

# „WIR GLAUBEN NICHT AN SCHÖNE WORTE“

## INTERVIEW MIT CCD-MITBEGRÜNDER

### RALF WESTPHAL

„Sauberer Code“ – das klingt nach traditionellen Werten aus dem deutschen Handwerkswesen und gewinnt mehr und mehr Anhänger. Die Initiatoren der CCD-Initiative [Clean-Code-Developer], Ralf Westphal und Stefan Lieser, sind gefragte Ratgeber auf Konferenzen und bei Unternehmenslenkern. Ihre XING-Gruppe hat inzwischen fast 3.000 Mitglieder, Tendenz steigend. Ein Grund für OBJEKTSpektrum, mit Ralf Westphal über die Ideen und die Zukunft von CCD zu sprechen.

**Die Clean-Code-Developer Initiative ist nicht mehr ganz neu. Was war damals eigentlich der Anlass für die Initiative?**

Der unmittelbare Anlass war, dass Stefan Lieser und ich das Buch „Clean Code“ von Robert C. Martin gelesen hatten. Das gefiel uns sehr gut. Wir hatten das Gefühl, davon könnten viele Entwickler profitieren. Doch wir waren skeptisch, ob die Lektüre des Buchs ausreichen würde, um wirklich eine Veränderung in der Praxis zu bewirken. Unser Hauptbedenken war, dass die Leser sich von der Fülle des Materials erschlagen fühlen könnten. Denn was nützt es, wenn man ein gutes Buch liest, zustimmend nickt, aber dann diese ganze Zustimmung nicht in den Alltag übertragen kann? Bei aller Qualität von Clean Code haben wir also ein didaktisch/methodisches Problem gesehen. Dazu kam, dass uns nicht alle Sichtweisen im Buch gleich wichtig erschienen und andere für uns bedeutsame ganz fehlten. Die Vision von „Clean Code“ war für uns größer. Insofern war das Buch wirklich nur Anlass und Ausgangspunkt, etwas, das in uns schon länger geschlummert hatte, endlich in eine Form zu gießen und zu veröffentlichen.

Deshalb ist CCD nicht einfach eine Liste der Prinzipien und Praktiken aus Martins Buch. Uns war es wichtig, mit der Initiative eine Grundhaltung zu transportieren, die über „Clean Code“ hinausgeht. Wir haben von den konkreten Buchinhalten abstrahiert. Das hat zum Wertesystem von CCD geführt. Aus unserer Sicht ist das essenziell, um nicht mittelfristig mit einer Liste von Prinzipien und Praktiken stecken zu bleiben, sondern fähig zur Selbstkritik zu sein. Jedes Regelsystem birgt ja die Gefahr, zum Dogma zu werden. Durch das Wertesystem glauben wir, dem gegenzusteuern.

Martins Buch allein tendiert zum Dogma – es ist das Wort eines Gurus der Branche. Die CCD-Initiative hat den Anspruch, dem

entgegenzuwirken. Durch unser öffentliches Wiki, in dem jeder mitwirken kann, und Diskussionsforen entmystifizieren wir Clean Code und machen die Weiterentwicklung transparent. Wie das Wertesystem halten wir das für einen Aspekt professionellen Vorgehens.

**Ihr sprecht von einer Hierarchie von Kundenbedürfnissen und erhebt die Befriedigung dieser Qualitätsbedürfnisse zur Basis eines Wertesystems für die Softwareentwicklung. Provozierende Frage: Was ist aus eurer Sicht wichtiger: Das Commitment der Beteiligten zu diesen Werten oder deren praktische Umsetzung?** Kein Werkzeug, keine Methode und kein Konzept hat eine Berechtigung in professioneller Arbeit, wenn es nicht dazu beiträgt, Kundenbedürfnisse zu befriedigen. Und etwas, das mehr dazu beiträgt, ist besser als etwas anderes. Deshalb ist es erstens wichtig, überhaupt einmal festzustellen, was Kundenbedürfnisse jenseits der platten Funktionalität sind. Und zweitens ist bei jedem Hilfsangebot zu prüfen, welches Bedürfnis es inwiefern bedient.

Die CCD-Initiative hat sich bemüht, das für einen Ausschnitt an Prinzipien und Praktiken zu tun. Wir haben nicht einfach die frohe Botschaft des „Single Responsibility Principle“ (SRP) oder von „Test Driven Design“ (TDD) wiederkauen wollen. Stattdessen haben wir die Werte bzw. die dahinter stehenden Bedürfnisse an beide angelegt. Dabei hat sich SRP als wertvoll herausgestellt und TDD haben wir für zu leicht befunden. Über das Ergebnis können wir diskutieren – der Prozess, d. h. die Bewertung in Bezug auf Kundenbedürfnisse, ist für uns unverbrüchlich.

