

Woher kommt die (softwaretechnologische) Zukunft?

Anna Sabine Hauptmann¹

Abstract: Da Softwaresysteme in nahezu allen Lebensbereichen eine Rolle spielen, beteiligen sich die Softwareentwickler maßgeblich an unserer Lebensgestaltung. Demzufolge sollten wir als Lehrende auf diesem Gebiet auch ethische und soziale Aspekte in den Unterricht einbeziehen, allem voran die Frage nach dem Sinn des Tuns. Während der gemeinsamen Projektarbeit auch im Studium treten durchaus Konflikte auf, deren Ursachen meist im Verborgenen liegen. Diese Konflikte lassen sich auch zurückführen auf den Umgang mit und die Ausübung von Macht bzw. Autorität. Brücken erlauben es, Erkenntnisse aus der Softwareentwicklung mit Erkenntnissen aus anderen Fachgebieten, insbesondere aus der Hirnforschung, zu verbinden und auch in den Alltag zu übertragen. Wollen wir den disruptiven Veränderungen in unserer Gesellschaft nicht ohnmächtig gegenüberstehen, dann werden wir Proaktive.

Keywords: Soziale Kompetenz; Ethik; Macht in Projekten; Haltung; Autorität; Reflexion; Annahmen

1 Einleitung

Technischer Fortschritt und verbunden damit wirtschaftliches Wachstum sind die angestrebten Ziele in unsere Gesellschaft. Demgegenüber machen wir als Menschen gerade die Erfahrung, dass sich eine sogenannte Krise nach der anderen zeigt. Das sollte uns zum Innehalten bewegen und zum Reflektieren. Vielleicht stecken wir mit unserer hohen technischen Kompetenz bei der gesellschaftlichen Entwicklung noch in den Kinderschuhen. Bau und Einsatz von Softwaresystemen berühren, wie technische Systeme überhaupt, gleichermaßen auch ethische und soziale Aspekte. Beziehungen zwischen allen Beteiligten beeinflussen den Projekterfolg auch innerhalb der studentischen Projekte. Die Projektarbeit offenbart die Haltungen, mit der die Teilnehmer agieren, Macht und Autorität sind spürbar, nicht kommunizierte Annahmen wirken und zeigen sich oft erst am Ende des Projektes. Die Lehrveranstaltungen zum Software Engineering bieten Raum, auch diese Perspektiven zu beleuchten. Im Grunde führt dieser Perspektivwechsel auch zu Synergien. Der Beitrag gibt einigen dieser Perspektiven Raum und lädt ein, darüber zu diskutieren.

2 Macht in Projekten

An jeder Universität und Hochschule gehört Projektarbeit zum Software Engineering. Studierende arbeiten in einer Gruppe und nicht selten kommt es zu Konflikten. Welche

¹ HTW Dresden, Fakultät Informatik/Mathematik, Friedrich-List-Straße 1, 01067 Dresden, Deutschland
hauptman@informatik.htw-dresden.de

Rolle spielt Macht dabei? Was eigentlich ist Macht? Jegliche Macht ist zunächst völlig neutral; sie ist weder gut noch böse. Studierende erwarben zum Beispiel im bisherigen Studium die Kompetenz, ein Datenmodell zu entwickeln. Im Rahmen des Softwareprojektes erhält nun eine Person die Aufgabe, die erforderliche Datenbank zu modellieren und zu implementieren. Zur Kompetenz, etwas tun zu können, kommt also die Befugnis, dies auch zu dürfen. Aber welche Intention, welche Absicht steht hinter dem Tun? Diese Intention weist unseren Handlungen die Richtung. Und mit diesem dritten Aspekt verliert die Macht ihre Neutralität. Je nach Intention kann aus Macht dabei auch Machtmissbrauch werden. Die Frage ist also, mit welcher Absicht baut das beauftragte Gruppenmitglied die Datenbank. In dem betrachteten Beispiel gibt es mehrere Intentionen. Abb. 1 zeigt dies beispielhaft. Macht in Projekten hat also nicht nur etwas mit der Projektleitung zu tun.

