

## Softwareentwickler – die unbekannte Spezies

### Leicht verrückt oder einfach nur anders?

Warum sind Soft Skills in der IT von so besonderer Bedeutung, dass es sich lohnt dazu ein Webinar durchzuführen und dieses White Paper zu verfassen? Es sind knallharte wirtschaftliche Fakten, die solch ein Thema in den Fokus bringt. Eine Studie zu Erfolgsfaktoren in Softwareprojekten brachte die intensive direkte Zusammenarbeit von



Uwe Vigenschow

Softwareentwicklern und Stakeholdern (Anwender, Fachexperten, Entscheider auf Managementebene, Vertriebs- und Marketingmitarbeiter usw.) als die für den Projekterfolg signifikanteste Einzeltechnik zutage [PM-Studie]. Mindestens einmal pro Woche, besser aber täglich, arbeiten Menschen zusammen, wie sie unterschiedlicher kaum sein könnten. Das führt uns direkt zum nächsten Problem. Unsere Kommunikation ist anfällig für Missverständnisse. Bei der Unterschiedlichkeit der

beteiligten Personen in und um IT-Projekte führen diese Missverständnisse schnell zu Konflikten [Vigenschow2011]. Diese Konflikte konstruktiv zu lösen, ist ein weiterer offensichtlicher Erfolgsfaktor. Eine Studie der KPMG zeigt die enormen Kosten aus unangemessen behandelten Konflikten deutlich auf [KPMG-Studie]. Soft Skills, die wir z.B. für unsere Teamfähigkeit, unserem Kommunikations- oder Konfliktverhalten an den Tag legen, sind also ein zentraler Erfolgs- und Kostenfaktor bei der Bewältigung komplexer Projekte. Und komplex sind unsere IT-Projekte allemal.

**Was sind nun unsere Soft Skills?** Meist wird darunter unsere soziale Kompetenz verstanden. Die dazu benötigten Soft Skills lassen sich in drei Gruppen einteilen. Unter den **allgemeinen Soft Skills** versteht man Aspekte wie z. B. Menschenkenntnis, Durchsetzungs- und Einfühlungsvermögen. Sie basieren auf unserer **Persönlichkeit** mit ihren Faktoren wie Empathie, Durchsetzung, Sensibilität, Flexibilität usw. Daraus lassen sich nun **beruflich relevante Soft Skills** ableiten wie Kommunikations-, Konflikt- und Teamfähigkeit, die wir im Folgenden näher betrachten wollen.

**Missverständnisse in unserer Kommunikation** kommen oft vor und haben viele Ursachen. Häufig sind es Transformationsverluste, die entlang der Kommunikation zwischen Menschen entstehen. Auf Schulz von Thun geht eine Zerlegung zurück, die uns bei der Analyse von Kommunikationsmissverständnissen helfen kann [Vigenschow2011]:

gedacht	bedeutet nicht	gesagt
gesagt	bedeutet nicht	gehört
gehört	bedeutet nicht	verstanden
verstanden	bedeutet nicht	einverstanden

Gehen wir das der Reihe nach durch. Was ich **gedacht** habe, habe ich nicht automatisch auch gesagt. Auf dem Weg aus unseren Gedanken in Sprache finden Tilgungen und Verzerrungen des Inhalts statt, weil wir den Inhalt verdichten müssen. Meist fällt uns das nicht weiter auf, doch sobald wir in einer Fremdsprache kommunizieren, merken wir sofort, welche Nuancen wir nicht wie gewünscht ausdrücken können. Wenn wir etwas **gesagt** haben, kann es aus verschiedenen Gründen sein, dass wir nicht gehört wurden. Vielleicht gab es Störgeräusche oder unser Gesprächspartner war abgelenkt. Auf jeden Fall fehlt ihm dann Inhalt. Aber auch wenn er alles richtig **gehört** hat, muss er es deswegen noch lange nicht in unserem Sinne verstanden haben. Gerade wenn Menschen mit unterschiedlichem fachlichen Hintergrund miteinander reden, passiert dieser Effekt häufig. Sammeln Sie doch mal die unterschiedlichen Definitionen in ihrer Firma für den Begriff *Kunde*. Softwareentwickler sind Experten für Programmierung. Sie analysieren Anforderungen mit Fachexperten aus ganz anderen, fremden Bereiche wie z. B. Steuerrecht, Medizin oder Betriebswirtschaft. Da sind Verständnisfehler vorprogrammiert. Und auch, wenn ich alles **verstanden** habe, muss ich nicht damit einverstanden sein, weil andere Prioritäten oder technische Möglichkeiten evtl. dagegen sprechen. Das Problem ist oft nicht nur, dass wir das nicht sofort kommunizieren, sondern dass wir unsere fehlende Kommunikation noch nicht einmal bemerken. Menschliche Kommunikation ist also anfällig für Missverständnisse. Und als wäre das nicht schon genug, kommt dazu, dass die Menschen, die in IT-Projekten aufeinander treffen und zusammenarbeiten müssen, um erfolgreich zu sein, oft auch noch sehr verschiedenartig denken und kommunizieren.

