



3.

Auflage



Uwe Vigerschow · Björn Schneider · Ines Meyrose

Soft Skills

für IT-Führungskräfte und Projektleiter

Softwareentwickler führen und coachen,
Hochleistungsteams aufbauen

dpunkt.verlag

Inhaltsverzeichnis

I Kontext

1 Soft Skills

1.1 Soft Skills vs. Hard Skills

1.2 Bewusstsein und Umwelt

»Ich entscheide immer alles aus dem Bauch heraus!«

»Ich weiß einfach nicht, was die alle von mir verlangen!«

»Unser Organigramm ist nicht das Papier wert ...«

2 Kommunikation

2.1 Das Metamodell der Sprache

2.2 Körpersprache

Das TALK-Modell

»Den kann ich nicht riechen!«

2.3 Metaphern - Sprachblumen pflanzen

3 Komplexe Systeme

3.1 Komplexe Systeme - komplexe Teams

»Bei uns regiert das Chaos, wir löschen nur noch Brände!«

3.2 Retrospektive Kohärenz

3.3 Gruppendynamik

»Meine Kollegen sind super, ich komme toll mit ihnen aus.«

3.4 Teamentwicklung

»Warum lernen wir nie aus unseren Fehlern?«

3.5 Aufgaben: komplex oder kompliziert?

»Das ist mir zu kompliziert, das kann ich nicht bearbeiten!«

4 Selbstorganisation und Troja-Prinzip

4.1 Selbstorganisation von Gruppen

4.2 Das Troja-Prinzip

Das Darwin'sche Prinzip

4.3 Organisation von Teams

Feedback, Supervision und Reflecting-Team

II Organisatorische Grundlagen

5 Ziele und Prioritäten

5.1 Ziele definieren

5.2 Ziele schriftlich erarbeiten

»Ziele sind gut, aber Zielvereinbarungen funktionieren nicht!«

5.3 Prioritäten setzen

6 Erfolgreiche Besprechungen

6.1 Grundstruktur von Besprechungen

»Verschwenden wir keine Zeit, kommen wir gleich zur Sache!«

6.2 Allgemeine Regeln für die Durchführung

»Eine Agenda ist überflüssig, ein guter Titel reicht!«

6.3 Eine Besprechung vorbereiten

6.4 Eine Besprechung durchführen

»Die meisten unserer Meetings sind so langweilig!«

6.5 Eine Besprechung nachbereiten

7 Zeitmanagement

7.1 Freiräume schaffen - Aufgaben delegieren

»Ich kann nichts delegieren, weil das immer in die Hose geht!«

7.2 Störungen blocken - goldene Stunde

III Entwickler führen

8 Wie funktioniert Führung?

8.1 Aspekte von Führung

»Informationen bedeuten Macht!«

8.2 Managementfolklore und Realität

8.3 Führungsebenen

9 Kontakt und Motivation

9.1 Führung und Nähe

»Nähe heißt Smalltalk. Das ist so aufgesetzt und überflüssig!«

9.2 Mythos Motivation

»Den Job vom Chef könnte ich auch machen, und zwar besser!«

10 Führung und Selbstorganisation

10.1 Wie sieht eine evolutionäre Führung aus?

10.2 Prozesskompetente Führungskraft

»Führung hat das Ziel, sich selbst überflüssig zu machen!«

10.3 Heterarchie und Hierarchie

»Flache Hierarchien sind toll, dann kann ich tun, was ich will!«

11 Entscheidungen: Über den Rubicon

11.1 Entscheidungen treffen

»Mein Chef entscheidet nichts, er sitzt Probleme aus!«

11.2 Spieltheoretische Grundlagen

11.3 Anwendungsmöglichkeiten der Spieltheorie

11.4 Man müsste mal ...: die Umsetzung

11.5 Entscheidungen in Gruppen

Das Tetralemma

IV Mitarbeiter weiterentwickeln

12 Möglichkeiten der Weiterentwicklung

12.1 Schulungen und Workshops

12.2 Mentoring und Coaching

»Als Führungskraft bin ich gleichzeitig Coach und Mentor!«

13 GROW - der Coaching-Prozess

- 13.1 GROW im Überblick
- 13.2 Stolperstein Zieldefinition

14 Das Individuum im Team

- 14.1 Grundlagen: Die Typologie weiter vertiefen
- 14.2 Einzelne Mitarbeiter weiterentwickeln
 - »Ich setze meine Mitarbeiter nur gemäß ihren Stärken ein!«

15 Weiterentwicklung nach dem Troja-Prinzip

- 15.1 Wann ist was angemessen?
- 15.2 Identität wahren
 - »Meinen Job mache ich jetzt, später jedoch etwas anderes!«
- 15.3 Eine professionelle Arbeitsethik entwickeln

V Hochleistungsteams aufbauen

16 Voraussetzungen für Spitzenteams

- 16.1 Ein Team weiterentwickeln
 - »Manche Menschen sind wie Hund und Katze. Warum nur?«
- 16.2 Was macht ein Team zu einem Spitzenteam?
 - »Mit dem/der kann ich nicht zusammenarbeiten!«
- 16.3 Das Umfeld für Hochleistungsteams
 - Critical-Chain-Planungsoptimierung*
- 16.4 Was motiviert die Teammitglieder?
 - Das Flow-Erlebnis*

17 Hochleistungsteams aufbauen

- 17.1 Die innere Struktur von Hochleistungsteams
- 17.2 Aufbau und Weiterentwicklung von Teams
- 17.3 Größe von Hochleistungsteams
 - »Zu den Kollegen im Führungskreis finde ich keinen Zugang!«

18 Gruppendynamik in Teams

18.1 Gruppen und Teams

»Manche meiner Mitarbeiter rutschen mir ins Abseits!«

18.2 Der bewusste Weg zum Hochleistungsteam

Das Wertequadrat

19 Systemische Ordnung in Teams

19.1 Was bedeutet Systemik?

19.2 Wie sind wir aufgestellt?

»Ich will zuerst meine persönlichen Unklarheiten beseitigen!«

VI Anhang

A Theoretische Hintergründe

A.1 Agile Vorgehensweisen

A.2 Die 16 Grundtypen nach Myers und Briggs

A.3 Selbstorganisation von Gruppen

A.4 Körpersprache

A.5 Stress und Stressbewältigung

A.6 Das Rubicon-Modell der Handlungsphasen

A.7 Das vereinigte Feld

B Übungen

B.1 Metapher: Führen durch Nähe

B.2 Coaching: Time Line

Danksagung

Literatur

Index

4 Selbstorganisation und Troja-Prinzip

4.1 Selbstorganisation von Gruppen

Führung wird oft in Form einer Hierarchie installiert und gelebt (Abb. 4.1 links). Es gibt schnelle und klare Entscheidungswege. Eine in einzelne, möglichst unabhängige Teile zerlegbare Aufgabe lässt sich gut verteilen. Eine Hierarchie ist also zur Bearbeitung gut unterteilbarer, *komplizierter* Aufgaben geeignet (Abb. 3.13 auf Seite 58).

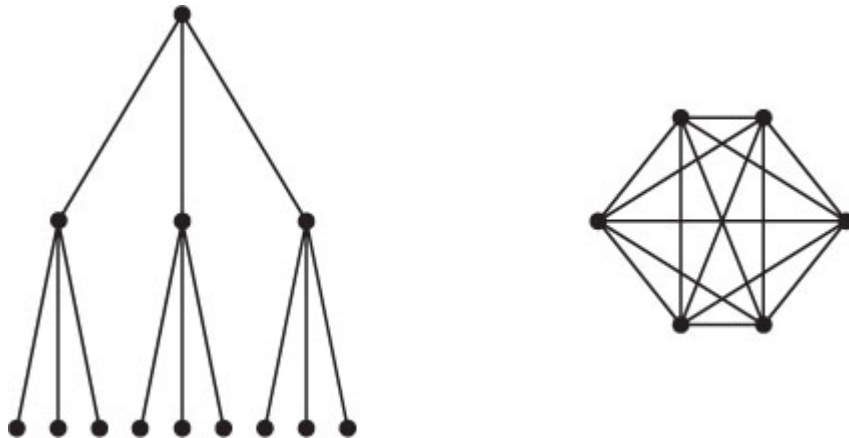


Abbildung 4.1: Die Anzahl der Kommunikationsschnittstellen in einer Hierarchie ist deutlich geringer als in einer gleichberechtigt kommunizierenden Gruppe. Eine Hierarchie mit z.B. 13 Mitarbeitern über drei Ebenen hat zwölf Kommunikationsschnittstellen zwischen zwei Personen (links), während eine 6er-Gruppe bereits 15 solcher Kommunikationsschnittstellen besitzt (rechts).

In einem gleichberechtigten Netz (Heterarchie) dagegen werden mehr Menschen eingebunden, was die Anzahl der Kommunikationswege deutlich erhöht (Abb. 4.1 rechts). Die Stärke solcher Netze liegt in der Beteiligung möglichst vieler Menschen an einer lösungsorientierten Zusammenarbeit, z.B. bei der Entwicklung kreativer, optimierter und individueller Lösungen für *komplexe* Aufgaben. So werden aus Betroffenen Beteiligte, die sich mit der Lösung identifizieren. !

Bei der Selbstorganisation von Gruppen geht es primär um die zweite Variante, das Netz. Die Effekte komplexer Systeme und der daraus resultierenden Gruppendynamik können wir nutzen, um Gruppen ausgesprochen effizient und ökonomisch zu organisieren: Eine Gruppe kann sich selbst organisieren! Da viele Gruppen dieses anspruchsvolle Konzept von vorneherein nicht ausreichend

effizient und ökonomisch beherrschen, fördern wir die Selbstorganisation mit dem Mittel der retrospektiven Kohärenz. Im Prinzip läuft der Regelkreis aus Abbildung 4.2 ab, den wir bereits in Abschnitt 3.5.2 ab Seite 61 für komplexe Situationen kennengelernt haben.

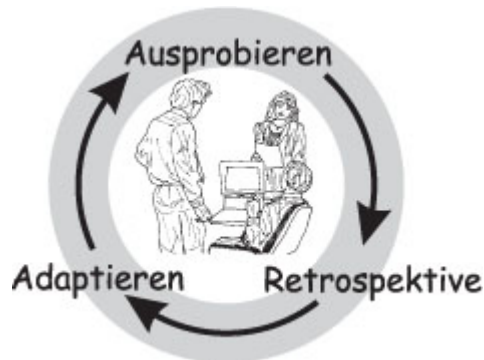


Abbildung 4.2: Regelkreis für selbstorganisierende Gruppen: Etwas wird ausprobiert und dann bewertet. Danach wird entschieden, was ggf. wie weiter umgesetzt wird.

