

Erfolgreiche Softwareeinführung

Praxiserprobtes Vorgehen zur
Prozessdigitalisierung

» Hier geht's
direkt
zum Buch

DAS VORWORT

1

Einleitung

1.1 Disclaimer

Dieses Buch schreibe ich als Mann und es liegt mir nicht ferner, als für ein bestimmtes Geschlecht oder persönliche Identität zu schreiben. Daher versuche ich auch schon seit längerem, die Geschlechtsformulierung in meinen Konzepten oder Dokumentationen zu vermeiden. Als Beispiel schreibe ich anstelle von ...Man sollte ...“ wo es geht etwas wie „...Es sollte ...“. Wenn dies nicht möglich ist – es gibt Satzaufbauten, in denen es die deutsche Sprache (mir) nicht erlaubt – werde ich dies in beiden Geschlechtern nicht mit der Erweiterung „... |innen“ beschreiben. Und selbstverständlich sollen auch Diverse sich mit dieser Beschreibung angesprochen fühlen.

1.2 Warum dieses Buch

Hallo, mein Name ist Stefan Scheer, ich habe meinen jetzt erwachsenen Kindern zwei wichtige Worte oft wiederholt, die sie in ihrem Leben berücksichtigen sollten. Das sind die Worte **Machen** und **Durchhalten**.

Viele Menschen reden über etwas, was sie anders machen würden oder gerne verändern oder haben wollen, nur starten sie nie oder viel zu selten damit. Die Gründe sind vielfältig. Aber wer nicht mit etwas startet, weiss auch nicht, ob es gelingen kann. Daher ist der Start des **Machen** das Wichtigste im Leben, wenn mich etwas – legal vorausgesetzt – bewegt.

Das zweite Wort **Durchhalten** ist das wichtigste gleich danach. Ich kann viele Dinge starten. Es wird erst einmal mit Enthusiasmus und Euphorie beginnen, dann kommt aber die wichtigste Zeit. Die der Zweifel, Rückschläge, Hindernisse, fehlenden Motiva-

tion, andere wichtigere Themen und vieles mehr. Daher ist ein Weitermachen, auch wenn es hart ist, der Schlüssel zum Erfolg für das, was ich angefangen habe.

Nach über 20 Jahren in der Softwareeinführung auf der Seite des Implementierungspartners beschäftigt es mich schon länger, meine Erfahrungen für mich selbst und als Idee für andere zusammenzufassen.

Mein Ziel war es, aus der Praxis der Softwareeinführung eine Vorgehensweise zu erarbeiten, die für mich und andere funktioniert. Es gibt viele Vorgehensweisen in der Softwareeinführung. Ich durfte schon bei mehreren Softwarehäusern arbeiten und deren Vorgehensweisen in vielen Projekten anwenden. Auch durfte ich die Vorgehensweisen bei Unternehmen schon mitgestalten. Aus den Erfahrungen habe ich meine eigene Vorgehensweise mit Schritten, die sich als erfolgreich herausgestellt haben, entwickelt und ich wende sie auch aktuell an.

Somit nehme ich meine Ratschläge nun für mich selbst wahr und ich habe mit dem Machen gestartet. Ich habe meine Arbeitszeit als Senior Solution Architect und Managing Consultant reduziert und angefangen, jeden Freitag daran zu arbeiten. Ob ich auch durchhalten werde, darüber war ich mir zu Beginn nicht sicher. Über so eine lange Zeit jede Woche dran zu bleiben und von einigen belächelt und nach dem Sinn gefragt zu werden, hat viel Eigenmotivation gebraucht. Hier kann ich jedem und jeder nur empfehlen – macht es und lasst euch nicht entmutigen, wenn ihr ein Thema habt, an dem ihr mit Leidenschaft arbeiten wollt. Wenn ihr ein Ziel habt, geht es an und bleibt dran. Es lohnt sich.

Spannenderweise ist mir aufgefallen, wie wir zu Beginn meiner Zeit in der Softwareeinführung bei Firmen ganz intuitiv oder naiv an die Softwareeinführung herangegangen sind. Dann über die Jahre kamen viele verschiedene Projektmethoden mit oft recht harten Vorgaben und vielen Details und noch mehr Fremdworten. Jetzt und mit der konkreten Auseinandersetzung mit der Thematik würde ich sagen, es ist eine Frage der Umgebung und Voraussetzungen und Menschen, die gemeinsam an einem Thema zusammenarbeiten. Daher kann nicht immer gleich vorgegangen werden.

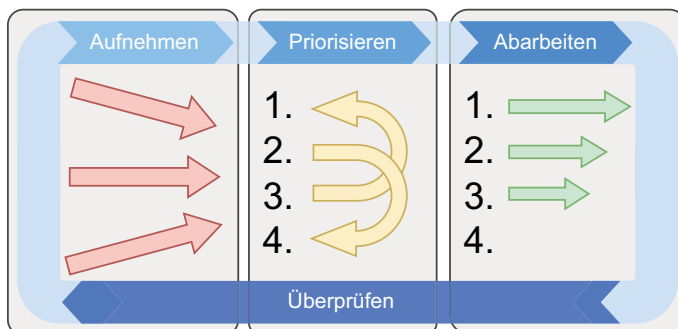


Bild 1.1 Ablauf Aufnehmen Priorisieren Abarbeiten Überprüfen

Die Zusammenfassung meiner Erfahrungen in verschiedensten Positionen in der Begleitung von Softwareeinführungen über die Jahre habe ich hier niedergeschrieben. Ich kann sie in einem Satz beschreiben:

Konzentriertes **Aufnehmen**, ständiges **Priorisieren**, fokussiertes **Abarbeiten** und kontinuierliches **Überprüfen**, wie in Bild 1.1 angezeigt.

Eigentlich ist mit diesem Satz alles gesagt und das Buch ist schon fertig. Mit dieser Herangehensweise kann ich jeden Tag im Detail, aber auch grosse und komplexe Aufgabenstellungen und Projekte angehen.

