

Urs Egli

## Agile Softwareprojekte: Rechtliche Qualifikation und vertragliche Umsetzung

---

Das Werk ist der zentrale Begriff des Werkvertrags. Wie präzise muss das Werk beschrieben sein, damit von einem Werkvertrag ausgegangen werden kann? Und wie verhält es sich, wenn gerade die Beschreibung des Werks einen wesentlichen Teil der Leistung ausmacht? Im Baurecht ist unbestritten, dass der Planervertrag werkvertraglichen Charakter hat. Für agile Softwareprojekte, bei welchen die Grenzen zwischen Planung und Ausführung verfließen, wird das in Frage gestellt. Zu Unrecht, wie der Beitrag aufzeigt.

---

Beitragsarten: Beiträge

Rechtsgebiete: Obligationenrecht; Werkvertrag

Zitiervorschlag: Urs Egli, Agile Softwareprojekte: Rechtliche Qualifikation und vertragliche Umsetzung, in: Jusletter 31. August 2015

## Inhaltsübersicht

1. Einleitung
2. Charakteristische Merkmale von Softwareprojekten
  - 2.1. Arten von Softwareprojekten
  - 2.2. Die Leistungsbeschreibung bei Softwareprojekten
  - 2.3. Lasten- und Pflichtenheft
  - 2.4. Klassische Projektmethoden
  - 2.5. Agile Projektmethoden
3. Mitwirkungspflichten des Kunden bei Softwareprojekten
  - 3.1. Inhalt der Mitwirkungspflichten
  - 3.2. Rechtsfolgen bei einer Verletzung der Mitwirkungspflichten
  - 3.3. Mitwirkungshandlungen bei agilen Softwareprojekten
4. Rechtliche Qualifikation agiler Softwareprojekte
  - 4.1. Einfache Gesellschaft
  - 4.2. Werkvertrag
  - 4.3. Auftrag und Personalstellung
  - 4.4. Zusammengesetzte Verträge
  - 4.5. Zurufsprojekte
5. Vertragsauslegung
6. Projektrisiken
7. Vertragsgestaltung
8. Schlussbemerkung

### 1. Einleitung

[Rz 1] Die Abgrenzung der werkvertraglichen Erfolgshaftung von der auftragsrechtlichen Sorgfalts-haftung gehört zu den zentralen Themen des Vertragsrechts. Schwierigkeiten entstehen vor allem dann, wenn zwar ein konkretes Ergebnis abzuliefern ist, sich dieses aber im Voraus nicht oder nur vage beschreiben lässt. Diese Konstellation zeigt sich im Baurecht bei den Planerverträgen,<sup>1</sup> im industriellen Bereich bei Forschungs- und Entwicklungsverträgen<sup>2</sup> und ganz besonders in der Informatik bei Verträgen über die Entwicklung und Einführung von Software.

[Rz 2] Bisher war allgemein anerkannt, dass auf Softwareprojektverträge in der Regel Werkvertrags-recht Anwendung findet.<sup>3</sup> In den letzten Jahren haben sich in der Softwareentwicklung jedoch agile Projektmethoden etabliert. Bei solchen Methoden wird einerseits auf eine spezielle, der Entwick-lungsphase zeitlich vorgeschaltete Planungsphase verzichtet. Andererseits übernimmt der Kunde eine grössere Verantwortung. Es stellt sich die Frage, ob nach agilen Methoden abgewickelte Pro-jekte nach wie vor dem Werkvertragsrecht unterstehen oder ob von einem Auftrag, von einem Personalverleihvertrag oder gar von einer einfachen Gesellschaft auszugehen ist.

[Rz 3] Davon hängen wesentliche Rechtsfolgen ab. So hat im Werkvertragsrecht ein Unternehmer erst mit der Abnahme des vereinbarten Arbeitsergebnisses durch den Besteller erfüllt und er hat mit einer Gewährleistung für dessen Mängelfreiheit einzustehen. Bei einem Auftrag hingegen ist kein Arbeitsergebnis, sondern nur ein sorgfältiges Tätigwerden geschuldet. Der Personalverleiher

---

<sup>1</sup> Zum Planervertrag und seiner Qualifikation siehe PETER GAUCH, Der Werkvertrag, 5. A., Zürich 2011, Rz 47 ff. und PETER GAUCH/PATRICK MIDDENDORF, in: Hubert Stöckli/Thomas Siegenthaler, Die Planerverträge, 3. A., Zürich 2013, Rz 1.1 ff.

<sup>2</sup> ANSLEM BRANDI-DOHRN, in: Röhrich/Graf von Westphalen/Haas (Hrsg.), Handelsgesetzbuch: HGB, 4. A. 2014, Forschungs- und Entwicklungsverträge, Rz 32.

<sup>3</sup> Für Nachweise siehe Ziffer 4.2.

ist nur für die sorgfältige Auswahl der eingesetzten Personen verantwortlich und nach dem Recht der einfachen Gesellschaft schliesslich sind die Gesellschafter zu gleichen Teilen an Gewinn und Verlust beteiligt, mithin also auch am Gelingen oder Scheitern des Projekts.

[Rz 4] Im Vertragsrecht gilt das Primat der Parteivereinbarung. Ist ein Vertrag vollständig und klar und verstösst er nicht gegen zwingendes Recht, so ergeben sich die Rechte und Pflichten der Parteien ausschliesslich aus dem Vertrag. Erst bei Unklarheiten oder Vertragslücken ist der Vertrag auszulegen oder zu ergänzen. Trotzdem ist eine Auseinandersetzung mit den gesetzlichen Vertragstypen eine unabdingbare Voraussetzung für eine gute Vertragsgestaltung.

[Rz 5] Dieser Beitrag hat erstens zum Ziel, agile Softwareprojekte im schweizerischen Recht zu verorten (Ziffern 4–6). Dies geschah bisher nur ansatzweise.<sup>4</sup> Dabei wird auch die deutsche Lehre<sup>5</sup> analysiert, die sich bereits intensiv mit diesem Thema befasst hat.<sup>6</sup> Und zweitens werden Hinweise zur vertraglichen Umsetzung gegeben (Ziffer 7), denn in der Praxis werden Vertragskonzepte gefordert, um agile Projekte rechtssicher abzuwickeln.<sup>7</sup> Zunächst wird aber auf ein paar charakteristische Merkmale von Softwareprojekten eingegangen (Ziffern 2 und 3),<sup>8</sup> welche für die rechtliche Analyse relevant sind.

---

<sup>4</sup> GIANNI FRÖHLICH-BLEULER, *Softwareverträge*, 2. A., Bern 2014, Rz 343 ff.; URSULA SURY, *Informatikrecht*, Bern 2013, 43 f.

<sup>5</sup> Allen voran das Sammelwerk von JOCHEN SCHNEIDER/FRIEDRICH GRAF VON WESTPHALEN (Hrsg.), *Software-Erstellungsverträge*, 2. A., Köln 2014; des weiteren die folgende Aufsatzliteratur: ASTRID AUER-REINSDORFF, *Feststellung der versprochenen Leistung beim Einsatz agiler Projektmethoden*, ITRB 2010, 93–95; ELKE BISCHOF, *Formen der Zusammenarbeit im IT-Projekt*, ITRB 2014, 117–118; CHRISTIAN FRANK, *Bewegliche Vertragsgestaltung für agiles Programmieren*, CR 2011, 138–144; ANKE FUCHS/CHRISTINE MEIERHÖFER/JOCHEN MORSBACH/LOUIS PAHLOW, *Agile Programmierung – Neue Herausforderungen für das Softwarevertragsrecht?* MMR 7/2012, 427–433; ARNDT HENGSTLER, *Gestaltung der Leistungs- und Vertragsbeziehungen bei Scrum-Projekten*, ITRB 2012, 113–116; FRANK A. KOCH, *Agile Softwareentwicklung – Dokumentation, Qualitätssicherung und Kundenmitwirkung*, ITRB 2010, 114–119; SASCHA KREMER, *Gestaltung von Verträgen für die agile Softwareerstellung*, ITRB 2010, 283–289; THOMAS LAPP, *Interaktion und Kooperation bei IT-Projekten*, ITRB 2010, 69–71; JOCHEN SCHNEIDER, *«neue» IT-Projektmethoden und «altes» Vertragsrecht* (zit. SCHNEIDER, *«neue» Projektmethoden*), ITRB 2010, 18–23; THOMAS SÖBBING, *Agile Projekte in der IT-rechtlichen Praxis*, ITRB 2014, 214–219; ANDREAS WITTE, *Agiles Programmieren und § 651 BGB*, ITRB 2010, 44–47.

<sup>6</sup> Die Berücksichtigung der deutschen Literatur und Rechtsprechung zur Ermittlung der unter schweizerischem Recht geltenden Regeln ist zulässig, da sich das deutsche und das schweizerische Recht bei der interessierenden Frage – der Abgrenzung des Werkvertrags vom Auftrag (im deutschen Recht als Dienstvertrag bezeichnet) – nicht wesentlich unterscheiden.

<sup>7</sup> ANDREAS OPELT/BORIS GLOGER/WOLFGANG PFARL/RALF MITTERMAYR, *Agiler Festpreis*, 2. A., München 2014, Vorwort.

<sup>8</sup> Für die Wiedergabe der Informatiksicht stützt sich dieser Beitrag auf die folgenden, von Informatikern verfassten Werke und Beiträge: HELMUT BALZERT, *Lehrbuch der Softwaretechnik: Basiskonzepte und Requirements Engineering*, 3. A., Heidelberg 2009 (zit. BALZERT *«Basiskonzepte»*); HELMUT BALZERT, *Lehrbuch der Softwaretechnik: Softwaremanagement*, 2. A., Heidelberg 2008 (zit. BALZERT *«Softwaremanagement»*); MARIO BECKER/REINHARD HABERFELLNER/RICARDO PASTERNAK/STEFAN GRÜNWALD, *IT-Wissen für Anwender*, 14. A., Zürich 2013; JÖRG DIRBACH/MARKUS FLÜCKIGER/STEFFEN LENTZ, *Software entwickeln mit Verstand*, Heidelberg 2011; STEFAN JÄHNICHEN, *Formen Agilen Programmierens*, in: Helmut Redeker/Peter Hoppen (Hrsg.), *DGRI Jahrbuch 2011*, Köln 2012, 119–125; IAN SOMMERVILLE, *Software Engineering*, 9. A., München 2012; OLIVER STIEMERLING, *Das IT-Projekt im Konflikt mit dem vertraglich definierten Regelwerk*, ITRB 2010, 289–291.

## 2. Charakteristische Merkmale von Softwareprojekten

### 2.1. Arten von Softwareprojekten

[Rz 6] Softwareprojekte sind in zwei Erscheinungsformen anzutreffen: Als Softwareentwicklungs- und als Softwareintegrationsprojekte.<sup>9</sup> Während bei der Softwareentwicklung ein Anbieter Softwarecode neu entwickelt, führt bei der Softwareintegration ein Anbieter, der nicht mit dem Hersteller identisch sein muss, eine bereits existierende Standardsoftware ein, indem er sie auf die Bedürfnisse des Kunden anpasst. Das geschieht entweder durch eine Parametrisierung<sup>10</sup> – d.h. eine Einstellung der Software anhand von vorgegebenen Einstellungsmöglichkeiten – oder durch die Entwicklung von zusätzlicher, auf die Bedürfnisse des Kunden angepasster Individualsoftware.

[Rz 7] Die Problemstellung ist ähnlich: Das abzuliefernde Resultat steht bei Vertragsabschluss nicht abschliessend fest, sondern es ist bei der Vertragserfüllung weiter zu konkretisieren. Allerdings sind die damit verbundenen Unsicherheiten bei der Softwareentwicklung grösser als bei der Softwareintegration. Dort besteht mit der Funktionalität und der Struktur der einzuführenden Standardsoftware ein Anhaltspunkt dafür, was der Kunde erwarten kann.

[Rz 8] Das Konzept der agilen Projektmethoden stammt aus der Softwareentwicklung und nicht aus der Softwareintegration.<sup>11</sup> Wenn in der Folge von Softwareprojekten die Rede ist, dann wird dabei in erster Linie an Softwareentwicklungsprojekte gedacht. Für Softwareintegrationsprojekte gelten die Ausführungen jedoch mutatis mutandis. Die Abgrenzung von Softwareentwicklung und Softwareintegration ist ohnehin fließend, da auch bei Integrationsprojekten bisweilen zusätzlich Individualsoftware entwickelt wird und umgekehrt auch bei der Softwareentwicklung vorbestehende Standardkomponenten verwendet werden.

