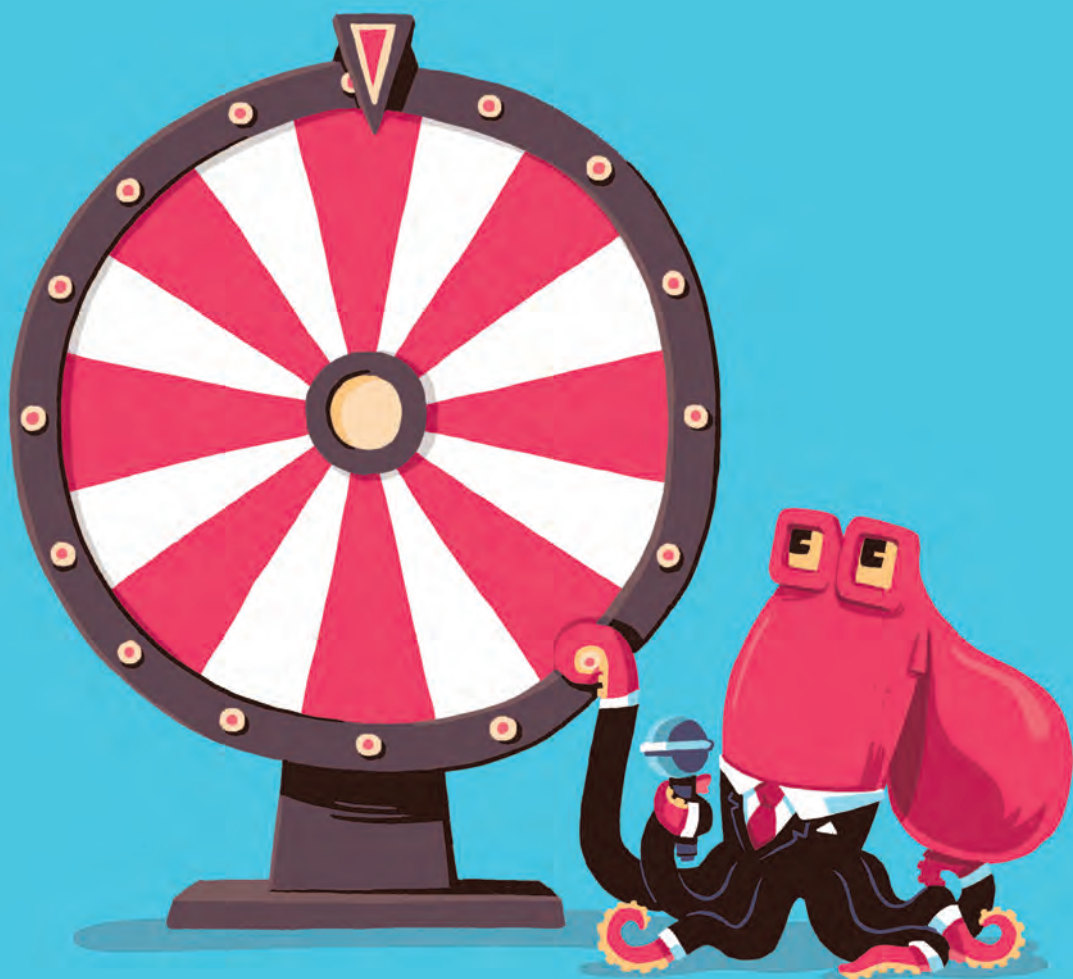


Kapitel 1

... wenn das Team in der Retrospektive voreilige Schlüsse zieht, indem es Symptome statt Probleme löst, und die moderierende Person die Teammitglieder dazu bringt, sich Zeit zu nehmen für die Ursachensuche hinter den Symptomen.

Das Glücksrad



Kontext

Bei der Planung ihrer ersten Retrospektive hat Sarah nicht viel Zeit. Sie muss eine Anleitung für die Moderation der Retrospektive finden, und ihre einzige Quelle, wie man eine Retrospektive moderiert, stammt aus dem Kurs für Scrum Master*innen. Der Kurs deckte alle Komponenten des Scrum-Frameworks ab, aber aufgrund seines breiten Umfangs war die Zeit, die für die Vermittlung eines einzelnen Themas, einschließlich der Moderation von Retrospektiven, aufgewendet wurde, begrenzt. Sarah entscheidet sich für die Methode »Anfangen-Aufhören-Weitermachen«, bei der das Team überlegt, womit es beginnen, womit es aufhören und womit es fortfahren soll. Sarah verwendet diese Aktivität als Kernstück der Retrospektive, und in der Phase »*Entscheiden, was zu tun ist*« stimmt das Team über die drei verschiedenen Themen ab und findet Dinge, mit denen es beginnen, aufhören und weitermachen möchte.

