

# Kapitel 7

*... wenn das Team glaubt, so gut geworden zu sein,  
dass es keine Retrospektiven mehr braucht, und die  
moderierende Person dem Team zeigt, wie es lernen  
kann, sich weiter zu verbessern.*

# *Nichts zu besprechen*



## Kontext

Das Team macht seine Sache wirklich gut. Den Teammitgliedern gefällt die agile Arbeitsweise, und ihr Arbeitsprodukt bringt den Nutzenden durchweg einen Mehrwert. In den letzten beiden Retrospektiven gab es keine Probleme zu lösen, und alle Post-it-Zettel waren grün, bis auf ein paar über schlechtes Wetter (es ist schließlich Dänemark). Sarah hört zufällig, wie sich René und Andrea darüber unterhalten, ob Sarah von ihnen erwartet, dass sie Probleme erfinden.

Es scheint, dass die Retrospektiven nicht mehr nötig sind, da alles gut läuft. Sarah fordert das Team auf, gründlicher nachzudenken, ob es Probleme gibt, die angegangen werden müssen, aber ohne Erfolg. Obwohl sie frustriert ist, muss Sarah dem Rest des Teams zustimmen, dass die Retrospektiven vielleicht nicht mehr erforderlich sind.

## Metakontext

Ich habe dieses Muster des Lebenszyklus von Retrospektiven in fast jeder Organisation und in jedem Team, in dem ich Retrospektiven eingeführt habe, beobachtet: Zunächst könnte das Team die Retrospektiven als Zeitverschwendung empfinden – bis es lernt, wie hilfreich und effizient (und unterhaltsam) sie sein können. Dann kommt eine Phase, in der die Teammitglieder viele Dinge finden, über die sie während der Retrospektiven sprechen können. Es fühlt sich gut an, etwas gegen die Probleme in der Teamarbeit, der Kommunikation, der Technologie oder dem System, an dem gearbeitet wird, zu unternehmen. Doch nach einiger Zeit gehen dem Team »die Probleme aus«. Die Diskussionen drehen sich um die gleichen Themen, und die Retrospektiven werden schal und nutzlos. Es stellt sich eine Routine ein, und die Retrospektiven werden zu einem weiteren Meeting, das man einfach hinter sich bringen möchte.

## Antipattern-Lösung

Die offensichtliche Lösung besteht darin, jedes Mal, wenn eine Retrospektive geplant ist, die Teammitglieder zu fragen, ob sie etwas Neues zu besprechen haben, und wenn die Antwort nein lautet, wird die Retrospektive ausgesetzt.

## Konsequenzen

Um zu verstehen, welche Folgen es hat, wenn auf eine Retrospektive verzichtet wird, sollte man sich zunächst die erwarteten Vorteile vor Augen führen: eine gemeinsame Geschichte über das, was geschehen ist oder geschieht, eine Möglichkeit, Dampf abzulassen, wenn etwas nicht gut läuft, und eine gemeinsame Ent-

scheidungsfindung über das weitere Vorgehen. Auch die persönlichen Faktoren sind bei Retrospektiven wichtig. Hier kommen oft alle zusammen, um zu lachen, Gefühle zu verschiedenen Themen auszutauschen und zu lernen, wie man sich gegenseitig oder sich selbst helfen kann. Richtig moderiert, können Retrospektiven dazu beitragen, als Team zu lernen und zu wachsen. Oft ist dies die einzige Gelegenheit für ein Team, über das Offensichtliche hinauszugehen und sich mit den tieferen Problemen zu befassen. Wie J. P. Morgan sagte, gibt es zwei Gründe für jede Entscheidung und jede Handlung: den offiziellen Grund und den wahren Grund.

## Symptome

Man hört im Team: »Wir brauchen keine Retrospektiven mehr«, »Es gibt nichts zu besprechen« oder »Wir sind jetzt ein gutes Team – wir möchten keine Zeit mehr damit verschwenden, Probleme zu erfinden«.