### 2.2. Die Leistungsbeschreibung bei Softwareprojekten

[Rz 9] Ein wesentliches Merkmal von Softwareprojekten besteht darin, dass das Projektziel unscharf ist und zwar sowohl hinsichtlich der Funktionalität der Lösung, als auch hinsichtlich des technischen Lösungswegs.<sup>12</sup> Hinsichtlich der funktionalen Anforderungen ist es unscharf, weil sich der Kunde zu Projektbeginn über seine Ziele oft noch im Unklaren ist. Die Erarbeitung seiner Anforderungen stellt damit einen wesentlichen Bestandteil der Leistung selber dar.<sup>13</sup> Hinsichtlich des technischen Lösungswegs ist es unscharf, weil es für ein bestimmtes Problem häufig mehrere Lösungswege gibt, wobei einige effizienter sind als andere.<sup>14</sup> Technische Unsicherheiten bestehen insbesondere bei gesteigerten Anforderungen hinsichtlich Datenmengen, Verarbeitungsgeschwindigkeiten sowie den Anpassungsmöglichkeiten nach der Fertigstellung.<sup>15</sup> Zwischen der Informatik und anderen

---

<sup>9</sup> FRÖHLICH-BLEULER (FN 4), Rz 368 ff.

<sup>10</sup> Zur Parametrisierung BECKER/HABERFELLNER/PASTERNAK/GRÜNWARD (FN 8), 31; FRÖHLICH-BLEULER (FN 4), Rz 369 und HELMUT REDEKER, in: Schneider/Graf von Westphalen (Hrsg.) (FN 5), Rz 91.

<sup>11</sup> BECKER/HABERFELLNER/PASTERNAK/GRÜNWARD (FN 8), 100.

<sup>12</sup> DIRBACH/FLÜCKIGER/LENTZ (FN 8), 15.

<sup>13</sup> Häufig sind so grundlegende Fragen offen, wie z.B. welche Unternehmensprozesse von der Software abgebildet werden sollen oder ob zur Einführung einer Software die Änderung von bestehenden Geschäftsprozessen erforderlich ist (STIEMERLING (FN 8), 289 f.).

<sup>14</sup> DIRBACH/FLÜCKIGER/LENTZ (FN 8), 16.

<sup>15</sup> STIEMERLING (FN 8), 290.

Ingenieursdisziplinen wie dem Maschinen- oder Bauingenieurwesen besteht ein Unterschied. Zwar gibt es heute auch in der Softwareentwicklung Standards.<sup>16</sup> Es besteht jedoch mehr Spielraum und die Konsequenzen einer Abweichung von einem Standard sind andere. Eine falsch konstruierte Brücke stürzt ein. Eine schlecht entwickelte Software läuft vielleicht trotzdem, verursacht aber höhere Kosten.

[Rz 10] Die Aufgabe des Herausfindens, Analysierens und Dokumentierens der Anforderungen an die Software wird Requirements Engineering (dt. Systemanalyse) genannt.<sup>17</sup> Das Requirements Engineering ist ein wichtiges und gleichzeitig anspruchsvolles Element der Softwareentwicklung.<sup>18</sup> Beim Sammeln der Anforderungen werden sämtliche verfügbaren Informationsquellen berücksichtigt. Neben der Analyse ähnlicher Software, des Geschäftsgebrauchs und der gesetzlichen Rahmenbedingungen geschieht das vor allem durch den Einbezug der Projektbeteiligten (Endbenutzer, Management, IT-Mitarbeiter) in Form von Beobachtungen und Interviews.<sup>19</sup>

[Rz 11] Der Anbieter trägt als Spezialist die Verantwortung für das Requirements Engineering.<sup>20</sup> Er ist jedoch auf die Mitwirkung des Kunden angewiesen. Dieser verfügt über die Fachkenntnisse und muss dem Anbieter mitteilen, welche Aufgaben die Software zu erfüllen hat. Die Spezifikation der Anforderungen an die Software erfolgt deshalb letztlich durch den Kunden. Die Aufgabe des Anbieters dabei besteht in dessen Anleitung und Unterstützung.<sup>21</sup> Zum Abschluss des Requirements Engineering hat der Kunde bei klassischen Projektmethoden zu bestätigen, dass die Software durch die erarbeiteten Anforderungen korrekt beschrieben wird.<sup>22</sup>

[Rz 12] Es wird zwischen funktionalen und nichtfunktionalen Anforderungen unterschieden. Funktionale Anforderungen beziehen sich auf die konkreten Aufgaben, welche die Software zu erledigen hat, nichtfunktionale auf allgemeine Kriterien wie z.B. Zuverlässigkeit, Antwortzeiten oder Speicherbedarf.<sup>23</sup> Letztere haben einen technischen Bezug und werden deshalb im Folgenden auch als softwaretechnische Anforderungen bezeichnet.

[Rz 13] Das Requirements Engineering erfolgt immer in einem vorgegebenen Umfeld mit technischen und wirtschaftlichen Rahmenbedingungen. Nicht alles Wünschbare ist auch machbar oder sinnvoll.<sup>24</sup> Es gibt eine Wechselwirkung zwischen fachlicher Anforderung und technischer Lösung und es kann sich herausstellen, dass ein Problem, so wie es sich der Kunde vorstellt, gar nicht gelöst werden kann. Dann muss das Problem modifiziert oder die Entwicklung muss eingestellt werden.<sup>25</sup>

[Rz 14] Auch nach Abschluss des Requirements Engineering verändert sich der Leistungsgegenstand

---

<sup>16</sup> BALZERT, «Basiskonzepte» (FN 8), 17 ff. mit einer zusammengefassten Beschreibung der Softwaretechnik.

<sup>17</sup> BALZERT, «Basiskonzepte» (FN 8), 434; SOMMERVILLE (FN 8), 114.

<sup>18</sup> BALZERT, «Basiskonzepte» (FN 8), 434. Requirements Engineering verursachte nach einer Untersuchung aus dem Jahr 2001 16% der Projektkosten und nahm 39% der Projektdauer in Anspruch (BALZERT, «Basiskonzepte» (FN 8), 441.

<sup>19</sup> Zu diesem sog. Requirements Discovery siehe SOMMERVILLE (FN 8), 136 ff.

<sup>20</sup> Zum Vorgehen beim Requirements Engineering siehe SOMMERVILLE (FN 8), 132 ff.

<sup>21</sup> MICHAELA WITZEL, in: Schneider/Graf von Westphalen (Hrsg.) (FN 5), Rz G 170 ff.

<sup>22</sup> Zur Validierung der Anforderungen siehe SOMMERVILLE (FN 8), 144 f.

<sup>23</sup> SOMMERVILLE (FN 8), 116 ff.; JOCHEN SCHNEIDER, in: Schneider/Graf von Westphalen (Hrsg.) (FN 5), Rz C 144.

<sup>24</sup> Interessant ist der Hinweis von Stiemerling (FN 8), 291, wonach bei Softwareintegrationsprojekten die Definition der Kundenanforderungen nicht im «produktfreien» Raum, sondern bereits mit Blick auf die von Standardsoftware zur Verfügung gestellte Funktionalität erfolgen sollte. Andernfalls würde eine Software ausgewählt, die umfangreich angepasst werden müsse, um die Anforderungen zu erfüllen.

<sup>25</sup> BALZERT, «Basiskonzepte» (FN 8), 438.

weiter. Leistungsänderungen (Change Requests) gibt es praktisch in jedem Softwareprojekt. Dafür gibt es mit dem Änderungsmanagement (Change Request Verfahren) einen separaten Prozess.<sup>26</sup> Die Änderung von Anforderungen während dem Projekt ist einer der wichtigsten Gründe für das Scheitern von Projekten.<sup>27</sup> Änderungen verursachen Zusatzkosten und meistens ist strittig, ob diese im Leistungsumfang enthalten sind und damit unter einen Festpreis fallen oder nicht.

### 2.3. Lasten- und Pflichtenheft

[Rz 15] Die Anforderungen an eine Software werden in einer frühen Projektphase in natürlicher Sprache geschrieben. Die Spezifikation der Software nimmt jedoch im Verlauf des Entwicklungsprojektes immer technischeren Charakter an, etwa in der Form von Architektur- und Datenbankmodellen.<sup>28</sup> Wichtige Dokumente sind das Lastenheft und das Pflichtenheft.<sup>29</sup>

[Rz 16] Ein Lastenheft (engl. User Requirements Document) wird vom Kunden verfasst. Es beschreibt die Anforderungen an die Software in einer so abstrakten Form, dass der technischen Lösung nicht vorgegriffen wird, d.h. dass mehrere Anbieter sich mit unterschiedlichen Lösungsansätzen um den Auftrag bewerben können. Das Pflichtenheft (engl. System Requirements Document) stellt eine genauere Systemdokumentation dar. Es wird vom Anbieter verfasst, damit der Kunde verstehen und beurteilen kann, was die Software tun wird. Im weiteren Projektverlauf wird dann noch eine Feinspezifikation (auch Detailspezifikation genannt) erstellt.<sup>30</sup>

[Rz 17] Die Begrifflichkeit für diese Artefakte kann allerdings variieren. So ist in der Schweiz der Begriff des Lastenhefts nicht gebräuchlich. Das Dokument, welches die Anforderungen aus der Sicht des Kunden beschreibt (recte also Lastenheft), wird meistens als Pflichtenheft bezeichnet.<sup>31</sup>

### 2.4. Klassische Projektmethoden

[Rz 18] Die klassische Projektmethode ist das plangesteuerte Phasenkonzept mit den Phasen Vorstudie, Grobkonzept, Detailkonzept, Realisierung und Nutzung.<sup>32</sup> Im Bereich der Softwareentwicklung hat dieses plangesteuerte Vorgehen seine Entsprechung im Wasserfallmodell.<sup>33</sup> Letzteres gliedert die Entwicklungstätigkeit in die Phasen Anforderungsdefinition, System- und Softwareentwurf, Implementierung und Modultests, Integration und Systemtests, Betrieb und Wartung. Die Phasen sind inhaltlich und zeitlich klar voneinander abgegrenzt und jede Phase endet mit einem

---

<sup>26</sup> BALZERT, «Basiskonzepte» (FN 8), 450 f.; SOMMERVILLE (FN 8), 148 f.

<sup>27</sup> BALZERT, «Basiskonzepte» (FN 8), 439.

<sup>28</sup> Zu den Architekturmodellen siehe SOMMERVILLE (FN 8), 184 ff.

<sup>29</sup> Zur den folgenden Ausführungen zum Lasten- und Pflichtenheft siehe BALZERT, «Basiskonzepte» (FN 8), 446 f.; SOMMERVILLE (FN 8), 114 f.; WITZEL (FN 21), Rz G 153.

<sup>30</sup> Zur Feinspezifikation siehe WITZEL (FN 21), Rz G 155.

<sup>31</sup> Zur Bedeutung des Begriffs «Pflichtenheft» in der Schweiz siehe BECKER/HABERFELLNER/PASTERNAK/GRÜN WALD (FN 8), 57 und 82.

<sup>32</sup> BECKER/HABERFELLNER/PASTERNAK/GRÜN WALD (FN 8), 57; BALZERT «Softwaremanagement» (FN 8), 519 ff.; siehe auch die Projektmethodik Hermes des Informatiksteuerungsorgans des Bundes ISB, [hermes.admin.ch](http://hermes.admin.ch) (zuletzt besucht am 9. Juli 2015).

<sup>33</sup> Zum Wasserfallmodell siehe SOMMERVILLE (FN 8), 57.