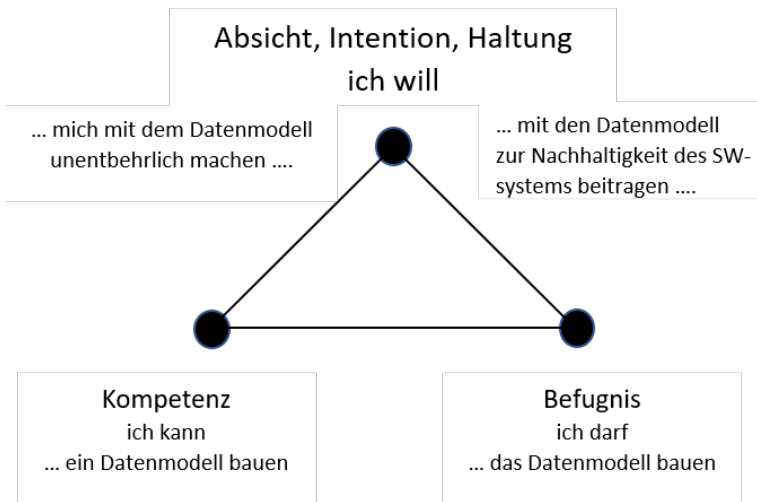


Abb. 1: Die Absicht (Haltung) gibt der Handlung ein Richtung.

Macht betrifft jede Person in der Gruppe. Und dabei ist es gleichgültig, ab diese Tatsache den Mitgliedern der Projektgruppe bewusst ist oder nicht. Das jeweilige Verhalten wirkt immer mit Konsequenzen auf die weitere Arbeit. Nimmt ein Projektmitglied seine Macht nicht wahr, ist es ohne Macht, quasi ohnmächtig. So kann Ohnmacht von außen verursacht aber auch selbst angenommen sein. Alles, was wir tun oder auch nicht tun, wird bestimmt von unserer Intention oder anders formuliert von unserer Haltung. Wir können also sagen, Softwareentwicklung ist eine Frage der Haltung wie die Lebensgestaltung überhaupt.

3 Haltung als Thema in Lehrveranstaltungen

Wenn Softwareentwicklung eine Frage der Haltung ist, dann stellt sich sofort die Frage: Wie lässt sich Haltung als wichtiger Faktor bei der Softwareentwicklung und auch beim Thema Digitalisierung vermitteln? Neurobiologisch gesehen tragen wir unsere Haltung in uns als

strukturelle Verankerung von Erfahrungen in neuronalen Netzwerken des präfrontalen Cortex im Gehirn. Die Haltung entspricht einer festen inneren Überzeugung, einer Einstellung. [TG20]

Diese quasi gespeicherte Haltung entsteht demnach aus unseren eigenen Erfahrungen; sie offenbart sich in unseren Worten, Aktionen und Reaktionen, ob es uns nun bewusst ist oder nicht. Haltung lässt sich also nicht wirklich vermitteln wie zum Beispiel die Syntax des Klassendiagrammes. Nach meiner Erfahrung ist es wichtig, zunächst auf diese Zusammenhänge hinzuweisen. Es geht darum anzuregen, die eigene Haltung zu ergründen und dabei auch zu prüfen, ob es sich wirklich um eine eigene Erfahrung handelt oder ob sie eventuell von einer Bezugsperson unreflektiert übernommen wurde. Praktisch bedeutet das, immer wieder die Fragen nach dem *Warum?* und dem *Wozu?* zu stellen.