## Refactoring-Lösung

Wie der Untertitel des Buches von Diana Larsen und Esther Derby, *Making Good Teams Great* [Larsen & Derby 2006], andeutet, sind Retrospektiven für alle Teams nützlich, nicht nur für Teams, die in irgendeiner Weise dysfunktional sind. Wir können immer besser werden in dem, was wir tun: Wir können uns von gut zu großartig entwickeln.

Weltklasse-Skifahrer\*innen versuchen immer noch, ihr Skifahren zu verbessern: Sie überprüfen ständig, was sie tun, und passen ihre Technik, ihren Schlaf oder ihre Ernährungsgewohnheiten entsprechend den Ergebnissen an. Auch wenn Sie in eine Zahnarztpraxis gehen, um Ihre Zähne reinigen zu lassen, oder in eine Werkstatt, um die Räder Ihres Autos auswuchten zu lassen, müssen Sie das regelmäßig wiederholen. Das sage ich manchmal auch zu einem Team, das sich in der Phase seiner Retrospektivenreise befindet, wo es glaubt, keine Retrospektiven mehr zu brauchen.

Und dann führe ich etwas Neues in ihre Retrospektiven ein. Ich probiere eine neue Art von Retrospektive aus, z. B. eine *positive Retrospektive* oder eine *Teamradar-Retrospektive* (siehe Abb. 7-1), oder ich mache eine *Zukunftsperspektive*, die aufzeigt, was sie befürchten oder hoffen, das passieren wird. Wie im Antipattem »*Negatives Team*« (Kap. 21) beschrieben, kann das Team nicht nur aus den positiven Aspekten eines Problems lernen, sondern auch aus den negativen – und welche guten Praktiken noch besser gemacht werden können.

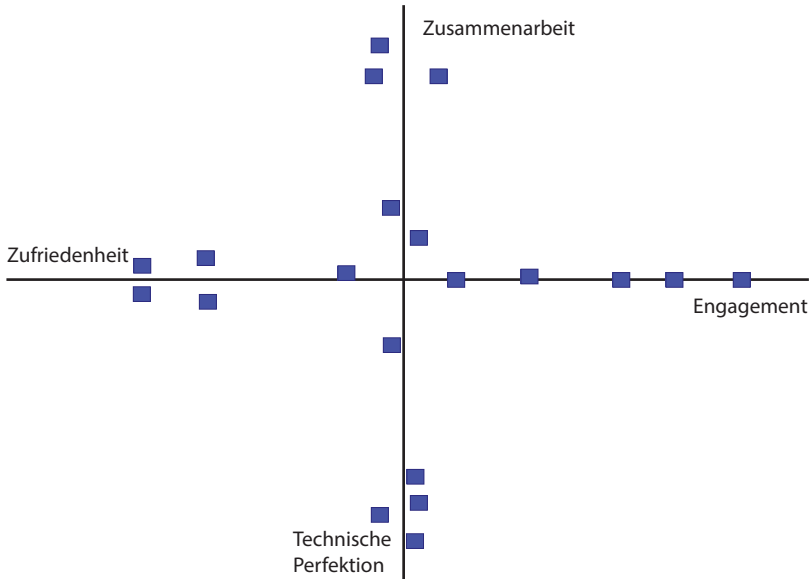


Abb. 7-1 Teamradar-Retrospektive mit vier Themen

### Zukunftsperspektive

Eine *Zukunftsperspektive* ist wie eine Retrospektive, aber der Fokus liegt auf der Zukunft, nicht auf der Vergangenheit. Für eine Zukunftsperspektive erstelle ich an dem Board einen Zeitstrahl, der heute (am Tag der Zukunftsperspektive) beginnt und in die Zukunft reicht, vielleicht drei Monate, vielleicht ein Jahr oder vielleicht bis zum nächsten Release. Dann bitte ich die Teammitglieder, sich vorzustellen, sie befänden sich in der Zukunft, am Ende des Zeitstrahls. Wenn ich mich traue, bitte ich sie, die Augen zu schließen und zu versuchen, sich vorzustellen, wie die Welt »jetzt« aussieht. Dann öffnen sie die Augen und kleben Post-it-Zettel an das Board, auf denen sie alle positiven und negativen Ereignisse beschreiben, die vom Anfang bis zum Ende des Zeitstrahls stattgefunden haben.