Anfangen-Aufhören-Weitermachen

Zur Vorbereitung dieser Aktivität erstellt die Person, die moderiert, drei Plakate auf Flipchart-Papier oder Whiteboards, eines mit der Überschrift »Anfangen«, eines mit der Überschrift »Aufhören« und eines mit der Überschrift »Weitermachen«. Die Teammitglieder erhalten Post-it-Zettel und Stifte. Einige ziehen es vor, dass alle Zettel dieselbe Farbe haben; andere bevorzugen Zettel in drei verschiedenen Farben, z.B. in Ampelfarben, damit die Zettel von den Boards entfernt werden können, ohne dass der Bezug zu einer der drei Möglichkeiten verloren geht. Die moderierende Person kann die Teilnehmenden auch auffordern, »Anfangen«, »Aufhören« oder »Weitermachen« oben auf die jeweiligen Zettel zu schreiben. Die Teammitglieder haben etwa 10 Minuten Zeit, um auf die Post-its zu schreiben, was ihrer Meinung nach begonnen, gestoppt und fortgesetzt werden sollte. Anschließend kleben sie die Post-its auf die vorbereiteten Boards. Das Team muss nun auf der Grundlage dieser Notizen seine Aktionspunkte auswählen, und wie dies geschieht, hängt sehr stark von der moderierenden Person, dem Inhalt der Post-it-Zettel und den Teammitgliedern ab.

Bei Sarahs erster Retrospektive hängt an dem »Anfangen«-Board ein Post-it-Zettel mit der Aufschrift »mehr Pair Programming¹«. Das lässt sich leicht verwirklichen: Das Team erstellt einfach einen Zeitplan, der jeden Tag drei Stunden Pair Programming vorsieht. Nach der Retrospektive sind alle zufrieden, aber im nächsten Sprint stellen sie fest, dass der Umfang des Pair Programming nicht wesentlich zugenommen hat. In der ersten Woche haben Bo und Peter etwas Pair Programming gemacht – allerdings nicht die 15 Stunden, die der neue Zeitplan vorsah. Kim und René haben sogar noch weniger gemeinsam im Pair Programming erar-

1. Anm. d. Übers.: Programmieren in Paaren, wobei zwei Entwickler*innen gleichberechtigt an einer Aufgabe arbeiten

beitet. Beide scheinen sich damit unwohl zu fühlen, sodass niemand von ihnen damit beginnen möchte.

Metakontext

Als Moderator* in einer Retrospektive treffen Sie manchmal auf Menschen, die der Meinung sind, dass eine Retrospektive Zeit von der eigentlichen Arbeit abzieht. Dieser Standpunkt könnte Sie dazu verleiten, so wenig Zeit wie möglich für die Retrospektive aufzuwenden, um alle zufriedenzustellen, sodass eine »Alles schnell beheben«-Aktivität die beste Idee zu sein scheint.

Antipattern-Lösung

Am einfachsten lässt sich eine Retrospektive durchführen, indem man drei Plakate mit den Überschriften Anfangen, Aufhören und Weitermachen aufhängt. Der nächste Schritt besteht darin, dass das Team ein Brainstorming durchführt, etwas auf Post-it-Zettel schreibt und die Zettel auf das entsprechende Plakat klebt. Der letzte Schritt ist noch einfacher. Sie beginnen mit den Punkten, die auf dem Anfangen-Plakat aufgelistet sind, hören mit den Punkten auf dem Aufhören-Plakat auf und fahren mit den Punkten auf dem Weitermachen-Plakat fort. Die Retrospektive ist in 15 Minuten beendet, und das Team kann sich wieder seiner »eigentlichen« Arbeit widmen. Selbst wenn Sie ein »Mache mehr davon«-Plakat hinzufügen würden, wären alle in kürzester Zeit wieder an ihren Schreibtischen.