Ergebnis, auf welchem in der folgenden Phase aufgesetzt wird.<sup>34</sup> Das Requirements Engineering findet ausschliesslich in der ersten Phase (Anforderungsdefinition) statt.<sup>35</sup>

[Rz 19] Im Hinblick auf die Vertragsgestaltung haben klassische Projektmethoden den Vorteil, dass ein Lastenheft vorliegt, welches die Leistung mit hinreichender Präzision beschreibt.<sup>36</sup> Besteht beim Kunden noch Unsicherheit über seine Anforderungen, so sind diese auszuräumen.<sup>37</sup> Dazu sind mit einem ersten Vertrag zuerst die Anforderungen weiter zu konkretisieren (Planung), um sie anschliessend unter einem zweiten Vertrag zu realisieren (Realisierung).<sup>38</sup>

## 2.5. Agile Projektmethoden

[Rz 20] Das plangesteuerte Vorgehen wird für Softwareentwicklungsprojekte bisweilen als zu schwerfällig empfunden.<sup>39</sup> Deshalb wurden in den vergangenen Jahren Projektmethoden entwickelt, welche die Anforderungsanalyse (Requirements Engineering) und die Entwicklung von Code als einheitliche Aufgabe betrachten.<sup>40</sup> Solche Methoden werden als agil bezeichnet.<sup>41</sup> Der wichtigste Unterschied zu den klassischen Methoden ist der Verzicht auf eine der Entwicklung vorgeschaltete Planungsphase.<sup>42</sup>

[Rz 21] Es gibt verschiedene agile Projektmethoden, z.B. exTreme Programming (XP), Scrum und Cristal, wobei jede dieser Methoden ihre Eigenheiten hat und teilweise auch in mehreren Ausprägungen existiert.<sup>43</sup> Der de-facto-Standard der agilen Softwareentwicklung ist heute Scrum.<sup>44</sup>

[Rz 22] Unter Scrum wird das Entwicklungsprojekt in mehrere Sprints von je ca. 30 Tagen Länge aufgeteilt.<sup>45</sup> Die Anforderungen an die Software werden aus der Sicht der Nutzer in Form von User Stories formuliert und im Product Backlog gesammelt.<sup>46</sup> Der Product Backlog wird ständig weiter

---

<sup>34</sup> Eine übersichtliche Beschreibung der einzelnen Phasen mit ihren Ergebnissen findet sich bei BECKER/HABERFELLNER/PASTERNAK/GRÜNWARD (FN 8), 56 ff.

<sup>35</sup> SOMMERVILLE (FN 8), 57.

<sup>36</sup> Siehe BECKER/HABERFELLNER/PASTERNAK/GRÜNWARD (FN 8), 57 und 82 ff., welche den Abschluss der Beschaffungsverträge erst auf der Basis des Pflichtenhefts vorsehen.

<sup>37</sup> STIEMERLING (FN 8), 290.

<sup>38</sup> Zur vertraglichen Zuteilung des Projekts in Planung und Realisierung ausführlich SCHNEIDER (FN 23), Rz C 1 ff.

<sup>39</sup> BECKER/HABERFELLNER/PASTERNAK/GRÜNWARD (FN 8), 100; SOMMERVILLE (FN 8), 88.

<sup>40</sup> SOMMERVILLE (FN 8), 92 f.

<sup>41</sup> Der Begriff geht auf das «Manifesto for Agile Software Development» (Agiles Manifest) von 2001 zurück (WITZEL (FN 21), Rz H 61). Ausführlich zum agilen Manifest siehe OPELT/GLOGER/PFARL/MITTERMAYR (FN 7), 6 ff.

<sup>42</sup> FUCHS/MEIERHÖFER/MORSBACH/PAHLOW (FN 5), 432 bezeichnen die Planung bei agilen Projekten als nicht abtrennbaren, integrierten Teil der Entwicklung, wobei diese Verschränkung von Planung und Entwicklung den Kern des Vertrags ausmache. Siehe auch REDEKER (FN 10), Rz D 108 und SCHNEIDER (FN 23), Rz C 119.

<sup>43</sup> BECKER/HABERFELLNER/PASTERNAK/GRÜNWARD (FN 8), 103 ff.

<sup>44</sup> OPELT/GLOGER/PFARL/MITTERMAYR (FN 7), 10.

<sup>45</sup> Zu der im Folgenden zusammengefassten Darstellung von Scrum siehe BECKER/HABERFELLNER/PASTERNAK/GRÜNWARD (FN 8), 104 f.; OPELT/GLOGER/PFARL/MITTERMAYR (FN 7), 10 ff.; SCHNEIDER (FN 23), Rz C 131 ff.; SÖBBING (FN 5), 214 ff.; SOMMERVILLE (FN 8), 104 f. sowie die bei Wikipedia.de wiedergegebene Beschreibung, <http://de.wikipedia.org/wiki/Scrum> (zuletzt besucht am 9. Juli 2015).

<sup>46</sup> Eine User Story ist in Anwendersprache formuliert und beschreibt mit einer Überschrift und in wenigen Sätzen eine Anforderung an die Software (AUER-REINSDORFF (FN 5), 94). Die folgenden zwei Beispiele für handschriftlich auf Zettel geschriebene User Stories finden sich bei BALZERT, «Basiskonzepte» (FN 8), 498. Beispiel 1: Interessenten und Kunden müssen sich über das Web über Seminare und Veranstaltungen austauschen kön-

entwickelt. Dabei handelt es sich jedoch nicht um eine Anforderungsspezifikation im herkömmlichen Sinn.<sup>47</sup> Darauf wird unter agilen Methoden verzichtet. User Stories werden in eine sinnvolle Reihenfolge gebracht und auf die einzelnen Sprints aufgeteilt. In einem Sprint wird die Software gleichzeitig spezifiziert, entwickelt und getestet. Auf diese Weise wird der Produkt Backlog laufend abgearbeitet, bis die fertige Software vorliegt.

[Rz 23] Ein weiteres wesentliches Merkmal ist, dass bei agilen Methoden auf eine formale Programmdokumentation im herkömmlichen Sinn verzichtet wird. Ob der Kunde auch bei agilen Projekten nicht trotzdem Anspruch auf eine Dokumentation hat, ist aus rechtlicher Sicht umstritten.<sup>48</sup> Aber auch aus Informatiksicht wird eine fehlende Dokumentation im Hinblick auf die Wartbarkeit und Integrationsfähigkeit der Software als grosser Nachteil empfunden.<sup>49</sup>

[Rz 24] Anstelle der Prozesse legt Scrum das Gewicht auf Interaktion und Flexibilität. Es gilt das Leitmotiv: Menschen statt Prozesse. Ein strukturelles Gegengewicht wird durch eine klare Rollenbeschreibung erreicht. Die oberste Planungsverantwortung liegt beim Product Owner. Dieser ist zuständig für die Anforderungsdefinition, den Auslieferungszeitpunkt und die Budgetierung.<sup>50</sup> Die Umsetzung erfolgt im Entwicklungsteam, welches durch den Scrum Master koordiniert wird.

[Rz 25] Und schliesslich ist die starke Einbindung des Kunden in den Prozess der Softwareentwicklung zu erwähnen.<sup>51</sup> Dies gilt nicht nur für die Anforderungsspezifikation, sondern auch für die Prüfung der Arbeitsergebnisse, welche laufend zu erfolgen hat.<sup>52</sup> Der Kunde soll sich über das Arbeitsergebnis frühzeitig ein Bild machen können, damit allfällige Änderungen wiederum in den Entwicklungsprozess einfließen können.<sup>53</sup>

[Rz 26] Agile Projektmethoden werden in der Softwareentwicklung eingesetzt;<sup>54</sup> und dort häufiger in firmeninternen Entwicklungsprojekten als in Kundenprojekten.<sup>55</sup> Das Prototyping als weitere, nicht plangesteuerte Methode eignet sich auch für Integrationsprojekte.<sup>56</sup> In der Form des explorativen Prototyping dient es als Entwurfshilfe zur effizienten Kommunikation zwischen Kunden und Anbietern. In der Form des evolutionären Prototyping ermöglicht es die schnelle Implementation einer Lösung, die nicht vollständig ist und die laufend verbessert wird.

[Rz 27] Welche Projektmethode sich eignet, ist eine technisch-kommerzielle Entscheidung. Aus softwaretechnischer Sicht wird der Einsatz agiler Entwicklungsmethoden eher bei kleinen Systeme-

---

nen; Priorität: 4; Aufwand: 20 Stunden. Beispiel 2: Edit SR Details Screen: add «cancel»-button to undo changes and return to previous page; check that browser returns to correct previous page; estimated: 2 h; actual.

<sup>47</sup> Da sich das Product Backlog laufend ändert, ist es sowohl als Lasten- wie auch als Pflichtenheft ungeeignet (HENGSTLER (FN 5), 114).

<sup>48</sup> KOCH (FN 5), 117; KREMER (FN 5), 284; SCHNEIDER, «neue» Projektmethoden (FN 5), 19; eine Dokumentationspflicht fordernd REDEKER (FN 10), Rz D 111.

<sup>49</sup> SOMMERVILLE (FN 8), 91; JÄHNICHEN (FN 8), 125 bezeichnet den Verzicht auf eine Dokumentation sogar als «leichtfertig».

<sup>50</sup> Zu den umfassenden Kompetenzen des Product Owners unter Scrum siehe HENGSTLER (FN 5), 115; JÄHNICHEN (FN 8), 123 und SÖBBING (FN 5), 216.

<sup>51</sup> Zur «partizipativen Entwicklung» siehe JÄHNICHEN (FN 8), 122 f.; zur starken Einbindung des Kunden auch LAPP (FN 5), 70 und SCHNEIDER, «neue» Projektmethoden (FN 5), 18 f.

<sup>52</sup> KOCH (FN 5), 118.

<sup>53</sup> SÖBBING (FN 5), 216.

<sup>54</sup> BECKER/HABERFELLNER/PASTERNAK/GRÜNWARD (FN 8), 100.

<sup>55</sup> JÄHNICHEN (FN 8), 124.

<sup>56</sup> Zum Prototyping siehe BECKER/HABERFELLNER/PASTERNAK/GRÜNWARD (FN 8), 100 f.; SCHNEIDER (FN 23), Rz C 120 ff.

men gesehen.<sup>57</sup> Agile Methoden wurden für kleine Programmiererteams entwickelt, die idealerweise im gleichen Raum arbeiten und sich informell austauschen können.<sup>58</sup> Hingegen wird es als schwierig erachtet, agile Methoden in grossen Projekten einzusetzen.<sup>59</sup> Auch wird in Frage gestellt, ob agile Methoden zur Entwicklung kritischer Softwaresysteme (z.B. für Systeme mit gesteigerten Anforderungen an die Betriebs- und Informatiksicherheit) eingesetzt werden können.<sup>60</sup> Gegen agile Methoden sprechen auch Bedenken bezüglich Wartbarkeit; dies aufgrund der fehlenden Dokumentation.<sup>61</sup> Nicht geeignet sind agile Methoden zudem für stark standardisierte IT-Projekte wie die Einführung von ERP-Systemen.<sup>62</sup> Erwähnenswert ist der Vorschlag eines zweistufigen Vorgehens, wobei in einem ersten Schritt die Anforderungen mit agilen Methoden konkretisiert werden, um auf der Basis der gewonnenen Erkenntnisse in einem zweiten Schritt ein Projekt mit festen Preisen und Terminen zu realisieren.<sup>63</sup> Interessanterweise entspricht dies der herkömmlichen Trennung von Planung und Realisierung.<sup>64</sup>

[Rz 28] Aus rechtlicher Sicht überwiegen die Bedenken.<sup>65</sup> Das überrascht nicht, denn die klassischen Projektmethoden entsprechen dem strukturorientierten Denken der Juristen besser als das stark von Interaktion geprägte und dynamische agile Vorgehen.

[Rz 29] Zusammenfassend kommen für agile Methoden Projekte in Frage, bei denen kumulativ die folgenden Voraussetzungen erfüllt sind: (1) die Anforderungen stehen nicht genau fest, (2) der Auftraggeber ist in der Lage, intensiv mitzuarbeiten und (3) das Projekt ist zeitlich und inhaltlich überschaubar.<sup>66</sup>

[Rz 30] Nicht sinnvoll sind agile Methoden bei Projekten, bei denen alternativ eines der folgenden Merkmale vorliegt: (1) Qualitätsvorgaben, die zwingend und dokumentierbar erfüllt werden müssen, insb. bei kritischen Softwaresystemen im medizinischen oder technischen Bereich, (2) klar fixierte Termine und Kosten, (3) lange Entscheidungswege beim Kunden, (4) Kunden, die an Pflichthefte gewöhnt sind sowie (5) wenig Kontakt zwischen Anbieter und Kunde.<sup>67</sup>

---

<sup>57</sup> BECKER/HABERFELLNER/PASTERNAK/GRÜNWARD (FN 8), 107 f.; SOMMERVILLE (FN 8), 106 ff.