Ein typischer Einwand gegen diese Übung ist, dass die Menschen nicht in die Zukunft sehen können. Als moderierende Person können Sie zugeben, dass sie die Zukunft nicht vorhersagen können, aber erklären, dass sie ihre Hoffnungen und Befürchtungen in Bezug auf das, was geschehen wird, mitteilen werden.

Wenn die Teilnehmenden ihre erwarteten positiven und negativen Ereignisse eingetragen haben, gehen Sie diese durch und besprechen, was zu diesen Ereignissen geführt hat – die Ursachen oder die Geschichten dahinter. Das Ergebnis einer Zukunftsperspektive kann sein, dass die Teammitglieder Richtlinien beschließen, die es ihnen ermöglichen, so zu arbeiten, wie es für sie am besten funktioniert, z.B. keine morgendlichen Treffen abhalten. Es könnte auch ein Hinweis an das Management sein, was es braucht, damit dieses Team erfolgreich ist – zum Beispiel sollte das Team zusammenbleiben und nicht von anderen Projekten im Unternehmen in seiner Arbeit behindert werden. Außerdem



könnten die Teammitglieder, wie bei einer normalen Retrospektive, Experimente identifizieren: Dinge, die sie ein paar Wochen lang tun können, und sich die Auswirkungen ansehen, wie z.B. ein Peer-Review aller Arbeiten oder Mob Programming<sup>3</sup> einmal im Monat.

Für mich ist ein zusätzlicher Vorteil, dass ich viel über die Vergangenheit der Teammitglieder erfahre. In ihren Befürchtungen beschreiben sie, welche negativen Dinge sie zuvor erlebt haben könnten. In ihren Hoffnungen beschreiben sie, wie sie im Wesentlichen arbeiten möchten. Daher führe ich oft zu Beginn der Zusammenarbeit eine Zukunftsperspektive mit einem neuen Team durch.

- a. Wie beim Pair Programming, nur dass das ganze Team gemeinsam programmiert und dabei die von Woody Zuill aufgestellten Regeln befolgt (Anm. d. Übers.: siehe z.B. <https://woodyzuill.com/>).

Eine weitere nützliche Übung ist »Die Schiff-Retrospektive« (Abb. 7–2), die auch eine gute Möglichkeit ist, in einer Retrospektive etwas Neues auszuprobieren.