Eigentlich ist dieser Ansatz nicht der einfachste Weg, eine effektive Retrospektive durchzuführen. Ich habe schon von »Retrospektiven« gehört (und ich verwende den Begriff »Retrospektive« hier sehr weit gefasst), bei denen das Team in einen Besprechungsraum ging, die Person mit der Verantwortlichkeit Scrum Master oder Projektleitung fragte, ob etwas geändert werden müsse, und als niemand etwas sagte, gingen alle wieder an die Arbeit.

Konsequenzen

Der Vorteil dieser Antipattern-Lösung besteht darin, dass sie eine sehr schnelle Methode zur Durchführung einer Retrospektive darstellt, aber die negativen Folgen können schwerwiegend sein. Manchmal hat man Glück und die Notizen auf den Post-its zeigen die wirklichen Probleme auf. Um sie zu beheben, hört das Team einfach mit den »Aufhören«-Elementen auf und widmet sich verstärkt den »Mache mehr davon«-Elementen. Oft beschreiben die Post-its jedoch nur die Symptome größerer Probleme, die eine grundlegendere Veränderung erfordern, als es die bloße Behebung der Symptome vermag. Daher sollte verhindert werden, dass die Symptome überhaupt auftreten, indem Sie das zugrunde liegende Problem beseitigen.

In unserem Beispiel von Titanic Software A/S funktioniert dieses Antipattern gut, wenn der einzige Grund dafür, dass nicht mehr im Pair Programming gearbeitet wird, darin besteht, dass die Leute es einfach vergessen und zu früheren Gewohnheiten zurückkehren. Die Schwierigkeiten des Teams können jedoch durchaus ein Symptom für ein interessanteres und herausfordernder zu lösendes Problem sein.

Hätten die Teammitglieder während der Retrospektive Zeit damit verbracht, die Ursache des Problems zu erforschen, hätten sie vielleicht die Gründe dafür herausgefunden, warum sie weniger im Pair Programming gearbeitet haben, als gemeinsam beschlossen wurde. Vielleicht konnten die Entwickler*innen den Nutzen nicht erkennen und wurden nie gefragt, ob sie Pair Programming machen wollten. Oder sie sahen zwar den Nutzen, aber die meisten von ihnen waren introvertiert und brauchten daher Stunden für sich allein, um über Informationen nachzudenken, bevor sie sie mit ihren Kolleg*innen teilten. Ein anderer Grund könnte sein, dass die Teammitglieder nicht wissen, wie Pair Programming funktioniert, also müssen sie erst lernen, wie das geht, bevor sie damit anfangen können. In unserem kleinen Team, in dem René eher ein *Großmaul* und Kim eine *Schweigsame* ist, könnte auch ein Mangel an psychologischer Sicherheit vorliegen. Keines dieser Probleme lässt sich allein dadurch lösen, dass man Menschen zur Zusammenarbeit zwingt.

Symptome

Wenn Sie Dinge hören wie »Warum reden wir immer wieder in der Retrospektive darüber?« und »Die Retrospektiven funktionieren nicht – nichts ändert sich außer der Art und Weise der Problembeschreibung«, dann sollten Sie sich fragen, ob Sie sich in einem *Glücksrad* befinden.

Wie bei einem echten Glücksrad erzielen Sie manchmal einen Gewinn, wenn Sie das Rad drehen – vorausgesetzt, das Rad landet auf dem eigentlichen Problem und nicht auf den Symptomen des Problems. Und die Wahrscheinlichkeit, dass das passiert, ist ungefähr so groß wie beim Drehen eines Glücksrads auf einem Jahrmakkt. Daraus resultiert als weiteres Symptom dieses Antipatterns, dass man bei den Retrospektiven immer wieder dieselben Themen diskutiert, und zwar aus dem einfachen Grund, dass die Probleme nie gelöst, sondern nur Pflaster auf die Symptome geklebt werden.

Refactoring-Lösung

Es ist verlockend, direkt von den Problemen zu den Lösungen überzugehen, und die meisten Entwickler*innen sind darauf trainiert, genau das zu tun. Leider werden die Probleme, die bei den Retrospektiven auftauchen, manchmal nicht sofort verstanden, sodass man sie untersuchen muss, bevor mit der Arbeit an den Lösungen begonnen werden kann. Nachdem die Daten gesammelt wurden, müssen Sie die Ursachen für die Probleme analysieren. Diese Phase wird, wie in der Einleitung beschrieben, als »*Erkenntnisse gewinnen*« bezeichnet und kann nicht übersprungen werden.