Eine kurze Erklärung:

- **Aufnehmen:** Konzentriertes Zuhören ist der erste Schlüssel für das Aufnehmen. Erst wenn ich verstanden habe, was überhaupt die Anforderung ist, kann ich mich gedanklich damit auseinandersetzen. Dazu muss ich konzentriert und bei dem Thema sein. Als Zweites muss dies dann in einer für alle verständlichen Weise niedergeschrieben werden. Es gibt viele sehr genaue und komplexe Vorgehensweisen, etwas aufzuschreiben oder zu visualisieren. Hier ist meiner Meinung nach weniger mehr. Es sind nicht immer Spezialisten in der Runde der Teilnehmenden, die eine Visualisierungsdarstellung bis ins Detail verstehen, daher sollte es einfach dargestellt und klar und mit einfachen Worten beschrieben sein. Und abschliessend muss ein Konsens/eine Einigkeit des Visualisierten und Aufgeschriebenen bei allen Beteiligten bestehen. Es gibt nichts Schlimmeres als eine Anforderung, die falsch verstanden, schlecht dokumentiert und nicht gewollt/gewünscht ist.
- **Priorisieren:** Es geht nicht immer alles zur gleichen Zeit. Eine Schwäche von vielen ist Ungeduld. In der heutigen Zeit fühlt es sich so an, dass immer alles und sofort umgesetzt und verfügbar ist, und so ist auch die Erwartungshaltung an Themen in einem Projekt. Ist etwas besprochen, sollte es schon fast fertig sein. Aber reell braucht es bei vielen Themen „Zeit“ und es hat oft zu wenig von genau dieser „Zeit“. Unabhängig ob kalendarisch oder der Teilnehmenden. Daher muss es immer mit der Frage „Was ist heute das Wichtigste“ eine Priorisierung geben. Und dass vor jeder Tätigkeit immer wieder aufs Neue.
- **Abarbeiten:** Wenn etwas realisiert werden soll, muss es genau beschrieben sein. Eine nicht klare Aufgabe führt zu Rückfragen, Zeitverlust oder gar zu falscher Umsetzung. Beim Abarbeiten kommt es schliesslich zu der Abklärung, Dokumentation, Umsetzung etc. selbst. Jemand kann sich einen ganzen Tag mit einem Thema beschäftigen und keinen Schritt vorankommen, es kann aber auch kurz sehr fokussiert ein Thema bearbeitet werden und es ist gelöst oder realisiert. Daher ist die Konzentration auf das Abarbeiten einer Aufgabe immens wichtig.
- **Überprüfen:** Jederzeit in allen Phasen eines Projekts muss hinterfragt werden können, ob das aktuelle Bild noch passt. Das kann eine Fragestellung, eine kleine Entscheidung sein, aber auch aufgrund von dem aktuellen neuen Wissensstand

heissen, dass kleine, aber auch grosse Entscheidungen infrage gestellt werden können und sollen. Ein Festhalten an einer Entscheidung, die mit „altem Wissen“ getroffen wurde, ist manchmal zu einem späteren Zeitpunkt nicht zielführend. Ein Projekt einer Softwareeinführung ist wie eine Reise. Und wer schon mal gereist ist, weiss, dass es immer etwas anders kommt als geplant. Es muss regelmässig proaktiv überprüft werden, ob die getroffenen Entscheidungen dem aktuellen Wissensstand noch standhalten.

Die Schritte **Aufnehmen**, **Priorisieren**, **Abarbeiten** und **Überprüfen** sollten in der täglichen Projektarbeit immer wieder berücksichtigt werden und wenn schwierige Herausforderungen anstehen, sollte nicht in Aktionismus verfallen werden, sondern genau diese Schritte angewendet werden. Es hilft nicht immer, aber oft.

Denn eins habe ich in den vielen Jahren in Projekten gelernt: **Nichts ist so beständig wie der Wandel**¹ und jedes Projekt muss neu in dem Vorgehen betrachtet werden. Einige grundlegenden Themen sind jedoch immer gleich. In einem Projekt, in dem ein Prozess digitalisiert werden soll, ist es zuerst wichtig, ein Projekt im Setup zu starten. Dann das IST zu verstehen und das Soll zu formulieren. Mit diesem Wissen kann eine Übersetzung in Systeme erfolgen, bevor mit der Umsetzung von grob nach fein begonnen werden kann. Ist alles umgesetzt, sollte integrativ getestet werden und abschliessend kann der Prozess mit digitalisierter Unterstützung angewendet werden. Alle diese Schritte sind Teil der Vorgehensweise in diesem Buch.

1.3 An wen richtet sich dieses Buch

Wie im vorigen Kapitel beschrieben, war es die Idee, meine Erfahrungen aus der Softwareeinführung zusammenzufassen und als Ideengeber zur Verfügung zu stellen.

Der Titel **Erfolgreiche Softwareeinführung** sollte der Antrieb eines jeden sein, der oder die in einem Projekt mitarbeitet. Und der Untertitel **Praxiserprobtes Vorgehen zur Prozessdigitalisierung** beschreibt gut, was dieses Buch ausmacht. Es soll keine theoretische Beschreibung einer Vorgehensweise sein, sondern ein Vorgehen aufzeigen, das sich aus vielen Projekten ergeben hat, in denen Prozesse mit Systembegleitung umgesetzt wurden. Jedes Projekt ist anders. Die Mischung der Anforderungen, des Zielbilds, der Teilnehmenden und der organisatorischen, fachlichen und technischen Umgebung gibt teilweise das Vorgehen vor.

Daher gibt es nur eine Zielgruppe, für die das Buch geschrieben ist. **Alle, die mit der Digitalisierung von Prozessen und einem Softwareeinführungsprojekt in irgendeiner Form zu tun haben.**

¹ Zitat von Heraklit von Ephesus, 535 – 475 v. Chr.

Wenn jemand ein Projekt plant, in der Verantwortung ist oder an einem Projekt mitarbeitet, können Themen dieses Buchs als Inspiration zur Vorgehensweise von geplanten oder laufenden Projekten dienen.

Die Vorgehensweise und deren Teilnehmende und Tools in diesem Buch sind eine Zusammenfassung von vielen Projekten über viele Jahre, in denen es gut, aber auch mal etwas harzig gelaufen ist. Jedes Holpern in einem Projekt hat Gründe und dieses Buch geht auf die wichtigsten Themen ein und wie sie aus der Erfahrung idealerweise gehandhabt werden können.