Einige Leserinnen und Leser werden diesen Regelkreis bereits aus dem Scrum-Umfeld kennen (Anhang A.1.2). Dabei beschreibt der Regelkreis eine Sicht auf die retrospektive Kohärenz komplexer Systeme. So haben wir über die Retrospektive die Möglichkeit, die Prozesse, Regeln und sonstigen Abläufe innerhalb eines Teams zu optimieren bzw. an veränderte Rahmenbedingungen anzupassen [106].

Idealerweise brauchen wir als Projektleiter, Gruppenleiter, Scrum Master oder wie auch immer unsere Führungsrolle und Bezeichnung ausgestaltet ist, nur dafür zu sorgen, dass dieser Regelkreis zyklisch und regelmäßig abläuft, um eine Gruppe nah am Optimum zu führen. Wir nutzen das Insiderwissen und die Ideenvielfalt der Teammitglieder zur Selbstorganisation der Gruppe.

Diese hohe Ausprägung eigenverantwortlichen Arbeitens ist anspruchsvoll und stellt hohe Anforderungen an jedes einzelne Teammitglied. In der Softwareentwicklung haben wir es zwar mit hoch qualifizierten Technikern zu tun, dennoch wird selbst in diesem Umfeld bei Weitem nicht jede Gruppe in der Lage sein, sich selbst zu organisieren. Auch stellen wir mit dieser Art des Arbeitens Ansprüche an unsere Umfeldgruppen wie Fachbereiche oder Management bezüglich der Art der Zusammenarbeit, die diese nicht immer erfüllen können oder wollen. Die Arbeit in einem *echten* Team setzt einen eher höheren Anteil universalistisch veranlagter Mitarbeiter voraus als viele spezialisierte Experten. Fassen wir abschließend die Stärken und Risiken in selbstorganisierten Teams zusammen [141].

Stärken

Motivation: Die eigenen Interessen haben mehr Bedeutung, wodurch auch die Arbeit als bedeutungsvoller und sinnhafter erlebt wird. Die Aufgaben sind ganzheitlicher und abwechslungsreicher. Die einzelnen Mitarbeiter können ihre Potenziale besser entfalten.

Flexibilität: Es herrscht eine hohe Anpassungsfähigkeit an unterschiedliche äußere und innere Situationen vor. Auch wird ein möglicher Anpassungsbedarf früher erkannt, da mehr Transparenz in den Prozessen gegeben ist.

Lenkbarkeit: Gerade in komplexen Unternehmen ist der Druck zu kooperieren auf die Mitarbeiter sehr groß und damit die Anforderung, sich selbst zu lenken. Selbstorganisation ist die Konsequenz daraus.

Zeitaufwand und Kosten: Anpassungen an veränderte Umstände und andere Veränderungen verlieren viel von ihrer Reibung und laufen so schneller ab.

Risiken

Überforderung: Selbstorganisation ist ein sehr anspruchsvolles Prinzip, das hohe Ansprüche an das Sozialverhalten jedes Teammitglieds stellt. Die damit verbundenen Freiheiten können zu Ängsten bei den Beteiligten führen.

Konflikte: Das Konfliktpotenzial ist grundsätzlich höher. Daraus erwächst gerade ein Großteil des Potenzials für die kreativen und optimierten Lösungen. Insbesondere fehlen jedoch von außen vorgegebene Regeln zur Lösung von Verteilungs- und Kompetenzkonflikten, die nun individuell aufzulösen sind.

Hohe Anforderung an Führung: Die formale Führung einer selbstorganisierten Gruppe muss mit der Unsicherheit über die Kenntnis der tatsächlich aktuell geltenden Ordnung leben. Dies erfordert viel Vertrauen in das Team.

Zeitaufwand und Kosten: Aufgrund der Prozesse innerhalb einer Selbstorganisation können Lösungs- und Entscheidungsfindung deutlich länger dauern als in der auf Schnelligkeit optimierten Hierarchie. Das Prinzip des Satisficing aus Abschnitt 11.3.3 ab Seite 188 hilft bei solchen Problemen.

Die Voraussetzungen für eine funktionierende Selbstorganisation in einem Team sind also im Wesentlichen durch drei Punkte gegeben:

Toleranz den anderen Mitgliedern und ihren Meinungen und Bedürfnissen gegenüber,

Sozialkompetenz im Umgang mit den anderen Teammitgliedern und

Mut, sich darauf einzulassen.

! Wenn das Optimum nicht erreichbar ist, brauchen wir nicht zu resignieren. Einzelne Aspekte lassen sich so gut wie immer umsetzen, und in diesem Buch finden Sie dazu reichlich Anregungen. Und darum geht es: flexibel zu bleiben als Individuum und als Team und sich in kleinen Schritten permanent weiterentwickeln.

4.2 Das Troja-Prinzip

Die antike Stadt Troja weist eine Geschichte auf, die sehr gut unsere Ansicht über Veränderungsprozesse in Gruppen widerspiegelt. Jede Ausprägung ihrer Entwicklung ist auf der jeweiligen Vorgängerversion errichtet worden und nutzte z.B. noch alte funktionierende Teile. Die Stadtentwicklung erfolgte also über die Jahrhunderte hinweg schichtweise (Abb. 4.3).

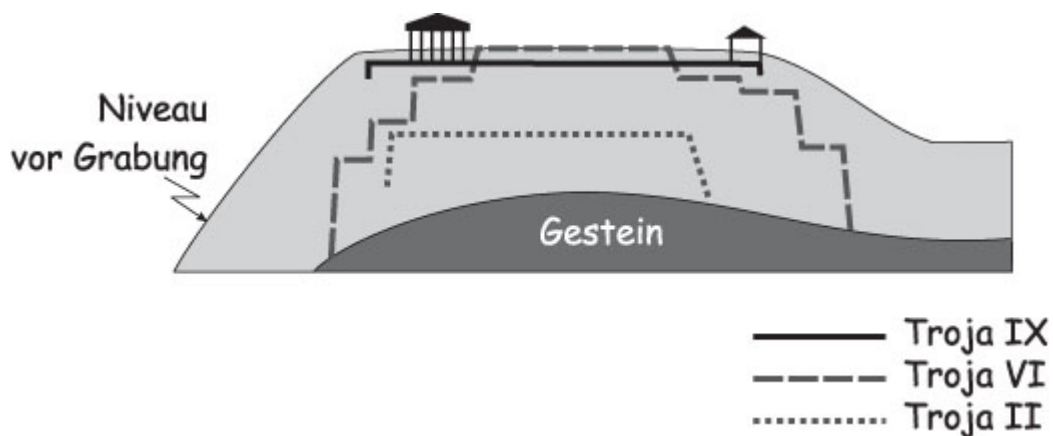


Abbildung 4.3: Das Troja-Prinzip: Neues entsteht auf und aus dem Alten. Dabei entwickelt sich das System weiter und passt sich veränderten Umweltbedingungen an. Als Metapher nehmen wir die Stadt Troja (Hisarlık), die in mindestens zehn Schichten übereinanderliegt und aufeinander aufbaut (Skizze nach Bibi Saint-Pol [141]).

Innerhalb einer Schicht erfolgte über vergleichsweise lange Zeit eine *evolutionäre* Weiterentwicklung. Durch äußere oder innere Auslöser ergab sich jedoch auch hier und da eine *revolutionäre* Entwicklung. Meist grenzte diese die


eine von der anderen archeologischen Schicht ab. Was steckt hinter den beiden ähnlichen und doch gegensätzlichen Begriffen genau [141]?

Evolution: Ein evolutionärer Prozess beschreibt eine zeitliche Entwicklung von einer einfachen hin zu einer höheren Komplexität. Dabei wird ein an die aktuellen Gegebenheiten angepasstes Optimum angestrebt. Obwohl sich diesem Ziel mal mit vielen kleinen, mal mit wenigen großen Schritten deutlich genähert wird, kann es wohl kaum je erreicht werden (siehe den Kasten auf Seite 69).

Revolution: heißt eigentlich Umwälzen bzw. drehen. Im Allgemeinen wird es als Bezeichnung für eine Veränderung, einen plötzlichen Wandel bzw. eine Neuerung gebraucht. Dies schließt auch einen gewaltsamen politischen Umsturz mit ein.

Viele Auslöser bewirken nicht per se eine evolutionäre oder eine revolutionäre Entwicklung. Bei einer Stadt wie Troja waren dies z.B. die Vorbereitungen auf Kriege oder auch innere Auslöser wie etwa das starke Wachstum der Bevölkerung. Beides kann in kleinen Schritten oder als drastische Umwälzung erfolgen. Maßgeblich ist die Länge der Vorlaufzeit innerhalb derer reagiert werden muss. Die Zerstörung in einem Krieg wird dagegen wohl stets eine revolutionäre Veränderung nach sich ziehen.

In der Entwicklung von Organisationseinheiten, die aus Gruppen von Menschen bestehen und auf Regeln basieren, finden wir Parallelen dazu. Der größte Anteil der Weiterentwicklung findet in evolutionären Verbesserungen mit kleinen Schritten unter Einbindung des bisher Erreichten statt. Dennoch lassen sich einige revolutionäre Veränderungen in großen Sprüngen nicht vollständig vermeiden. Trends wurden zu spät erkannt oder der Markt hat sich schnell in eine nicht vorhergesehene Richtung entwickelt. Der daraus resultierende Zeitdruck lässt dann eine evolutionäre Lösung oft nicht mehr zu (Abb. 4.4).

Eine weitere Parallele zur Arbeit mit Entwicklerteams ist die zeitlich begrenzte Lebensdauer eines einmal gefundenen lokalen Optimums. Nichts  ist von langer Dauer, da sich das Umfeld und die Rahmenbedingungen immer wieder ändern. Erfolgreiche Teams wie auch eine erfolgreiche Stadtentwicklung passen sich an und entwickeln sich mit. Für dieses Bild haben wir die Metapher *Troja-Prinzip* gewählt.

Fundamental bei diesem kombinierten evolutionär-revolutionären Prozess ist die Form der Führung, um ein solches lokales Optimum zu erreichen. Eine zentrale Führung legt die strategische Ausrichtung fest, um die wesentlichen

Herausforderungen zu meistern. So werden in unserer Metapher z.B. die Stadtmauern geplant, erweitert oder abgerissen und in diesem Zusammenhang die Frage beantwortet, was innen liegt und was außen, also wer dazugehört und wer nicht.