Ausgehend davon bietet die Anforderungsanalyse die Gelegenheit, vor der Modellierung von Anwendungsfällen diesen wichtigen Fragen Raum zu geben. Manchmal wird viel zu schnell das Ziel formuliert. So ist es weit verbreitet, den Gewinn als Unternehmensziel zu betrachten. Aber ist der Gewinn nicht vielmehr eine Bedingung für den Fortbestand des Unternehmens? Das Ziel des Unternehmens leitet sich doch eigentlich aus der Gründungsvision aus dem Inhalt der Arbeit ab. Im Trailer zum Film „Die stille Revolution“ ist zu hören bzw. auf der zugehörigen Internetseite zu lesen: „Wir haben in der Vergangenheit viel *Know-how* gewonnen. Aber wir haben das *Know-why* verloren.“ [StRe18]. Allein das allgegenwärtige Thema Digitalisierung verliert ohne die Antwort auf die Frage *Wozu?* seinen Sinn. Der Focus muss sich also auch auf die Fragen richten: *Warum tun wir etwas?* oder *Wozu?* bzw. *Worin liegt der Sinn?*. Wenn wir etwas tun, nur weil wir es können, genügt dies nicht wirklich als Begründung.

Die Anregung, eigene Haltungen zu ergründen und zu prüfen, kann in den Lehrveranstaltungen auch unterstützt werden durch Hinweis auf die Erfolgsmuster (der Proaktive, der Vielsehende, ...) und Antipattern (zu viel des Guten, Toolistan, ...) von Peter Hruschka und Gernot Starke [HS18]. Ihre These ist: Gute Systeme entstehen durch gute Arbeit der Architekten. Diese gute Arbeit aber basiert neben den Kompetenzen eben auch auf entsprechenden Haltungen, die in den oberen Hirnschichten als Erfahrungen strukturell verankert sind. Neue Erfahrungen führen auch zu neuen Haltungen.

4 Autorität begegnen, Autorität ausüben

Auftretende Konflikte in einer Projektgruppe können zum Thema Autorität führen. Der Autoritätskonflikt ist wohl der erste Konflikt, dem wir Menschen gegenüberstehen. Schlussendlich resultiert er aus unserem Bedürfnis nach eigenem Willen (Autonomie) einerseits und unserem Bedürfnis nach Zugehörigkeit andererseits. Wir alle entwickeln abhängig vom Kontext Strategien, die sich beim Autoritätskonflikt (scheinbar) bewähren und die wir dann quasi als abrufbare Muster im Hirn speichern. Ob uns diese Muster bewusst sind oder nicht, spielt dabei keine Rolle. Nahezu jeder Softwareentwickler begegnet der Situation,

mit Autorität umgehen zu müssen. Im Falle der Übereinstimmung zum Beispiel bei Entwurfsentscheidungen entsteht kein Konflikt, wohl aber, wenn der Konsens abhandenkommt. Welche Möglichkeiten ergeben sich dann für uns, mit diesem Autoritätskonflikt umzugehen? Im Grunde wählen wir als einzelne Person aus nur zwei Möglichkeiten: sich *anpassen* oder dagegen *rebellieren* (mit vielen Graustufen dazwischen). Sowohl das völlige Anpassen als auch das Rebellieren führen über kurz oder lang in der jeweiligen Person zu Inkohärenz, die sich als Stress zeigt. Abb. 2 verdeutlicht dies. Die Projektgruppe verfügt aber noch

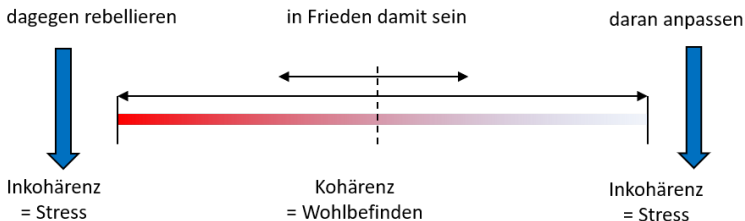


Abb. 2: Mögliche Reaktionen in einem Autoritätskonflikt.

über eine dritte Möglichkeit. Diese dritte Variante stellt Jürgen Pasch [Pa94] in seinem schon etwas älteren Buch vor als den dialogischen Softwareentwurf bzw. das dialogische Prinzip in der Projektgestaltung. Auf diese Weise entstehen nachhaltige Softwaresysteme. Vertreter der modernen Hirnforschung, allen voran Gerald Hüther, bezeichnen dies als Potentialentfaltung [Pot18]. Ist in einem Dialog ein nachhaltiger Entwurf entstanden, dann profitieren das Softwaresystem und die Projektgruppe gleichermaßen davon: die Qualität des Entwurfes ist hoch und die Projektmitglieder erreichen Kohärenz zwischen dem, was sie wollen und dem, was sie tun. Schlichtweg fühlen sie sich jenseits von Stress wohl bei ihrer Arbeit.