Um die Frage, an wen sich das Buch richtet, zu beantworten, sind hier einige Rollen in einem Projekt konkret ausgeführt:

- **Initiatoren eines Projekts:** Wer ein Projekt plant oder schon in der Planung ist, einen Prozess zu digitalisieren, also mit einer Software zu unterstützen, findet hier die ersten Schritte und einen Überblick der Vorgehensweise sowie der benötigten Teilnehmenden und Tools.
- **Projektverantwortung:** Wer in der Projektverantwortung ist oder zukünftig sein will oder soll, findet hier eine Vorgehensweise zur Softwareeinführung im Detail. Alle Schritte von dem Umgang der Prozesse, der Vorgehensweise in allen ihren Phasen im Detail und wie sie administriert und durchgeführt werden. Über die Rollen, Verantwortungen und Aufgaben aller Projektbeteiligten, bis hin zu den notwendigen Tools und ihrer Nutzung.
- **Fachverantwortung:** Neben der Vorgehensweise mit allen Phasen und den Anforderungen an den Fachbereich, dem Umgang der Prozesse in ihrer Analyse, Dokumentation, Umsetzungsbegleitung bis hin zum erfolgreichen Testen und wie die Prozesse in die Anwendung kommen.
- **Beratung:** Die Beratung teilt sich in Prozessberatung, Umsetzungsberatung und technische Beratung auf. Die Prozessberatung ist hier nur angedeutet, die Bereiche Umsetzungsberatung und technische Beratung sind im Detail enthalten und geben allen in der Beratung ein gutes Bild, wie ein Prozess aufgenommen, optimiert, in eine Softwarelösung übersetzt und technisch umgesetzt werden kann. Die Prozessentwicklung in der Unternehmensberatung ist hier nicht weiter detailliert.
- **Weitere Teilnehmende:** Egal in welcher Rolle jemand an einem Projekt teilnimmt oder involviert ist, er oder sie bekommt mit diesem Buch eine Übersicht der Prozessaufnahme und Dokumentation, der Vorgehensweise in einem Softwareeinführungsprojekt, der Rollen und ihren Mitwirkungspflichten und der benötigten Tools mit Erklärung der Nutzung.

1.4 Anglizismen und Fachbegriffe

Anglizismen, also englische Worte in der deutschen Sprache und/oder Fachbegriffe, werden oft in Fachbereichen oder im Kontext eines bestimmten Themas verwendet. Als nicht studierter Mensch oder Fachfremder sind sie oft „suspekt“, also fragwürdig, und es stellt sich die Frage, warum sie nicht mit einfachen gängigen und deutschen Worten beschrieben werden können.

Ich hatte recht früh in meiner Laufbahn ein Aha-Erlebnis in einem Arbeitstermin (Neudeutsch „Workshop“) in Österreich (einem deutschsprachigen Land). In dem hatten wir sehr viele englische Worte bei unserer Präsentation verwendet. Mitten in dem Termin hat ein Teilnehmer, der nicht in dem Fachthema war wie wir, die präsentiert haben, genau diese eine Frage gestellt: Warum können Sie das alles nicht so erklären und beschreiben, dass wir es auch verstehen? – eine valide (gültige) Frage.

Seit langem versuche ich in Terminen, die nicht mit reinen Fachspezialisten sind, eine einfache Wortwahl zu treffen und wo nötig die fachspezifischen Worte idealerweise auf ein allgemeines Verständnis zu vereinfachen, oder wenn es nicht möglich oder explizit nicht gewollt ist, diese zu auch nicht zu verwenden.

Zudem kommt es bei langjährig gewachsenen Prozessen und Vorgehen bei Unternehmen zu einer eigenen Wortfindung. Diese spezielle Wortwahl sollte auf jeden Fall hinterfragt werden. Es kann hier schnell zu Missverständnissen kommen, wenn Menschen aufeinandertreffen, die nicht die gleiche Wortwahl bei Themen haben oder sich nicht in der Fach-Blase bewegen. Beispielsweise war ein „Kontakt“ in dem Altsystem eines Kunden eine „Aktivität“ wie ein Telefonanruf und im neu einzuführenden System war ein „Kontakt“ ein „Ansprechpartner“, also eine natürliche Person. So musste das Wort „Kontakt“ bei jeder Erwähnung kurz erklärt werden, um Missverständnisse zu vermeiden.

Komplizierte, dem allgemeinen Sprachgebrauch fremde oder englische Worte verwirren Teilnehmende mehr als sie vermeintlich „beeindrucken“. Dass sollte immer im Hinterkopf sein, wenn solche Worte verwendet werden.

Aber was ist mit einem System, das Englisch als Basis-Sprache hat, während die Projektsprache Deutsch ist? Hier ist es ganz einfach – reden und beschreiben in Deutsch, aber jede Eigenschaft des Systems muss vom verantwortlichen Bereich in deren Verständnis auf Englisch und Deutsch vorgegeben werden.

Es sollte auch immer ein klares und einheitliches Verständnis über die Festlegung der Benennung von Bereichen in einem System festgehalten werden. Es gibt nichts Verwirrenderes, als dass in einer Dokumentation ein Bereich unterschiedlich benannt war. Hier sollten bei Änderungen von Bezeichnungen alle Dokumentationen akribisch nachgearbeitet werden. Das kann sehr aufwendig werden, verhindert aber Verwirrungen. Und wenn es grössere Änderungen in einer Bezeichnung gibt, sollte sie an einem Ort, einem Glossar dokumentiert werden.

In diesem Buch werden Anglizismen und Fachbegriffe verwendet. Und ab hier in dem Buch sind solche Worte wie Anglizismen (eingedeutschte englische Worte) und Fachbegriffe in Klammern kurz beschrieben.

1.5 Methodik, Methode und Methodologie

Um diese Worte einleitend noch zu erklären, wobei es nur der Vollständigkeit halber erwähnt sein soll. Zum einen ist hier die Erfahrung und erarbeitete Vorgehensweise in Softwareprojekten beschrieben und zum anderen sind Schritte, dies durchzuführen, wen es dazu braucht und welche Tools dazu idealerweise genutzt werden aufgeführt.

Eine Methodik stellt die Gesamtheit von Vorgehensweisen dar, wobei eine Methode eine spezifische Vorgehensweise ist. Eine Methodologie ist die Theorie oder Lehre der Methodik und Methoden.

Somit ist dieses Buch als Methodologie (der Projektvorgehensweise) von einer Methodik (dem Vorgehen) mit Methoden (den konkreten Schritten und ihren umgebenden Themen) zu verstehen.

Damit schliesst der Ausflug in Wortgefechte und es geht wieder um den konkreten Inhalt.