Softwareentwickler – die unbekannte Spezies  
Leicht verrückt oder einfach nur anders?

in Zusammenarbeit mit:



Whitepaper zum Webinar „Soft Skills in der IT“  
Oktober 2012, korrigierte Fassung 4. 8. 2013  
© Uwe Vigenschow

GoToAssist  
by citrix

Abb. 1: Vier-Quadranten-Typologie

<p>Suche nach Optionen und Möglichkeiten Bedürfnis: Freiheit und Wandel Metapher: Visionär Bezug: Zukunft Sprache: visuell, unverbindlich, offen <i>„Entscheidung ist Mord an Möglichkeiten!“</i></p> <h2 style="text-align: center;">Wohin noch?</h2> <p>Falle: Ablenken lassen und nichts zu Ende bringen Lösung: Fokus auf Gegenwart und Priorität, „Wenn-dann“-Argumentation Beispiel: <i>„Wenn wir in Zukunft diese wichtige Möglichkeit weiter haben wollen, brauchen wir jetzt die folgende technologische Basis dafür!“</i> Konfliktvermeidungsmuster: Ablenker – <i>„Nichts ist wirklich wichtig!“</i> Stärke: kreativ, langfristig denkend, charismatisch, Probleme relativierend</p>	<p>Suche nach dem Sinn (für einen selbst) Bedürfnis: Nähe und Ethik Metapher: Philosoph Bezug: Vergangenheit Sprache: pauschal, verallgemeinernd, emotional Beispiel: <i>„Das haben wir noch nie so gemacht!“</i></p> <h2 style="text-align: center;">Warum?</h2> <p>Falle: innere Blockade Lösung: Wertschätzung und konkret werden Beispiel: <i>„Ihre Erfahrungen sind mir wichtig für die Lösung! Was genau hat es bisher verhindert, so zu arbeiten?“</i> Konfliktvermeidungsmuster: Beschwichtiger – <i>„Ich habe Schuld!“</i> Stärke: de-eskalierend, gutes Miteinander</p>
<p>Suche nach Handlungsanweisungen und „Kochrezepten“ Bedürfnis: Ergebnisse und Individualität bzw. Distanz zur Gruppe Metapher: Improvisator Bezug: Gegenwart und nahe Zukunft Sprache: schnell, kurz, knapp, ungeduldig <i>„Das ist doch ganz einfach!“</i></p> <h2 style="text-align: center;">Wie?</h2> <p>Falle: Aktionismus und „ungare“ Lösungen Lösung: bremsen und erste Handlungsanweisungen geben oder erarbeiten lassen Beispiel: <i>„Wir verstehen das Problem noch nicht gut genug! Was kann ein erster Schritt sein, mehr Information darüber bis 15 Uhr zu bekommen?“</i> Konfliktvermeidungsmuster: Ankläger – <i>„Du bist schuld!“</i> Stärke: ergebnisorientiert, durchsetzungsstark, Konflikte kommen auf den Tisch</p>	<p>Suche nach Information, Wissen und Beweisen Bedürfnis: Sicherheit und Stabilität Metapher: Wissenschaftler Bezug: zeitlos Sprache: ruhig, exakt, sachlich <i>„Es dauert so lange, wie es dauert!“</i></p> <h2 style="text-align: center;">Was?</h2> <p>Falle: Theoretisieren, Verzetteln und Analyse-Paralyse Lösung: ruhig, geduldig bleiben, Aufgaben mit Timebox, methodisch arbeiten Beispiel: <i>„Was brauchen Sie, damit Sie mir nach einer Standard-Schätzmethode bis 18 Uhr eine Aussage zur Dauer machen können?“</i> Konfliktvermeidungsmuster: Rationalisierer – <i>„Gefühle stören!“</i> Stärke: lösungsorientiert, qualitätsorientiert</p>

**Was steckt hinter der Unterschiedlichkeit der Menschen**, die direkt oder indirekt bei einem Softwareprojekt zusammenarbeiten? Menschen haben teils genetisch bedingt, teils durch Sozialisation entwickelte, unterschiedliche Bedürfnisse, Werte und Charakterzüge. Neben diesen, sich eher langsam weiter entwickelnden Aspekten, treffen wir zusätzlich auf individuelle Interessen, die durch die konkrete Situation oder persönliche Rahmenbedingungen geprägt sind. Bedürfnisse, Werte, Wesenszüge und Interessen bilden die persönliche und sehr individuelle Basis für die Kommunikation eines Menschen. Ein einfaches typologisches Modell kann uns helfen, in konkreten Kommunikationssituationen mit dieser Individualität angemessener umzugehen und so erfolgreicher zu kommunizieren. Bevor wir tiefer in diese Typologie einsteigen, ist es mir wichtig, eine Warnung auszusprechen, die bereits C. G. Jung sinngemäß seinen typologischen Betrachtungen voran gestellt hat. Jede Typologie ist eine starke Vereinfachung auf zentrale, sichtbare Aspekte menschlichen Verhaltens. Sie darf auf keinen Fall dazu missbraucht werden, Menschen in Schubladen zu legen und z. B. zu sagen, der Paul ist ein XY-Typ. Die Anwendung einer Typologie in einer konkreten

Gesprächssituation ist nur auf diese kurze Situation beschränkt. Menschen sind zu vielfältig, um sich in eine Typologie pressen zu lassen, egal wie viele Schubladen diese bietet. Situativ eingesetzt sind typologische Grundkenntnisse hingegen für die Kraft und Zielführung unserer Kommunikation sehr hilfreich. Dazu kommt, dass wir durch typologische Betrachtungen ein besseres Gefühl und eine Herangehensweise erhalten, um mit unterschiedlichen Menschen angemessen umgehen zu können.