Inkohärenzen entstehen sicher nicht nur, wenn es um Entwürfe geht. Wie heilsam wäre es, wenn wir Menschen unsere Handlungsmuster in Bezug auf den Umgang mit Autorität und in Bezug auf das Ausüben von Autorität reflektieren und dann ggf. verändern könnten zum Nutzen aller. Unsere Hirne sind plastisch; einer strukturellen Veränderung steht also nichts im Weg. [Hü01]

5 Nicht kommunizierte Annahmen im Projekt

In der Diskussion von Entwürfen zeigen sich weitere Analogien: Bei der Beschreibung von Programmierschnittstellen verweisen wir neben Semantik und Syntax auch explizit auf Vor- und Nachbedingungen sowie gegebenenfalls auf Invarianten. Schlussendlich geht es um Annahmen. Aber wie sieht es mit unseren Annahmen aus, die wir in der Projektarbeit treffen? Kommunizieren wir diese Annahmen wirklich? Der gelegentliche Ausspruch „Ach, ich habe gedacht, dass . . . “ weist unmittelbar auf nicht kommunizierte Annahmen. Die Menge von uns umgebenden sogenannten Krisen zeigt wohl deutlich, dass wir auch im

gesellschaftlichen Leben Annahmen getroffen, sie quasi zum Gesetz erhoben haben und demzufolge nicht mehr hinterfragen. Diese Annahmen mögen zu ihrer Entstehungszeit und im Entstehungskontext durchaus berechtigt gewesen sein. Heute treffen sie vielleicht nicht mehr zu. Die Annahmen „Konkurrenz ist der tragende Entwicklungsmotor.“, „Zeit ist Geld.“ und „Märkte sind Mechanismen.“ scheinen wie fest verdrahtet in unseren Köpfen. Aber wohin haben uns diese und andere Annahmen geführt? Tom DeMarco antwortet auf die Frage: „Haben uns verirrt, kommen aber gut voran.“ [De01](S.43, Kapitel II). Wo bzw. was ist der Kompass, der uns aus dieser Situation herausführen kann? Gerald Hüther sieht in der Besinnung auf unsere *Würde* eine gute Möglichkeit dazu. [Hü18] Tom DeMarco [De01] und Gerald Hüther [Hü97], [Hü20] stimmen darin überein, dass die Kultur der Angst als essentielles Entwicklungshemmnis wirkt. Dies gilt offensichtlich sowohl für Softwareprojekte als auch für unsere Gesellschaft. Im Rahmen der Projektarbeit taucht häufig eine weitere nicht explizit kommunizierte Annahme auf: die Projektleitung habe die Aufgabe, Projektmitarbeiter zu motivieren. Nach meiner Erfahrung und Beobachtung gibt es keine extrinsische Motivation. Vielmehr schafft das, was wir extrinsische Motivation nennen, Abhängigkeit von Geld, von Noten, von Lob und manchem mehr. Allerdings können wir eine Arbeitskultur ohne Angst schaffen, die der (intrinsischen) Motivation der Projektmitarbeiter Raum bietet, eine Kultur, in der sich Potentiale entfalten können. Diese Kultur baut auf Wertschätzung, auf Achtung, ist verbunden mit Achtsamkeit und Gewährsein und führt unmittelbar zu der Frage *Was ist eigentlich Arbeit?*. Arbeit wurde im Laufe der Zeit immer mehr zum Gelderwerb, vielleicht ist sie aber eher soziale Teilhabe.