1.6 Aufteilung dieses Buchs

Dieses Buch soll die Lehre einer Projektmethode sein, in der es um Prozesse geht, die bearbeitet und mit einer Software unterstützt werden sollen. Aufgeteilt ist das Buch in drei gleichberechtigte Bereiche, die sich um den Prozess platzieren (siehe Bild 1.2).

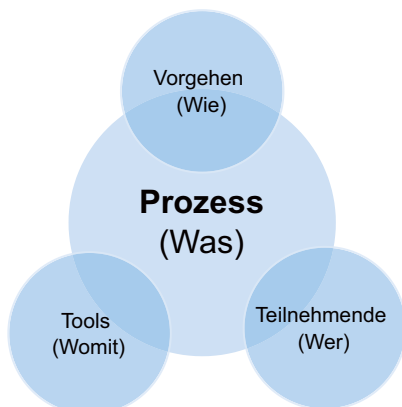


Bild 1.2
Aufteilung Buch

- Kapitel 2 Der Prozess: Herzstück ist der Prozess. Hier wird das **Was** beschrieben. Nicht, wie bei vielen Projektmethodologien die zu implementierende Software, sondern der Ablauf von Schritten, die mit oder auch ohne eine Software unterstützt werden sollen.
- Kapitel 3 Das Vorgehen: Hier wird das **Wie** beschrieben, in welchen Schritten idealerweise in einem Projekt vorgegangen werden sollte. Es kann allgemein definiert werden. Es wird unterschieden in eine einfache und komplexere Vorgehensweise für kleine Anforderungen bis hin zu einer grossen Veränderung mit allem, was prozessual dazu gehört. Zudem ist hier aufgeführt, was für die Durchführung von einem Projekt idealerweise benötigt wird.
- Kapitel 4 Die Teilnehmenden: Hier wird das **Wer** beschrieben. Wenn in einer Vorgehensweise eine Rolle mit klar definierten Aufgaben versehen wird, wird es schwierig, die richtigen Teilnehmenden zu finden. Projektgeschäfte sind, solange sie nicht von künstlicher Intelligenz komplett abgelöst werden, mit Menschen besetzt, die alle ihre eigenen Erfahrungen und Meinungen mitbringen. Daher sind der Mensch und die Zusammenarbeit und dann erst die Aufteilung der Tätigkeiten in einem Projektkontext wichtig und sie kommen gleich nach der Vorgehensweise.
- Kapitel 5 Die Tools: Hier wird das **Womit** beschrieben. Die Vorgehensweise und die kompetentesten Teilnehmenden sind nur so gut, wie die Nachhaltigkeit und das gemeinsame Verständnis. Eine Diskussion ist nur so lange allen klar, bis die nächste Aufgabe ansteht. Daher ist es unerlässlich, das gemeinsam Erarbeitete aufzuschreiben und wo möglich zu visualisieren. Die Tools fangen aber schon vorher an. Selbst bei der Planung und den Aufgaben oder Tätigkeiten ist es wichtig, eine Dokumentation zu erstellen. Hier ist es der beste Weg, im Vorfeld eine klare Struktur für das Projekt zu erstellen und in Gesprächen die Dokumentation einzubinden und nicht jeder für sich seine Notizen oder gar nur Gedanken zu machen. Eine gemeinsame Nutzung von Tools und der direkten Dokumentation von Themen bringt Transparenz und Klarheit in jedes Projekt.
- Kapitel 6 Fallbeispiele: geht auf die Vorgehensweise mit verschiedenen Fallbeispielen ein.

1.6.1 Der Prozess

Der Prozess steht im Mittelpunkt einer Einführung einer Software. Die einzuführende Software ist nur die Unterstützung des Prozesses. Daher müssen, bevor eine technische Lösung angegangen wird, zuerst die Anforderungen formuliert sein.

Um einen Prozess zu formulieren, müssen die Anforderungen an den Prozess allen klar sein und die Summe der Anforderungen ergeben den Prozess. Die Anforderungen und wie damit umgegangen werden sollte, ist später genauer aufgeführt.

Ein Prozess ist ein Arbeitsablauf für einen bestimmten Bereich. Dieser Ablauf kann mehrere Ausprägungen, also Anwendungsfälle, haben. Daher ist es zuerst wichtig zu verstehen, um was es genau geht.

Jeder Prozess hat einen Start, mindestens einen Zwischenschritt und ein Ende. Mit dieser Vorlage kann man jede Prozessbeschreibung beginnen. Prozesse haben natürlich mehr Schritte und komplexere Prozesse können Abzweigungen haben. Daher ist es wichtig, neben einem Prozess auch immer den Anwendungsfall oder die Anwendungsfälle zu verstehen. Ein Anwendungsfall ist eine Auflistung von Schritten, die in dem Prozess durchgeführt werden. Der Unterschied zu einem Prozess ist, dass der Prozess allgemein den Schritt beschreibt und der Anwendungsfall die Durchführung des Prozesses mit konkretisiert (siehe Bild 1.3).