**Was sagt unsere Typologie genau aus?** Werfen wir einen Blick auf eine einfache Typologie aus vier Quadranten (Abb. 1). Wir haben stets alle vier Ausprägungen in uns und das ist auch gut so, da sie für unsere charakterliche Vollständigkeit notwendig sind. Alle vier Quadranten weisen wertvolle Stärken auf, die wir für einen langfristigen Projekterfolg benötigen. Oft sind zwei dieser Präferenzen eher stark ausgeprägt und in einer konkreten Situation dominiert häufig genau eines dieser Muster unsere Kommunikation [Vigenschow2011]. Gehen wir die vier Quadranten der Reihe nach durch und finden heraus, wie wir mit Personen in einer bestimmten Präferenz in Kontakt kommen.



Im **Warum**-Quadranten beginnen wir oben rechts mit unseren Betrachtungen zu den Präferenzen in menschlicher Kommunikation. Die zugrunde liegenden Bedürfnisse sind die Nähe zu anderen Menschen in der Gruppe, ein harmonisches Miteinander und ethisch-moralisches Handeln. Wenn wir aus dieser Präferenz heraus kommunizieren, beziehen wir uns meist auf die Vergangenheit und pauschalisieren bzw. verallgemeinern in unserer Wortwahl. Als Metapher für diese Grundhaltung wähle ich gerne das Bild des Philosophen und bitte gleich bei allen um Verzeihung, die sich ernsthaft mit Philosophie auseinander gesetzt haben. Die Stärke dieser Präferenz liegt in ihrer de-eskalierenden Wirkung auf andere und dem guten Miteinander in der Gruppe, das so geschaffen wird. Ihre Falle, in die sie die Menschen führen kann, ist die innere Blockade bis hin zum Rückzug aus der Kommunikation. Betrachten wir zur Illustration ein paar Beispiele für typische Aussagen aus dieser Präferenz heraus:

- „Das haben wir noch nie so gemacht!“
- „Das haben wir schon immer so gemacht!“
- „Die machen doch immer, was sie wollen!“
- „Die Anwender haben alle keine Ahnung!“

Die Wortwahl zeichnet sich also durch die Verwendung von Universalquantoren wie alle, keiner, immer, nie usw. aus, ist meist hoch emotional und kann bis zum persönlichen Angriff gehen. Manchmal ist aber genau das Gegenteil der Fall und die innere Blockade hat zur Resignation geführt. Dann zieht sich ein Mensch mit dieser aktuellen Präferenz komplett aus der Kommunikation heraus und wird still.

Wie können wir mit einer Person in diesem Modus (wieder) konstruktiv kommunizieren? Dazu gehen wir zweistufig vor: Zum einen kommen wir dem Bedürfnis nach Nähe nach, indem wir sehr wertschätzend und individuell kommunizieren, um einen guten Kontakt herzustellen. Zum anderen ziehen wir den Fokus der Kommunikation aus der Verallgemeinerung wieder in das Konkrete. Das klingt einfach und

logisch, ist aber tatsächlich oft recht schwierig. Durch die durchaus hohe Emotionalität und die pauschalen K.O.-Argumente ist es oft für uns eine Herausforderung, wertschätzend zu kommunizieren. Der Schlüssel, liegt darin, sich immer wieder klar zu machen, dass sich hinter den verallgemeinernden Aussagen oft wertvolle Details und konkrete Informationen verbergen, die für den Erfolg zu entwickelnder Lösungen oder anstehender Veränderungen wichtig sind. Es kann sich also sehr wohl lohnen, den pauschalen Blockierer nicht einfach abzutun, sondern genau hinzuschauen.

Unsere Wertschätzung können wir meist am einfachsten ausdrücken, indem wir die Erfahrung und das (Detail-)Wissen der Person als wichtig für die Lösung des anstehenden Problems erachten. So bekommen wir dann auch den Dreh, aus der Pauschalisierung in die konkreten Details zu kommen: „Ihre konkrete Erfahrung mit der Materie ist mir besonders wichtig für die gemeinsame Lösung. Was genau verhindert aus Ihrer Sicht die Veränderung?“ So können wir unser ehrlichen Wertschätzung Ausdruck verleihen, ohne zu übertreiben, sind nah bei der anderen Person und kommen zu den konkreten Hindernissen.

**Die Basis einer wertschätzenden Kommunikation** bildet *echte* Wertschätzung. Das gilt für jede Präferenz, auch wenn sie für den *Warum?*-Quadranten aufgrund des Bedürfnisses nach Nähe besonders wichtig ist. Damit wir nicht aufgesetzt kommunizieren, sondern echte Wertschätzung ausdrücken können, kann es helfen, sich auf die Stärken der aktuellen Präferenz unseres Gesprächspartners zu besinnen oder generell auf die Stärken der Person. Dieses einfache *Reframing* wird in den meisten Situationen ausreichend sein. Unter schwierigen Bedingungen kann es helfen, wenn wir uns kurz aus der Kommunikation zurückziehen und eine unserer Kolleginnen oder Kollegen übernehmen kann. Generell kann eine andere Konstellation bei festgefahrenen Kommunikationsmustern helfen.