Wenn wir Softwareentwicklung im Kontext von Programmierparadigmen diskutieren, dann wird deutlich, dass diese damals neuen Paradigmen die Antwort auf spezielle Rahmenbedingungen waren. Genau genommen haben die Eltern der Objektorientierung Softwaresysteme damals quasi „neu gedacht“. Das führte zu erheblichen Vorteilen ebenso auch zu Konsequenzen. Wenn wir die uns umgebende Wirklichkeit betrachten, dann drängt sich der Gedanke auf: Sollten wir nicht auch in diesem Kontext *neu denken* [Gö21]? Neue Denkweisen schaffen Raum, um Probleme zu lösen, die durch bisherige Denkweisen entstanden sind. Albert Einstein gab uns diese Erkenntnis schon an die Hand. Neue Denkweisen finden wir aber nur, wenn wir veraltete, inzwischen hinderliche Denkweisen als solche erkennen und konsequent über Bord werfen. Und diese veralteten Denkweisen sind meist verbunden mit alten, nicht mehr in Frage gestellten Annahmen. Der gedankliche Weg von nicht kommunizierten Annahmen sowohl in der Anforderungsanalyse als auch im Softwareentwurf zu nicht (mehr) kommunizierten Annahmen in unserer Gesellschaft ist ein kurzer. Andreas Reckwitz diskutiert die Frage, ob digitale Technologien auch eine Kulturalisierung des Technologischen und des Sozialen betreiben. Um auf die Grenzen der These, wir befänden uns in einer Informations- und Wissensgesellschaft, hinzuweisen, unterscheidet er nicht nur Daten und Informationen, sondern auch Kulturformate: „Das Internet ist zu erheblichen Teilen eine Affektmaschine. Seine zirkulierenden Bestandteile erregen, unterhalten, stimmen freudig, entspannen, hetzen auf oder bewirken, dass man sich angenehm aufgehoben fühlt.“ ([Re17], S.234f.) Wäre es nicht notwendig, diese Denkanstöße

aus ethischer Perspektive insbesondere auch mit Studierenden der (Medien-)Informatik zu diskutieren?

6 Reflexion als wichtige Methode zur Überprüfung

Reflektieren, das Innehalten und Prüfen, ist wichtiger Bestandteil der Softwareentwicklung. Bewusst und regelmäßig in kurzen Abständen angewandt, trägt Reflexion agiles Arbeiten und bietet Verbesserungsmöglichkeiten insbesondere auch für Softwareentwürfe. So wie beim Entwickeln und Weiterentwickeln eines Softwaresystems durchaus auch Fehler entstehen können (technische Schulden), so kommt es zu fachlichen Schulden bei mangelnder Anforderungsanalyse. Durch bewusstes Innehalten und Anwendung von geeigneten Analyserwerkzeugen sind wir in der Lage, diese Schulden zu erkennen und darauf zu reagieren, so wie es Carola Lilienthal für Softwareentwürfe empfiehlt [Li16]. Das Analysieren und Verbessern setzt aber auch die entsprechenden Haltungen voraus; das äußert sich z.B. in der Planung, in konsequenter Durchführung, in einem Qualitätsbewusstsein.

Vielleicht kommt es analog auch zu ungewollten gesellschaftlichen Schulden, die nach Korrektur verlangen. Aus welcher Perspektive wir Softwareentwicklung bzw. unsere Gesellschaft auch betrachten, schlussendlich geht es auch darum, dass wir uns die Rahmenbedingungen und die Konsequenzen unseres Tuns in kleinen Schritten bewusst machen. Dabei spielt die Reflexion eine zentrale Rolle; die Reflexion wird zum Rhythmus der Arbeit, sie sollte auch zum Rhythmus des persönlichen und des gesellschaftlichen Lebens überhaupt werden. Aber Reflexion braucht Zeit und Raum; der Effizienzwahn der vergangenen Jahrzehnte hat mit dem Beseitigen der Spielräume diese wichtigen Voraussetzungen für reflektiertes Handeln in manchen Softwareprojekten und ebenso in anderen Bereichen verdrängt. [De01]