Innerhalb der Stadt bilden sich Untergruppen z.B. durch räumliche Nähe oder gemeinsam genutzte Bereiche wie Badehäuser oder Marktplätze. Diese Untergruppen bilden eine eigene Führung aus, die sich um die konkreten lokalen Probleme kümmert. Im Vergleich zu den eher langfristigen strategischen Themen handelt es hier hierbei um eher kurzfristige taktische Aufgaben. Innerhalb dieser lokalen Untergruppen können sich unterschiedliche Führungsformen etablieren wie z.B. eine hierarchische Ständeführung oder eine selbstorganisierte Struktur innerhalb einer Gruppe freier Bürger. Der wichtige Aspekt für uns ist dabei, dass sich die einzelnen Gruppen direkt oder indirekt gegenseitig beeinflussen. Wir haben es also mit einem komplexen System zu tun.

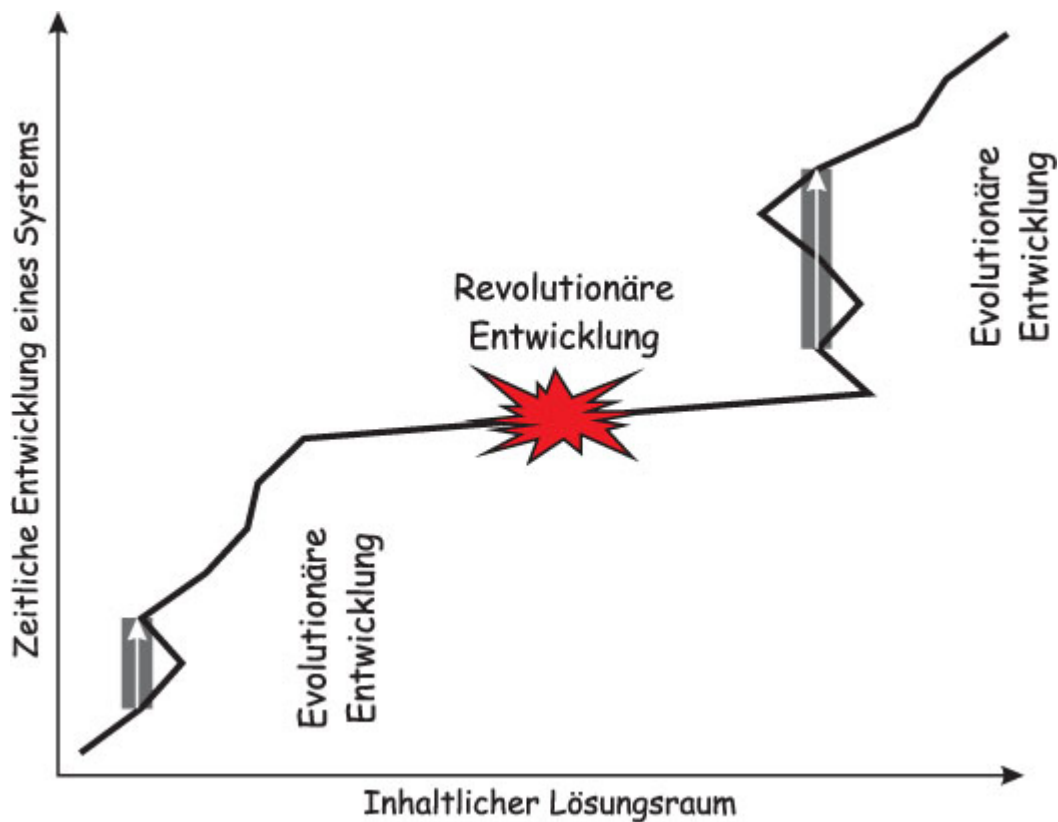


Abbildung 4.4: Das Troja-Prinzip beschreibt eine Kombination aus längeren evolutionären Anpassungs- und Veränderungsprozessen und kurzzeitigen, starken revolutionären Veränderungen. Die beiden grau hinterlegten senkrechten Pfeile markieren Bereiche, in denen die gleiche Struktur zu unterschiedlichen Zeiten realisiert wird.



Ein wichtiger Faktor kommt allerdings hinzu, der den Unterschied zum in der Natur vorkommenden evolutionären Prinzip ausmacht, das Charles Darwin (1809 – 1882) herausgefunden hat (siehe Kasten auf Seite 69). Das Troja-Prinzip beschreibt einen Weiterentwicklungsprozess in Gruppen, dessen Zusammenspiel auf sozialen Regeln beruht. Diese Regeln können explizit formuliert sein, z.B. in Form von Gesetzestexten, oder es können implizite gruppenspezifische Regeln sein, die wir in der Kindheit von den Eltern mehr oder weniger unbewusst gelehrt bekommen und übernommen haben. Auf jeden Fall gibt es Personen innerhalb der Gruppe, die eine Art Wächterfunktion ausüben, also auf die Einhaltung der Regeln achten. Die Metapher Troja-Prinzip beschreibt den dynamischen Entwicklungsprozess innerhalb von selbstorganisierten, regelbasierten komplexen sozialen Systemen.

Das Darwin'sche Prinzip

Darwins Evolutionsprinzip beruht auf dem Zusammenspiel von drei Schritten innerhalb des Evolutionsprozesses: Mutation, Selektion und Reproduktion. Dieses quasi *lebende* Versuch-und-Irrtum-Prinzip beschreibt ein zirkuläres Regelsystem. Es ist einerseits stabil genug, um seine Struktur zu bewahren, und andererseits flexibel genug, um durch Experimente zu Informationen zu gelangen, die ausreichen, sich im Rahmen seiner eigenen Dynamik weiterzuentwickeln [74].

Unter dem Fokus der Softwareentwicklung gelangen wir so zu einem inkrementell-iterativen Vorgehen. Wir entwickeln unsere Software in kleinen Schritten mit lauffähigen Zwischenergebnissen und messen diese regelmäßig im Kontext des dynamischen Umfelds an den zugrunde liegenden Anforderungen und aktuellen Prioritäten. Zu Projektbeginn erscheint dieses Vorgehen wie ein experimentelles Herantasten an ein pragmatisches Optimum. Aus der Projektretrospektive erkennen wir ein scheinbar zielgerichtetes, planvolles Handeln. Im Rahmen dieses Fortschreitens entwickeln sich einerseits die beteiligten Gruppen als Teams und andererseits auch die einzelnen beteiligten Personen weiter. Der Gruppeneffekt unterliegt den Regeln der Gruppendynamik. Unsere individuelle Weiterentwicklung ist Teil unserer persönlichen Entwicklung, die C.G. Jung als Individuation bezeichnet hat.

4.3 Organisation von Teams

4.3.1 Warum sind komplexe Systeme für uns wichtig?

Spätestens jetzt taucht die Frage nach dem *Warum* auf. Wenn das alles so schwierig und von so vielen Parametern abhängig ist, warum organisieren wir unsere Softwareentwicklung nicht einfach streng hierarchisch und sind die Probleme los?

Hierarchien haben ihre Vorteile und Stärken, ebenso selbstorganisierte Gruppen. Diese liegen nur in anderen Bereichen. Haben wir wenige Universalisten wie z.B. Softwarearchitekten im Team, die wir an zentralen Stellen einsetzen müssen, um viele Spezialisten, also z.B. Programmierer, zu dirigieren, bietet sich eine hierarchische Struktur an. Auch wenn wir kurze Reaktionszeiten in eher standardisierten Arbeitsabläufen brauchen, spielt eine Hierarchie aufgrund ihrer eher kurzen und wenigen Kommunikationswege ihre Stärken aus. Erfolgreich sind solche Strukturen in einem sich eher langsam verändernden Umfeld (Abb 4.1 auf Seite 63).

Besteht unser Team jedoch, wie in der Softwareentwicklung oft gegeben, sehr stark aus eigenmotivierten Mitarbeitern, die sich in mehreren Bereichen gut auskennen, können wir in selbstorganisierten Gruppen viel mehr Mitarbeiter an Entscheidungen und Ideenfindungen beteiligen. So erreichen wir häufig kreativere Lösungen, als wenn sich nur einige wenige Führungskräfte dieser Aufgabe stellen. Auch können solche Gruppen in einem stark dynamischen Umfeld deutlich flexibler auf Änderungen reagieren. Da wir uns im Softwareentwicklungsumfeld oft dieser Dynamik ausgesetzt sehen, kann es für uns von Vorteil sein, sich mit komplexen Systemen und selbstorganisierenden Gruppen auseinanderzusetzen.

Das praktische Problem der Selbstorganisation in Gruppen, das sich in der Realität schnell zeigt, betrifft die Führung solcher komplexen Systeme. Wie wir bereits erkannt haben, handelt es sich dabei nicht um eine hierarchische inhalts- und informationsbasierte Führung, da diese aufgrund der hohen Komplexität von Softwaresystemen nicht mehr realisierbar ist.

! Selbstorganisierte Gruppen sind regel- und wertebasiert, daher erfolgt die Führung über Regeln und das Vorleben von Werten. Auch erfolgt die Führung nicht ausschließlich durch einige wenige Führungskräfte, sondern ist bezogen auf einzelne Aspekte auf vielen Schultern verteilt. Besonders

anspruchsvoll wird diese Art der Führung, wenn das Team eine Gruppengröße von zehn Personen überschreitet.

4.3.2 Selbstorganisation in großen Gruppen

Das Thema Selbstorganisation ist für kleine Gruppen mit den geeigneten Personen gut umzusetzen. Sicherlich eignet sich nicht jede Person für ein solches Team. Diese Erfahrung haben vermutlich schon mehrere Anwender agiler Methoden wie XP oder Scrum gemacht (Anhang A.1). Wenn unser Team aus einer kleinen Anzahl motivierter und kommunikativer Entwickler besteht, funktionieren Gruppengrößen bis ca. zehn Personen ganz gut.

Die echte Herausforderung entsteht, wenn solche Systeme weiter wachsen. Es gibt nicht mehr ein zusammenhängendes Netz, das wir zu überblicken haben, sondern es bilden sich Untersysteme aus. In der Softwareentwicklung scheint dieser Übergang bei Teamgrößen zwischen zehn und zwölf Personen zu liegen [10, 33]. Diese Ballung können wir kaum noch als Einzelner überblicken (Abb. 4.5).