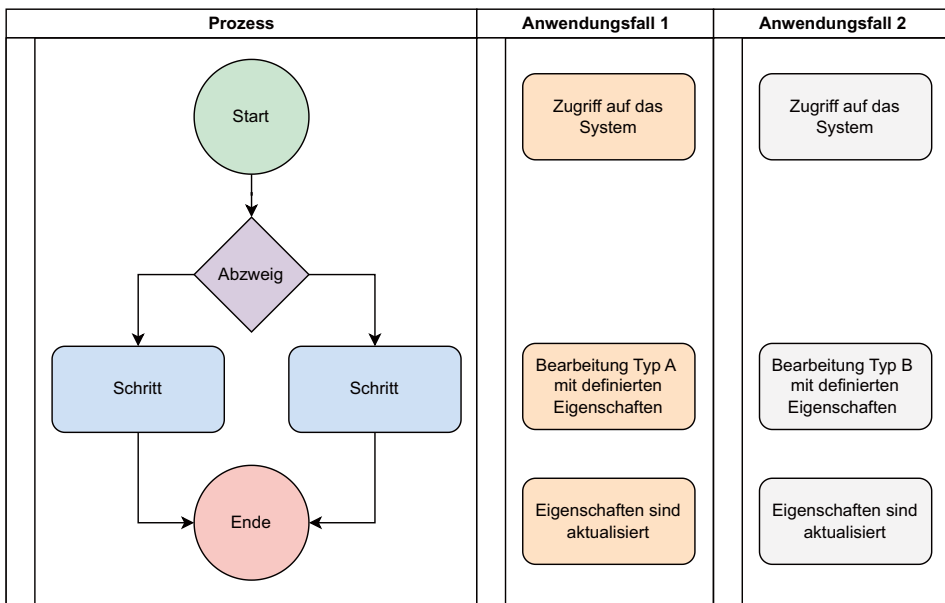


Bild 1.3 Unterscheidung Prozess zu Anwendungsfall

1.6.2 Das Vorgehen

Viele Projektmethodologien zur Softwareeinführung fokussieren sich auf die Einführung einer Software. Viel wichtiger ist aber bei einer Softwareeinführung der Prozess, in dem eine Software ein Teil des Ganzen ist. Somit geht es in dieser Vorgehensweise zum einen inhaltlich um den Prozess, welcher mit einer Software unterstützt werden kann oder soll. Und zum anderen organisatorisch, da es um Menschen geht, die miteinander arbeiten und ein Ziel verfolgen.

Bei dieser Vorgehensweise soll es nicht um im Markt gängige Themen wie Kanban, Wasserfall, Agil oder Ähnliches gehen. Vielmehr können Teile als Methoden in dieser Vorgehensweise genutzt werden.

Oft wird eine Softwareeinführung geplant und der Fachbereich/spätere Benutzerkreis ist nicht stark eingebunden. Es kann zur Folge haben, dass die Unterstützung des Prozesses mit Software nicht im Vordergrund steht. Dies kann aus Erfahrung von vielen Projekten, aus Gründen einer Harmonisierung von Software oder weil die alte Software aus dem Support läuft, kommen. Auch Umstrukturierungen im Unternehmen etc. können der Grund für eine Softwareeinführung sein.

Es wird somit eine Software erstellt, die im schlimmsten Fall im Ablauf und der flüssigen Arbeitsweise nicht hilfreich ist und nur bedingt von den Anwendern angenommen wird. Daher ist der Prozess das ausschlaggebende Element. Ein Prozess kann auch sein, dass etwas komplett ohne Software bearbeitet wird und nur der Start und das Ende oder ein kleiner Schritt in der Mitte in einer Software realisiert wird.

Daher gilt folgende Reihenfolge, welche Schritte aufeinanderfolgen (siehe Bild 1.4):



Bild 1.4 Anforderung definiert Prozess definiert Funktion definiert Lösung

Tabelle 1.1 Definition der Schritte

Schritt	Beschreibung
Anforderung aufnehmen	Eine oder mehrere Anforderungen, die für einen Prozess notwendig sind
Prozess formulieren	Ein Ablauf von Schritten mit Verantwortlichkeiten, die fachlich notwendig und genau beschrieben sind
Funktion identifizieren	Eine Funktionalität, die einen fachlichen Prozessschritt beschreibt
Lösung erarbeiten	Die Realisierung einer Funktionalität in einem System

Solange die Anforderungen in einem Prozess mit seinen Schritten und Verantwortlichkeiten nicht klar sind, braucht es keine funktionale Anforderung. Und solange eine funktionale Anforderung nicht klar ist, braucht es keine technische Lösung in einer Software.

Beispiele:

- **Funktion oder Lösung vor Anforderung:** Es kann keine Funktionalität oder Lösung erarbeitet werden, wenn die Anforderungen nicht klar formuliert sind.
- **Funktion vor Prozess:** Es kann keine Berechnungsfunktion definiert werden, wenn nicht klar ist, wann und mit welchen Voraussetzungen diese benötigt werden.
- **Lösung vor Funktion:** In einer Software für den Verkauf kann keine Angebotsfunktion umgesetzt werden, wenn sie durch den Innendienst im ERP-System erstellt werden muss.
- **Lösung vor Prozess:** Es kann keine Standard-Software für einen Bereich festgelegt werden, wenn der Ablauf noch nicht klar ist. Der Ablauf kann so stark von der Standard-Software abweichen, dass es ein grösserer Aufwand ist, die Standard-Software anzupassen, als eine andere zu verwenden oder eine Individualentwicklung vorzunehmen.

Zusammenfassung: Zuerst müssen die Anforderungen formuliert und anschließend in einem Ablauf flüssig und systemunabhängig aufgenommen werden. Im nächsten Schritt kann eine Ebene tiefer die funktionale Anforderung formuliert werden. Mit dieser Information kann entschieden werden, wo und wie die Anforderung in einer Lösung umgesetzt werden kann.

Hier gibt es folgende Lösungsmöglichkeiten:

- **Nutzung ohne System** gibt es immer noch. Beispielsweise können Menschen noch immer miteinander reden. Es muss nicht alles systemisch abgebildet werden – auch das ist ein möglicher Prozess(-Schritt).
- **Nutzung Standard-Softwarelösung**, wenn die Anforderungen mit überschaubarem Aufwand angewendet und integriert werden können
- **Individualentwicklung**, wenn es keine Standard-Software für die Anforderungen gibt
- **Der Prozess muss nochmal überdacht werden**, wenn der Aufwand der Umsetzung oder Integration zu gross ist

Mit den Voraussetzungen ergeben sich folgende Schritte in dieser Vorgehensweise (siehe Bild 1.5)