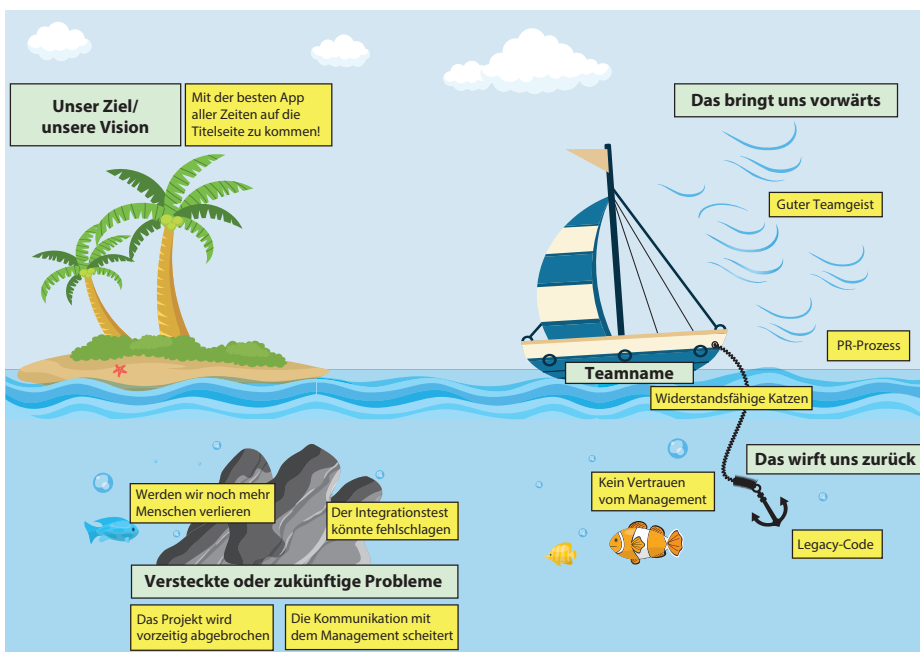


Abb. 7–2 Die Schiff-Retrospektive

### Die Schiff-Retrospektive

Die *Schiff-Retrospektive* ist um das Bild eines Schiffes vor einer unbewohnten Insel, mit Wind in den Segeln, einem Anker und einigen unter Wasser versteckten Klippen herum aufgebaut. Das Team muss sich zunächst darauf einigen, was sich auf der Insel befindet, d. h., was das Traumszenario, ihre Vision, ist. Dann sollte dem Schiff ein Name gegeben werden, denn die Entscheidung über den Namen für das Schiff macht deutlich, wie Menschen im selben Team sehr unterschiedliche Ansichten über den bestimmenden Faktor dieses Teams haben können. Ich überspringe diesen Schritt manchmal aus Zeitgründen. Nun schreiben die Teammitglieder einzelne Post-it-Zettel für die positiven Dinge und platzieren sie auf dem Wind, der die Segel in Bewegung setzt, damit das Schiff die Insel erreicht. Der nächste Schritt ist der Anker, der für die Dinge steht, die uns von der Insel fernhalten und das Vorankommen erschweren. Der letzte Schritt sind die Klippen, d.h. die Dinge, bei denen wir uns nicht sicher sind, aber befürchten, dass sie passieren könnten. Dann ist die Phase »Daten sammeln« abgeschlossen, und das Team kann mit den Phasen »Erkenntnisse gewinnen« und »Entscheiden, was zu tun ist« beginnen. Die Schiff-Retrospektive wird unter anderem in dem Buch *Getting Value Out of Agile Retrospectives: A Toolbox of Retrospective Exercises* [Gonçalves & Linders 2014] beschrieben.

Eine andere Möglichkeit besteht darin, die Teammitglieder zu fragen, was das Worst-Case-Szenario ist oder wie sie arbeiten würden, wenn sie weniger Zeit, mehr Zeit, weniger Leute, mehr Leute usw. hätten. Versuchen Sie, ihnen ein anderes Weltbild zu vermitteln, und beobachten Sie, was ihnen dazu einfällt. Wenn sie nicht erkennen können, was sie ändern müssen, um sich zu verbessern, liegt das in der Regel daran, dass sie sich so sehr an den jetzigen Zustand gewöhnt haben, dass es ihnen schwerfällt, sich eine andere Situation für das Team vorzustellen.

Manche Teams haben gelernt, sich in der Retrospektive nur mit ihrer Kommunikation, ihrer Zusammenarbeit und ihren Prozessen zu befassen. Obwohl diese Überlegungen wichtig sind, kann es sich auch lohnen, im Teil »Daten sammeln« der Retrospektive harte Daten zu betrachten wie z.B. Burndown-Charts, Übersichten über Regressionstests, Peer-Review-Statistiken, Feedback von Personen, die die Software nutzen.

Die Wahl eines Themas für die Retrospektive kann auch eine Möglichkeit sein, sich auf ein bestimmtes Thema zu konzentrieren und es zu vertiefen. Das Thema kann ein bestimmtes Release, die Teststrategie, die Art und Weise, wie wir als Team und individuell lernen, die Architektur oder was auch immer für dieses Team interessant sein könnte, sein.