7 Fazit

Es ist wohl kaum mehr zu übersehen, dass wir als Gesellschaft vor disruptiven Veränderungen stehen. Fragen wie: *Wie wollen wir eigentlich leben und arbeiten?*, *Was eigentlich ist Arbeit?*, *Wer wollen wir sein?* drängen sich auf. Softwareentwickler stehen mit ihrer Arbeit mitten in diesem Veränderungsprozess, vielleicht sogar an einer zentralen Stelle. In den Lehrveranstaltungen zum Software Engineering über technische Themen hinaus zu gehen, soziale und ethische wie auch ökologische Aspekte einzuschließen, bedeutet, die Studierenden einzuladen, der Ohnmacht die Bewusstheit entgegenzusetzen, um gemeinsam einen Weg in eine neue Gesellschaft zu finden, in eine Gemeinschaft, von der Maja Göpel behauptet „Wir können auch anders“ [Gö22]. Vielleicht entsteht diese neue Welt nicht mehr nur vordergründig durch Bearbeitung oder Verarbeitung (Ressourcenausnutzung), sondern durch Erarbeitung [MH20]. Wir alle sind Suchende auf diesem Weg. Auch die heutigen Studierenden werden es sein, die diese Zukunft maßgeblich gestalten und das auf der Basis ihrer Haltungen und damit auf der Basis ihrer heutigen Erfahrungen. Als Lehrende können

wir sie einladen, über den Tellerrand hinaus zu schauen, interdisziplinär zu denken und zu handeln und dabei neue Erfahrungen zu sammeln. Wir können sie mit unseren Fragen ein Stück des Wegs begleiten auch oder gerade im Software Engineering.

Literatur

- [De01] DeMarco, T.: Spielräume - Projektmanagement jenseits von Burnout, Stress und Effizienzwahn. Verlag Hanser, 2001.
- [Gö21] Göpel, M.: Unsere Welt neu denken - Eine Einladung. Verlag Ullstein, 2021.
- [Gö22] Göpel, M.: Wir können auch anders - Aufbruch in die Welt von morgen. Verlag Ullstein, 2022.
- [HS18] Hruschka, P.; Starke, G.: Knigge für Softwarearchitekten. Verlag entwickler.press, 2018.
- [Hü01] Hüther, G.: Bedienungsanleitung für ein menschliches Gehirn. Verlag Vandenhoeck & Ruprecht, 2001.
- [Hü18] Hüther, G.: Würde Was uns stark macht - als Einzelne und als Gesellschaft. Verlag Knaus, 2018.
- [Hü20] Hüther, G.: Wege aus der Angst - Über die Kunst, die Unvorhersehbarkeit des Lebens anzunehmen. Verlag Vandenhoeck & Ruprecht, 2020.
- [Hü97] Hüther, G.: Biologie der Angst - Wie aus Stress Gefühle werden. Verlag Vandenhoeck & Ruprecht, 1997.
- [Li16] Lilienthal, C.: Langlebige Softwarearchitekturen - Technische Schulden analysieren, begrenzen und abbauen. Verlag dpunkt, 2016.
- [MH20] Metelmann, J.; (Hg.), H. W.: Imagineering - Wie Zukunft gemacht wird. Verlag Fischer, 2020.
- [Pa94] Pasch, J.: Softwareentwicklung im Team – mehr Qualität durch das dialogische Prinzip bei der Projektarbeit. Verlag Springer, 1994.
- [Pot18] Akademie für Potentialentfaltung: Gemeinsam über sich hinauswachsen, 2018, URL: <https://akademiefuerpotentialentfaltung.org/>, Stand: 10.01.2023.
- [Re17] Reckwitz, A.: Die Gesellschaft der Singularitäten - Zum Strukturwandel der Moderne. Verlag Suhrkamp, 2017.
- [StRe18] Kristian Gründling and Bodo Janssen: Die stille Revolution, 2018, URL: <https://www.die-stille-revolution.de/#vision>, Stand: 06.01.2023.
- [TG20] Teobald Werner, Gerald Hüther: Eine Frage der Haltung!, 2020, URL: https://www.gerald-huether.de/wp-content/uploads/2020/06/Eine_Frage_der_Haltung.pdf, Stand: 06.01.2023.