<sup>58</sup> SOMMERVILLE (FN 8), 106.

<sup>59</sup> SOMMERVILLE (FN 8), 108.

<sup>60</sup> SOMMERVILLE (FN 8), 87, 90 und 106 ff.

<sup>61</sup> SOMMERVILLE (FN 8), 91.

<sup>62</sup> OPELT/GLOGER/PFARL/MITTERMAYR (FN 7), 32.

<sup>63</sup> STIEMERLING (FN 8), 290.

<sup>64</sup> Siehe SCHNEIDER (FN 23), Rz 1 ff.

<sup>65</sup> Ausführlich SCHNEIDER (FN 23), Rz C 113 ff., der bei agilen Methoden typische Projektsünden (unklare Vertragstypologie, fehlendes oder unvollständiges Pflichtenheft) erkennt; KOCH (FN 5), 118 wirft ein, dass der vom Kunden bei agilen Projekten zu betreibende, grosse Aufwand möglicherweise besser in die Erarbeitung eines Lastenhefts investiert würde.

<sup>66</sup> KREMER (FN 5), 285; KOCH (FN 5), 119; SCHNEIDER (FN 23), Rz C 118 sowie WITZEL (FN 21), Rz H 72 mit einer Entscheidungsmatrix.

<sup>67</sup> Siehe die Zitate zu FN 66.

### 3. Mitwirkungspflichten des Kunden bei Softwareprojekten

#### 3.1. Inhalt der Mitwirkungspflichten

[Rz 31] Aus rechtlicher Sicht ist es eine Selbstverständlichkeit, dass bei Softwareprojekten auch der Kunde handeln und zum Erfolg beitragen muss.<sup>68</sup> Der Aufwand des Kunden dafür beträgt unabhängig von der gewählten Projektmethode erfahrungsgemäss zusätzliche 30–50% der externen Projektkosten.<sup>69</sup> Dies gilt bereits für nach klassischen Methoden abgewickelte Projekte.<sup>70</sup>

[Rz 32] Die wichtigste Mitwirkungshandlung besteht in der Mitwirkung beim Requirements Engineering. In dieser Projektphase muss der Kunde dem Anbieter zahlreiche Informationen vermitteln, insbesondere über die eigene Betriebsorganisation, die Systemumgebung, die Geschäftsprozesse und die daraus resultierenden Anforderungen, die Datenmengen, Testfälle, die Abnahmekriterien sowie die anzubindenden Drittsysteme.<sup>71</sup> Weitere typische Mitwirkungshandlungen sind die Bereitstellung von Systemen, die nicht in den Verantwortungsbereich des Anbieters fallen, die Koordination von Nebenunternehmern, die Abstellung von qualifiziertem Personal, die Mitwirkung in der Projektorganisation sowie die Mitwirkung bei den Abnahmetests.<sup>72</sup>

[Rz 33] Noch wenig diskutiert, aber in der Praxis wichtig ist, dass die Einführung einer neuen Software gleichzeitig ein Reorganisationsprojekt im Betrieb des Kunden darstellt.<sup>73</sup> Die Umsetzung dieses Reorganisationsprojekts stellt ebenfalls eine Mitwirkungshandlung des Kunden dar. Das durchzusetzen, ist oft schwierig. War es ursprünglich noch die Absicht des Kunden, möglichst viele Standards zu übernehmen und dementsprechend seine internen Abläufe an die Software anzupassen, will er im Projektverlauf dann allzu oft doch nicht auf Gewohntes verzichten.

#### 3.2. Rechtsfolgen bei einer Verletzung der Mitwirkungspflichten

[Rz 34] Nimmt der Kunde die Mitwirkungshandlungen nicht vor, stellt sich die Frage, ob der Anbieter den verursachten Mehraufwand sowie weiteren Schadenersatz fordern und vom Vertrag zurück treten kann. Das hängt davon ab, ob eine Mitwirkungshandlung rechtlich eine Pflicht oder eine blosser Obliegenheit darstellt.

[Rz 35] Mitwirkungshandlungen werden im Normalfall als blosser Obliegenheit des Gläubigers betrachtet.<sup>74</sup> Auf Erfüllung einer Obliegenheit besteht kein klagbarer Anspruch.<sup>75</sup> Der Gläubiger

---

<sup>68</sup> FRÖHLICH-BLEULER (FN 4), Rz 760; REDEKER (FN 10), Rz D 223; WITZEL (FN 21), Rz G 205 und 215. So lässt sich bereits dem Urteil des BGH vom 13.7.1988, VIII ZR 292/87 (Registrierkassen), wiedergegeben in CR 1989, 102 (104), LS 2, die folgende Aussage zur Mitwirkung des Kunden entnehmen: Kann der Softwarehersteller die Verpflichtung zur Programmierung nur sinnvoll unter Mitwirkung des Kunden erfüllen, so besteht aus in der Natur der Sache liegenden Gründen, auch ohne ausdrückliche Regelung, eine entsprechende Mitwirkungspflicht.

<sup>69</sup> REDEKER (FN 10), Rz D 228; WITZEL (FN 21), Rz G 216.

<sup>70</sup> Zum grösseren Aufwand des Kunden bei agilen Projekten siehe hinten Ziffer 3.3.

<sup>71</sup> FRÖHLICH-BLEULER (FN 4), Rz 770; REDEKER (FN 10), Rz D 224; WITZEL (FN 21), Rz G 205.

<sup>72</sup> FRÖHLICH-BLEULER (FN 4), Rz 769 ff.; REDEKER (FN 10), Rz D 223 ff.; WITZEL (FN 21), Rz G 204 ff.

<sup>73</sup> JOCHEN SCHNEIDER, Mitwirkungspflichten des Auftraggebers bei Softwareanpassung – Organisatorische Änderungen beim Kunden als wesentliche Mitwirkungsleistung, ITRB 2008, 261 ff.

<sup>74</sup> ROLF H. WEBER, Berner Kommentar, Das Obligationenrecht, Art. 68-96, Bern 2005, Art. 91 OR N 50; GAUCH (FN 1), Rz 1328.

<sup>75</sup> PETER GAUCH/WALTER R. SCHLUEP/SUSAN EMMENEGGER, Schweizerisches Obligationenrecht, Allgemeiner Teil, Band II, 10. A., Zürich 2014, Rz 2390.

gerät jedoch bei einem Ausbleiben in Gläubigerverzug; im Normalfall ist dafür eine Abmahnung erforderlich.<sup>76</sup>

[Rz 36] Gläubigerverzug schliesst erstens Schuldnerverzug aus,<sup>77</sup> d.h. die Erfüllungsfristen stehen für den Anbieter während des Gläubigerverzugs still. Zweitens geht die Gefahr für den zufälligen Untergang auf den Gläubiger über. Und drittens kann der Schuldner ausser bei reinen Sachleistungen vom Vertrag zurücktreten (Art. 95 Obligationenrecht [OR]). Nicht entschieden ist, ob der Anbieter bei einer Verletzung einer Mitwirkungsobliegenheit auch Ersatz des dadurch verursachten Mehraufwands verlangen kann (sog. Mehrvergütung).<sup>78</sup> Allerdings stellt sich diese Frage nur bei einem vereinbarten Festpreis, weil bei einer Aufwandsentschädigung dieser Aufwand ohnehin mitberücksichtigt wird.<sup>79</sup> Hingegen besteht kein Schadenersatzanspruch für Verzögerungsschäden, z.B. wenn der Anbieter aufgrund der Verzögerung in anderen Projekten in Rückstand gerät oder um Folgeaufträge kommt.<sup>80</sup>

[Rz 37] Unter besonderen Voraussetzungen sind Mitwirkungshandlungen aber keine blossen Obliegenheiten, sondern echte Vertragspflichten des Gläubigers.<sup>81</sup> Auch steht es den Parteien frei, Mitwirkungshandlungen vertraglich als echte Vertragspflichten auszugestalten.<sup>82</sup> Ob eine solche Vereinbarung im Einzelfall getroffen wurde, ist eine Auslegungsfrage. Ein Indiz für das Vorliegen einer echten Mitwirkungspflicht ist ein für den Kunden erkennbares, starkes Eigeninteresse des Anbieters an der Ausführung des Werks, namentlich bei langfristigen Werkverträgen.<sup>83</sup> Im Unterschied zu den Mitwirkungsobliegenheiten hat der Anbieter bei Verletzung einer echten Mitwirkungspflicht einen durchsetzbaren Erfüllungsanspruch. Über die Verzugsrechte kann der Anbieter dann insbesondere auch den Ersatz von reinen Verzögerungsschäden geltend machen.

### 3.3. Mitwirkungshandlungen bei agilen Softwareprojekten

[Rz 38] Bei agilen Projekten hat der Kunde eine intensivierete Mitwirkungspflicht.<sup>84</sup> Seine Mitarbeiter sind stärker in die Projektorganisation eingebunden. Fachexperten des Kunden sind sogar Teil des eigentlichen Entwicklungsteams. Die Ergebnisse der Entwicklungszyklen sind laufend zu prüfen und abzunehmen.<sup>85</sup> Der Kunde leistet damit einen wesentlichen, über die Anforderungsspezifikation hinausgehenden Beitrag. Seine Mitwirkungshandlungen sind mit den Leistungspflichten des Anbieters eng verzahnt. Zudem hat der Anbieter ein Eigeninteresse, das Projekt zu realisieren, um damit sein Know-how im betroffenen Fachbereich weiterzuentwickeln und mit einer Projekt-

---

<sup>76</sup> Zur Abmahnung siehe ALFRED KOLLER, Schweizerisches Obligationenrecht, Allgemeiner Teil, Bern 2009, § 56 Rz 5; a.M. FRÖHLICH-BLEULER (FN 4), Rz 786.

<sup>77</sup> GAUCH/SCHLUEP/EMMENEGGER (FN 75), Rz 2449; Koller (FN 76), § 56 Rz 101.

<sup>78</sup> Bejahend GAUCH (FN 1), Rz 1336 ff., der unter Heranziehung des deutschen Rechts einen solchen verschuldensunabhängigen Mehrvergütungsanspruch durch richterliche Lückenfüllung fordert; zurückhaltend GAUCH/SCHLUEP/EMMENEGGER (FN 75), Rz 2471 f.

<sup>79</sup> GAUCH (FN 1), Rz 1336.

<sup>80</sup> GAUCH (FN 1) Rz 1340; GAUCH/SCHLUEP/EMMENEGGER (FN 75), Rz 2471.

<sup>81</sup> BK-WEBER (FN 74), Art. 91 OR N 56.

<sup>82</sup> GAUCH/SCHLUEP/EMMENEGGER (FN 75), Rz 2391; GAUCH (FN 1), Rz 1344.

<sup>83</sup> GAUCH (FN 1), Rz 1345.

<sup>84</sup> WITZEL (FN 21), Rz G 225; zu den folgenden Ausführungen über die besonderen Mitwirkungspflichten bei agilen Projekten siehe KOCH (FN 5), 118; JÄHNICHEN (FN 8), 124.

<sup>85</sup> FRÖHLICH-BLEULER (FN 4), 352.

referenz zu belegen. Die Mitwirkungshandlungen des Kunden sind bei agilen Projekten deshalb echte Vertragspflichten und nicht blosse Obliegenheiten.<sup>86</sup>

## 4. Rechtliche Qualifikation agiler Softwareprojekte

### 4.1. Einfache Gesellschaft

[Rz 39] Einige deutsche Autoren ziehen mit Verweis auf die ausgedehnte Mitwirkung des Kunden in Betracht, die Zusammenarbeit zwischen einem Anbieter und einem Kunden im Rahmen agiler Projekte als einfache Gesellschaft zu beurteilen.<sup>87</sup> Dem ist nicht zuzustimmen, denn es fehlt im Normalfall an der für Gesellschaftsverhältnisse konstitutiven gemeinsamen Zweckverfolgung.<sup>88</sup> Diese ist das ausschliessliche Kriterium zur Abgrenzung einer einfachen Gesellschaft gegenüber Austauschverträgen.<sup>89</sup>

[Rz 40] Bei einem Vertrag über ein Softwareprojekt zwischen einem Anbieter und einem Kunden handelt es sich vielmehr um einen Geschäftsbesorgungsvertrag.<sup>90</sup> Die Interessen des Anbieters und des Kunden zielen zwar vordergründig in die gleiche Richtung – beide wollen das Projekt erfolgreich realisieren – sind aber im Kern wie bei jedem Austauschvertrag gegensätzlich. Der Kunde will eine Software entwickelt oder implementiert haben, um sie nachher zu verwenden. Der Anbieter will diesen Auftrag ausführen und damit Umsatz erzielen. Ohnehin würden die mit einem Gesellschaftsverhältnis verbundenen Rechtsfolgen den Interessen nicht gerecht.<sup>91</sup> Der Anbieter will nicht, dass der Kunde gemeinsam mit ihm Gesamteigentum am Arbeitsergebnis erwirbt, und der Kunde will im Aussenverhältnis nicht solidarisch haften. Bei agilen Projekten liegt deshalb im Normalfall kein Gesellschaftsverhältnis vor.<sup>92</sup> Anders wäre es nur dann, wenn mit der Zusammenarbeit ein über die Projektrealisation hinausgehendes Ziel verfolgt wird, wie z.B. der gemeinsame Vertrieb oder der gemeinsame langfristige Unterhalt einer Software.