In dieser Phase können mehrere Aktivitäten durchgeführt werden. Eine einfache Möglichkeit besteht darin, nach der Geschichte hinter der Notiz auf dem Post-it zu fragen, um zu erfahren, was zu diesem Problem geführt hat. Andere Aktivitäten sind das Fishbone-Diagramm² (siehe Abb. 1–1 inkl. Beschreibung) und die »5 How«. Früher habe ich für die Ursachenanalyse die »5 Why³«-Methode verwendet, aber aufgrund der Erkenntnisse, die ich durch John Allspaw [Allspaw 2014] gewonnen habe, verwende ich jetzt »5 How«.

»5 How«

»5 How« ist eine iterative Fragetechnik, die dazu dient, die einem bestimmten Problem zugrunde liegenden Ursache-Wirkungs-Beziehungen zu untersuchen. Das Hauptziel dieser Technik besteht darin, die Ursache eines Fehlers oder Problems zu ermitteln, indem wiederholt gefragt wird: Wie ist es dazu gekommen? Jede Antwort bildet die Grundlage für die nächste Frage. Die Fünf im Namen leitet sich von einer anekdotischen Beobachtung über die Anzahl der Iterationen ab, die zur Lösung des Problems erforderlich sind.

Nur wenige Probleme haben eine einzige Ursache. Um mehrere Grundursachen aufzudecken, kann die Methode wiederholt werden, wobei jedes Mal eine andere Abfolge von Fragen gestellt wird.

Die Methode enthält keine festen Regeln darüber, wie die Reihenfolge der Fragen formuliert werden sollen oder wie lange die Suche nach weiteren Ursachen fortgesetzt werden soll. Selbst wenn die Methode genau befolgt wird, hängt das Ergebnis also immer noch vom Wissen und der Beharrlichkeit der beteiligten Personen ab.

John Allspaw erklärt, dass die Frage nach dem Warum dazu führen kann, dass man versucht, eine einzige Ursache für ein Problem zu finden, und sogar in Schuldzuweisungen enden kann [Allspaw 2014]. Die Frage nach dem »Wie« könnte stattdessen zu einer Geschichte und einer Reihe von Ursachen führen.

2. Anm. d. Übers.: Fishbone (engl.) bedeutet Gräte; die Bezeichnung Fishbone- oder Ishikawa-Diagramm wird im Deutschen analog zu Ursache-Wirkungs-Diagramm verwendet, siehe auch https://en.wikipedia.org/wiki/Ishikawa_diagram oder <https://de.wikipedia.org/wiki/Ursache-Wirkungs-Diagramm>.
3. Anm. d. Übers.: Bei dieser Methode werden fünf Warum-Fragen gestellt, um eine Ursache für ein Problem zu identifizieren.

Eine Schuldzuweisung ist natürlich nicht erwünscht, und die Suche nach einer einzigen Ursache in einem so komplexen Bereich wie der Softwareentwicklung und der Teamarbeit ist optimistisch und grenzt an Leichtgläubigkeit. Sie könnten dies sogar auf einer persönlichen Ebene tun: Versuchen Sie selbst das nächste Mal, wenn etwas nicht so läuft, wie Sie es sich vorgestellt haben, nach dem »Wie« und nicht nach dem »Warum« zu fragen.