Management ist anscheinend im Wesentlichen die Beherrschung von Komplexität. Ein häufig zu findender, leider oft untauglicher Versuch, ein komplexes System wie aus Abbildung 4.5 zu beherrschen, entspringt der Metapher einer *Maschine* (Abb. 4.6). Es wird versucht, die Großgruppenstruktur über Hierarchien zu konstruieren. Fredmund Malik (*1944) bezeichnet dieses Paradigma des Managements als *konstruktivistisch-technomorph* [74]. Wir finden in einer mechanischen Maschine eindeutige Ursache-Wirkung-Ketten und keine Rückkopplungsschleifen. Auch mechanische Meisterleistungen sind daher im Prinzip lineare Ketten und dadurch in ihren Verhaltensmöglichkeiten gegenüber einem komplexen System stark eingeschränkt [120].

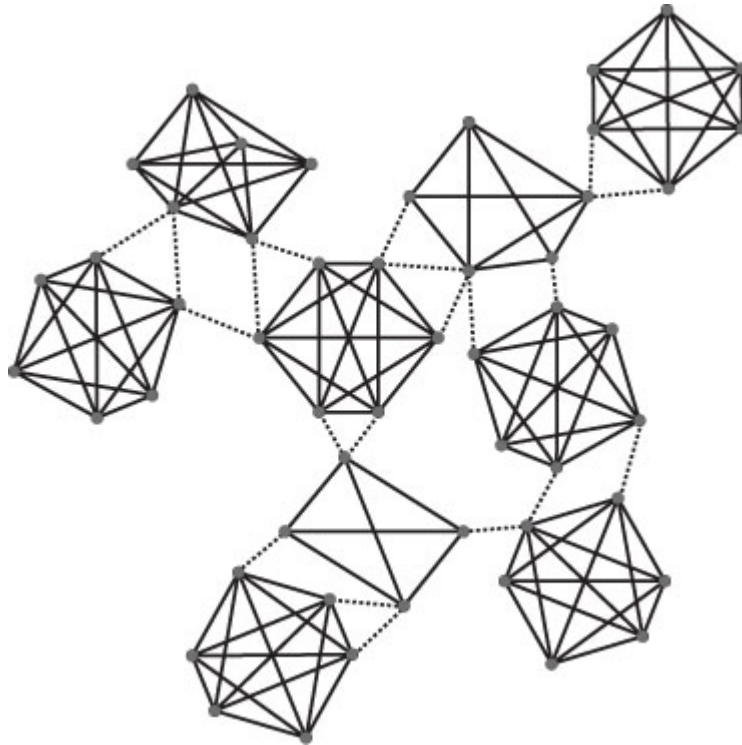


Abbildung 4.5: Wenn ein kleines einzelnes, funktionierendes Netz wächst, bildet es ab einem Grenzwert Unternetze aus. In agilen Softwareentwicklungsteams scheint diese Grenze bei ca. zehn Personen zu liegen.

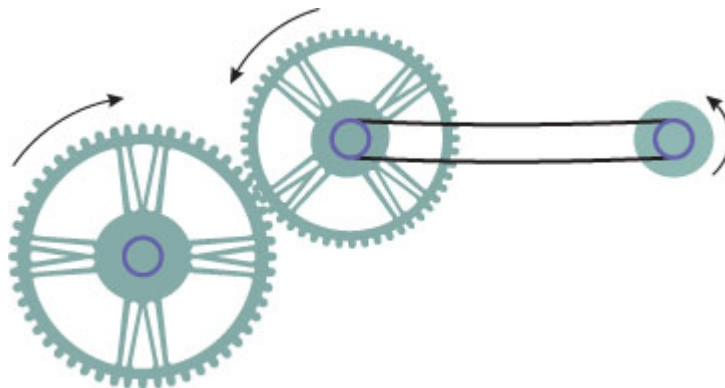


Abbildung 4.6: Der konstruktivistisch-technomorphe Managementansatz arbeitet mit der Metapher der Maschine.

4.3.3 Prozesse und Organisationsformen

In der Softwareentwicklung, die sich ja als besonders anspruchsvolle Herausforderung darstellt, stoßen wir damit schnell auf ein elementares Kommunikationsproblem: Die notwendige Kommunikation verläuft in einem hierarchischen System teilweise quer zu dieser vertikalen Ordnung entlang der Abläufe. Bereits Ende der 70er-Jahre wies Taiichi Ohno (1912 – 1990), der

Begründer der *Lean Production*, darauf hin, dass in einem idealen Produktionsprozess die Hierarchien entlang der Kommunikationswege entstehen sollten [90]. Dieses Grundprinzip wird in der IT oft ad absurdum geführt (Abb. 4.7).

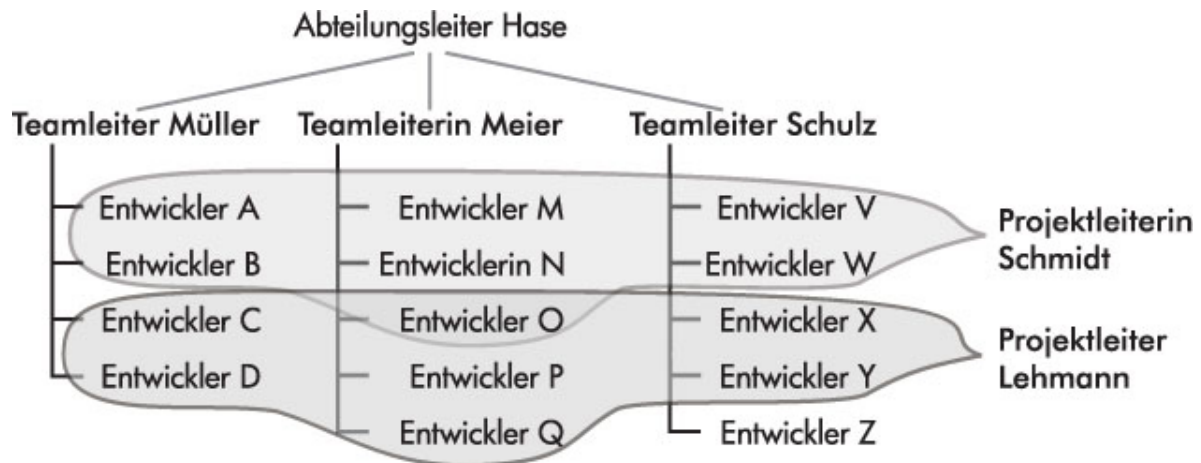


Abbildung 4.7: In einer IT-Matrixstruktur wird die senkrechte Personalführungshierarchie durch eine orthogonale Projekthierarchie erweitert. In der Praxis resultiert daraus oft ein organisatorischer und kommunikativer Mehraufwand, der einen innovativen Entwicklungsprozess hemmt. In diesem Beispiel ist Entwickler O beiden Projekten zugeordnet und Entwickler Z keinem.

Es wird versucht, eine Matrixstruktur zu implementieren, ohne sich genau über die Anforderungen und Risiken einer solchen Struktur im Klaren zu sein. Unter diesem Namen wird eine Organisationsform geschaffen, in der versäumt wird, die entsprechenden Kommunikationsschnittstellen zu etablieren. Dabei kommt oft erschwerend hinzu, dass einzelne Mitarbeiter mehreren Projekten parallel zugeordnet werden (Abb. 4.7). Die daraus resultierenden Task-Wechsel reduzieren dann die Leistungsfähigkeit dieser Projektteammitglieder weiter. Der wesentliche Vorteil dieser Organisationsform liegt unserer Erfahrung nach oft nur darin, dass zusätzliche Karrierepositionen für *verdiente* Mitarbeiter geschaffen werden. Eine solche Matrixstruktur erschwert die Kommunikationsprozesse und behindert damit das innovative Handeln des Gesamtsystems.

Doch was bedeutet *Matrix-Organisation* wirklich? Ab einer bestimmten Größe einer Organisation bildet sie einfach einen Aspekt der Realität ab. Auf den verschiedenen eher strategischen bzw. operativen Managementebenen fokussieren die Manager auf bestimmte querschnittliche Themen. Ein Entwicklungsleiter, in dessen Verantwortung die Umsetzung verschiedener Projekte fällt, ist ein Beispiel dafür. Eine einzelne Abteilung oder ein Projektteam hat eine eher ganzheitliche Sicht, aber nur auf ein oder wenige

zusammengehörige Produkte. Dadurch entstehen Kommunikationsinseln, deren Entstehung in Abbildung 4.8 schematisch dargestellt ist.

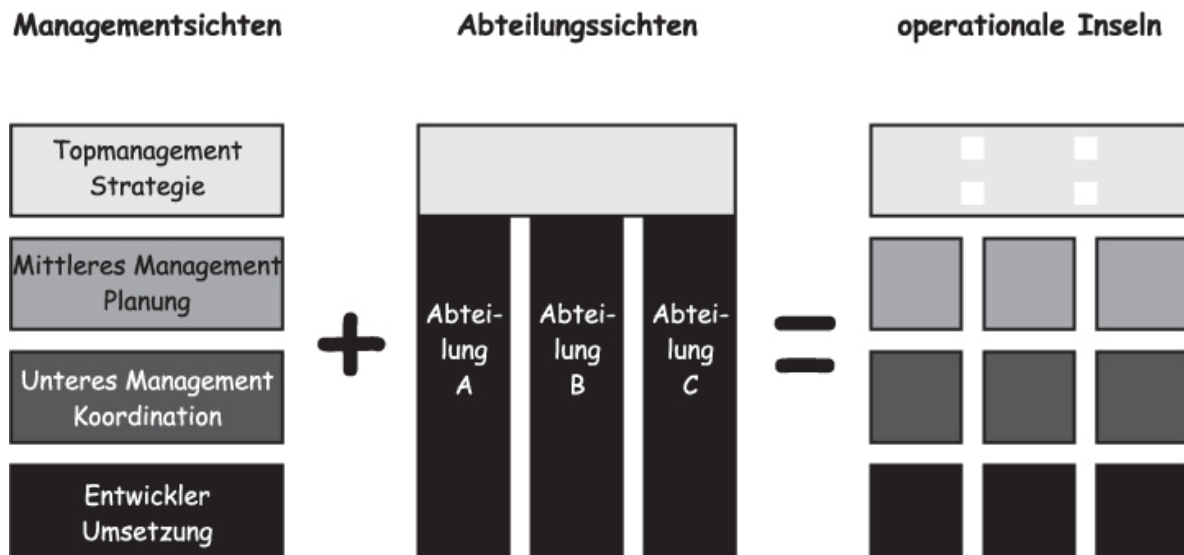


Abbildung 4.8: Auf den verschiedenen Managementebenen werden unterschiedliche strategische und operationale Sichten fokussiert (links). Auf den Abteilungsebenen wird sich dagegen auf fachliche Themen wie z.B. ein einzelnes Projekt konzentriert (Mitte). Dadurch entstehen mehr oder weniger isolierte operationale Inseln (rechts) [62].