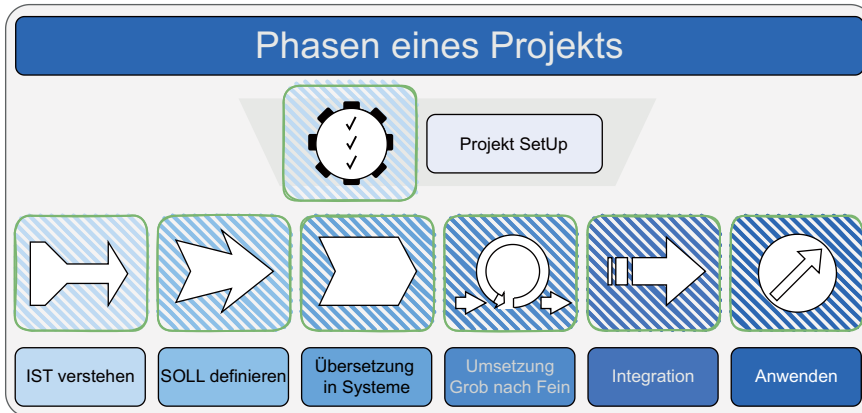


Bild 1.5 Die Phasen eines Projekts

- **Projekt-Set-up:** um was geht es und wie soll das Projekt realisiert werden.
- **IST-Prozess verstehen:** wie wird aktuell gearbeitet, dass die „Wahrheit“ mit allen ihren Feinheiten und schon genutzten oder nicht genutzten Systemen aufgenommen wird.
- **SOLL-Prozess definieren:** Wie sollen die zukünftigen Prozesse aussehen, mit denen gearbeitet werden soll. Jeder Prozess ist entweder ein eindeutiger Durchlauf oder komplexer mit mehreren Anwendungsfällen.
- **Übersetzung in Systeme:** Ab hier ist es überhaupt erst möglich, die Soll-Prozesse in mögliche Software zu übersetzen, eine Auswahl von möglichen Zielsystemen treffen und die Anpassungen grob skizzieren zu können.
- **Umsetzung von Grob nach Fein:** Die Realisierung und das Testen des Prozesses mit der oder den Softwarelösungen. Mit einem einfachen Fall beginnend und Schritt für Schritt mehr Komplexität realisieren – dies so lange wie nötig mit Optimierungsschritten.
- **Integration:** eine vollständige Testphase mit allen Komplexitätsstufen und mit Massendaten – idealerweise mit realen Daten und gleichzeitiger Schulung aller späteren Benutzer.
- **Anwenden:** ist aufgeteilt in folgende Schritte:
 - **Optional Pilot-Phase:** zum Testen in einer kleinen Umgebung
 - **Produktivstellung:** Mit Übergang vom alten Prozess zum neuen mit der neuen Softwarelösung. Hier ist eine intensive Phase der Begleitung und Behebung kleiner nicht aufgefallener Fehler.
 - **Begleitung und Optimierung:** Nachdem das neue System eingeführt wurde, muss es im Betrieb begleitet werden. Wenn nötig werden Kleinigkeiten direkt optimiert oder grössere Themen in neue Projekte ausgelagert.

1.6.3 Die Teilnehmenden

Aktuell arbeiten immer noch Menschen miteinander. Sie werden zwar punktuell von mehr oder weniger automatisierter Software und künstlicher Intelligenz unterstützt. Jedoch ist die künstliche Intelligenz im Jahr 2025 noch nicht so weit, eine Softwareeinführung ohne das Zutun von Menschen durchzuführen.

Bei der Organisation eines Projekts geht es darum, bestimmtes Wissen und Erfahrung in dem Projektteam zu haben. Ein Zusammenwürfeln von Teilnehmenden und schauen wir mal, wo wir rauskommen, wird nur selten funktionieren und hat mit Glück zu tun. Professionell sollte daher klar auf die Aufgabenstellung gesehen werden und folgende „Rollen“ sollten in der entsprechenden Seniorität vorhanden sein.

Die folgenden Rollen können grob einen Anhaltspunkt geben und sind später weiter detailliert aufgeführt.

- **Organisation:** die sogenannten Auftraggebenden, welche das Projekt finanzieren und die Projektleitung (Project Management), die sich um die Einhaltung der Ziele und den Fortschritt kümmert, sowie das gesamte Team zur Anforderungsdefinition, Umsetzung, Testen, Qualitätssicherung und das Produkt-Marketing und alle nicht im Team Beteiligten zusammenhält und einen Blick auf die Zeit und das Budget hat.
- **Begleitung Inhaltliche Anforderung:** der Fachbereich und die Initiatoren, Befürworter oder Motivatoren der Prozessänderungen. Zusammengehalten von einer Person, die den Zielprozess und die Lösung im Blick hat (Product Owner) und den Fachbereich, der den aktuellen Prozess nutzt und den neuen Prozess definiert oder begleitet und ihn fachlich versteht.
- **Betreuung Fachliche Prozesse:** die sogenannte Prozessberatung (Business Consulting). Zum Verstehen der Ist-Prozesse und zur Unterstützung bei der Formulierung der Soll-Prozesse.
- **Betreuung der Änderungen:** das sogenannte Änderungsmanagement (Change Management). Jede Änderung verursacht bei Beteiligten oft Bedenken oder gar Ängste, die begleitet werden müssen.
- **Betreuung Übersetzung in Zielbild:** die sogenannte Lösungsarchitektur (Solution Architecture). Die Übersetzung der neuen Prozesse in einen neuen technischen Lösungsansatz und Begleitung des gesamten Projekts als Übersetzer zwischen fachlicher Anforderung und technischer Lösung.
- **Betreuung technische Lösung:** begonnen bei der technischen Verantwortung (Lead Development), welche die Verantwortung für die technische Lösung hat. Und weiter die umsetzenden Teilnehmenden (Anpasser und Entwickler), welche das Ziel-System anpassen und Programmierungen vornehmen.

- **Testen:** eine der wichtigsten Rollen. Es geht um das ständige Testen der Umsetzung, Theoretisch und punktuell geht alles, aber funktioniert es auch praktisch und Ende zu Ende mit allen Schritten und vielen Daten. Diese Gruppe ist für den Erfolg des Projekts am wichtigsten.

1.6.4 Die Tools

Jede Arbeit sollte strukturiert und idealerweise dokumentiert erfolgen. Wenn mindestens jemand mit einer anderen Person arbeitet, also mindestens zwei Menschen zusammenarbeiten, sollten sie sich an einem gemeinsamen Ort austauschen können. In der heutigen Welt kann über das persönliche Gespräch über Online-Dienste sehr einfach zusammengearbeitet werden. Und Projektdokumentation sollte für alle je nach Rolle sofort zugänglich sein.

Daher ist es zum einen eine gemeinsame Entscheidung, WO dokumentiert und sich ausgetauscht wird und zum anderen WIE der Austausch aussieht.

Das Wo lässt sich anhand der Anforderungen klar formulieren und ist in dem Unterkapitel „Software“ beschrieben. Hier sind Anforderungen formuliert und jeweils mit einem realen Tool als Beispiel beschrieben.

Das WIE ist teilweise in den Tools beschrieben, es gibt aber auch noch die Anforderung an eine strukturierte Darstellung. Sei es ein Protokoll, eine Präsentation oder ein Konzept. All diese sollten auch hier Erwähnung finden und sind im Unterkapitel „Dokumente und Vorlagen“ beschrieben.