Im **Was**-Quadranten finden wir die wohl häufigste Präferenz bei der Kommunikation mit Entwicklern. Sie beruht auf dem Bedürfnis nach Sicherheit und Stabilität. Um diese zu befriedigen, wird nach Informationen, Belegen bzw. Beweisen gesucht und unsere Arbeit auf bewährte Methoden und Praktiken gestützt. Diese, von mir mit der Metapher des Wissenschaftlers beschriebene Präferenz kommuniziert daher sehr ruhig und sachlich. Zeit bzw. Termine spielen dabei oft eine eher untergeordnete Rolle oder werden gar ignoriert, da unter Druck die Gefahr besteht, etwas wichtiges zu übersehen. Es sei denn, es wird sich aus dieser Präferenz heraus rational mit Zeitmanagement befasst. Die Stärken in diesem Quadranten liegen in der hohen Rationalität, mit der Lösungen erarbeitet werden und einer Fokussierung auf Qualität. Leider, und damit kommen wir zu den dazugehörigen Fällen, sind die Qualitätsaspekte oft andere, als die einer Qualitätssicherung oder der

Eine **Timebox** ist ein vergleichsweise kurzer Zeitabschnitt, an dessen Ende wir den Stand der Ergebnisse gemeinsam sichten. Danach wird dann entschieden, ob und wie weitergearbeitet wird. Der Zeitrahmen einer Timebox ist stets fix. Wir können Workshops oder Besprechungen ebenso durch mehrere Timeboxes strukturieren wie die Entwicklung von Software. Im Rahmen einer iterativen Entwicklung sind die Timeboxes ineinander verschachtelt. Ein Release in drei Monaten (Timebox) wird in drei bis sechs Iterationen mit fester Dauer (Timeboxen) unterteilt. Die Arbeit der Entwickler wird dann weiter in Wochen- oder Tagesrhythmen auf Timeboxes aufgeteilt und dazu in testbare Teilaufgaben zerlegt. Die Menge der abgenommenen Ergebnisse macht dann das Inkrement einer Iteration aus.

Generell kann aus Sicht des Projektmanagements noch ergänzt werden, dass ein **inkrementell-iteratives Vorgehen** mit kurzen Timeboxen und regelmäßigen Feedbackschleifen die Zusammenarbeit aller Stakeholder und die Kommunikation über alle vier Quadranten unterstützt. Die regelmäßige Zusammenarbeit lässt sich schrittweise optimieren und alle beteiligten Personen können lernen, angemessen miteinander umzugehen. Dies ist auch meist notwendig, da der Erfolg iterativen Vorgehens wesentlich von unserer Kommunikationsfähigkeit abhängt.

Anwender. Entwickler stellen in dieser Präferenz oft die innere Qualität und die Eleganz des inneren Designs in den Vordergrund. Aspekte der Bedienbarkeit oder bestimmter Fehlerbilder, wie sie für Anwender oder Tester von Bedeutung sind, bleiben bei diesen Betrachtungen leicht außen vor. Die Falle einer übertriebenen Was-Präferenz liegt daher in einem Theoretisieren und Überanalysieren von Problemen und dem Ignorieren emotionaler Aspekte als Bestandteil von Lösungen. Diese bisweilen sehr hartnäckige Grundhaltung beim Lösen von technischen oder algorithmischen Problemen führt dann auch zu Aussagen wie:

- „Das kann ich nicht abschätzen. Es braucht so lange, wie es dauert!“
- „Bei der Bearbeitung der Aufgabe bin ich auf ein wichtiges Design-Problem gestoßen, dass ich unbedingt noch lösen sollte!“
- „Kannst Du diese Behauptung irgendwie belegen?“

Hier heißt es, in unserer Kommunikation ruhig und geduldig bleiben. Es gilt, sehr methodisch vorzugehen und alle relevanten Zusammenhänge sachlich zu erläutern. Das gilt auch für fachliche Prioritäten sowie für betriebswirtschaftliche oder emotionale Aspekte. Sobald sich ein Bereich rational erschließen lässt, kann er konstruktiv eingebunden werden. Um die etwas „zeitlose“ Art im Projektkontext besser in den Griff zu bekommen ist, lassen wir die Aufgaben in kleine Teilaufgaben zerlegen und arbeiten dann mit Timeboxen (s. Kasten). Die Entwickler bzw. Kollegen können sich dann selbst steuern und ggf. den Fokus wieder auf die übergeordneten Prioritäten lenken.

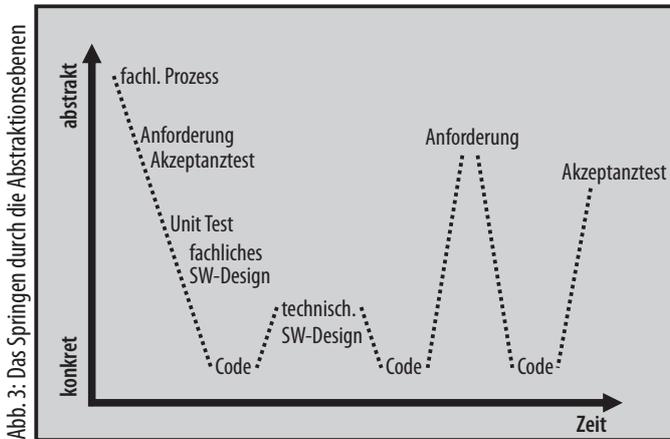
Warum ist die Betrachtung dieser beiden, eher introvertierten Präferenzen (Abb. 2) für die Zusammenarbeit mit Softwareentwicklern

so wertvoll? Weil sie so häufig vorkommen! Introvertierte Präferenzen kommen zwar im Bevölkerungsdurchschnitt etwa in einem Verhältnis 1/3 zu 2/3 zu extravertierten Präferenzen vor. In den Ingenieursdisziplinen und Naturwissenschaften, aus denen viele Quereinsteiger in die IT kommen, kehrt sich dieses Verhältnis aber in etwa um [Briggs1995]. Das erklärt bereits eine Reihe typischer Missverständnisse und Kommunikationsprobleme zwischen Entwicklern und anderen Stakeholdern in einem IT-Projekt. Die Kommunikationsmuster passen nicht zusammen. Bevor wir uns weitere Lösungen für dieses Problem anschauen, werfen wir noch einen Blick auf die beiden extravertierten Präferenzen.