Eine Matrix-Organisation bildet dann einfach die Realität ab. Wieso kommt es dabei so oft zu den eingangs geschilderten Problemen? Die implizit in der Organisation durch die Größe, eine minimal notwendige Hierarchie und aus der Vielfalt an Aufgaben mit den daraus resultierenden Priorisierungen existierenden Spannungen werden jetzt explizit sichtbar. Wenn diese Schwierigkeiten nicht angemessen gelöst werden, wachsen sie sich zu ernsthaften Konflikten aus. Das Beispiel aus Abbildung 4.7 zeigt dieses Problem auf. Eine Organisationsmatrix ist primär eine Kommunikationsmatrix! Matrix-Organisation bedeutet daher stets, dass Kommunikation zur Chefsache wird. Es gilt, die Kommunikationskanäle explizit aus- und umzubauen [36].

4.3.4 Lebende Organisation – kooperative Konflikte

Eine gelebte Matrix-Organisation weist eine hohe Kommunikationsrate entlang den horizontalen und vertikalen Kommunikationskanälen auf. Eine souveräne Matrix-Führungspersönlichkeit schafft mit dem Bau einer dieser operationalen Inseln einen Rahmen und Freiraum, in dem das dort beheimatete Team seine Arbeitsabläufe selbst organisiert. Wie ein Softwarearchitekt definiert und

optimiert es beim Gestalten dieser *Komponenten* deren Schnittstellen. Wenn dies gelingt, ist nicht nur eine einzelne *Insel* eine lernende, flexible Einheit, sondern auch weite Teile der Organisation. Dies kann auch bedeuten, dass die Struktur der *Inseln* regelmäßig an die sich verändernden Anforderungen angepasst wird.

Im Optimum durchdringen Organisation und Kommunikation sich gegenseitig. Wir erkennen den Qualitätsgrad oft schon an der gelebten Transparenz und den geringen Kommunikationsbarrieren gemischt mit Arbeitsrückzugsgebieten, wenn wir uns nur anschauen, wie das Arbeitsumfeld aussieht. Nichts anderes sind im Kleinen die Beschreibungen agiler Arbeitsplätze [8].

In einer *echten* Matrix-Organisation muss eine konstruktive Fehlerkultur herrschen, da sie sowohl innerhalb der Inseln als auch und vor allen Dingen an deren Schnittstellen Konflikte sichtbar macht. Dies ist so gewollt. Wichtig ist dabei, diese Konflikte kooperativ zu lösen, damit sie das gesamte Unternehmen weiterbringen. Konflikte werden konstruktiv genutzt, und es wird kein Schuldiger gesucht, sondern die aktuell beste Lösung. So ergibt sich ein Vorsprung vor der Konkurrenz oder ein aktueller Nachteil wird wettgemacht.

Eine Falle, in die besonders gerne Entwicklerteams tappen, ist es, die kooperativen Konfliktlösungen mit Harmoniesucht zu verwechseln. Genau das ist damit nicht gemeint. In einer harmoniesüchtigen Gruppe bleiben bestehende Konflikte *unter der Decke*. So sind sie jedoch nicht lösbar und hemmen die Aktions- und Reaktionsgeschwindigkeit der Mitarbeiter. Kooperative Konfliktlösungen dürfen gerne laut und engagiert ausgetragen werden. Ihr Unterschied zur häufig anzutreffenden, nichtkooperativen Konfliktlösung liegt im Ziel. Bei kooperativen Konflikten geht es um die Sache und nicht um den Sieg einer einzelnen Person. Es werden bewusst Win-win-Situationen angestrebt. Um diese zu erreichen, brauchen wir die konstruktive Mitarbeit aller Beteiligten.

4.3.5 Katalysatoren

Wenn wir das Troja-Prinzip auf große Gruppen anwenden, kommen wir von einer anderen Seite her auf ähnliche Ideen, die eher dem *systemisch-evolutionären* Managementparadigma nach Fredmund Malik zuzuordnen sind [74]. In der Chemie finden wir z.B. bei der Steuerung komplexer Reaktionen sog.

Katalysatoren. Das sind Stoffe, die den Veränderungsprozess begleiten, die Reaktionsgeschwindigkeit verändern, jedoch am Ende wieder unverbraucht aus dem Experiment hervorgehen. Chemische Katalysatoren verändern die Aktivierungsenergien in einer Reaktion. Ähnlich können wir auch menschliche Katalysatoren in ein komplexes System einbringen, die dort die notwendigen Führungsaufgaben wahrnehmen (Abb. 4.9). Konkret sehen diese Führungsaufgaben Moderation, Mentoring und Coaching vor, auf die wir später noch ausführlich zu sprechen kommen.

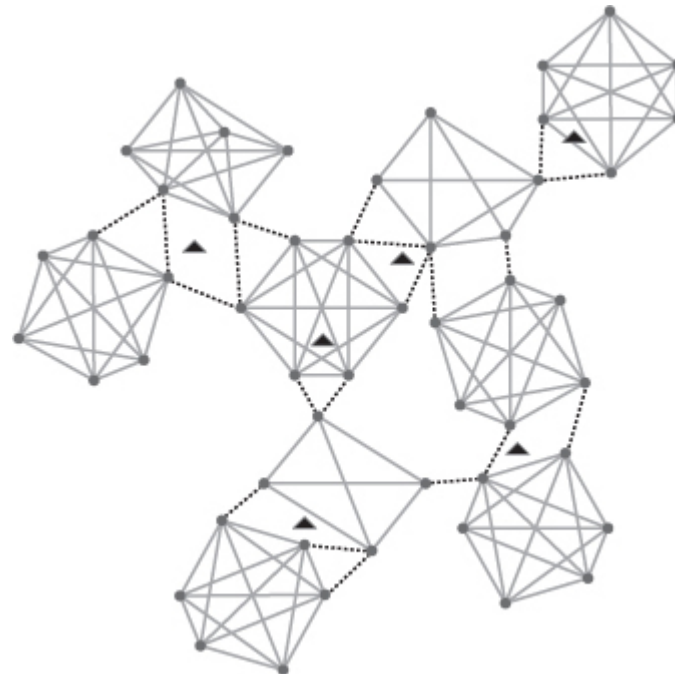


Abbildung 4.9: In großen Gruppen vernetzen sich mehrere kleinere Netze (durchgezogene Linien) zu einem dynamischen, heterogenen Gesamtgebilde (gepunktete Linien), einer dynamischen Matrix. Zur Ausbildung einer Gesamtkultur und zur Steuerung der hohen Dynamik werden spezielle Mitarbeiter als Katalysatoren in das System eingebracht (Dreiecke).

Malik fasst die dominierenden Denkmuster von konstruktivistisch-technomorphem und systemisch-evolutionärem Management wie in Tabelle 4.1 auf Seite 76 gezeigt zusammen [74]. Wie kann so eine Struktur aussehen?

Stellen wir uns eine Softwareentwicklungsabteilung vor, die als interner Dienstleister die Software für die eigene Firma entwickelt (Inhouse-Entwicklung). Sie besteht oft aus 30 bis 50 Mitarbeitern, die verschiedene Rollen wahrnehmen wie Abteilungsleiter, Teamleiter, Projektleiter, Architekt und Entwickler. Manchmal kommen noch Analytiker hinzu, doch oft wird diese Rolle zwischen Teamleiter, Projektleiter, Architekt und Entwickler aufgeteilt.

Eine streng hierarchische Struktur wird der Komplexität der miteinander vernetzten Entwicklungsvorhaben kaum gerecht werden. Also werden vielleicht

verschiedene Scrum-Teams gebildet, die über einen *Scrum-of-Scrums*-Ansatz miteinander gekoppelt werden. Die Scrum Master bilden dabei die Rolle der Katalysatoren ab. Dies löst verschiedene häufig existierende Probleme. Die fachliche Kompetenz der Mitarbeiter über das Verständnis moderner Softwarekonzepte ist oft sehr unterschiedlich verteilt. Auch die Motivation, sich damit näher zu befassen, variiert stark.

konstruktivistisch-technomorph	systemisch-evolutionär
Manager führen Menschen	↔ Manager gestalten und lenken ganze Institutionen
Manager führen wenige Personen	↔ Manager führen viele Personen
wenige Personen agieren als Manager	↔ viele Personen agieren als Manager
Manager wirken direkt	↔ Manager wirken indirekt
Management ist auf Optimierung ausgerichtet	↔ Management ist auf Steuerbarkeit ausgerichtet
Manager haben im Großen und Ganzen ausreichende Informationen	↔ Manager haben nie ausreichende Informationen
Management hat das Ziel der Gewinnmaximierung	↔ Management hat das Ziel der Maximierung der Lebensfähigkeit

Tabelle 4.1: Gegenüberstellung der dominierenden Denkmuster der Managementdefinitionen in der konstruktivistisch-technomorphen (links) und der systemisch-evolutionären Sichtweise (rechts) [74]

Noch stärker sind meistens die Defizite auf der kommunikativen Ebene und der Prozesskompetenz, Besprechungen, Workshops oder Retrospektiven effizient durchführen zu können. Die Mitarbeiter, die auf beiden Ebenen stark sind, übernehmen die Rolle des Scrum Master bzw. Katalysators und betreuen ein oder zwei Teams. Die Scrum Master stimmen sich dann zwischendurch in einem *Scrum of Scrums* ab und koordinieren gemeinsam mit dem Abteilungsleiter die Abteilung (Anhang A.1.2).



Scrum haben wir hier als eine weitverbreitete Variante dieser Idee aufgegriffen. Es gibt auch konventionellere Vorgehensweisen. Häufig werden die hoffentlich geeignet ausgewählten Team- und Projektleiter eine Katalysatorrolle übernehmen. Sie werden dann neben einer nur noch geringen inhaltlichen Mitarbeit in ihrem Team die Kommunikationskultur entwickeln und entsprechen daher weniger dem klassischen Führungsbild als einem Moderator und Coach, der Dienstleistungen gegenüber seinem Team erbringt. Auch in diesem Beispiel bedarf es einer regelmäßigen inhaltlichen und methodischen Abstimmung dieser *Teamcoaches* untereinander im Abgleich mit dem Abteilungsleiter. Hier erscheint es uns eine gute Idee, als oberste Instanz den Personalvorgesetzten zu nehmen und nicht den Hauptprojektleiter, um eine klare Trennung zwischen der langfristigen Weiterentwicklung im Sinne der Softwarestrategie und der Umsetzung von eher kurzfristigen Projektzielen zu erreichen.