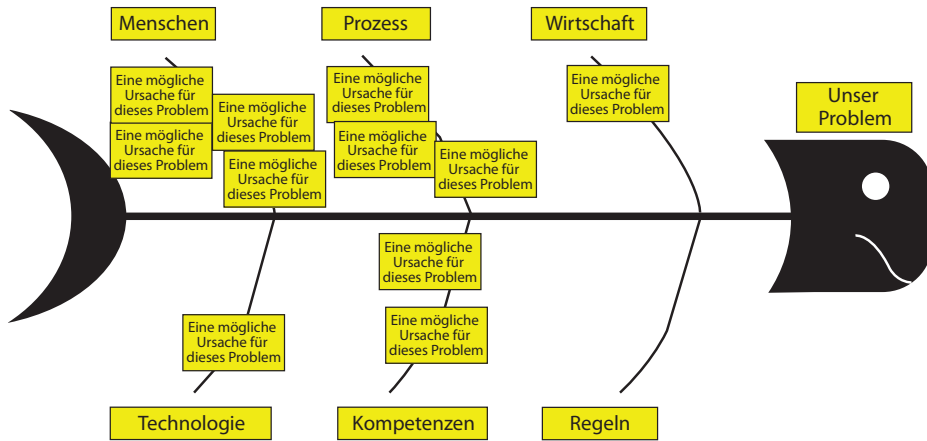


Abb. 1-1 Fishbone-Diagramm zur Ermittlung mehrerer Ursachen

Fishbone

Ein *Fishbone* ist eine Aktivität zur Ursachenanalyse, bei der in der Moderation ein Fischskelett mit verschiedenen Themenbereichen wie Menschen, Prozesse, Wirtschaft und Technologie als Fischgräten gezeichnet wird, um die verschiedenen Arten von Ursachen für ein Problem zu visualisieren. Dann bittet die Person, die moderiert, die Teammitglieder auf Post-it-Zettel die verschiedenen Ursachen aufzuschreiben, die ihnen einfallen, und sie auf der Fischgräte zu platzieren. Das Ergebnis ist ein Fisch, der viele verschiedene Ideen über die mögliche Ursache des Problems zeigt. Vielleicht stellen Sie fest, dass sich die meisten Ideen in einer bestimmten Kategorie bündeln, z.B. Prozess oder Technologie, und das gibt Ihnen Hinweise, an welcher Ursache Sie arbeiten sollten. Diese Technik wird im Allgemeinen in der Phase »Erkenntnisse gewinnen« einer Retrospektive eingesetzt. Sie wird auch als *Ishikawa-Diagramm* oder *Ursache-Wirkungs-Diagramm* bezeichnet.

Alle Aktivitäten der Phase »Erkenntnisse gewinnen« zur Ursachensuche beinhalten eine Vertiefung dessen, was auf den Post-it-Zetteln steht, um die wahren Geschichten dahinter zu finden.

Jedes Meeting, auch eine Retrospektive, hat einen Lebenszyklus von Divergenz und Konvergenz⁴ (siehe Abb. 1–2), wie im Buch *Facilitator's Guide to Participatory Decision-Making* von Sam Kaner [Kaner 2007] beschrieben. Wir beginnen hoffentlich in einem kleinen Raum, in dem wir uns über den Zweck des Treffens und die erwarteten Ergebnisse im Klaren sind; zumindest über das Thema des Treffens sollte man sich einig sein. Dann kommen wir in die Divergenzphase, in der die Diskussionen zu Meinungsverschiedenheiten und Meinungskonflikten führen. Als Nächstes kommt die Diskussionsphase, in der die Teilnehmenden weiter diskutieren, nicht um eine Einigung zu erzielen, sondern um mehr über das Thema zu erfahren. Nach einer gewissen Zeit für die Diskussionsphase muss das Treffen in eine Konvergenzphase übergehen, in der wir versuchen, den Umfang einzugrenzen und/oder eine Einigung zu erzielen. Wenn es zu keiner Einigung kommt, kann sich die Gruppe zumindest darauf einigen, nicht einer Meinung zu sein, und die Bedingungen und Gründe für die Uneinigkeit festlegen.

Zu Beginn ähnelt die Einigung auf das Thema der Phase »*Den Rahmen schaffen*« der Retrospektive. Der Divergenzteil einer Retrospektive ist die Phase des Sammelns von Daten und der Gewinnung von Erkenntnissen. Dies ist kein einfacher, aber ein sehr wichtiger Teil einer Retrospektive. Die Diskussionsphase kann mit dem letzten Teil der Phase »*Erkenntnisse gewinnen*« und dem Beginn der Phase »*Entscheiden, was zu tun ist*« zusammenhängen. Die Konvergenz ist der Kern der Phase »*Entscheiden, was zu tun ist*«, und wenn Sie am Ende der Retrospektive eine Entscheidung getroffen haben, schließen Sie die Retrospektive ab. Achten Sie darauf, alle fünf Phasen einer Retrospektive zu durchlaufen und Erkenntnisse zu gewinnen, anstatt voreilige Schlüsse zu ziehen und so zu einer verfrühten Konvergenz zu gelangen, nur um schnell zum Punkt zu kommen.