Disclaimer der angegebenen Software in diesem Buch

Die in diesem Buch beschriebene Software nutzt der Autor in 2025 aktiv selbst. Sie sind teilweise lizenzpflichtig. Es gab weder von einem Hersteller noch von Dritten Zuwendungen, um sie hier zu erwähnen. Gibt es andere Software, welche die gleichen oder mehr Anforderungen erfüllen, dann bitte gerne diese nutzen. Die angegebenen Software sollen nur eine Idee geben, wie die Anforderungen konkret genutzt werden können.

1.6.5 Fallbeispiele

Um die ganze Theorie mit etwas Leben zu versehen, geht das Kapitel Fallbeispiele auf die Nutzung der Vorgehensweise mit vier Beispielen ein (siehe Bild 1.6).

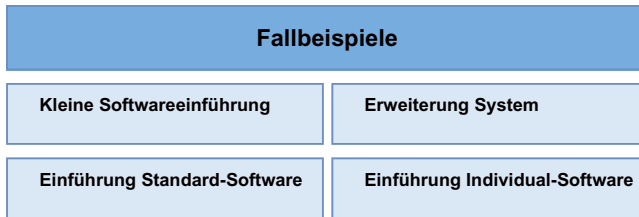


Bild 1.6
Fallbeispiele

1. Ein kleines Softwareeinführungsprojekt (siehe Abschnitt 6.1), das auf den ersten Blick keine Vorgehensweise braucht. Selbst bei einer Softwareeinführung von einem Kleinstprojekt sollte strukturiert und methodisch vorgegangen werden, was hier beschrieben wird.
2. Eine Erweiterung/Anpassung eines bestehenden Systems (siehe Abschnitt 6.2) ist auch ein kleines Projekt, das seine Eigenheiten hat und ein Vorgehen braucht.
3. Ein mittleres Softwareeinführungsprojekt (siehe Abschnitt 6.3), das oftmals mit der Nutzung einer Standardsoftware, mit einem schon im System vorhandenen Ablauf und Funktionalität vorhanden ist. Das System muss an die lokalen Gegebenheiten angepasst und in die bestehende Infrastruktur integriert werden.
4. Ein Grossprojekt ist in der Softwareeinführung (siehe Abschnitt 6.4), wenn es sehr komplex ist und auf teilweise oder gar nicht vorhandene Funktionalität der Software aufbaut und innerhalb der Software mehrere Bereiche oder sogar mehrere Softwaresysteme betrifft.

Die Beispiele teilen sich auf in eine kurze Beschreibung, ein oder zwei konkrete Beispiele, die Konzeption, die Umsetzung, die Integration und das Anwenden und eine Zusammenfassung.

1.7 Mehr als nur eine Softwareeinführung

Wenn aus welchem Grund auch immer in einer Organisation eine neue Software benötigt wird, geht es um mehr als nur eine technische Unterstützung eines Prozesses oder einer Arbeitsweise mittels einer Software.

Es geht in erster Linie um einen Prozess, der mit richtigen Teilnehmenden, einem klaren Vorgehen und bestimmten Tools realisiert werden soll.

Wichtige Blickwinkel zum Prozess sind jedoch, wie die neue Softwarelösung organisatorisch, fachlich und technisch definiert oder angepasst und realisiert werden soll (siehe Bild 1.7).

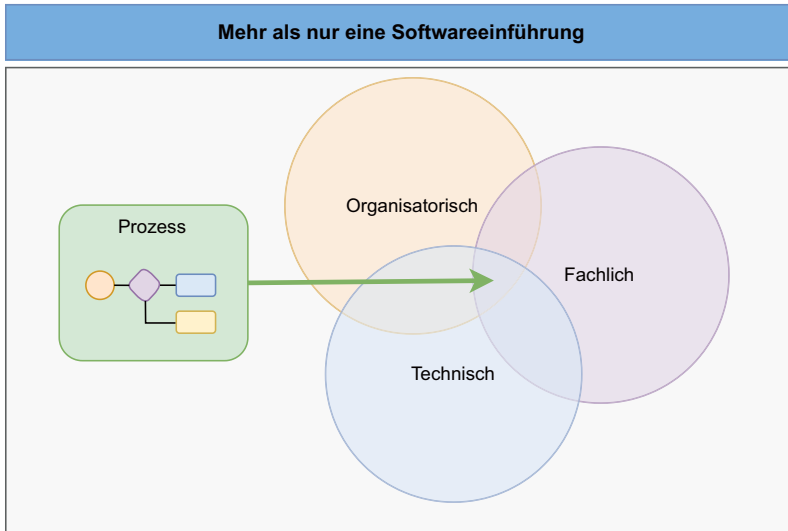


Bild 1.7 Prozess mit organisatorischen, fachlichen und technischen Anforderungen

Eine neue Software soll das Ergebnis eines definierten neu erstellten oder angepassten oder optimierten Prozesses darstellen und hat eben diese drei Bereiche als Grundlage. Oft wird in Projekten aus einem der drei Bereiche gestartet und auch der Fokus liegt auf einem der Bereiche.

So gibt es eine Organisationsänderung durch ein Zusammenlegen oder Verändern von Abteilungen oder Verantwortungsbereichen oder Software wird durch Zukäufe von Firmen zusammengelegt und der Fachbereich muss sich danach richten.

Wenn beispielsweise die eingesetzte Software aus dem Support läuft, also so alt ist, dass der Hersteller sie nicht mehr mit Updates versorgt, ist der Treiber für das Projekt ein technischer Grund und neue Software arbeitet ganz anders als die aktuell eingesetzte.

Es gibt auch Fälle, in denen der Fachbereich mit der aktuellen Lösung nicht mehr zurechtkommt, weil sich der Markt oder die Anforderungen an die Software geändert haben.

Jeder der Bereiche kann ein Initiator für ein Softwareprojekt sein. Alle drei sollten aber immer betrachtet werden, egal woher die Anforderung an das Projekt kommt.

Daher sollten Softwareprojekte aus diesen drei Blickwinkeln betrachtet werden, egal woher sie initiiert wurden:

- **Organisatorisch:** sind mit dem Projekt organisatorische Änderungen geplant und sind die Auswirkungen schon klar benannt. Oft wird in einem Projekt erst klar, dass die Änderung eines Prozesses auch organisatorischer Optimierungen bedarf oder diese empfehlenswert sind. Dies sollte im Vorfeld schon angedacht und im Projektverlauf berücksichtigt werden.