Diese eher universell aufgestellten Teamcoaches benötigen einerseits regelmäßiges Feedback untereinander bis hin zu Konzepten wie Supervision durch andere Teamcoaches oder ein Reflecting-Team (siehe Definitionen im Kasten). Andererseits wird auch regelmäßiger externer Input notwendig sein, um die Teamcoaches in einem solchen komplexen Netz angemessen weiterzuentwickeln [102].

Feedback, Supervision und Reflecting-Team

Über ein **Feedback** teile ich einer anderen Person in nicht verletzender Weise konstruktiv mit, wie ihr konkretes Verhalten auf mich wirkt. Ein Feedback erfolgt nach festen Regeln, die u. a. sicherstellen sollen, dass nur *Ich*-Botschaften gesendet werden [133]. Ein kurzes Beispiel dient zur Illustration: »Du hast mich in meiner Kundenpräsentation gestern abend zweimal unterbrochen « (konkrete Wahrnehmung), »Ich fühle mich dadurch verunsichert und verliere meinen roten Faden!« (Wirkung auf mich).

Supervision bezeichnet das Angebot, unter einer fachlicher Anleitung über die Arbeit zu reflektieren. Dies kann einzeln oder in der Gruppe erfolgen.


Mit **Reflexion** bezeichnet man in der Psychologie den Prozess des Nachdenkens und Besinnens sowie die Vertiefung in die eigenen

Gedanken. Hier kann z.B. das eigene Verhalten kritisch hinterfragt oder Klarheit über die eigene Persönlichkeit gewonnen werden.

Ein **Reflecting-Team** ist eine von Tom Andersen (1936 – 2007) entwickelte Methode des Reflexionsgesprächs, in dem in unserem übertragenen Sinn zusätzlich anwesende Teamcoaches gemeinsam mit dem für ein spezielles Team verantwortlichen Teamcoach und unter Beisein des Teams über das Team reflektieren [141].

4.3.6 Zielorientierung in lebenden Organisationen

Das klingt so weit recht verlockend. Es macht sicherlich Spaß, in einem solchen Netz zu arbeiten oder als *Katalysator* Entwicklungsprozesse zu begleiten. Doch wie erfolgen die Ausrichtung und Steuerung? Wie kann die Geschäftsführung sicherstellen, dass langfristig erfolgreiche Produkte erstellt werden?

Anders als in eher hierarchischen Organisationen, in denen Varianten des *Befehl-Gehorsam-Prinzips* gelebt werden, kann eine Steuerung einer *echten*  Matrix nur über gemeinsame Ziele und Visionen erfolgen. Dabei ist die Qualität der Botschaft entscheidend. Die Motivation ergibt sich daraus, dass sich die Mitarbeiter mit der Vision und den Zielen identifizieren. Dazu braucht es eine Vision und daraus abgeleitete Ziele, die die Mitarbeiter emotional berühren [36].

Diese Emotionalität drückt sich dann in dem Wunsch aus, dabei sein zu wollen. Dazu gibt die Vision dem Handeln jedes einzelnen Mitarbeiters Sinn und schafft so eine Bedeutung für jeden Einzelnen. Daraus ergibt sich dann der hohe Aktivitätslevel unter den Mitarbeitern, den wir für die Umsetzung anspruchsvoller Projekte brauchen. Dieser Wert führt direkt zu wirtschaftlicheren Resultaten, da jeder Mitarbeiter für das Erreichen der Ziele nach den besten Mitteln und Wegen sucht. Ohne diese gemeinsame Ausrichtung verfolgt jeder Mitarbeiter seine eigenen Ziele, die mehr oder weniger zu den Unternehmenszielen passen.

Eine solche Vision zu erschaffen und daraus im obigen Sinn brauchbare Ziele abzuleiten ist kreative Arbeit und nicht durch Rhetorik zu ersetzen. Nach unserer Erfahrung ist es dabei hilfreich, oft und regelmäßig im Kontakt mit möglichst vielen Mitarbeitern zu sein, um erkennen zu können, welche eigenen

Ideen Kraft haben oder welche kraftvollen Ideen sich in den Köpfen befinden. Um die Menschen herum, die eine Vision kraftvoll verfolgen, können dann die entsprechenden Teams aufgebaut werden. Wichtig bei der Teambildung ist, dass die Team-Kristallisationskeime maßgeblichen Einfluss auf die Teambildung haben, um die Kraft und Motivation des Teams zu erhalten.

! Eine solche Organisationsform ist schwierig und für jeden Beteiligten herausfordernd. Wir sind davon überzeugt, dass sich der Aufwand in jeder Hinsicht und damit vor allen Dingen auch wirtschaftlich rechnet. Dieser Menschen-zentrierte Ansatz drückt unsere Überzeugung aus, dass Softwareentwicklung primär *People Driven Development* ist [133]: Software wird von Menschen, mit Menschen für Menschen erstellt!

Index

Symbole

6-Stufen-Fragetechnik, 217

A

A-Minus-Strategie, 156

ABC-Analyse, 90, 94

Abwägephase, 320

Advocatus Diaboli, 197

Affekt, 290

Agenda, 98, 100, 101, 103, 104, 107

 Daily Scrum, 106

Agendastruktur, 104, 105

Agent, 35

Agiles Manifest, 154, 307

Agilität, 153, 189, 190, 307

aktives Zuhören, 271

Analyse, 61

Analytische Psychologie, 7

Andersen, Tom, 77

Anerkennung, 265, 266

Anforderung, 145

Anforderungsdynamik, 60

Angebot, 145

Angst, 50, 154, 318

APM, 311

Arbeit, 239
Arbeitsethik, 241, 242
Arbeitsteilung, 254
Arbeitsverteilung, 271
Architektur, 261, 263
Ashby, William Ross, 87
Ashby'sches Gesetz, 87, 162
asynchrone Kommunikation, 145
auditiv, 22
Aufgabe, 84
Aufgabenorientierung, 127
Aufstellung, 294, 299, 300
Auftreten, 5
Aufwand, 155
Ausdrucksvermögen, 5
Ausgleich, 292–294
Ausprobieren, 40, 43, 61
Außenseiter, 284
auswählender Entwickler, 269
Autonomie, 314
Autor-Kritiker-Zyklus, 277
Autorität, 130

B

Backlog, 307
Balance, 286, 287, 324
Balanced Scorecard, 81
Bedienbarkeit, 85

Bedürfnis, 148, 265, 279, 283, 291
Bedürfnispyramide, 148
Beobachter, 131
Berater, 269, 270
Beschluss, 112
Besprechung, 97, 101, 103, 107
Best Practices, 61
beurteilend, 227
Beurteilung, 222, 224
 analytisch, 222
 gefühlsmäßig, 222
Bewertung, 322
Bewertungsphase, 322
Bewusstsein, 8, 9
Bewusstseinsfunktion, 9–11
 irrational, 10
 rational, 10
Beziehung, 96, 97, 272
Beziehungsebene, 50, 253, 255, 283
Beziehungsgeflecht, 50
Beziehungsnetzwerker, 130
Big Five, 312
Bindung, 292-294
Bindung – Ordnung – Ausgleich, 292
Break-even, 175
Briggs, Katherine, 222, 271, 312
Budgetplanung, 145
Buridan, Johannes, 178

Buridans Esel, 178, 201
Burn-out, 166
Business Motivation Model, 81

C

C-Plus-Strategie, 156
Change Agent, 20
Change Control Board, 160
Chaos, 61
Chaos-Report, 46
Chaostheorie, 35
Chesterton, Gilbert Keith, 185
Co-Mentoring, 170
Coach, 76, 161, 206
Coach the Coach, 234
Coach the Mentor, 234
Coachee, 206
Coaching, 75, 205, 206, 209, 219, 231
 extern, 232
 intern, 232
 Ziele, 213
Coaching-Prozess, 211
Collins, Jim, 134, 137
Constantine, Larry, 152, 267, 268
Controller, 269
Coping, 318
Critical Chain, 258, 259
Critical Path, 259

Csikszentmihalyi, Mihaly, 265

Cynefin, 61

D

Daily Scrum, 41, 106, 143, 147, 167, 308

Darwin, Charles, 68

Dauer, 84

Definition of Done, 308

Defizitbedürfnisse, 148

Delegation, 114, 133, 271

Denken, 9, 10, 223, 228

deterministisch, 35

Development Team, 308

Differenzierung, 281

Dilemma, 198

Dilts, Robert, 236, 323

Disney, Roy, 274

Disney, Walt, 274

Dissoziation, 180

Distanz, 167, 180

Double Bind, 29

Drama-Dreieck, 230

Drei-Schichten-Architektur, 261

Druck, 50, 166, 167, 180

Durchhaltevermögen, 254

Durchsetzungsvermögen, 48

E

E-Plus-Strategie, 156
effektiv, 102
effizient, 102
Einarbeitung, 209
Einfachheit, 61, 189
Einfluss, 51
Einfühlungsvermögen, 6
Einstellung, 11, 323
 äußere, 222, 224
 innere, 222, 223
Eisbergmodell, 18, 24, 49, 96
Eisenhower-Methode, 113, 192
Emotion, 125
Empfinden, 10
Engagement, 47, 265
entdeckender Promoter, 269
Entdecker, 269
Entscheiderrollen, 131
Entscheidung, 114, 171, 174, 176, 180, 182, 199, 222, 272
 in Gruppen, 193
 unter Unsicherheit, 174
Entscheidungsprozess, 171, 172
Entscheidungsregeln, 168
Entscheidungsverhalten, 129
Entwickler, 269
Entwicklungsgeschwindigkeit, 263
Entwicklungsprozess, 40
Entwicklungsteam, 267

ergebnisorientiert, 224
Ergebnistyp, 100, 102, 159
Erkenntnis, 43
Erscheinungsbild, 5
Erstarren, 37, 317
Erstkontakt, 28
Ertrag, 155
Erweiterbarkeit, 85
Ethik, 242
Evolution, 67
Evolutionsprinzip, 69
Experte, 175
Extraversion, 11, 222–224, 226

F

fachliche Architektur, 263
Fachwissen, 3, 4, 207
Fähigkeiten, 323
Fazit-Tendenz, 321
Feature, 41
Feature-Team, 261, 264
Feedback, 8, 76, 77, 109, 167
visuell, 44
Feeling, 222
Fehlerkultur, 43, 74, 153, 264
Feld, 302, 322
Fiat-Tendenz, 321
Firmenkultur, 127