Der **Wie**-Quadrant zeichnet sich eine kurze, knappe, klare und primär ergebnisorientierte Sprache und Verhaltensweise aus. Damit werden sowohl die Bedürfnisse nach greifbaren Ergebnissen wie auch nach dem Ausleben der eigenen Individualität im Gegensatz bzw. in Distanz zur Gruppe befriedigt. Hier werden Ergebnisse produziert und vor allem auch sichtbar gemacht. Damit dies schnell erfolgen kann, braucht es klare, einfache Handlungsanweisungen, die manchmal eher den Charakter eines Kochrezepts haben als eines fundierten und abgesicherten Vorgehens. Der zeitliche Bezug der Kommunikation liegt daher im Hier und Jetzt bzw. in der nahen Zukunft. Typische Beispiele für Aussagen aus dieser Präferenz heraus sind:

- „Die laufende Iteration ist wichtig! Danach sehen wir weiter.“
- „Es ist doch alles klar. Lass uns endlich anfangen!“
- „Das passt schon und die Dokumentation ist bis morgen auch ausreichend. Und ins Handbuch schaut doch eh keiner.“
- „Im Großen und Ganzen ist es fertig!“

Die Stärke dieser Präferenz liegt darin, schnell brauchbare Ergebnisse zu produzieren, Dinge ins Laufen zu bringen und Impulse zu geben. In schwierigen Situationen wird eine ausreichende Durchsetzungsstärke an den Tag gelegt. In diesem Modus wird anderen Personen Orientierung gegeben und für Klarheit im Projekt gesorgt. In der Übertreibung führt diese Präferenz zu Aktionismus und halben Lösungen, die dann leider doch nicht brauchbar sind. Dann gilt es, auf die Bremse zu treten und mit klaren Handlungsanweisungen den Fokus wieder auf die noch fehlenden Aspekte zu lenken. Ähnlich wie in Warum-Quadranten ist auch hier unsere Emotionalität gefordert. Jedoch ist für die Wie-Präferenz nicht unsere verständnisvolle Seite sondern unser Willen und unsere Durchsetzungsstärke gefragt. Das notwendige Bremsen muss mit der gleichen Energie erfolgen, wie aus der Wie-Präferenz in das vorschnelle Handeln gedrängt wird. Lange rationale Erklärungen sind dabei nicht zielführend, sondern die Veränderung des Handlungsfokus auf die nächste notwendige Aufgabe, indem klare Handlungsanweisungen erfolgen: „Stopp, wir verstehen das Problem an dieser Stelle noch nicht ausreichend, um eine Entscheidung treffen zu



können. Was kann dein erster Schritt sein, um in 2 Stunden mehr Informationen zu haben?“ Es gilt also nicht unbedingt, die Lösung vorzugeben, sondern oft reicht es aus, den Weg zu korrigieren. Im **Wohin-noch**-Quadranten treffen wir auf unsere charismatische, visionäre und kreative Seite. Die dahinter stehenden Bedürfnisse sind Freiheit und Wandel bzw. Veränderung. Dazu werden möglichst lange Optionen und Möglichkeiten offen gehalten. Sprache und Wortwahl richten sich daher auf die Zukunft, sind unverbindlich, offen und visuell („Ich sehe ...“ oder „Vor meinen Augen ergibt sich folgendes Bild: ...“). Die Prioritäten können sich schnell ändern und als emotionale Seite wird Spaß und Freude sichtbar. Typische Aussagen lauten:

- „Müssen wir das heute entscheiden?“
- „Wenn wir die neue Version hätten, könnten wir ...“
- „Das sieht spannend aus. Was können wir damit noch machen?“

Die Stärken im **Wohin-noch**-Quadranten liegen in der langfristig zukunftsorientierten Art des Denkens und der damit verbundenen Kreativität. Häufig wirkt man in dieser Präferenz charismatisch auf andere Menschen und kann diese begeistern. In schwierigen Situationen ist es auch hilfreich, dass in dieser Präferenz Probleme nicht als erdrückend wahrgenommen, sondern im Zusammenhang relativiert werden, so dass man handlungsfähig bleibt. Die Falle in diesem Modus ist das Verzetteln bis hin zum Bauen von Luftschlössern und dem Verdrängen von Schwierigkeiten. Unsere Aufgabe ist es dann, den Fokus wieder auf die Gegenwart und die Lösung der anstehenden Probleme zu richten. Dabei kann uns eine Wenn-Dann-Argumentation helfen: „Wenn wir in Zukunft die Unabhängigkeit unserer Software von einer konkreten Datenbank behalten wollen, müssen wir uns heute für eines der beiden Frameworks entscheiden!“