Wir glauben also nicht an schöne Worte, sondern an die Manifestation von Nutzen. Ultimativ bedeutet das sogar: Wenn ein Team demonstrieren kann, dass seine

Software alle Kundenbedürfnisse heute und in Zukunft erfüllt bzw. leicht erfüllen kann, dann ist es egal, wie es zu dem Ergebnis kommt. Falls dahinter allerdings keine CCD-Bausteine stecken, würden wir gern lernen, wie das Team so etwas schafft.

So ist es auch mit dem Commitment zu den CCD-Werten. Schön, wenn sich ein Entwickler oder eine ganze Firma dazu bekennen. Falls sich das jedoch nicht in der Entwicklungspraxis niederschlägt, nützt es nichts. Nur der Effekt zählt. Erst wenn durch eine Veränderung der Praxis Bedürfnisse des Kunden besser befriedigt werden, ist etwas erreicht. Und das ist nicht so leicht, wie es aussieht. Ein Armband ist schnell übergestreift. Die Anzeige, die einen CCD-Entwickler sucht, ist schnell geschrieben. Und dann? CCD steht für eine Haltung, für eine Kultur, eine Denkweise. Wer sich CCD „ins Haus holt“, der muss verstehen, dass er damit womöglich einen Kulturkampf auslöst. Wenn er es nicht versteht, weil er CCD nur als Strategie ansieht, ohne seine Kultur zu reflektieren, dann wird sich Peter Druckers Ausspruch bewahrheiten: „Culture eats strategy for breakfast.“ In „Manifest Speak“ könnte ich sagen: Praxis vor Commitment. Das drückt Rainer Maria Rilke für mich sehr prägnant aus mit: „Du musst dein Ändern leben.“

Allerdings ist die Praxis allein auch nichts. Die Gefahr, Geschäftigkeit mit Nützlichkeit zu verwechseln, ist groß. Praxis muss mit Theorie und Reflexion ausbalanciert werden. Theorie liefert Ideen für noch bessere Praxis. Reflexion beobachtet, ob Praxis so nützlich wie geglaubt ist.

**Ihr betont die Nachhaltigkeit bei der Softwareentwicklung. Erleben wir gerade ein Revival des Substanziellen und Nachhaltigen im Strudel des schnell**

### Vergänglichen? Spielt die zu erwartende Lebensdauer gerade entstehender Software nicht eine Rolle bei der Beurteilung der Wichtigkeit der Werte?

Auf die Frage des Kunden, „Sollen wir auf X oder Y setzen?“, lautet die sprichwörtliche Antwort des Beraters, „Es kommt darauf an.“ Die mag grundsätzlich richtig sein – nur ist sie leider meist wenig hilfreich.

Natürlich spielt die Lebensdauer eine Rolle bei der Abwägung der CCD-Werte in Bezug auf eine konkrete Software. Natürlich müssen die Investitionen in die verschiedenen CCD-Werte von Projekt zu Projekt immer wieder neu austariert werden. Ein solches Austarieren hat jedoch eine Voraussetzung: Ein Team muss überhaupt fähig sein, sich unterschiedlich zu verhalten.

Es mag hart klingen, aber so zeigt sich uns die Projektrealität in vielen Teams: Es gibt keine Bandbreite an Verhaltensweisen. Es gibt kein Methodenspektrum, keine konzeptionelle Vielfalt, keine diversen praktischen Erfahrungen. Wie eingefahren Teams sind, wie eng ihr Horizont ist, weil sie sich unter Druck durch die verhärteten Strukturen des immer selben Softwaresystems hangeln, ist kaum zu überschätzen. Deshalb tritt die CCD-Initiative fest auf, ja sie riskiert sogar, als arrogant angesehen zu werden. Wir sagen nicht, „Es kommt darauf an“, sondern „Mach es genau so!“ Bemühe dich jeden Tag um jeden CCD-Baustein. Fang nicht mit Abwägen an, sondern mit Bemühen und Konzentration. Und wenn du genügend Praxis und Einsicht gewonnen hast, wenn du eine hohe professionelle Fitness erreicht hast, dann kannst du dir Abwägen erlauben.

Entwickler, die nicht fähig sind, verlässlich Software mit hoher Qualität aus dem Stand in guter Geschwindigkeit auf der grünen Wiese zu produzieren, brauchen in ihrer täglichen Arbeit eine klare Rahmenvorgabe. Das enthebt sie vieler Entscheidungen, weil immer schon so viel vorgegeben ist. Einerseits mag das suggerieren, man könne deshalb produktiv sein. Andererseits jedoch führt das eben zur Atrophie der geistigen Entwicklermuskeln. Entwickler verlieren die Freiheit und Kraft, dieses System zu verlassen. Damit verlieren sie aber auch die Fähigkeit zur Innovation. Sie geben sich zufrieden mit dem Bewahren. Größere Veränderungen werden dann sehr, sehr schwer und schmerzhaft. Was wäre aber nötiger bei den heutigen Märkten, als innovativ und flexibel zu sein?