Flexibilität, 5, 48, 65, 114, 165
Flow, 265, 266
Flucht, 37, 317
Forming, 55, 208, 255, 281
Formulierung, 100
Fotoprotokoll, 110
Fragekultur, 264
Fragenkatalog, 103
Freiheitsgrad, 168, 169, 219
Freizeit, 239
Freud, Sigmund, 18, 24, 30
Fühlen, 10, 223, 228
Führen, 327
Führer, 284
Führung, 5, 48, 63, 67, 70, 75, 76, 88, 123, 124, 127, 143, 153, 159, 254
 evolutionär, 160
Führungsaspekte, 127
Führungsaufgaben, 160
Führungsebene, 138
Führungshierarchie, 138
Führungskraft, 130, 137
Führungspersönlichkeit, 134
Führungsrahmen, 167–169, 212
Führungsrolle, 127
Führungsstil, 126, 128, 129, 210
Führungsverhalten, 127
Funktion
 Beurteilung, 222, 223

Wahrnehmung, 222, 223

Funktionalität, 84

Future Pace, 211

G

Gate, 258, 260

Geborgenheit, 166

Gefangenendilemma, 183, 185

Gefühl, 171

Gegenkopplung, 36

Gegensätze

innere, 226

Team, 250

Gegner, 284

Gehirn, 315

Gehorsam, 154

Gemeinschaft, 282, 283

geordnete Systeme, 35

Gerechtigkeit, 292

Geschäftsordnung, 50, 96, 323

Gesprächsstruktur, 219

Glaubenssätze, 249, 323

Goal setting (GROW), 211

goldene Stunde, 118, 119

Gollwitzer, Peter, 320

Großgruppe, 45, 70

Großhirn, 316

Großprojekt, 261

GROW, 211, 212
Grundbedürfnisse, 148
Gruppe, 45, 267
Gruppenarten, 282
Gruppendruck, 237
Gruppendynamik, 45, 49, 54, 69, 279, 299
gruppendynamischer Raum, 51
Gruppenentscheidung, 194
Gruppengröße, 274
Gruppenhistorie, 299
Gruppenidentität, 240, 242, 324
Gruppenkonsens, 195
Gruppenphasen, 54
Gruppenziel, 279
gustatorisch, 23

H

Hackman, J. Richard, 150
Handlungsphase, 322
Handlungsspielraum, 46
Hard Skills, 3
Harmonie, 249
Haufen, 282
Heckhausen, Heinz, 320
Herzberg, Frederick, 148
Heterarchie, 63, 164
Heterogenität, 283
Hierarchie, 63, 69, 164

Hochleistungsteam, 155, 252, 253, 264, 265, 267, 271, 276, 279, 283, 290, 292, 303
Holzfäller-Metapher, 113, 115
homo oeconomicus, 182, 183
Hygienefaktoren, 148, 150

I

Ich, 8
Ich-Ideal, 13
Identität, 53, 235, 238, 239, 250, 324
 5 Säulen, 238
 Gruppe, 236
 Säulen, 238
Identitätserleben, 236
Identitätskrise, 238
Imaginationstechnik, 180
Individuation, 69
Informant, 131
Informationsmanager, 267, 268
Informationsnetz, 132
Informationsrollen, 131
informierter Berater, 270
Initiative, 5
Inkrement, 190, 311
inkrementell, 69
inneres Team, 304
Innovator, 256, 269
Integration, 281, 310
Intellektualisierung, 290

Intention, 162, 190
Interaktionsphänomen, 124
interpersonelle Rollen, 130
Intimität, 51
Introversion, 11, 222-224, 227
Intuieren, 10
Intuition, 5, 177, 180, 222, 223, 227
Iteration, 39, 40, 44, 115, 146, 307, 311
Iterationsdauer, 260
Iterationsebenen, 41
iterativ, 69
iteratives Vorgehen, 133

J

Job Characteristics Model, 150, 151
JOHARI-Fenster, 9
Judging, 222, 224
Jung, Carl Gustav, 7, 8, 11, 69, 221, 222, 225, 226, 268, 271, 312

K 315, 317, 318

Kampf, 37, 317
Kanban, 116
Kanban-Board, 116, 119
Katalysator, 74, 137, 161
Kategorie, 61
Kernkonflikt, 51
kinästhetisch, 22
Klarheit, 107-109

Kleinhirn, 316

Klubmanager, 129

Kohärenz, 40, 41

kollektives Code-Eigentum, 310

Komfortzone, 37, 50, 161

Kommunikationsfähigkeit, 48

Kommunikation, 73, 268

- hierarchiefrei, 264

Kommunikationsinseln, 73

Kommunikationsfähigkeit, 4, 6

Kommunikationsproblem, 19

Kompensation, 155

Komplexität, 58, 60-62, 70, 133, 142, 160, 173, 265, 314

- dynamisch, 59
- statisch, 59

Komplexitätsfaktor, 60

kompliziert, 58, 61, 162

Konflikt, 13, 55, 65, 182, 243, 249, 253

Konfliktfähigkeit, 4, 6

Konfliktkultur, 263

Konfliktmanagement, 200

Konkurrenz, 276

Können, 142

Konsens, 84, 195, 198, 199

- Daumenabfrage, 201
- Fist-to-Five, 200

Konsequenz, 48

konstruktive Fehlerkultur, 43

Kontakt, 143
Kontaktpfleger, 130
Kontaktzeitpunkt, 143
Kontextmodell
 Beratung, 219
 Teamentwicklung, 247
Kontextwechsel, 259
kontrollierender Überwacher, 270
Kooperation, 55
Kooperationsfähigkeit, 4, 47, 254
Koordinationsprobleme, 276
Körpersignale, 191
Körpersprache, 24, 26,
Kosten, 84
Kotter, John P., 133
kreativer Innovator, 269
Kreativität, 5
Kreativitätskultur, 264
Krise, 94, 239
Krisenmanager, 131
Kritiker, 268
Kritikfähigkeit, 6, 47
Kybernetik, 87

L

Laissez-faire-Manager, 129
Leadership, 133
Lean Production, 72

Leiblichkeit, 239
Leistung, 239
Leistungsfähigkeit, 167
Leistungsgewinn, 55
Lenkbarkeit, 65
Lernen, 265
Lernkultur, 264
Lernprozess, 14, 39, 40
Lernzone, 38
Level-5-Unternehmensführer, 134, 137
Lewin, Kurt, 322
Limit, 116, 119
Loyalität, 290

M

Macht, 51, 125
Machtbasis

- Festlegung, 125
- Funktion, 125
- Persönlichkeit, 125
- Sache, 125

Machtstruktur, 125
Malik, Fredmund, 70, 74, 75
Management, 70
Management-Leitlinien, 132
Managementparadigma

- konstruktivistisch-technomorph, 70, 76
- systemisch-evolutionär, 74, 76

Managementrollen, 130
Managementtheorie, 130
Manager, 130, 134
Mannschaft, 282
Margerison, Charles, 268, 271
Martin, Robert C., 242
Maslow, Abraham, 148
Maßnahme, 81, 83
Matrix-Organisation, 72, 73
Matrixstruktur, 72
McCann, Dick, 268, 271
Mediation, 162, 253
Mehrabian, Albert, 24
Mehrheit, 198
Meilenstein, 41, 146, 311
Meinungsbildner, 237
Menschenbild, 153
Menschenkenntnis, 47
Mentee, 169
Mentor, 169, 209
Mentoring, 75, 169, 205, 206, 208, 209, 231, 233
Meta-Agenda, 104
Metamodell, 21
Metapher, 31-33, 61
Methodenkompetenz, 4
Mindmap, 219
Mintzberg, Henry, 130, 133
Mission, 82

Missverständnis, 145
Mitarbeiter, 284
Mitarbeiterorientierung, 128, 143
Mitarbeiterqualifizierung, 114
Mitgefühl, 47
Mitkopplung, 36
Mitwirkungsorientierung, 128
Modellierung, 61
Moderation, 4, 75, 100, 127, 194
Moderator, 76, 101, 107, 108, 161
Molcho, Samy, 25
Morgenstern, Oskar, 182
Motiv, 81, 82, 190
Motivation, 65, 77, 124, 148, 155, 260, 264, 266, 271
 extrinsisch, 149, 151
 intrinsisch, 150, 151, 155
Motivationsprobleme, 276
Motivatoren, 148, 150
Multiplikator, 234
Multiprojektmanagement, 257, 258
Multitasking, 118
Mut, 66, 298
Mutation, 69
Myers, Isabel, 222, 226, 271, 312
Myers-Briggs Type Indicator[®], 18, 221, 222, 226, 272, 312

N 327

Nachfolge, 137

Nachkontakt, 98, 109
Nachverfolgung, 112
Nähe, 67, 143, 166,
Nash, John, 181, 183
Nash-Gleichgewicht, 183
Netz, 63, 64
Neugierkultur, 264
Neumann, John von, 182
NLP, 21, 23
Norm, 46, 52, 236, 317
Norming, 55, 208, 242, 275, 281, 297

O

Oberflächenstruktur, 21
Oestereich, Bernd, 33
Offenheit, 290, 291, 314
Ohno, Taiichi, 72
Öko-Check, 211
Oldham, Greg R., 150
olfaktorisch, 23
Open-Source-Entwicklung, 155
Optimierung, 189
Optimismus, 254
Options (GROW), 212, 215
Ordnung, 292–294
Organisationsberatung, 291
Organisationseinheit, 67
Organisationsentwicklung, 264

Organisationsform, 267
Organisator, 128, 129, 133, 269, 270
Orientierung, 84, 115, 159, 166, 254
Orientierungsphase, 40
Outsourcing, 238

P

Pair Programming, 310
Panik, 37
Pareto, Vilfredo Frederico, 89
Pareto-Prinzip, 89, 90
Pascal, Blaise, 181
People Driven Development, vii, 78, 303
Perceiving, 222, 224
Performance, 85
Performing, 55, 201, 208, 242, 250, 253-255, 258, 275, 297
Perls, Frederick S., 238
Persona, 12
Personalentwicklung, 114
persönliche Kompetenz, 5
Persönlichkeitseigenschaften, 222
Persönlichkeitstypen, 225
Phantasiekultur, 264
Plan, 82
Planung, 193
Planungsphase, 321
Planungsspiel, 310
Portfoliomanagement, 257, 258, 260

Post-mortem-Analyse, 42
präaktionale Vorbereitung, 193
Präferenz, 223, 226, 302
Prämie, 152
Präsentation, 4
Primärgruppe, 45
Priorität, 84, 92, 159, 254
Problemlösekultur, 264
Problemlöser, 131
Product Backlog, 307
Product Owner, 308
Produktionsmanager, 129, 133
Produktkarton, 33
Programmierstandards, 310
Programmmanagement, 257, 258, 260
Projekt, 93, 311
Projektüberwacher, 131
Projektabschätzung, 145
Projektabschluss, 145
Projektbeginn, 145
Projekterfolg, 46
Projektleiter, 130
Projektmanagement, viii, 4
Projektvorbereitung, 144
Promoter, 269
Protokoll, 110
Prozesskompetenz, 161
Prozessoptimierer, 267, 268