**Heterogene Teams** – Probleme ergeben sich dann, wenn Menschen aus unterschiedlichen Präferenzen heraus kommunizieren und dabei die entsprechenden, dahinter liegenden Bedürfnisse nicht ausreichend

adressieren. In den Diagonalen des Vier-Quadranten-Modells (Abb. 1) ist das offensichtlich. Hier liegen die Bedürfnisse diametral entgegengesetzt. Nähe zu anderen Menschen mit dem Ziel einer harmonischen Gemeinsamkeit und ethisch-moralisches Handeln (Warum?) trifft auf das Ausleben der Individualität, um sich in der Wahrnehmung anderer Menschen von der Gruppe hervorzuheben und das unbedingte Erreichen von Ergebnissen (Wie?). Aus beiden Quadranten fordern wir Wertschätzung ein: im Warum-Quadranten für das, was man für den Zusammenhalt und die Harmonie der Gruppe macht, in der eigene Bedürfnisse auch zurückgestellt werden und im Wie-Quadranten in Form von Respekt für die eigene Leistung. Ähnlich verhält es sich in der anderen Diagonalen. Im Wohin-noch-Quadranten steht die eigene Freiheit und die Veränderung im Vordergrund, aus der Was-Präferenz heraus stehen Sicherheit und Stabilität an erster Stelle. Auch das passt nicht auf den ersten Blick zusammen. Bevor wir uns weiter mit Kommunikationsproblemen befassen, werfen wir einen kurzen Blick auf eine damit im Zusammenhang stehende Problematik, die sich für viele Softwareentwickler in ihrer täglichen Arbeit stellt.

**Das Springen durch die Abstraktionsebenen** stellt eine massive Herausforderung für viele Entwickler dar. Insbesondere in agilen Projekten wird diese geistige Anforderung an die Entwickler immer stärker. Was hat es damit auf sich? Besonders im Rahmen der engen und intensiven Zusammenarbeit mit Stakeholdern werden Entwickler immer wieder aus ihrer auf die Details fokussierten Implementierung der Anforderungen herausgerissen. In Gedanken springen wir alle paar Minuten zwischen Programmierdetails, Software-Design, Architekturprinzipien, Anforderungsszenarien, Akzeptanztests und fachlichen Abläufen bzw. Prozessen hin und her. Das schematische Beispiel aus Abbildung 3 stellt ein testgetriebenes Vorgehen dar, das typischerweise mehrmals pro Stunde durchlaufen wird [Vigenschow2010]. Das ist einerseits anstrengend und fordert andererseits bei jedem Wechsel kurze Rüstzeiten. Die Vorteile einer solchen Arbeit überwiegen sicherlich den Preis, aber diese geistige Mehrarbeit ist auf jeden Fall zu leisten. Da wir in den höheren Abstraktionsebenen Details ausblenden und Entwickler in den Diskussionen mit vielen Nicht-Entwicklern eher etwas extravertierter auftreten (sollten), entspricht die tägliche Arbeit eines Entwicklers einem ständigen Wechsel zwischen meist benachbarten Quadranten. Und das fällt manchen Entwicklern leichter als anderen. Über diese Betrachtung kommen wir jetzt als nächstes zu Aspekten der Teamfähigkeit.

**Teamfähigkeit** wird schätzungsweise in über 50 % aller IT-Stellenanzeigen von Bewerbern gefordert, ohne näher auszuführen, was damit eigentlich gemeint ist. In unserem typologischen Kontext bedeutet das primär zwei Dinge. Zum einen bedarf es einer echten

Wertschätzung der Unterschiedlichkeit der Menschen in einem leistungsfähigen und flexiblen Projektteam als eigener Wert und Grundvoraussetzung für dessen besondere Leistungsfähigkeit. Zum anderen bedeutet Teamfähigkeit, über angemessene Wege zu verfügen, um mit Personen, die aus einem anderen Quadranten kommunizieren als der aktuell eigene, erfolgreich und zielgerichtet zu kommunizieren. Letzteres gilt insbesondere in Konfliktsituationen, die in Projekten zwangsläufig immer wieder zu lösen sind. Ein Team aus Softwareentwicklern wird einen vergleichsweise hohen Anteil an introvertierten Kommunikationsmustern aufweisen. Da Menschen in der Regel zwei benachbarte Präferenzen besonders stark ausgeprägt haben, finden wir also zum einen Kombinationen aus Warum?- und Was?-Verhaltensweisen. Jede andere Kombination benachbarter Quadranten ist natürlich auch möglich. In Entwicklungsteams finden wir vergleichsweise häufig noch die analytisch geprägten Was- und Wie-Verhaltensweisen in ihrer Kombination stark ausgeprägt [Briggs1995]. Teams, die mit diesen Präferenzen stark besetzt sind, sind wahre Problemlösungsexperten. Hier liegen ihre Stärken. Doch für sich alleine ist das leider wenig Erfolg versprechend. Es fehlen die übergeordneten Prioritäten und die langfristige, kraftvolle Vision, wo es eigentlich hingehen soll. Erst daraus kann abgeleitet werden, welche Probleme überhaupt wichtig und in welcher Reihenfolge zu lösen sind. Hier stoßen dann auch gute Problemlösungsteams, wenn ihre Teamfähigkeit im oben beschriebenen Sinn nicht ausreichend ausgeprägt ist, an ihre Grenzen. Dafür brauchen sie die Anwender, fachliche Experten und Entscheider auf Managementebene. Wie kommunizieren nun Nicht-ITler am besten mit den Entwicklern? In dem sich die (meist extravertierten) Nicht-ITler darauf vorbereiten, mit Menschen in einer Was-Wie- oder Warum-Was-Präferenz zu reden. Ein wertschätzender Umgang, geduldige Erläuterungen, methodisches Vorgehen und klare Entscheidungen sind hilfreich. Und stets sachlich bleiben. Kundenmitarbeiter brauchen in der Regel keine großen Emotionen bei Entwicklern aufzubauen, um sie zu motivieren. Viele Entwickler sind in großem Maße eigenmotiviert. Ihnen macht ihre Arbeit Spaß. Ganz im Gegenteil nehmen Menschen im Was-Quadranten das Aufbauschen von Information um emotionale Aspekte als Blendwerk wahr. Sie nehmen dann den fachlichen Ansprechpartner für die konkreten Details als nicht kompetent wahr und reduzieren die Kommunikation auf ein Minimum. Und genau das darf nicht passieren! Bringen Sie Verständnis auf für die schwierige Arbeit, die Entwickler zu leisten haben, wenn Sie in kürzester Zeit zwischen den Abstraktionsebenen hin- und herspringen. Jeder Mensch braucht dafür entsprechende Rüstzeiten.