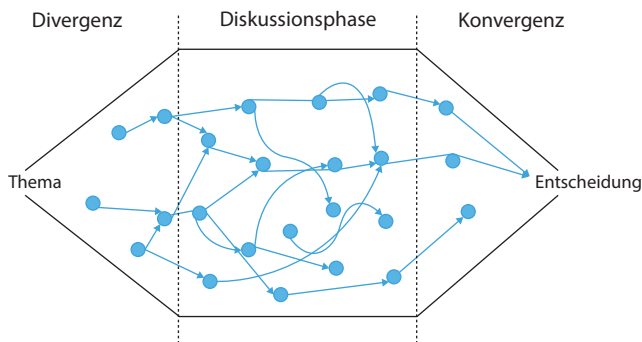


Abb. 1–2 Divergenz und Konvergenz in einem Meeting

4. Anm. d. Übers.: Divergenz = Unterschiedlichkeit/Widersprüchlichkeit, Konvergenz = Annäherung, Übereinstimmung, siehe auch das »Diamant-Modell« von Sam Kaner

Online-Aspekt

Wenn die Retrospektive online durchgeführt wird, können Sie ein Online-Tool verwenden, das das Team dazu zwingt, bei der Phase »Erkenntnisse gewinnen« zu beginnen und dort zu bleiben, bevor es zur Problemlösung übergeht. Wenn Sie Ihr eigenes Online-Dokument erstellen, können Sie jederzeit visualisieren, in welcher Phase der Retrospektive Sie sich befinden. Ich verwende oft ein mit »Google Drawings« erstelltes Diagramm für die Phase »Erkenntnisse gewinnen« und streiche das Feld für Vorschläge durch. Auf diese Weise werden die Teammitglieder visuell daran erinnert, keinen Vorschlag für ein Experiment hinzuzufügen, bevor wir die Phase »Entscheiden, was zu tun ist« erreichen (siehe Abb. 1–3).

Sie können die Phasen der Retrospektive auch visualisieren, indem Sie für jede Phase einen anderen Hut aufsetzen. Wenn die Teilnehmenden weiterhin vergessen, dass sie keine voreiligen Schlüsse ziehen sollen, können Sie lächelnd auf Ihren Hut zeigen.

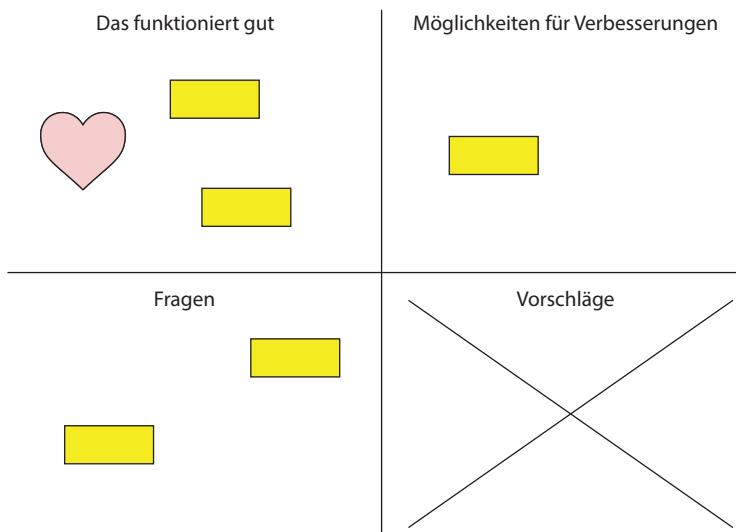


Abb. 1–3 Ein Online-Retrospektiven-Dokument mit durchgestrichenem Feld für Vorschläge

Persönliche Anekdote

Ich leitete ein Team in einem Unternehmen in Dänemark, das von einer großen ausländischen Organisation übernommen worden war. Das Unternehmen hatte seine eigene Art, wie es Dinge erledigte, und mochte es auch so. Leider beschloss die neue Organisation, das ursprüngliche Unternehmen in drei Teams aufzuteilen. Bei einer der Retrospektiven, die nicht lange nach der Übernahme stattfand, wurde unter anderem »Weniger Arbeit in Silos« auf die Tagesordnung gesetzt.

Wir alle verstanden, was damit gemeint war: Die Teammitglieder wollten zusammenarbeiten, anstatt als drei separate Teams parallel zu arbeiten, ohne sich gegenseitig zu helfen.

