Unsicherheit kann beispielsweise durch Faktoren wie „Unsicherheit über Kundenwünschbarkeit“, „Unsicherheit der technischen Machbarkeit“ oder „Unsicherheit über wirtschaftliche Machbarkeit“ beeinflusst werden. Diese

Unsicherheiten werden vor allem in frühen Phasen der PGE – Produktgenerationsentwicklung versucht durch ein gezieltes iteratives Vorgehen zu reduzieren. In Kontrast dazu stehen die kurzfristigen bzw. ungeplanten Iterationen, welche in den meisten Fällen durch eine nicht vorhergesehene Abweichung des IST- und SOLL-Zustandes ausgelöst werden. Faktoren die hierbei eine Rolle spielen sind beispielsweise die Qualität oder der Reifegrad der Ergebnisse. Zusätzlich dazu kann jegliche, unvorhergesehene Veränderung eines Elementes im ZHO Triple zu einer ungeplanten Iteration führen. Hierzu zählt beispielsweise ein neues Ziel, das durch das Management während des Projektes eingefordert wird oder fehlende Kompetenzen im Handlungssystem. Demnach gibt es eine Vielzahl an Faktoren, die als Auslöser für eine Iteration fungieren können. Dennoch bedarf es weiteren Untersuchungen, um valide Aussagen über den direkten Einfluss der einzelnen Faktoren auf die unterschiedlichen Iterationsarten zu identifizieren.

5. Diskussion und Ausblick

Basierend auf der Analyse der verschiedenen Iterationsarten in den beiden Entwicklungsprojekten in AIL – Agile Innovation Lab konnte festgestellt werden, dass auf den vier Ebenen des ASD – Agile Systems Design Ansatzes (siehe Abbildung 1) unterschiedliche Iterationsarten auftreten. Es ist zu erwarten, dass je nach Projektart und Kontext bestimmte Iterationsarten vermehrt oder vermindert auftreten. Aus der durchgeführten Untersuchung konnte beobachtet werden, dass nicht jede Iterationsart auf jeder Ebene des ASD Ansatzes auftritt. Es bedarf daher weiterer Forschung, um eine mögliche Korrelation zwischen den Iterationsarten und den Ebenen des ASD Ansatzes zu verifizieren. Aufgrund der Rahmenbedingungen der untersuchten Entwicklungsprojekte konnten lediglich die zwei Kategorien progressive Iterationen und korrektive Iterationen untersucht werden. Zur Untersuchung der dritten Kategorie koordinative Iterationen wird es erforderlich sein, diverse Entwicklungsprojekte in unterschiedlichen Unternehmenskontexten zu untersuchen. Der vorliegende Beitrag basiert auf der Annahme, dass durch den gezielten und vorausschauenden Einsatz der passenden Iterationsarten die Planungsunsicherheit reduziert werden kann, um so eine möglichst konstante Planungsstabilität zu erzielen. Da die Art des Zusammenhangs zwischen den zwei Elementen Planungsunsicherheit und Planungsstabilität

noch nicht ausreichend bekannt ist, bedarf es hierzu ebenfalls weiterer Forschung. Somit soll ein weiterer Beitrag zur Verbesserung der Projektplanung in einer frühen Phase der Mechatroniksystementwicklung geleistet werden.