---

<sup>86</sup> So auch FRÖHLICH-BLEULER (FN 4), Rz 352; a.M. KOCH (FN 5), nach welchem das ähnlich wie nach GAUCH (siehe dazu FN 83) nur gilt, wenn der Anbieter ein über das Projekt hinaus reichendes Interesse hat (z.B. bei markterschliessenden Pilotprojekten).

<sup>87</sup> FRANK (FN 5), 138 f.; KOCH (FN 5), 119; LAPP (FN 5), 70; SCHNEIDER (FN 23), Rz C 154; kritisch für das deutsche Recht hingegen FUCHS/MEIERHÖFER/MORSBACH/PAHLOW (FN 5), 430.

<sup>88</sup> Zur gemeinsamen Zweckverfolgung (*animus societatis*) siehe PETER JUNG, in: Marc Amstutz et al. (Hrsg.), Handkommentar zum Schweizer Privatrecht, 2. A., Zürich 2012, Art. 530 OR N 4; LUKAS HANDSCHIN, in: Heinrich Honsell/Nedim Peter Vogt/Rolf Watter (Hrsg.), Basler Kommentar, OR II, 4. A, Art. 530 OR N 4 f.

<sup>89</sup> HANDSCHIN (FN 88), Art. 530 OR N 10.

<sup>90</sup> Zur Abgrenzung der einfachen Gesellschaft von den Geschäftsbesorgungsverträgen siehe HANDSCHIN (FN 88), Art. 530 OR N 10a und JUNG (FN 88), CHK-Art. 530 OR N 6.

<sup>91</sup> LAPP (FN 5), 70.

<sup>92</sup> Siehe auch HENGSTLER (FN 5), 114.

## 4.2. Werkvertrag

[Rz 41] Softwareentwicklungsverträge werden im schweizerischen Recht von Lehre und Rechtsprechung als Werkverträge beurteilt.<sup>93</sup> Im deutschen Recht verhält es sich ähnlich.<sup>94</sup> Auch bei Softwareintegrationsverträgen handelt es sich im Wesentlichen um Werkverträge. Zwar tritt mit dem Erwerb der zu integrierenden Software ein kauf-, respektive lizenzrechtliches Element hinzu.<sup>95</sup> So hat das Bundesgericht einen Integrationsvertrag für ein ERP-System als gemischten Vertrag mit Elementen des Lizenz-, Kauf- und Werkvertrags beurteilt.<sup>96</sup> Im vorliegenden Kontext ist das jedoch ohne Belang, da sich dadurch an der werkvertraglichen Natur der hier interessierenden Integrationsleistung nichts ändert.

[Rz 42] Diese einheitliche Einordnung unter das Werkvertragsrecht wird nun für agile Softwareprojekte teilweise in Frage gestellt.<sup>97</sup> Dabei werden hauptsächlich die folgenden Argumente vorgebracht: (1) Es fehle im Zeitpunkt des Vertragsabschlusses eine Leistungsbeschreibung, denn diese werde erst später konkretisiert und (2) bei der Leistungsbeschreibung wirke der Kunde intensiv mit. Dem ist aus folgenden Überlegungen nicht zuzustimmen.

[Rz 43] Zu (1): Die Verpflichtung des Unternehmers zur Herstellung eines Werks ist ein wesentlicher Vertragspunkt für die Qualifikation als Werkvertrag.<sup>98</sup> Dies bedeutet aber nicht, dass im Zeitpunkt des Vertragsabschlusses bereits eine detaillierte Leistungsbeschreibung vorliegen muss. Das Werk muss nicht bestimmt, sondern lediglich bestimmbar sein.<sup>99</sup> Zum einen kann sich die Leistungsbeschreibung auch aus den Umständen ergeben. So kann bei fehlendem Pflichtenheft auf einen mittleren Ausführungsstandard zurückgegriffen werden, der sich am Stand der Technik und am Vertragszweck orientiert.<sup>100</sup> Deshalb werden Erstellungs- oder Anpassungsverträge auch dann als Werkverträge beurteilt, wenn das Pflichtenheft erst nach Vertragsabschluss erstellt wird.<sup>101</sup> Zum anderen kann es sogar gänzlich dem Besteller überlassen werden, ein bei Vertragsabschluss

---

<sup>93</sup> FRÖHLICH-BLEULER (FN 4), Rz 392 mit Verweisen.

<sup>94</sup> REDEKER (FN 10), Rz D 73 ff.; SCHNEIDER (FN 23), Rz C 73. Allerdings wird im deutschen Recht auch die Anwendung von Kaufrecht in Betracht gezogen: siehe FUCHS/MEIERHÖFER/MORSBACH/PAHLOW (FN 5), 432 und LAPP (FN 5), 70; ausführlich und kritisch REDEKER (FN 10), Rz D 74 ff.; grundlegend WITTE (FN 5), 44 ff.; siehe auch Urteil des BGH VII ZR 151/08 vom 23. Juli 2009, E. 31 ff., wonach Werkvertragsrecht nur zur Anwendung gelangt, wenn die Planungsleistung den Schwerpunkt des Vertrags bildet. Im vorliegenden Kontext spielt dies indessen keine Rolle, weil vorab die Abgrenzung gegenüber dem Dienstvertrag (in der Schweiz: Auftrag) interessiert.

<sup>95</sup> FRÖHLICH-BLEULER (FN 4), Rz 373 ff.

<sup>96</sup> Urteil des Bundesgerichts 4C.393/2006 vom 27. April 2007, E. 3.1.

<sup>97</sup> FRÖHLICH-BLEULER (FN 4), Rz 347 lehnt unter schweizerischem Recht eine werkvertragliche Resultatverantwortung ab, wenn die Parteien das zu erreichende Resultat nicht im Voraus bestimmen, sondern während der Zusammenarbeit (auf Zuruf) gemeinsam entwickeln. Für das Deutsche Recht zusammenfassend: FUCHS/MEIERHÖFER/MORSBACH/PAHLOW (FN 5), 429 sowie REDEKER (FN 10), Rz 108 ff.; für das deutsche Recht auf Werkvertrag schliessend: FUCHS/MEIERHÖFER/MORSBACH/PAHLOW (FN 5), 432 f.; KREMER (FN 5), 286 ff.; REDEKER (FN 10), Rz D 115; kritisch zur werkvertraglichen Ausgestaltung agiler Softwareverträge: FRANK (FN 5), 140; auch eine dienstvertragliche Einordnung in Betracht ziehend HENGSTLER (FN 5), 115 und Lapp (FN 5), 70 sowie SCHNEIDER (FN 23), Rz C 153 ff. und SCHNEIDER «Neue Projektmethoden» (FN 5), 19.

<sup>98</sup> GAUCH (FN 1), Rz 381. Zu den wesentlichen Vertragspunkten (*essentialia negotii*) als Voraussetzung für das Zustandekommen eines Vertrags siehe statt vieler CLAIRE HUGUENIN, *Obligationenrecht*, 2. A., Zürich 2014, Rz 256.

<sup>99</sup> GAUCH (FN 1), Rz 382.

<sup>100</sup> AUER-REINSORFF (FN 5), 95; KOCH (FN 5), 118; SCHNEIDER (FN 23), Rz C 142.

<sup>101</sup> Für das deutsche Recht: SCHNEIDER (FN 23), Rz C 153 mit Verweis auf ein Urteil des OLG Düsseldorf vom 10. Juni 1992 – 19 U 23/91, publiziert in CR 1993, 361.

nur in den Grundzügen beschriebenes Werk durch einseitige Erklärungen zu konkretisieren.<sup>102</sup> Aus diesem Grund ist es im Bauwesen möglich, einen Totalunternehmervertrag über die Errichtung eines Bauwerks abzuschliessen, welches aufgrund einer Bauidee des Bestellers zuerst noch geplant werden muss.<sup>103</sup>

[Rz 44] Betrifft die werkvertragliche Leistung ein Ergebnis, das vorher nicht oder nicht in vergleichbarer Form existierte, so ergibt sich aus der Natur der Sache, dass dieses Ergebnis im Zeitpunkt des Vertragsabschlusses nicht vollständig beschrieben werden kann. Die Beschreibung des Ergebnisses kann geradezu den Kern der vertraglichen Leistung ausmachen. Dies gilt zum Beispiel für die Erstellung von Plänen durch einen Architekten, was nach konstanter Rechtsprechung als werkvertragliche Leistung beurteilt wird.<sup>104</sup> Auch der Architekt arbeitet eng mit dem Bauherrn zusammen, nimmt dessen Bedürfnisse auf und erstellt daraus ein Resultat, welches zum Zeitpunkt des Vertragsabschlusses erst dem Grundsatz nach, aber nicht detailliert beschrieben werden konnte. Das Gleiche muss auch für die Entwicklung von Individualsoftware nach den Anforderungen des Kunden gelten. Die Leistungskomponente des Requirements Engineering ist mit der Planungsleistung des Architekten durchaus vergleichbar. Sie kann somit auch Gegenstand einer werkvertraglichen Leistung sein.<sup>105</sup>

[Rz 45] Dies bedeutet nicht, dass der Kunde alles umgesetzt haben kann, was er als wünschbar erachtet. Seine Wünsche sind unter objektiven Aspekten möglicherweise nicht auf sinnvolle Weise oder überhaupt nicht umsetzbar. Wie sich der Bauplaner an Normen und Planungsgrundsätze halten muss, orientiert sich auch der Softwareentwickler an Standards und Best Practices. Das Recht des Kunden ist deshalb kein absolutes, einseitiges Leistungsbestimmungsrecht, sondern ein Recht zur Konkretisierung der Leistung im Rahmen des nach der Natur des Projektes zu erwarteten Umfangs.<sup>106</sup>

[Rz 46] Die Übernahme einer werkvertraglichen Verantwortung für ein noch nicht konkretisiertes Resultat stellt für den Anbieter zweifellos ein Risiko dar, insbesondere dann, wenn er seine Leistungen zu einem Festpreis anbietet. Diese Risikoverteilung ist für den Werkvertrag jedoch typisch. Das ergibt sich e contrario aus Art. 373 Abs. 2 OR, wonach der Richter bei ausserordentlichen Umständen, welche die Vertragserfüllung übermässig erschweren, nach seinem Ermessen entweder den Werklohn angemessen erhöhen oder die Vertragsauflösung bewilligen kann. Angesichts dieser Spezialnorm bleibt auch wenig Raum, die Übernahme des Erfüllungsrisikos bei unspezifizierten Leistungen als übermässige Bindung im Sinne von Art. 27 Abs. 2 Zivilgesetzbuch (ZGB) zu beurteilen.<sup>107</sup>

[Rz 47] Die werkvertragliche Risikoverteilung ist auch für agile Softwareprojekte sachgerecht, denn der Anbieter kann die Risiken als Spezialist besser kontrollieren als der Kunde. Erstens muss ein erfahrener Anbieter in der Lage sein, die Anforderungen des Kunden an das werkvertragliche Resultat bereits im Vorfeld abzuschätzen. Die Informatik gilt als tendenziell beherrschbare Ingenieursdiszi-

---

<sup>102</sup>GAUCH (FN 1), Rz 382, der dieses Konkretisierungsrecht für den Anlagebau als geradezu typisch erachtet. Allerdings meint FRANK (FN 5), 140, dass sich Softwarehersteller nicht auf ein solches einseitiges Leistungsbestimmungsrecht einlassen werden.