Manchmal können Sie sich erlauben, in der Retrospektive eine etwas aktivere Rolle einzunehmen. Wenn ich zum Beispiel Fragen und Kommentare höre wie: »Warum werden wir nicht einfach besser darin, Zeitpläne für Aufgaben zu schätzen? Dann brauchen wir nicht so viel zu reden, sondern melden uns einfach,

wenn die Arbeit fertig ist« oder: »Müssen wir besprechen, was gut läuft? Können wir uns nicht einfach auf die Dinge konzentrieren, die schlecht laufen?«, dann stelle ich manchmal gerne Fragen dazu, wie die Teammitglieder wirklich arbeiten wollen. Glauben sie an Agilität? Und an Wertschätzung? Und welche Ziele verfolgen sie mit diesem Team?

Sie können auch eines der Modelle zum Ermitteln der Wirksamkeit, wie z.B. »Agile Fluency«, verwenden, um dem Team bewusst zu machen, wie es in Bezug auf agile Methoden, Standards für die Codequalität oder Teamzufriedenheit arbeitet. Dann können Sie eine Diskussion darüber beginnen, wo es hinwill oder wo es dachte, auf welcher Stufe es sich befindet.

### **Agile Fluency**

Agiles Arbeiten zu lernen ist wie das Erlernen einer neuen Sprache. Man lernt zwar die Sprache, aber wenn etwas Unerwartetes passiert, wechselt man wieder zurück zur gewohnten Sprache. Ich kann zum Beispiel Deutsch sprechen, aber wenn ich mir den Zeh an einem Tischbein anstoße, fluche ich auf Dänisch. Wenn ich eine Sprache nicht fließend spreche, wechsele ich zu einer Sprache, die ich fließend beherrsche, wenn ich überrascht, wütend oder ängstlich bin. Das Gleiche gilt für die agile Arbeitsweisen in Teams. Die Teammitglieder haben vielleicht einige Kurse besucht und gelernt, wie man mit Scrum arbeitet, aber wenn etwas Unerwartetes passiert, wechseln sie zum Wasserfallmodell, weil sie das kennen und verstehen. Das Ziel von Agile Fluency ist es, zu ermitteln und zu visualisieren, in welchem Maße ein Team mit agilen Methoden vertraut ist.

Diana Larsen und James Shore haben das »Agile Fluency Project« mitbegründet, mit dem Ziel, Coaches dabei zu unterstützen, Teams in die Lage zu versetzen, auf der Stufe von Agile Fluency zu arbeiten, die den Bedürfnissen ihres Unternehmens am besten entspricht.

Der Sinn dieser überarbeiteten Lösung besteht darin, Teammitglieder dabei zu unterstützen, Wege zu finden, um jeden Aspekt des Teams und seiner Arbeit zu verbessern, und sei es auch nur in kleinen Schritten.

## **Online-Aspekt**

Wenn die Retrospektive online durchgeführt wird, gelten die gleichen Punkte. Der einzige Unterschied besteht darin, dass es schwieriger ist, die Agenda einer Online-Retrospektive zu ändern, sodass Sie dieses Antipattern vielleicht nicht ändern können, wenn Sie darauf stoßen, aber Sie können es für die nächste Retrospektive einplanen.



## Persönliche Anekdote

Wie bereits erwähnt, habe ich diesen speziellen Teil des Lebenszyklus von Retrospektiven bei zahlreichen Teams erlebt. Bei einem bestimmten Team war es auch für mich schwer zu erkennen, wie es sich verbessern könnte. Es schien alles genau richtig zu machen und die Teammitglieder waren dabei sehr nett zueinander.