Literaturverzeichnis

- Albers, A., Bursac, N., Heimicke, J., Walter, B. & Reiß, N. 2017: 20 years of co-creation using case based learning. An integrated approach for teaching innovation and research in Product Generation Engineering. In M. E. Auer, D. Guralnick & J. Uhomibhi (Hrsg.), Proceedings of the 20th ICL Conference, 636-647.
- Albers, A., Heimicke, J., Spadinger, M., Reiss, N., Breitschuh, J., Richter, T. et al. 2019: Eine Systematik zur situationsadäquaten Mechatroniksystementwicklung durch ASD – Agile Systems Design. In KIT Scientific Working Papers (Bd. 113). Karlsruhe: KIT.
- Albers, A., Heimicke, J., Walter, B., Basedow, G. N., Reiß, N., Heitger, N. et al. 2018: Product Profiles. Modelling customer benefits as a foundation to bring inventions to innovations. *Procedia CIRP*, 70 (1), 253-258.
- Albers, A., Lohmeyer, Q. & Ebel, B. 2011: DIMENSIONS OF OBJECTIVES IN INTERDISCIPLINARY PRODUCT DEVELOPMENT PROJECTS. INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED11, Copenhagen, Denmark.
- Albers, A., Reiß, N., Bursac, N. & Breitschuh, J. 2016a: 15 Years of SPALTEN Problem Solving Methodology in Product Development. In C. Boks (Ed.), Proceedings of NordDesign 2016. August 10-12, 2016, Trondheim, Norway (pp. 411-420). Bristol, United Kingdom: The Design Society.
- Albers, A., Reiß, N., Bursac, N. & Richter, T. 2016b: iPeM – Integrated Product Engineering Model in Context of Product Generation Engineering. *Procedia CIRP*, 50, 100-105.
- Albers, A., Wilmsen, M., Duehr, K. & Heimicke, J. 2019: The first steps towards innovation: A reference process model for developing product profiles. International Conference on Engineering Design, ICED19.
- Boehm, B. & Turner, R. 2003: Using risk to balance agile and plan-driven methods. *Computer*, 36 (6), 57-66.
- Breitschuh, J., Albers, A., Seyb, P., Hohler, S., Benz, J., Reiß, N. et al. 2018: Scaling agile practices on different time scopes for complex problem-solving. Proceedings of NordDesign 2018.
- Cooper, R. G. 1990: Stage-gate systems: A new tool for managing new products. *Business Horizons*, 33 (3), 44-54.
- Cooper, R. G. 2016: Agile–Stage-Gate Hybrids. *Research-Technology Management*, 59 (1), 21-29. Zugriff am 07.12.2017. Verfügbar unter <http://www.five-is.com/wp-content/uploads/Agile-Stage-Gate-Cooper-in-RTM-20162.pdf>
- Fowler, M. & Highsmith, J. 2001: The Agile Manifesto. *Softw. Dev.* 9, 28-35.

- Heimicke, J., Reiss, N., Albers, A., Walter, B., Breitschuh, J., Knoche, S. et al. 2018: Agile Innovative Impulses in the Product Generation Engineering. Creativity by Intentional Forgetting. Proceedings of 5th International Conference on Design Creativity (ICDC 2018) (8), 183-190.
- Plattner, H., Meinel, C. & Leifer, L. 2011: Design Thinking. Understand – Improve – Apply. Berlin, Heidelberg: Springer Berlin Heidelberg.
- Ropohl, G. 1979: Eine Systemtheorie der Technik: zur Grundlegung der allgemeinen Technologie.
- Schmidt, T. S., Paetzold, K. & Weiss, S. 2018: Agile Entwicklung physischer Produkte. VDI-Statusreport 09/2018.
- Schwaber, K. & Sutherland, J. 2017: The Scrum Guide.
- Smith, R. P. & Morrow, J. A. 1999: Product development process modeling. *Design Studies*, 20 (3), 237-261.
- Spreiter, L., Böhmer, A. I. & Lindemann, U. 2018: EVALUATION OF TAF AGILE FRAMEWORK BASED ON THE DEVELOPMENT OF AN INNOVATIVE EMERGENCY WEARABLE FOR SENIORS. In DS92: Proceedings of the DESIGN 2018 15th International Design Conference, 1345-1356.
- Thomke, S. & Reinertsen, D. 1998: Agile Product Development: Managing Development Flexibility in Uncertain Environments. *California Management Review*, 41 (1), 8-30.
- Wynn, D. C. & Clarkson, P. J. 2018: Process models in design and development. *Research in Engineering Design*, 29 (2), 161-202.
- Wynn, D. C., & Eckert, C. M. 2017: Perspectives on iteration in design and development. *Research in Engineering Design*, 28(2), 153-184.

Kontakt

M.Sc. Miriam Wilmsen
M.Sc. Markus Spadinger
Univ.-Prof. Dr.-Ing. Dr. h. c. Albert Albers
B.Sc. Cong Minh Nguyen
M.Sc. Jonas Heimicke
KIT – Karlsruher Institut für Technologie
IPEK – Institut für Produktentwicklung
Kaiserstr. 10
76131 Karlsruhe
<http://www.ipek.kit.edu/index.php>