Da wir das Problem verstanden hatten, wurde ohne große Diskussion beschlossen, diesen Aktionspunkt bei dieser Retrospektive umzusetzen. Und bei der folgenden Retrospektive und der danach. Schließlich beschlossen wir, die Ursache für das Silo-Arbeiten zu untersuchen. Es stellte sich heraus, dass die für jedes Team festgelegten Leistungsindikatoren der Zusammenarbeit im Wege standen. Da jedes Team seine eigenen Ziele verfolgte, wurde es zu einem Nullsummenspiel, bei dem der Gewinn des einen Teams auf dem Verlust des anderen Teams beruhte. Auch die Planungsmeetings für die Teams fanden mit verschiedenen Personen in der neuen Organisation statt. Es wäre gut gewesen, dies von Anfang an erkannt zu haben, um zu verstehen, ob dies etwas war, das sie ändern konnten, oder etwas, mit dem sie lernen mussten, umzugehen (siehe auch Kap. 3 »Die Ursuppe«).

Nun zu etwas ganz anderem: Ich arbeite hauptsächlich mit IT-Leuten und finde es immer schwierig, die Menschen dazu zu bringen, in der offenen Diskussion oder in der Diskussionsphase zu bleiben. Es fühlt sich schon fast körperlich schmerzhaft an, diese Phase der Retrospektive offenzuhalten, bevor ich ihnen erlaube, sich in die Lösungsphase zu stürzen. Meiner Erfahrung nach ist jedoch jede investierte Minute in der offenen Diskussionsphase wertvoll. Hier kommen oft neue Einsichten und Ideen zum Vorschein. Dave Snowden hat oft über verfrühte Konvergenz als Teil der Komplexitätstheorie geschrieben, und er erläutert, was uns dabei entgehen könnte: »Dinge offenzuhalten, ihnen erlauben, in feingranularere Objekte zu verfallen, und dann zu sehen, wie sie sich wieder zusammenfügen und gemeinsam entwickeln« [Snowden 2015].

Als ich schon viel Erfahrung mit IT-Teams hatte, wurde ich gebeten, die Mitarbeitenden eines Kunstmuseums dabei zu unterstützen, Ideen zu generieren. Ich war nicht auf das vorbereitet, was dann passierte. Es war extrem einfach, die Teilnehmenden in die offene Diskussionsphase zu bringen, in der alles möglich ist und in der sie sich nicht auf eine Lösung oder einen Weg festlegen sollten. Ich hatte das Gefühl, dass es sehr gut lief und dass es mir gelungen war, sie an diese Arbeitsweise heranzuführen.

Aber dann begannen meine Probleme. Es war fast unmöglich, sie dazu zu bringen, Entscheidungen zu treffen, Ideen einzuschränken und Möglichkeiten auszuschließen. Ich ertappte mich dabei, wie ich von Gruppe zu Gruppe lief und Post-it-Zettel vom Board nahm, die sie in dem Moment, in dem ich mich einer anderen Gruppe zuwandte, wieder anbrachten. Als ich jemanden sagen hörte: »Benötigen wir überhaupt ein Kunstmuseum?«, verspürte ich leichte Panik, denn wir waren hier, um ein Brainstorming für eine App durchzuführen und nicht, um eine existierende App zu verbessern.

tenzielle Diskussion darüber zu führen, ob wir ein Kunstmuseum brauchen. Ich musste mich auf dieses neue Publikum einstellen, und zwar schnell.

Für mich fühlte es sich fast so an, als würde ich mit einer anderen Spezies zusammenarbeiten. Ich musste viel direkter sein als bei anderen Gruppen, mit denen ich bisher gearbeitet hatte, und ich musste mehr Energie aufwenden, um diese Gruppe dazu zu bringen, in die letzte Phase eines Meetings zu gehen und zu einer Art Konvergenz zu kommen. Am Ende gelang es uns nur durch das Versprechen, dass die anderen Ideen für die Zukunft aufbewahrt werden, uns auf einen Schwerpunkt für die nächste Phase zu einigen. Es ist interessant, wie kulturell abhängig dieses Verhalten oder Denkmuster ist. Dies wird auch in Kapitel 23 »*Unterschiedliche Kulturen*« beschrieben. In manchen Kulturen versucht man, Konflikte oder Meinungsverschiedenheiten zu vermeiden. Menschen aus einer solchen Kultur (und das kann innerhalb eines bestimmten Landes, Unternehmens oder Teams der Fall sein) werden so schnell wie möglich einen Konsens anstreben und damit vielleicht die Chance verpassen, dass aus ihrer Interaktion neue Ideen entstehen.