Ich beschloss, eine Retrospektive rund um das Wort »mutig« vorzubereiten. Die Phase »*Den Rahmen schaffen*« beinhaltete eine Frage dazu, wie mutig jedes Teammitglied im letzten Monat gewesen war. Ich habe nur nach einem Beispiel gefragt, das sowohl bei der Arbeit als auch im Privatleben vorkommen konnte. (Natürlich sind wir auch die Aktionspunkte der letzten Retrospektive durchgegangen und haben besprochen, ob die Experimente erfolgreich waren, was sie gelernt hatten und was sie weiterhin tun werden.)

In der Phase »*Daten sammeln*« habe ich die Teammitglieder gebeten, sich drei Dinge zu überlegen, die sie tun würden, wenn sie eines Morgens aufwachen und zehnmal mutiger als sonst wären. Mindestens zwei dieser Dinge sollten einen Bezug zu ihrer Arbeit haben. Man könnte einwenden, dass alle drei Dinge mit ihrer Arbeit zusammenhängen sollten, aber ich habe mich aus zwei Gründen dafür entschieden, einen möglichen privaten Rahmen einzubeziehen.

Erstens kann diese Frage zu einigen lustigen Fakten übereinander führen, zu Dingen, die sie schon immer einmal ausprobieren wollten, sich aber nie getraut haben. Spaßige Fakten können das Team zum Lachen bringen und den Zusammenhalt der Gruppe stärken. Manchmal finden die Teammitglieder sogar heraus, dass sie sich gegenseitig helfen können, ein »mutiges Ziel« zu erreichen. Als zum Beispiel jemand sagte, er wolle fliegen lernen, lud ihn eine Teamkollegin ein, bei ihrem nächsten Flug ihr Co-Pilot zu sein.

Zweitens: Indem Sie in der Frage Raum für Gedanken jenseits der Arbeit lassen, ermutigen Sie die Teilnehmenden vielleicht dazu, über einen wichtigen persönlichen Wunsch zu sprechen, über den sie bei der Arbeit normalerweise nicht reden würden, der sie aber dennoch in Gedanken beschäftigt. Wenn sie ihre wichtigen Träume oder Ziele mitteilen, anstatt zu versuchen, ihre Gedanken zu unterdrücken, sind sie weniger abgelenkt und können sich besser auf ihre Arbeit konzentrieren.

Das Ergebnis dieser Übung waren so unterschiedliche Antworten wie »direkt mit den künftigen Nutzenden des Systems im nächsten Einkaufszentrum sprechen«, »Mob Programmierung ausprobieren«, »um Hilfe bitten, wenn ich auf etwas stoße, das ich nicht verstehe« und »mich für die Backend-Entwicklung umschulen zu lassen«. In der nächsten Phase, »*Erkenntnisse gewinnen*«, ging es um die Geschichten hinter ihren Antworten. Wenn die Personen wollten, konnten sie die Erfahrungen mitteilen, die zu ihrer Liste der Dinge führten, die sie tun würden,

wenn sie zehnmal mutiger wären. Hatten sie es bereits versucht und waren gescheitert? Was ist passiert, als sie es versucht haben? Was hat sie dazu bewogen, es zu versuchen, und was haben sie sich davon versprochen?

In der Phase »*Entscheiden, was zu tun ist*« wählte jede Person eine Idee aus, auf die sie sich konzentrieren wollte oder jemand anderem dabei helfen wollte, sich darauf zu konzentrieren. Und dann unterstützte ich sie dabei, herauszufinden, was ein weniger beängstigender erster Schritt sein könnte. Die Retrospektive brachte einige neue Themen an die Oberfläche, da die Teammitglieder gezwungen waren, auf eine andere Art zu denken. Einigen fiel es leichter als anderen, aber alle Anwesenden machten sich Gedanken über ihr normales Verhalten und darüber, ob etwas auf persönlicher Ebene verbessert werden könnte.