<sup>103</sup>GAUCH (FN 1), Rz 382.

<sup>104</sup>GAUCH (FN 1), Rz 49 ff.; GAUCH/MIDDENDORF (FN 1), Rz 1.30 ff.

<sup>105</sup>So auch REDEKER (FN 10), Rz D 115.

<sup>106</sup>Nach AUER-REINSDORFF (FN 5), 95, ist bei agilen Projektmethoden ein mittlerer Ausführungsstandard geschuldet, der sich durch Priorisierungen und Zusatzwünsche des Kunden nach und nach verfeinert.

<sup>107</sup>GAUCH (FN 1), Rz 382.

plin.<sup>108</sup> Ist das überhaupt nicht möglich, so muss er Vorbehalte anbringen oder er darf sich nicht auf einen Werkvertrag einlassen. Und zweitens muss der Anbieter den Kunden beim Requirements Engineering anleiten. Dies beinhaltet auch die Pflicht des Anbieters, den Kunden abzumachen, wenn dieser beim Requirements Engineering Anforderungen stellt, die sich nicht sinnvoll umsetzen lassen.

[Rz 48] Zu (2): In Bezug auf die Mitwirkung des Kunden unterscheiden sich klassische und agile Projektmethoden nicht grundsätzlich, sondern nur graduell.<sup>109</sup> Jedes Softwareprojekt setzt die Mitwirkung des Kunden voraus, denn nur der Kunde ist in der Lage, seine Anforderungen zu spezifizieren. Ob dies in einer separaten, vorgeschalteten Projektphase geschieht oder später im Gleichlauf mit der Programmierung, spielt für die rechtliche Qualifikation keine Rolle.

[Rz 49] Verträge über agile Softwareprojekte sind deshalb ebenfalls als Werkverträge zu qualifizieren, sobald der Vertrag dem Anbieter eine Ergebnisverantwortung zuweist.<sup>110</sup> Dies ist immer dann der Fall, wenn der Anbieter Softwarecode liefert.

### 4.3. Auftrag und Personalstellung

[Rz 50] Es gibt Projekte, bei welchen der Kunde die Gesamtverantwortung für das Softwareprojekt selber trägt und externe Leistungserbringer nur für Teilaufgaben bezieht. Solche Konstellationen sind aber die Ausnahme, weil dem Kunden in der Regel die Kompetenz fehlt, den Entwicklungsprozess selber zu gestalten.<sup>111</sup> Im Normalfall will der Kunde, dass der Anbieter für das Ergebnis einzustehen hat.<sup>112</sup>

[Rz 51] Selbstverständlich können jedoch Verträge über agile Softwareprojekte auch auftragsrechtlicher Natur sein.<sup>113</sup> Dies ist z.B. dann der Fall, wenn der Kunde Beratungsleistungen einkauft. Bezieht der Kunde beim Anbieter hingegen lediglich Personal, welches er selbständig anweist, handelt es sich um einen Personalverleihvertrag.

### 4.4. Zusammengesetzte Verträge

[Rz 52] Einige Autoren propagieren eine Aufteilung des Vertragsverhältnisses über agile Projekte in einen Rahmenvertrag einerseits und Teilprojektverträge andererseits.<sup>114</sup> Der Rahmenvertrag würde die Grundlage der Zusammenarbeit regeln und wäre dienstvertraglicher, respektive nach schweizerischer Terminologie auftragsrechtlicher Natur. Für die einzelnen Abschnitte (Sprints) würden Teilprojektverträge abgeschlossen. Sie hätten werkvertraglichen Charakter. Dabei würde es sich um einen zusammengesetzten Vertrag handeln ähnlich einem Sukzessivlieferungsvertrag.<sup>115</sup>

---

<sup>108</sup>Ebenso wie die Mechanik und die Elektrotechnik, jedoch im Unterschied zur Chemie, Biologie und Pharmazie (BRANDI-DOHRN (FN 2), Rz 35).

<sup>109</sup>SCHNEIDER (FN 23), Rz C 142; zur Mitwirkung des Kunden bei agilen Methoden siehe vorne Ziffer 3.3.

<sup>110</sup>Zu den Ausnahmen sowie zur Vertragsauslegung siehe die nachfolgenden Ziffern 4.3–4.5 und 5.

<sup>111</sup>FUCHS/MEIERHÖFER/MORSBACH/PAHLOW (FN 5), 429; LAPP (FN 5), 70.

<sup>112</sup>FRANK (FN 5), 140.

<sup>113</sup>FRANK (FN 5), 139 f.

<sup>114</sup>FRANK (FN 5), 141; für das schweizerische Recht FRÖHLICH-BLEULER (FN 4), 346 ff. unter Berufung auf FRANK.

<sup>115</sup>Zu den zusammengesetzten Verträgen siehe HUGUENIN (FN 98), Rz 3676 ff.

[Rz 53] Das führt jedoch zu einem erheblichen administrativen und rechtlichen Mehraufwand und ist nicht praktikabel.<sup>116</sup> Zudem bleibt der Zielkonflikt zwischen Flexibilität, Ergebnisverantwortung und Kostensicherheit ungelöst, denn die Parteien betrachten die vertraglichen Leistungen als untrennbares Ganzes. Aus der Sicht des Kunden ist die Leistung erst dann erbracht, wenn das angestrebte Endresultat erreicht wurde. Mit einer blossen Teilleistung in Bezug auf einen einzelnen Sprint ist dem Kunden nicht gedient.<sup>117</sup> Aber auch der Anbieter will nicht, dass der Kunde nur einen Teil der Leistungen bezieht und sich dann aus dem Projekt zurück zieht, denn er muss ja seine Entwicklungsressourcen für die gesamte Projektdauer planen. Auch könnten die Parteien in Anwendung der zwingenden Bestimmung von Art. 404 Abs. 1 OR<sup>118</sup> jederzeit vom auftragsrechtlichen Rahmenvertrag zurücktreten, was nicht im Interesse der Parteien sein kann. Und schliesslich widerspricht der Formalismus, der mit der Ausarbeitung der Teilprojektverträge verbunden wäre, dem Anspruch agiler Methoden nach einer flexiblen Zusammenarbeit grundsätzlich.<sup>119</sup>

#### 4.5. Zurufsprojekte

[Rz 54] Schwierig ist die rechtliche Einordnung, wenn der Kunde zwar die Gesamtverantwortung trägt, beim Anbieter jedoch nicht nur singuläre Beratungsleistungen oder Personal einkauft, sondern ganze Entwicklungsteams, die möglicherweise sogar ihre eigenen Entwicklungswerkzeuge einsetzen. Bei einer solchen Leistung ist von einem Innominatvertrag (sui generis) auszugehen, der neben auftragsrechtlichen und werkvertraglichen unter Umständen auch lizenzvertragliche Elemente enthält.<sup>120</sup> Ein reiner Personalverleih hingegen liegt nicht mehr vor, sobald weitere Leistungskomponenten zur Arbeitnehmerüberlassung hinzutreten (z.B. Qualitätssicherung oder Projektleitung).

[Rz 55] In der deutschen Praxis werden solche Konstellationen unter dem Begriff der «Zurufsprojekte» diskutiert.<sup>121</sup> Bei «Zurufsprojekten» äussern Mitarbeiter des Kunden mehr oder weniger vorbereitete ihre Wünsche und der Anbieter setzt diese technisch um.<sup>122</sup> Dann ist unklar, wie weit die Erfolgshaftung des Anbieters geht. Zwar trägt dieser bei «Zurufsprojekten» die Verantwortung für die technisch einwandfreie Ausführung der einzelnen Aufträge, sofern sich der Kunde nicht auch in den technischen Bereich einmischt. Keine Verantwortung übernimmt der Anbieter jedoch für das Requirements Engineering. Er haftet deshalb höchstens für die softwaretechnisch fachgerechte Ausführung der Entwicklungsleistungen, nicht jedoch für das Gesamtergebnis. Derartige Projekte

---

<sup>116</sup>So auch FUCHS/MEIERHÖFER/MORSBACH/PAHLOW (FN 5), 430.

<sup>117</sup>Gemäss HUGUENIN (FN 98), Rz 3678 rechtfertigt es sich bei zusammengesetzten Verträgen angesichts der Koppelung der Verträge nur selten, die einzelnen Verträge gesondert zu behandeln. Vielmehr sei über alle gekoppelten Verträge hinweg der vertragliche Regelungsschwerpunkt zu ermitteln, um diese Regeln sodann auf das gesamte Vertragsnetz anzuwenden.

<sup>118</sup>Zur zwingenden Natur von Art. 404 Abs. 1 OR siehe ROLF H. WEBER, in: Heinrich Honsell/Nedim Peter Vogt/Wolfgang Wiegand (Hrsg.), Basler Kommentar, OR I, 5. A., Basel 2011, Art. 404 OR N 9.

<sup>119</sup>HENGSTLER (FN 5), 116; WITTE (FN 5), 47.

<sup>120</sup>Zu den Innominatverträgen sowie deren Einteilung HUGUENIN (FN 98), Rz 3681 ff.; MARC AMSTUTZ/ARIANE MORIN/WALTER R. SCHLUEP, BSK OR I (FN 118), Einl. vor Art. 184 ff. OR, N 5 ff.

<sup>121</sup>In einem Referat an einer Fachveranstaltung des Zürcher Anwaltsverbands wurde eine ähnliche Ausgangslage von Rechtsanwalt Stephan Kronbichler, Genf, auch schon als «Miete einer Softwarefabrik» bezeichnet.

<sup>122</sup>SCHNEIDER (FN 23), Rz C 170. Unter deutschem Recht werden «Zurufsprojekte» als Werkverträge beurteilt (Urteil des OLG Karlsruhe 1 U 250/01 vom 16. August 2002, publiziert in CR 2003, 95.) In der deutschen Literatur wird dies kritisiert und es wird die Anwendung von Dienstvertragsrecht gefordert (JOCHEN SCHNEIDER, Softwareerstellung und Softwareanpassung – Wo bleibt der Dienstvertrag, CR 2003, 318 f.).

sind aufgrund der wechselseitigen Leistungspflichten kaum mehr justiziabel.<sup>123</sup>

## 5. Vertragsauslegung

[Rz 56] Unter welchem Vertragstyp ein konkreter Vertrag einzuordnen ist, bestimmt sich anhand des Vertragsinhalts. Ist dieser unklar, ist der Vertrag auszulegen. Dabei sind auch die äusseren Umstände massgebend wie z.B. das gegen aussen kommunizierte Geschäftsmodell des beauftragten Anbieters,<sup>124</sup> beim Kunden vorhandene softwaretechnische Kompetenzen,<sup>125</sup> die Art und Weise der Instruktion und der Überwachung der Programmierstätigkeit<sup>126</sup> sowie die Natur der abzuliefernden Arbeitsresultate.<sup>127</sup>

[Rz 57] Oft werden ein Festpreis mit einer werkvertraglichen Verpflichtung und eine Aufwandsentschädigung mit einem Auftrag gleichgesetzt. Zwar ist auch die Entschädigungsart ein zu berücksichtigender Umstand,<sup>128</sup> jedoch nur einer unter mehreren. Zwingend mit einem Werkvertrag verbunden ist einzig, dass der Unternehmer das Mängelrisiko trägt, d.h. eine Leistung erbringt, die durch bestimmte oder bestimmbarere Erfüllungskriterien definiert ist und für deren Mängelfreiheit er mit einer Gewährleistung einzustehen hat.

## 6. Projektrisiken

[Rz 58] Die in der Praxis auftretenden Risiken bei der Abwicklung agiler Softwareprojekte lassen sich im Wesentlichen auf drei Bereiche reduzieren: Wurden die Projektziele erreicht (Entwicklungsrisiko)? Sind die Termine eingehalten worden (Terminrisiko)? Gibt es Kostenüberschreitungen (Kostenrisiko)?<sup>129</sup> Diese Projektrisiken sind je nach Vertrag entweder vom Anbieter oder vom Kunden zu tragen. Nur bei einer einfachen Gesellschaft tragen die Gesellschafter die Risiken gemeinsam.<sup>130</sup>

[Rz 59] Das Entwicklungsrisiko trägt im Werkvertragsrecht der Anbieter. Er verpflichtet sich zur Herstellung eines Werkes (Art. 363 OR), für dessen Mängelfreiheit er einzustehen hat (Art. 368 OR). Der Anbieter hat seine werkvertragliche Leistungspflicht erst mit der Ablieferung des vertraglich vereinbarten Werks erfüllt.<sup>131</sup> Gelingt ihm dies nicht, kann der Kunde Verzug geltend machen und vom Vertrag zurücktreten (Art. 107 Abs. 2 OR). Bei einem Auftrag hingegen trägt der Kunde

---

<sup>123</sup>KREMER (FN 5), 287; LAPP (FN 5), 71.