**Ein grundsätzlicher Tipp bei der Kommunikation** über mehrere Quadranten hinweg kann auch hier helfen, erfolgreich zu kommunizieren. Beginnen Sie Ihre Ausführungen stets mit Aspekten

aus dem Warum-Quadranten und beantworten Sie am Anfang die Frage nach dem Sinn und Wert des Folgenden. Danach geben Sie ausreichend tiefe Informationen (Was-Quadrant), um die notwendige Sicherheit zu erreichen, um in den Wie-Quadranten zu springen. Erst jetzt zeigen sie die konkreten Handlungen auf. Beenden Sie ihre Ausführungen dann mit einem kurzen Ausblick darauf, wie sich diese Ergebnisse in das Gesamtbild einpassen und zum Erfolg beitragen. Was bedeutet das konkret? Wenn z. B. Entwickler im Rahmen eines Ergebnis-Reviews anderen Stakeholdern die neuen Features ihrer Software vorstellen, erfolgt erst eine kurze Demonstration der neuen Abläufe (Was-Quadrant), bevor die Stakeholder selbst die Software ausprobieren (Wie-Quadrant). So haben alle Stakeholder die ausreichende Sicherheit, um selbst ins Handeln bzw. Testen zu gehen.

**Kommen wir zurück zur Projektkommunikation.** Was kann nun ein Entwicklerteam machen, um sich den Stakeholdern gegenüber angemessener zu verhalten? Hier stellt sich oft die umgekehrte Situation, in der sich eher introvertierte auf eher extravertierte Menschen einstellen. Es ist genau zu klären, welche Details und Informationen für die Stakeholder wirklich wichtig und nützlich sind und wie diese Information so aufzubereiten ist, dass sie schnell erfasst werden kann. Visualisierungen und Trendanalysen wie z. B. Burndown-Charts sind typische Möglichkeiten dafür und die eben gezeigte Präsentationsreihenfolge durch die vier Quadranten ist dabei auch hilfreich. Welche Fragen gilt es zu klären? Wie soll der Ergebnistyp für die Antworten aussehen? Bis wann braucht das Team welche Entscheidung? Über diese und ähnliche Fragen wird in der Vorbereitung intern diskutiert. Mit den Ergebnissen aus dieser Diskussion kann dann häufig viel klarer und zielführender kommuniziert werden. Es ist offensichtlich, dass diese Vorbereitung ebenso wie die oben beschriebene umgekehrte Richtung Zeit braucht. Doch sie ist es wert, investiert zu werden. Je empfängerorientierter kommuniziert wird, desto schneller und erfolgreicher läuft ein Projekt.

Widmen wir uns abschließend noch einem Aspekt, der für den Projekterfolg besonders wichtig ist: der Umgang mit Konflikten. Gehen wir dazu noch einmal die vier Quadranten durch und schauen uns **die typischen Konfliktmuster** an [Satir2000]. Im Warum-Quadranten finden wir ein beschwichtigendes Vorgehen, in dem die Existenz eines Konflikts oft geleugnet wird. Auch wird Schuld sofort übernommen, ob es gerechtfertigt ist oder nicht. Die Harmonie im Team ist wichtiger. Dies kann im Extremfall bis zur Überlastung der Person mit einer ausgeprägten Warum-Präferenz führen. Aus der Was-Präferenz heraus wird ein Konflikt rein sachlich-rational betrachtet und seine starke emotionale Seite ignoriert. Gefühle stören nur das saubere Lösungskonstrukt und sind so gar nicht berechenbar. Daher fehlt der Lösung aber meist die entscheidende Komponente. Im Wie-Quadranten finden

wir oft ein eher aggressives, anklagendes Muster, in dem alle Schuld nach außen weggeschoben wird und die Lösung daher oft einseitig zu Lasten anderer geht. Und im Wohin-noch-Modus wird ein Konflikt leicht komplett ignoriert und auf andere, angenehmere Themen abgelenkt. Er ist nicht so wichtig und so kommt es auch zu keiner Lösung. Wenn wir solche oder ähnliche Verhaltensweisen erkennen, können wir über die bereits skizzierten Lösungswege einerseits die jeweiligen Stärken zu Lösung nutzen und die Schwächen minimieren, so dass eine tragfähige Lösung entstehen kann. Falls ein Konflikt bereits zu weit eskaliert ist, um ihn aus der Gruppe der beteiligten Personen heraus lösen zu können, lohnt es sich, einen Mediator hinzuzuziehen. Der Mediator steuert dann einen Prozess, in dem die Konfliktpartner gemeinsam eine tragfähige Lösung erarbeiten können.