- **Fachlich:** sind in dem Projekt fachlich prozessuale Änderungen geplant. Selbst wenn keine Änderungen an den Prozessen geplant sind. Das kann nur der Austausch einer Software sein, es kann Optimierungen oder Anpassungen geben, weil die Software anders funktioniert oder andere Möglichkeiten hat. Sind fachliche Anpassungen am Prozess geplant, müssen sie formuliert und im Projekt angegangen werden.
- **Technisch:** Jede Änderung an einer Software hat technische Auswirkungen. Das kann von Feldern, Funktionen oder Programmierungen, die gebaut werden müssen, über Datenmengen, die sich ändern, hin zu Integrations Szenarien zwischen Softwaresystemen reichen. Wichtig ist, dass die Software, wenn nicht die Anforderung „Wir nutzen 100 % den Standard einer Software“ ist, als Werkzeug gesehen werden soll und sich nach den organisatorischen und fachlichen Anforderungen richten sollte. Dagegen steht der oft geäußerte Wunsch oder die klare Vorgabe „So nah wie möglich am Standard zu bleiben“ und die Software nicht zu stark an die organisatorischen oder fachlichen Anforderungen anzupassen. Dies muss je nach Zielen des Projekts berücksichtigt werden.

Wie sich hieraus ergibt, ist es ein Abwägen der drei Einflussthemata auf ein Software-einführungsprojekt, das jederzeit berücksichtigt, behandelt und klar entschieden und kommuniziert werden muss.

1.8 Künstliche Intelligenz

Wenn in 2025 über Arbeit an Software gesprochen wird, kommt sofort die Frage der Unterstützung von künstlicher Intelligenz auf. In Softwareprojekten wird die künstliche Intelligenz immer mehr genutzt und in Zukunft sehr wahrscheinlich auch viel mehr Raum in Softwareprojekten einnehmen. Einige Aufgaben kann sie jetzt schon übernehmen und zukünftig werden es bestimmt mehr.

Grundsätzlich ist künstliche Intelligenz, abgekürzt KI oder in Englisch AI, artificial intelligence ein Oberbegriff für Computersysteme, die anhand von angelernten Mustern Ergebnisse ausgeben können. Das reicht von einfachen Antworten auf Fragen in einem Chat, bis hin zu beispielsweise autonomem Fahren über die automatische Videoanalyse beim Fahren eines Autos oder Generierung von fast echt wirkenden Videos mit Menschen und Natur, die jedoch künstlich generiert wurden.

Dies ist ein grosser Unterschied zu Programmierungen, denen genaue Regeln zur Abarbeitung und Verarbeitung gegeben werden.

Somit ist eine Künstliche Intelligenz also eine eigenständige Entscheidungsfindung, die mit den eingehenden Daten beeinflusst werden kann. Sie beruht aber, und das ist wichtig zu bedenken, auf zuvor eingegebenen oder automatisch aufgenommenen Daten. Somit kann eine Künstliche Intelligenz zwar über Kombinieren von Daten Ergeb-

nisse generieren, aber (noch) nicht wie wir Menschen, eigenständig neue Themen erfinden. Was jetzt genau der Unterschied zwischen neuen Themen erfinden und Kombinieren von bestehenden Themen und daraus Neues zu erschaffen ist, vermag der Autor nicht zu beurteilen.

Es sollte auch unterschieden werden, ob eine künstliche Intelligenz zur Nutzung selbst angelernt oder eines der vielen Tools genutzt werden soll. Hier geht es in erster Linie um die Nutzung von Tools, die auf einer KI-Technologie beruhen.

Datensicherheit ist oft ein Thema, wenn es darum geht, die eigenen Daten als Grundlage für die Unterstützung durch eine künstliche Intelligenz freizugeben. Daher sollte genau darauf geachtet werden, welche Daten für die künstliche Intelligenz freigegeben werden und welche Benutzergruppen auf diese Daten durch die künstliche Intelligenz freigegeben werden. In Systemen gibt es oft Berechtigungseinschränkungen, die aufgrund von Rolle oder Abteilung/Bereich gegeben sind. Diese sollten unbedingt auch bei den Daten für eine künstliche Intelligenz berücksichtigt werden.

Beispielsweise wenn Personalakten der künstlichen Intelligenz zur Verfügung gestellt werden und jeder Mitarbeitende diese abfragen könnte, was normalerweise nur der Personalabteilung vorbehalten ist.

Der Wahrheitsgehalt oder die Richtigkeit des Ergebnisses basiert auf den wie weiter oben erwähnten angelernten Daten. Somit kann es sein, dass eine Antwort von einer künstlichen Intelligenz einfach falsch ist. Das kann eine falsche Schlussfolgerung sein oder ganz einfach auf einer falschen Datengrundlage beruhen. Ist ein Modell nicht gut genug angelernt (auf das Anlernen von künstlichen Intelligenzen gehe ich hier nicht weiter ein), kann es somit sein, dass die Daten nicht korrekt sind. Daher sollte eine Antwort immer kritisch hinterfragt und nicht als „Wahrheit“ gesehen werden. Die Qualität der Ergebnisse ist somit abhängig von den zugrunde liegenden Daten und der Fragestellung.

Ethik ist ein anderes Thema, das bei der Nutzung von künstlicher Intelligenz immer wieder aufkommt. Sind keine ethischen Algorithmen in der künstlichen Intelligenz berücksichtigt, wird auch die Antwort anhand der eingegebenen Daten analytisch und basierend auf genau den vorhandenen Daten sein. Das kann zu diskriminierenden oder nicht „menschlichen“ Antworten führen. Ein eher unverfängliches Beispiel: Wenn viele Mitarbeitende für eine anstehende Produktivstellung einmalig Überstunden in der Nacht leisten und dort die wichtigen Arbeiten realisiert werden und die künstliche Intelligenz nach der optimalen Zeit für ein manuelles Produktivstellen von Anpassungen gefragt wird, kommt bestimmt eine Zeit in der Nacht, obwohl zu der Zeit geplant niemand regulär arbeitet. Das als ein Beispiel, wenn Daten als Grundlage genommen werden und ethische oder organisatorische Aspekte nicht berücksichtigt sind.

In Softwareprojekten kann die künstliche Intelligenz viele Aufgaben übernehmen und automatisieren. Das kann von Vorschlägen, über Unterstützung bis hin zur kompletten Übernahme von Aufgaben reichen. Es sollte aber nicht jede Frage, die auf-