Prüfung, 206
psychodynamische Ebene, 50
Pull-Prinzip, 249
Pünktlichkeit, 109
Push-Prinzip, 249, 250

Q

Qualifikation, 166, 262
Qualität, 84, 85
Qualitätsstandards, 271

R

radikale Neupositionierung, 231
Rahmenbedingung, 67
rangdynamisches Modell, 53
räumliche Trennung, 260
Rationalität, 189
Reaktion, 61
Realisierungsdauer, 84
Realitäts-Check, 211
Reality Check (GROW), 212, 213, 215, 216
Redundanz, 314
Refactoring, 39, 288, 310
Reflecting-Team, 77
Reflexion, 77
Reforming, 55, 208, 281, 297
Regelkreis, 25, 26, 35, 36, 61, 64, 142
Regeln, 67, 68, 70, 162

relative Einordnung, 92, 189
Release, 312
Repräsentant, 130
Repräsentationssystem, 22
Reproduktion, 69
Reserve, 177
Retrospektive, 40-42, 44, 64, 69, 115, 132, 133, 142, 189, 259, 264, 322
retrospektive Kohärenz, 40, 41, 45, 49, 64
Retter, 230
Return on Investment, 94, 260
Review, 40, 115, 309
Revolution, 67
Risiko, 145, 172
Risikomanagement, 101, 160
Rolle, 46, 125, 267, 268
Rollenkonflikt, 15
Rollenkonzept, 273
Rollenmodell, 53
Rubicon, 114, 190
Rubicon-Modell, 320
Rubicon-Prozess, 191
Rückkopplung, 36
 Gegenkopplung, 36
 Mitkopplung, 36
Rückkopplungskreislauf, 36

S

Sachebene, 49

Satisficing, 65, 188, 199

Scham, 154

Schätzung, 145, 146, 181

Schindler, Raoul, 53

Schlüsselqualifikationen, 4

Schmerzensgeld, 156

Schmetterlingseffekt, 35, 36

Schnittstellenkoordinator, 268

Schuld, 154

Schulz von Thun, Friedemann, 27, 286

Scrum, 60, 64, 70, 75, 76, 264, 307

Scrum of Scrums, 75, 76

Scrum Master, 75, 127, 161, 167, 170, 308

Sekundärgruppe, 45

Selbstbewusstsein, 48

Selbstdisziplin, 47

Selbstkompetenz, 5

Selbstmanagement, 4

Selbstorganisation, 62, 64–66, 68, 70, 88, 127, 137, 159, 160, 162, 164, 167, 168, 199, 264, 268, 314

- autogen, 315
- autonom, 315

Selbstreferenz, 314

Selbstreflexion, 254

Selbstregulierung, 47

Selbstverwirklichung, 148, 265

Selbstwirksamkeit, 254

Selektion, 69

Seminar, 205, 206
Sensing, 222
Sensorik, 223, 227
Sicherheit, 148, 166, 169, 172, 239, 318
Sichtbarkeit, 150
Silver Bullet, 196
Simon, Fritz B., 123
Simon, Herbert A., 182, 188
Sinn, 150, 153
slack time, 177
Smalltalk, 19, 96, 144
Sneed, Harry March, 84
Snowdon, David J., 61
Soft Skills, 3
somatische Marker, 191, 193
soziale Anerkennung, 148
soziale Beziehungen, 148
soziales Netz, 239
Sozialkompetenz, 3, 5, 47, 66
soziodynamische Ebene, 50
soziokognitive Systeme, 37
Spaß, 265
Spannungsfeld, 286
Spezialist, 59, 164, 175, 254
Spiel, 183
Spielraum, 46, 85, 93
Spieltheorie, 181, 184, 188
Spiritualität, 324

Sprachkompetenz, 47
Sprecher, 131
Sprenger, Reinhard K., 148
Sprint, 106, 307
Sprint Backlog, 307
Stabilisator, 270
Stakeholder, vii, 6, 94, 308
Stammhirn, 316
Standish Group, 46
Standup-Meeting, 41, 308, 311
Stärken, 225, 314
Statusbesprechung, 147
Stimmigkeit, 107
Storming, 55, 208, 242, 248, 253, 255, 275, 281, 297
Störung, 108, 109, 118
Störungskultur, 118
Strategie, 82, 83, 163, 211, 271
Strategietreffen, 115
Strategieverhalten
 sequenziell, 183
 simultan, 183
strategische Aufgabe, 114
Streitkultur, 264
Stress, 18, 37, 166, 318
Stressoren, 319
Structured Open Team, 267, 268, 270
Struktur, 95
Sündenbock, 44, 284

Supervision, 77
Synergie, 47
System
 komplex, viii, 35, 49, 64, 70, 291
 soziokognitiv, 37
systematischer Umsetzer, 270
Systemik, 291
systemische Grundregeln, 292, 297
systemische Ordnung, 291, 294, 298
systemisches Denken, 291

T

Tabu, 50
Tagesgeschäft, 113, 192
Taktik, 82, 83
TALK, 27
Task-Wechsel, 118
Team, 46, 54, 84, 128, 247, 250, 267, 304
Team Management Systems, 268, 270
Teamarbeit, 271
Teambildung, 78, 201, 205, 208, 228, 265, 281
Teamentwicklung, 54, 247, 250, 253, 271, 291
Teamfähigkeit, 4, 6, 47, 48, 229, 247, 298
Teamgröße, 274, 275
Teamkultur, 263
Teamleiter, 253, 267, 268
Teammanager, 128, 129, 133
Teamrad, 269-271

Teamstruktur, 261, 299
Teamuhr, 54, 247, 281
Teamzusammenstellung, 150, 263
technische Schulden, 85
Technologie, 60
Telefonkonferenz, 145
Termin, 84
Tetralemma, 196, 198
Teufelskreis, 138
Teufelsquadrat, 84, 88
Themenpool, 285
Thinking, 222, 223
Thomsett, Rob, 152, 268
Tiefenstruktur, 21
Tilgung, 22
Time Line, 328
Timebox, 108, 109, 146, 201, 311
Toleranz, 47, 66
Transaktionsanalyse, 252
Transformation, 231
Transparenz, 107, 108
Trennung, 260
Triage, 94, 115
Troja-Prinzip, 67, 68, 74, 231, 238, 282
Truppe, 282, 283
Tuckman, Bruce Wayne, 54
Typologie, 7, 16, 18, 140, 221, 225, 268, 302, 312

U

Überforderung, 65, 166
Überwacher, 270
Überzeichnung, 92
Überzeugung, 249
Umgebung, 323
UML, 59
Umsetzer, 270
Umsetzung, 190
Umwelt, 323
Umweltzustand, 172
Unbewusstes, 8
Ungewissheit, 172
Unit Test, 159, 243, 310
Universalist, 59, 164, 274
Unordnung, 61
Unsicherheit, 65, 167, 172
Unterhändler, 131
Unternehmer, 131
unterstützender Stabilisator, 270
Ursache, 61
Ursache-Wirkung-Prinzip, 35
Ursachenforschung, 44
Urteilen, 224
Utilizing Strengths, 229

V

Varietät, 87

Verallgemeinerung, 22
Veränderung, 20, 231, 234, 297, 318
Veränderungsbegleiter, 20
Veränderungsprozess, 234
Veränderungstypen, 231
verantwortliche Führungskraft, 130
Verantwortung, 47, 48, 150
vereinigtes Feld, 323
verfahrensorientiert, 224
Verfolger, 256
Verhalten, 323
Verteiler von Ressourcen, 131
Vertrauen, 194, 211, 249, 254
Verwechslungen, 50
Verzerrung, 22
Veto, 199, 200
Vier-Quadranten-Modell, 16-18, 223, 225, 282
Vision, 77, 163
Visualisierung, 4, 100, 107, 111, 181
visuell, 22
Volition, 321
Volitionsstärke, 322
Vorbildfunktion, 48
Vorgehensmodell, 31, 32
Vorkontakt, 96, 98, 107
Vorstellungsrunde, 95
Vorurteil, 29

W

Wachstumsbedürfnisse, 148

Wächterfunktion, 68

Wahrnehmen, 224

wahrnehmend, 227

Wahrnehmung, 61, 222, 224, 288

intuitiv, 222

selektiv, 29

sinnlich, 222

Wartbarkeit, 85

Warum (Typ), 59, 140

Warum-Führungskraft, 99

Was (Typ), 59, 141

Was-Führungskraft, 99

Weinberg, Gerald Marvin, 100

Weiterentwicklung, 38, 205, 229, 266

Wert, 46, 70, 126, 236, 239, 242, 265, 283, 286

Wertekonflikt, 243

Wertepaar, 287

Wertequadrat, 286, 287

Wertschätzung, 29, 154, 255, 260, 266

Westphal, Ralf, 242

What, when, who, will (GROW), 212

Whitmore, John, 211

Widerstandsfähigkeit, 319

Wie (Typ), 59, 141

Wie-Führungskraft, 99

Wiedererkennungsheuristik, 92, 189

Willenstärke, 321
WIP-Limit, 116, 119
Wir-Gefühl, 46
Wirkung, 43, 61
Wissen, 142
Wohin noch (Typ), 59, 141
Wohin-noch-Führungskraft, 99
Work in progress, 116
Workshop, 101, 208, 260

X

XP, 70, 170, 177, 189, 309
XP-Coach, 127, 161, 170

Z

Zaleznik, Abraham, 130, 133, 137
Zeitaufwand und Kosten, 65
Zeitdruck, 145
Zeitmanagement, 4, 109, 113, 192
Zertifizierung, 207
Ziel, 77, 81-84, 162, 196, 211, 212, 214, 217, 253, 255, 279, 287
 Goal (BMM), 82
 Kriterien, 217
 Objective (BMM), 82
 Planziel, 82, 98
 Vision, 82
 Zielstellung, 82, 98
Zieldefinition, 217, 219

Zielgeflecht, 81
Zielinitiierung, 321
Zielintention, 320
Zielkonflikt, 84-86
Zielpool, 280, 283, 285
Zielsetzung, 271
zielstrebigem Organisator, 270
Zielvereinbarung, 86-88
Zörner, Stefan, 220
Zugehörigkeit, 51, 292
Zwei-Faktoren-Theorie, 148
Zwischenhirn, 316
zyklische Mehrheiten, 195