Ein weiterer Aspekt dieses Antipatterns ist, dass manche Teams von vornherein denken, dass Retrospektiven aus einem bestimmten Grund keinen Wert für sie hätten. Kürzlich hat mir jemand gesagt, dass »Agile Coaches und Scrum Master\*innen nervös werden, weil überall in ihrem Umfeld DevOps-Methoden verwendet werden. Wenn Scrum Master\*innen und Agile Coaches über DevOps Bescheid wüssten, würde sich ihre gesamte Sichtweise auf Retrospektiven ändern. Bei DevOps-Methoden geht es darum, Ergebnisse zu messen. In diesem Fall sollte sich die Retrospektive auf diese Ergebnisse konzentrieren, und viele DevOps-Ergebnisse sind technischer Natur: Fehlerrate in der Produktion, Feature-Zykluszeit, mittlere Zeit bis zur Wiederherstellung des Service. Man kann keine Diskussionen über diese Dinge führen, ohne etwas über DevOps zu wissen.«

Ich mag Aussagen wie diese sehr, weil sie mir die Möglichkeit geben, zu erklären, warum ich Retrospektiven für wertvoll halte. Zumindest wird die Person, die diese Aussage gemacht hat, Retrospektiven nicht nur um der Retrospektiven willen einführen. Ich habe auf diesen Kommentar geantwortet, dass erstens, auch wenn die Leute DevOps-Methoden einsetzen, sie immer noch Menschen sind und die gleichen Probleme haben wie alle anderen auch.

Zweitens, auch wenn ich weiß, dass die meisten Leute bei einer Retrospektive beim Sammeln von Daten an Daten zu Vorfällen und Gefühlen denken, sehe ich oft auch Burndown-Charts, Regressionstestergebnisse, Wiederherstellungszeiten und Antwortzeiten. Manchmal leite ich Retrospektiven an, bei denen das Team beschlossen hat, sich nur mit technischen Themen zu befassen, was sehr gut funktionieren kann. Oft sprechen wir aber auch über zwischenmenschliche Themen, weil wir uns Fragen stellen wie: »Wie kam es zu dieser Antwortzeit?« oder: »Warum haben wir dieses Testergebnis erhalten?«

Drittens weiß ich, dass ich von meinem technischen Hintergrund profitiere, und wenn Leute in technische Diskussionen eintauchen, kann ich sie manchmal abkürzen, weil ich verstehe, was sie sagen, und weiß, wann sie anfangen, sich zu wiederholen. Aber viele moderierende Personen haben diesen Hintergrund nicht

und sind trotzdem perfekt in der Lage, Retrospektiven zu moderieren. Wir als Moderator\*innen sind dazu angehalten, unsichtbar zu sein und die Diskussionen zu unterstützen. Mit dem richtigen Verständnis für Körpersprache und Teamdynamik ist es dennoch möglich, ein Meeting zu moderieren, bei dem es um etwas geht, von dem man thematisch nichts weiß.

Viertens, und das ist spezifisch für DevOps-Methoden, moderiere ich derzeit Retrospektiven für zwei Teams, ein Betriebsteam und ein Entwicklungsteam, die gerade zu einem DevOps-Team zusammengelegt werden. Sie haben spezifische Probleme im Zusammenhang mit der Zusammenführung ihrer separaten Teams zu einem einzigen. Die Traumvorstellung ist natürlich, dass sie als ein Team arbeiten und dass die Entwickler\*innen nicht nur Sympathie für den Betrieb haben, sondern auch Empathie, weil sie lernen, was im Betrieb gemacht wird – und vice versa. Das bringt einige Herausforderungen mit sich, wenn es darum geht, das Wissen und die Fähigkeiten der anderen zu respektieren und eine Lernumgebung zu schaffen, in der sie voneinander lernen, damit sie Daten sammeln und auf diese Daten reagieren können. Es gibt viele Spannungen zwischen diesen beiden Teams, und die Retrospektiven helfen ihnen, sich gegenseitig zu respektieren, wenn auch langsam.