Als abschließende Betrachtung gehen wir die vier Quadranten noch einmal durch und **formulieren ein Nein**, das entsprechend zu der jeweiligen Präferenz passt und daher mit hoher Wahrscheinlichkeit auch angenommen wird. Empfängerorientiertes Kommunizieren bedeutet eben auf keinen Fall, einer anderen Person nach dem Mund zu reden, sondern seine eigenen Interessen und Bedürfnisse angemessen zu kommunizieren. So könnte also ein Nein für die vier Präferenzen aussehen:

- Warum – Das wertschätzende Nein: „Ich weiß, dass das neue Vorgehen für Dich schmerzhaft Nachteile hat. Versuche bitte zu verstehen, dass ich so im Sinne des Projekts entschieden habe.“
  - Was – Das begründete, sachliche Nein: „Die Ergebnisse des Prototypen sind ausreichend und die aktuelle fachliche Priorität erfordert, dass wir mit Aufgabe 15 weitermachen und den Prototypen nicht weiter verbessern.“
  - Wie – Das kurze, knappe und klare Nein: „Nein!“
  - Wohin noch – Das Wenn-Dann-Nein: „Wenn wir uns unsere Flexibilität erhalten wollen, dann brauchen wir sofort das XY-Framework und dürfen uns auf keinen Fall davon ablenken lassen!“
- Interessanterweise kommt das Wort *Nein* nur in einer Variante vor.

**Fazit:** Natürlich lässt sich noch viel mehr zu Typologien und Kommunikation sagen und typologische Aspekte machen nur einen Teil der spannenden Gruppendynamik aus, mit der wir es im Kontext einer Softwareprojekte zu tun haben. Diese Typologie ist jedoch ein kraftvolles Mittel, um situativ angemessen und schnell reagieren zu können und einige typische Kommunikationsmissverständnisse zu vermeiden. Darüber hinaus ist sie ein wertvolles Instrument beim Aufbau von heterogenen Hochleistungsteams und zur Optimierung der Zusammenarbeit von Softwareentwicklern untereinander und mit den unterschiedlichen Stakeholdern. Mit diesen Werkzeugen können wir unsere eigenen Fallen, in die wir oft tappen, besser umgehen und gleichzeitig mit anderen Menschen enger und zielgerichteter

kommunizieren. Wir haben eine leicht erfassbare Basis für die Teamfähigkeit und die gegenseitige Wertschätzung von Heterogenität.

## Referenzen

- [Briggs1995] Briggs Myers: *Gifts Differing – Understanding Personality Type*, Davies-Black Publishing, 1995
- [KPMG-Studie] KPMG AG Wirtschaftsprüfungsgesellschaft: *Konfliktkostenstudie – Die Kosten von Reibungsverlusten in Industrieunternehmen*, 2009, <http://www.kpmg.de/Presse/14276.htm>
- [PM-Studie] Toth, Vigenschow, Wittwer: *Einfluss klassischer und agiler Techniken auf den Projekterfolg – 1. und 2. Ergebnisbericht 2009*, <http://www.oose.de/nuetzliches/fachliches/pm-studie/>
- [Satir2000] Satir, Banmen, Gerber, Gomori: *Das Satir-Modell*, Junfermann, 2000
- [Vigenschow2010] Vigenschow: *Testen von Software und Embedded Systems*, 2. Aufl., dpunkt, 2010
- [Vigenschow2011] Vigenschow, Schneider, Meyrose: *Soft Skills für Softwareentwickler – Fragetechniken, Konfliktmanagement, Kommunikationstypen und -modelle*, 2. Aufl., dpunkt, 2011
- [Vigenschow2012] Vigenschow, Schneider, Meyrose: *Soft Skills für IT-Führungskräfte und Projektleiter – Softwareentwickler führen und coachen, Hochleistungsteams aufbauen*, 2. Aufl., dpunkt, 2012

## Weiterführende Literatur

- Dirbach, Flückiger, Lentz: *Software entwickeln mit Verstand*, dpunkt, 2011
- Jung: *Typologie*, 7. Aufl., dtv, 2003
- Rammilmair: *IT-Verkaufsberatung in der Praxis*, dpunkt, 2012
- Riemann: *Grundformen der Angst*, 39. Aufl., Reinhardt, 2009
- Siwon: *Die menschliche Seite des Projekterfolgs*, dpunkt, 2011
- Schulz von Thun: *Miteinander reden 1-3*, rororo, 2010
- Vigenschow, Schneider: *Soft Skills für IT-Berater – Workshops durchführen, Kunden methodisch beraten und Veränderungen aktiv gestalten*, dpunkt, 2012
- Watzlawick: *Anleitung zum Unglücklichsein*, 15. Aufl., Piper, 2009
- Watzlawick: *Wie wirklich ist die Wirklichkeit?*, 11. Aufl., Piper, 2005
- Weinberg: *Die Psychologie des Programmierers*, mitp, 2004

**Seminare zu „Soft Skills in der IT“** wie beispielsweise zu Moderation, Kommunikation oder Konfliktmanagement finden Sie unter <http://www.oose.de/leistungen/seminare/offene-seminare/>

**Uwe Vigenschow** ist Berater, Trainer und Coach bei der oose Innovative Informatik GmbH in Hamburg mit den Schwerpunkten agiles Projektmanagement und Soft Skills tätig. Seit Ende der 1980er Jahre hat er Erfahrung als Softwareentwickler und -designer, Testmanager, Projektleiter und Führungskraft. Er ist verheiratet, lebt in Hamburg und spielt gerne Tennis und E-Gitarre.