<sup>124</sup>Ein Personalverleiher wird keine softwaretechnische Gesamtverantwortung tragen.

<sup>125</sup>Verfügt der Kunde über keine eigene softwaretechnische Kompetenz, so ist eine Übernahme der Gesamtverantwortung von vornherein ausgeschlossen.

<sup>126</sup>Übt der Anbieter selber eine Kontrollfunktion aus, indem er nicht nur Programmierer zur Verfügung stellt, sondern auch Projektleitungsaufgaben übernimmt, so spricht dies für eine Übernahme der Gesamtverantwortung zumindest im softwaretechnischen Bereich.

<sup>127</sup>Liefert der Anbieter prüfbarer Softwarecode ab, so übernimmt er dafür immer eine softwaretechnische Verantwortung.

<sup>128</sup>Gemäss REDEKER (FN 10), Rz D 57 ist ein vereinbarter Festpreis ein Indiz für einen Werkvertrag.

<sup>129</sup>Zu den Risikoarten siehe BRANDI-DOHRN (FN 2), Rz 32 ff.

<sup>130</sup>Zum Grundsatz der gemeinsamen Risikotragung bei der einfachen Gesellschaft siehe Art. 533 Abs. 1 OR; zur eingeschränkten Haftung des Gesellschafters bei Leistungsstörungen siehe JUNG (FN 88), CHK-Art. 531 OR N 10.

<sup>131</sup>HUGUENIN (FN 98), Rz 3138.

das Entwicklungsrisiko. Der Anbieter schuldet kein Resultat, sondern die getreue und sorgfältige Ausführung des Auftrags (Art. 398 Abs. 2 OR). Gleich verhält es sich beim Personalverleih, wobei hier der Anbieter gar nur für die sorgfältige Auswahl der eingesetzten Arbeitnehmer einzustehen hat.<sup>132</sup>

[Rz 60] Bei einem agilen Softwareprojekt ist der werkvertraglich haftende Anbieter mit der besonderen Schwierigkeit konfrontiert, dass keine detaillierte Leistungsbeschreibung existiert, anhand welcher eine Abnahmeprüfung stattfinden kann. Stellt sich der Kunde auf den Standpunkt, der Anbieter habe die werkvertragliche Leistung nicht erbracht, muss der Anbieter anhand objektiver Kriterien darlegen, dass er erfüllt hat. Die daraus resultierenden Beweisrisiken trägt der Anbieter (Art. 8 ZGB).

[Rz 61] Auch das Terminrisiko liegt unter dem Werkvertragsrecht beim Anbieter. In der Regel vereinbaren die Parteien einen Ablieferungstermin. Andernfalls ist ein solcher gemäss dem hypothetischen Parteiwillen zu ermitteln.<sup>133</sup> Liefert der Anbieter das Werk nicht zum vereinbarten Termin ab, stehen dem Kunden die Verzugsrechte offen. Zwar ist die Mitwirkung des Kunden besonders bei agilen Projekten für den Projekterfolg entscheidend.<sup>134</sup> Projektverzögerungen belasten jedoch einseitig den Anbieter, da nur diesem bei Nichteinhaltung des Ablieferungstermins Rechtsnachteile drohen. Will sich der Anbieter auf eine Verletzung der Mitwirkungspflichten berufen, muss er diese abmahnen, was eine straffe und somit dem agilen Ansatz widersprechende Projektführung bedingt.

[Rz 62] Auch nach Auftragsrecht tritt bei Nichteinhaltung von Terminen Verzug ein.<sup>135</sup> Da jedoch kein messbares Resultat geschuldet ist, liegt selten Verzug vor – ausgenommen es wird überhaupt nicht geleistet.<sup>136</sup> Das Gleiche gilt beim Personalverleih.

[Rz 63] Die Zuordnung des Kostenrisikos schliesslich hängt davon ab, ob eine pauschale Entschädigung oder eine Entschädigung nach Aufwand vereinbart ist. Sowohl nach Werkvertragsrecht (Art. 374 OR), wie auch nach Auftragsrecht wird der Anbieter mangels anderer Abrede nach Aufwand entschädigt.<sup>137</sup> Oft wird bei Werkverträgen jedoch ein Festpreis (Art. 373 OR) oder ein Kostendach vereinbart.<sup>138</sup> Auch unter dem Auftragsrecht sind pauschale Entschädigungen denkbar,<sup>139</sup> wenn auch in der Praxis eher unüblich. Ein Personalverleih hingegen wird in der Praxis immer nach Zeitaufwand entschädigt.

[Rz 64] Bei einer Aufwandsentschädigung trägt der Kunde, bei einem Festpreis oder einem Kostendach der Anbieter das Kostenrisiko. Eine Zusatzentschädigung kann der Anbieter bei einem Festpreis nur fordern, wenn er eine Beststellungsänderung, schuldhaftes Verhalten des Kunden, einen Annahmeverzug oder ausserordentliche Umstände darlegen kann.<sup>140</sup> Sich als Anbieter bei agi-

---

<sup>132</sup>Zur Haftung des Personalverleihers MANFRED REHBINDER, Berner Kommentar, Der Arbeitsvertrag: Der Einzelarbeitsvertrag, Art. 319–330b OR, 2. A., Bern 2010, BK-OR Art. 319 N 57.

<sup>133</sup>GAUCH (FN 1), Rz 646 und 649.

<sup>134</sup>Siehe Ziffern 2.5 und 3.3.

<sup>135</sup>BSK OR I-WEBER (FN 118), Art. 398 OR N 20.

<sup>136</sup>Zu einem Anwendungsfall einer Nichtleistung bei einem Planervertrag siehe Urteil des Bundesgerichts 4C.18/2005 vom 30. Mai 2005.

<sup>137</sup>Massgebend ist nach Werkvertragsrecht der Aufwand des Unternehmers (Selbstkosten plus Zuschlag für Risiko und Gewinn) und nicht der Wert des fertiggestellten Werkes (zum Ganzen GAUCH (FN 1), Rz 947 ff.). Für das Auftragsrecht siehe BSK OR I-WEBER (FN 118), Art. 394 N 39.

<sup>138</sup>Zum gesetzlich nicht ausdrücklich geregelten, verbindlichen Kostendach siehe GAUCH (FN 1), Rz 1036 ff.

<sup>139</sup>BSK OR I-WEBER (FN 118), Art. 394 N 37.

<sup>140</sup>GAUCH (FN 1), Rz 904; zur Preisanpassung bei ausserordentlichen Umständen im Werkvertragsrecht siehe zudem Art. 373 Abs. 2 OR.

len Projekten auf Bestellungenänderungen (Change Requests) zu berufen, ist jedoch sehr schwierig, weil die Leistung nur rudimentär beschrieben ist.<sup>141</sup> Auch ist die Ergebnisoffenheit ein wesentliches Kriterium agiler Methoden und laufende Änderungen und Neu-Priorisierungen sind beabsichtigt und Bestandteil des Prozesses.<sup>142</sup> Für kostenpflichtige Change Requests bleibt damit nur wenig Raum.<sup>143</sup>

[Rz 65] Zu berücksichtigen ist auch die Regelung der Zahlungsbedingungen. Wenn der Kunde sich nicht einen Zahlungsrückbehalt sichert, wird der Anbieter laufend für seinen gesamten Aufwand entschädigt. Dann kann der Anbieter im Streitfall die weitere Leistungserbringung verweigern und sich zur Rechtfertigung z.B. auf eine Bestellungenänderung oder auf eine Verletzung von Mitwirkungspflichten berufen. Damit kann zumindest die Klägerrolle – wenn auch nicht die Beweislast – dem Kunden zugeschoben werden.

[Rz 66] Als Zwischenergebnis kann somit festgehalten werden, dass unter dem Werkvertragsrecht die Projektrisiken vom Anbieter getragen werden. Das gilt jedoch nur dann uneingeschränkt, wenn ein Festpreis mit Zahlungsrückbehalt vereinbart wird. Bei einem Auftrag oder einem Personalverleihvertrag hingegen werden die Projektrisiken vom Kunden getragen.

[Rz 67] Diese vertragstypische Risikoverteilung wird unter Umständen durch die Mitwirkungspflichten des Kunden überlagert, welche bei agilen Projekten wie dargelegt besonders intensiv sind; dies aus zwei Gründen. Erstens fallen die Mängelrechte des Kunden dahin, wenn der Kunde die Mängel durch Weisungen oder auf andere Weise selber verschuldet hat (Art. 369 OR). Das Gleiche gilt mutatis mutandis für Terminverzug und Kostenüberschreitungen.<sup>144</sup> Da der Kunde institutionalisiert und mit entscheidungsbefugten Vertretern in die Projektorganisation eingebunden ist, kann im Normalfall auch die von Art. 369 OR vorausgesetzte Abmahnung entfallen.<sup>145</sup> Und zweitens genehmigen die in die Entwicklung eingebundenen Mitarbeiter des Kunden laufend die Entwicklungsergebnisse sowie allfällige Projektänderungen.<sup>146</sup>

## 7. Vertragsgestaltung

[Rz 68] Verträge sollen Klarheit schaffen. Das gelingt am besten durch die Einordnung unter einen gesetzlichen Vertragstyp oder einen anerkannten Innominatvertrag.<sup>147</sup> Vollständig neu entwickelte Vertragsmodelle bergen demgegenüber erhebliche Risiken, da in einem Streitfall nicht auf eine Rechtsprechung und die Lehre zurückgegriffen werden kann.

[Rz 69] Verträge über agile Softwareprojekte lassen sich als Werkvertrag oder als Personalverleihver-

---

<sup>141</sup>Siehe Ziffer 2.5.

<sup>142</sup>HENGSTLER (FN 5), 115; KREMER (FN 5), 284.

<sup>143</sup>Nach AUER-REINSDORFF (FN 5), 93 sind klassische Change Requests bei agilen Methoden im Allgemeinen ausgeschlossen.

<sup>144</sup>Zu den Mehrkosten bei Bestellungenänderungen siehe GAUCH (FN 1), 905 ff.; zum Ausschluss des Schuldnerverzugs bei Ereignissen aus dem Verantwortungsbereich des Bestellers siehe GAUCH (FN) 1, 678 ff.

<sup>145</sup>Ausführlich zur Haftungsbefreiung ohne Abmahnung bei sachverständig erteilter Weisung GAUCH (FN 1), 1955 ff. Ausgenommen ist eine zwingend erforderliche Abmahnung bei einer Weisung von grosser Tragweite, welche die Kompetenz der im Projekt eingebundenen Kundenvertreter erkennbar übersteigt.

<sup>146</sup>KOCH (FN 5), 118; LAPP (FN 5), 70. Häufig wird auch vertraglich vereinbart, entscheidungsbefugte Mitarbeiter in die Projektgremien zu entsenden.

<sup>147</sup>SCHNEIDER (FN 23), Rz C 7 bezeichnet eine unklare Vertragstypologie als eine der Projektsünden, die zum Scheitern von IT-Projekten führen.

trag abbilden. Im ersten Fall trägt der Anbieter, im zweiten der Kunde das Projektrisiko. Auch die Ausgestaltung als Auftrag ist möglich.<sup>148</sup> Ein Auftrag mit einer reinen Sorgfaltshaftung ist jedoch nur bei Leistungen mit Beratungscharakter denkbar. Liefert der Anbieter Softwarecode, wird er im Normalfall – auch ohne Ergebnisverantwortung für die funktionalen Anforderungen – eine Haftung für die softwaretechnisch fachgerechte Entwicklung des Codes übernehmen. Dann liegt kein Auftragsverhältnis mehr vor, sondern ein dem Werkvertrag angenähertes «Zurufsprojekt». Solche Verträge sind vertragsgestalterisch besonders anspruchsvoll, weil bei den Gewährleistungsbestimmungen zwischen den funktionalen und den softwaretechnischen Anforderungen zu differenzieren ist.