Also: Alles spielt bei Entwurf und Implementierung von Software eine Rolle. Um darauf aber wirklich angemessen reagieren zu können, braucht es Fitness. Die Bausteine der CCD-Initiative sind in diesem Sinne Fitnessgeräte. Wer daran seinen täglichen Workout macht, versetzt sich über die Zeit in die Lage, Abwägungen verantwortungsvoll vornehmen zu können.

### Einer der aktuellen Hypes unserer Szene ist Agilität. Vor allem die Beschreibung der Kundenbedürfnisse von CCD erinnern stark an die agilen Prinzipien. Ist CCD ein Teil der derzeitigen agilen Revolution in der Softwareentwicklung? Oder vielleicht umgekehrt? Oder sind die Ähnlichkeiten rein zufällig?

Natürlich gibt es Berührungspunkte zwischen Agilität und CCD. Der Agilität geht es um Kundennutzen, CCD auch. Insofern kann man CCD grob als Teil der agilen Revolution ansehen. Doch ich möchte genauer hinschauen – durchaus im Sinne des SRP. Denn dann tut sich eine Lücke in der Agilität auf. Die schweigt nämlich, was die codebezogene Praxis angeht. Scrum und Kanban als diesseits des Atlantiks prominenteste Vertreter agiler Vorgehensmodelle geben keinen Hinweis darauf, wie ein Team von einer ausgewählten User-Story zu lauffähigem Code kommt. Sie machen auch keine Aussage darüber, wie dieser Code aussehen sollte.

Damit will ich diese Vorgehensmodelle nicht kritisieren. Die Lücke ist kein Bug, sondern ein Feature. Vorgehensmodelle sind eben Vorgehensmodelle. Wir sollten deshalb nicht mehr von ihnen verlangen, als sie leisten können. Und wir sollten deshalb nicht glauben, mit ihnen sei alles gesagt. Erkennen wir an, dass sie sich im Sinne des SRP konzentrieren.

Scrum und Kanban wollen keine Aussage über Codierungspraxis und Codestrukturen machen. Auch dem Agilen Manifest ist das kein Anliegen. Doch irgendwer muss etwas dazu sagen, denn das Codieren erledigt sich nicht von allein und nicht jeder Code hat dieselbe Qualität in Bezug auf die Bedürfnisse des Kunden. Also versucht die CCD-Initiative, die Lücke zu schließen. Sie formuliert einen Anspruch daran, was in einem Sprint in Bezug auf Code mindestens passieren sollte.

Ob damit schon genug gesagt ist für das Codieren in Agilität, lasse ich hier mal dahingestellt. Ich glaube, dem Software-

Engineering steht da noch ein Aufwachen bevor. Denn wenn wir Agilität wirklich ernst nehmen, d. h. das kleinschrittige, nutzenorientierte Entwickeln, dann spiegelt sich das noch nicht in der Codierungsmethode wider, glaube ich. Code kann man noch zu wenig ansehen, ob er agil entwickelt wurde. Dazu wird die CCD-Initiative noch mehr Beitrag liefern müssen.

### Ihr habt schon viel erreicht. Welche Ziele habt ihr für die nächsten Jahre?

Konkrete Ziele gibt es nicht. Dennoch wollen wir nicht stehenbleiben. Stefan Lieser und mir schwebt Bewegung in drei Richtungen vor:

Wir glauben, Entwickler brauchen noch bessere Anleitung, um CCD-Bausteine in der Praxis einzusetzen. Wir basteln deshalb an einem weiteren didaktisch-methodischen Ansatz, der sich ums Üben dreht. Wir wollen es dem Einzelnen, aber auch einer Community leichter machen, systematisch Fortschritte zu erzielen.

Dann erscheint uns die aktuelle Bausteinliste zu lang und zu unpräzise. Wir spielen daher immer wieder mit dem Gedanken, ein Destillat herzustellen. Manche Prinzipien sind schlicht grundlegender oder gar universeller als andere. Manche Praktiken sind unmittelbar nützlicher als andere. Wenn wir CCD besser fokussieren, dann kann die Akzeptanz nur steigen.

Und schließlich ist die agile Haltung in CCD noch zu wenig vertreten. Wir fühlen uns aufgefordert, in dieser Hinsicht gerade bei den Bausteinen zum Wert Evolvierbarkeit nachzubessern. Da sind mehr Praktiken gefragt. Denn Praktiken oder Codierregeln nehmen die Entwickler mehr an die Hand. Unter Projektdruck über ein Prinzip nachzudenken, fällt schwer. Darüber kann man ja womöglich lange diskutieren. Aber nach einer Praktik handeln, das ist einfacher.

Also: Die CCD-Initiative ist lebendig. Die Community wächst beständig. Und die Ideen für eine Weiterentwicklung gehen uns nicht aus.

**Dann wünsche ich euch weiterhin viel Erfolg. Vielen Dank für das Gespräch.**

*Das Interview führte Thorsten Janning, Chefredakteur von OBJEKTSpektrum.*