[Rz 70] Ob der Anbieter sich auf einen Werkvertrag über ein agiles Softwareprojekt einlassen soll, ist seine unternehmerische Entscheidung, die er in Abwägung aller Chancen und Risiken zu treffen hat. Wenn bereits ein detailliertes Pflichtenheft besteht, ist davon abzuraten, denn agile Projekte setzen voraus, dass die Anforderungen an die Software bis zu einem gewissen Grad verhandelbar sind.<sup>149</sup> Das Gleiche gilt umgekehrt, wenn das Resultat nicht einmal in einer rudimentären Form beschrieben werden kann.<sup>150</sup>

[Rz 71] In der Praxis wird nach Lösungen gesucht, um die werkvertragliche Erfolgshaftung, den Anspruch des Kunden auf Kostensicherheit und die agile Methodik zu kombinieren. Erwähnenswert ist das nachfolgend beschriebene Modell des agilen Festpreises.<sup>151</sup> Ausgehend von den Anforderungen, die bei Vertragsabschluss bekannt sind, wird ein Projektumfang definiert. Diesem Projektumfang wird eine bestimmte Anzahl Leistungsmasseinheiten (Storypoints) zugeordnet. Jeder Storypoint hat einen Wert (Stunden- oder Tagessatz). Zu den einzelnen Anforderungen wird angegeben, wie viele Storypoints verbraucht werden. Streicht der Kunde eine Anforderung, kann er die frei gewordenen Storypoints für andere Anforderungen verwenden.

[Rz 72] Ob sich Vorschläge wie der agile Festpreis durchsetzen können, wird sich zeigen. Sie entlasten nicht von Diskussionen über Projektänderungen, denn die vom Kunden neu gewünschte Anforderung muss ja auch mit Storypoints bewertet werden.

[Rz 73] Die zuverlässigste Lösung bleibt, Verträge über agile Projekte als klassische Werkverträge auszugestalten. Dies bringt dem Kunden die gewünschte Sicherheit. Aber auch für den kompetenten Anbieter hält das Werkvertragsrecht Rechtsbehelfe bereit, allerdings nur dann, wenn der Anbieter seine Führungsrolle wahrnimmt.<sup>152</sup>

[Rz 74] Bei der Formulierung eines Werkvertrags ist den Besonderheiten agiler Projektmethoden Rechnung zu tragen. Dabei gilt es die folgenden Punkte zu beachten:<sup>153</sup>

1. Die Resultatbeschreibung muss so detailliert sein, dass sich daraus – allenfalls unter Rückgriff auf den Geschäftsgebrauch – Abnahmekriterien ableiten lassen.<sup>154</sup>

---

<sup>148</sup>Für HENGSTLER (FN 5), 116 ist die Ausgestaltung als Dienstvertrag (Auftrag) die einzige Möglichkeit, um die Vorteile des Scrum-Prozesses beibehalten zu können.

<sup>149</sup>Zu den weiteren negativen Indikationen, die gegen den Einsatz agiler Methoden sprechen, siehe vorne FN 66 und FN 67.

<sup>150</sup>KREMER (FN 5), 288 verlangt eine Beschreibung der Kernfunktionalität, der Designanforderungen und des Zwecks, für welchen der Kunde die Software einsetzen will.

<sup>151</sup>AUER-REINSDORFF (FN 5), Fussnote 9 sowie ausführlich OPELT/GLOGER/PFARL/MITTERMAYR (FN 7), 45 ff.

<sup>152</sup>Zur Führungsrolle des Anbieters vgl. BECKER/HABERFELLNER/PASTERNAK/GRÜNWARD (FN 8), 60.

<sup>153</sup>Hinweise zur Vertragsgestaltung finden sich auch bei FRÖHLICH-BLEULER (FN 4), Rz 349 ff. sowie zahlreich in der in FN 5 zitierten deutschen Literatur, in zusammengefasster Form namentlich bei KREMER (FN 5), 288 f.

<sup>154</sup>Zu den Minimalanforderungen an die Resultatbeschreibung siehe FN 150.

2. Die eingesetzte Projektmethode ist in den Grundzügen zu beschreiben.
3. Es braucht eine vertragliche Regelung des Vorgehens bei Vertragsänderungen (Change Requests).<sup>155</sup> Bei agilen Projekten hat diese Vertragsbestimmung eine besondere Bedeutung, weil Änderungen des Leistungsumfangs bei agilen Projekten typisch sind.<sup>156</sup> Innerhalb einer gewissen Toleranzgrenze ist deshalb von einer durch die agile Methodik abgedeckten Leistungskonkretisierung und nicht von einer Vertragsänderung auszugehen. Daneben gibt es jedoch «echte» Change Requests,<sup>157</sup> die eine Neuverhandlung der Leistung, der Terminen und der Kosten voraussetzen.
4. Art und Umfang der Dokumentation sind zu vereinbaren. Bei agilen Softwareprojekten ist das besonders wichtig, weil agile Projektmethoden weitgehend auf eine Dokumentation im herkömmlichen Sinn verzichten.<sup>158</sup>
5. Die Rolle des Product Owners muss von einem Vertreter des Anbieters wahrgenommen werden. Will der Kunde den Product Owner stellen, sollte sich ein Anbieter nicht auf einen Werkvertrag einlassen.<sup>159</sup>
6. Die Mitwirkung des Kunden ist als Pflicht und nicht als bloße Obliegenheit zu formulieren.
7. Wie bei jedem Werkvertrag ist die Entschädigung zu bestimmen (Pauschalpreis, Kostendach oder Aufwandsentschädigung).
8. Wird auf einen Pauschalpreis oder ein Kostendach verzichtet, verbleibt das Kostenrisiko beim Kunden. Die Parteien können jedoch ein Risksharing bei Kostenüberschreitungen vereinbaren, etwa indem der Anbieter seine Leistungen zu einem reduzierten Ansatz erbringt, wenn die Gesamtkosten einen bestimmten Betrag überschreiten.<sup>160</sup>
9. Es ist zu regeln, ob die in den einzelnen Sprints vorgenommenen Abnahmehandlungen verbindlich sind. Schweigt sich der Vertrag darüber aus, gelten Sprintabnahmen als rechtlich verbindliche Teilabnahmen.<sup>161</sup> Das ist gerechtfertigt. Dass damit jedoch die Gewährleistungsfristen zu laufen beginnen, ist von den Parteien weder beabsichtigt noch gerechtfertigt.<sup>162</sup> Dies sollte erst mit der Gesamtabnahme der Fall sein, weshalb ein diesbezüglicher Vorbehalt anzubringen ist. Eine Gesamtabnahme wird als unverzichtbar erachtet.<sup>163</sup> Sie hat sich jedoch auf die Integration des Gesamtsystems und auf die Gesamtperformance zu beschränken.<sup>164</sup>
10. Der Zahlungsplan soll sich wie bei jedem Projektvertrag am Projektfortschritt orientieren mit einem Zahlungsrückbehalt bis zur Schlussabnahme.
11. Es ist festzuhalten, ob die Immaterialgüterrechte an den Projektergebnissen auf den Kunden

---

<sup>155</sup>AUER-REINSORFF (FN 5), 95; KREMER (FN 5), 288.

<sup>156</sup>HENGSTLER (FN 5), 115.

<sup>157</sup>Siehe auch AUER-REINSORFF (FN 5), 95.

<sup>158</sup>Siehe Ziffer 2.5. Zu den Minimalanforderungen an die Dokumentation ohne besondere vertragliche Abrede siehe KREMER (FN 5), 287. Nach KOCH (FN 5), 117 darf der Kunde ohne anderslautende vertragliche Abrede eine Dokumentation erwarten.

<sup>159</sup>Zur herausragenden Bedeutung des Product Owners für die Leistungsbestimmung siehe vorne Ziffer 2.5; vgl. jedoch HENGSTLER (FN 5), 115, der die Übernahme der Rolle des Product Owners durch den Kunden fordert, um die Vorteile von Scrum realisieren zu können. Konsequenterweise sieht dieser Autor jedoch eine dienstvertragliche (auftragsrechtliche) Ausgestaltung als einzige sinnvolle Lösung für agile Softwareprojekte (HENGSTLER (FN 5), 116).

<sup>160</sup>Zu weiteren alternativen Vergütungsmodellen siehe FRANK (FN 5), 143 und SÖBBING (FN 5), 218.

<sup>161</sup>FRÖHLICH-BLEULER (FN 4), Rz 354 und SCHNEIDER, «neue» Projektmethoden (FN 5), 21.

<sup>162</sup>FUCHS/MEIERHÖFER/MORSBACH/PAHLOW (FN 5), 430.

<sup>163</sup>KREMER (FN 5), 288 und SCHNEIDER, «neue» Projektmethoden (FN 5), 21.

<sup>164</sup>KREMER (FN 5), 288 und SCHNEIDER, «neue» Projektmethoden (FN 5), 21.

übertragen werden oder ob sie beim Anbieter verbleiben, unter Einräumung eines Nutzungsrechts an den Kunden. Im Normalfall ist letzteres gerechtfertigt, da der Anbieter sich durch eine Übertragung der Rechte in seiner Geschäftsentwicklung zu sehr einschränken würde. Ohne explizite vertragliche Regelung besteht angesichts der engen Zusammenarbeit das Risiko, dass von einer gemeinsamen Rechteinhaberschaft auszugehen ist.<sup>165</sup>

12. Es kann sinnvoll sein, ein beidseitiges Kündigungsrecht vorzusehen.<sup>166</sup> Die Ausübung des Kündigungsrechts kann auf einen bestimmten Zeitpunkt (Checkpoint) beschränkt werden.<sup>167</sup> Der Checkpoint ist so anzusetzen, dass die Parteien einerseits Gelegenheit haben, die Zusammenarbeit zu testen, und dass andererseits die Kostenfolgen überschaubar bleiben. Die Auswirkungen der Kündigung auf die Entschädigung sowie die Verwendung der Arbeitsergebnisse sind zu regeln. Sinnvoll ist, dass der Anbieter die bis dahin bezahlte Entschädigung behalten kann und dass der Kunde umgekehrt die erarbeiteten Resultate weiterverwenden darf.

## 8. Schlussbemerkung

[Rz 75] Es ist bei Softwareprojekten nicht möglich, die vom Anbieter geschuldete Leistung bereits im Zeitpunkt des Vertragsabschlusses bis ins Detail zu beschreiben. Dies führt in jedem grösseren Softwareprojekt zu Diskussionen über den Leistungsumfang (Scope) und Leistungsänderungen (Change Requests). Agile Projektmethoden akzeptieren diese Unschärfe als gegeben und bieten Ansätze, um damit auf strukturierte Art und Weise umzugehen. Agile Projektmethoden sind jedoch nur Projektmethoden. Als solche beziehen sie sich in erster Linie auf das Vorgehen. Die für die rechtliche Qualifikation grundlegenden Parameter wie Leistung und Gegenleistung bleiben jedoch unberührt. Auch wenn agile Methoden auf den ersten Blick Neues bringen, so ist bei genauerer Betrachtung Vieles schon bei klassischen Projekten zumindest im Ansatz vorhanden. Die Unterschiede zwischen agilen und klassischen Projekten sind deshalb nicht grundsätzlicher Art, sondern nur graduell. Es rechtfertigt sich somit nicht, bei der rechtlichen Qualifikation von Verträgen über agile Softwareprojekte und bei der Vertragsgestaltung von der bisherigen rechtlichen Bewertung und Methodik abzuweichen.

---

Dr. URS EGLI, Rechtsanwalt, ist Partner in einer Wirtschaftskanzlei in Zürich ([www.epartners.ch](http://www.epartners.ch)) und hat sich auf Informatik- und Technologierecht spezialisiert.

---

<sup>165</sup>FRÖHLICH-BLEULER (FN 4), Rz 353.

<sup>166</sup>Nach dem Werkvertragsrecht steht ein solches Kündigungsrecht sonst nur dem Kunden zu (Art. 377 OR).

<sup>167</sup>Zu den Checkpoints und den Ausstiegsmöglichkeiten unter Scrum ausführlich OPELT/GLOGER/PFARL/MITTERMAYR (FN 7), 54 ff.