

Der dornige Weg zum agilen Team – Teil 4: Selbstorganisation

Chaos oder Wunderwaffe?

■ VON JIRI LUNDAK

In einem agilen Team wird normalerweise Selbstorganisation groß geschrieben [1]. Dem Team wird die Autorität eingeräumt, selbst bestimmen zu können, was es tut und was es lässt. Niemand ist da, der ihm Vorschriften macht, wie es vorzugehen hat. Niemand ist da, der für das Team die Verantwortung übernehmen kann. Viele Manager, die Softwareentwicklung im traditionellen Stil gewohnt sind, betrachten dieses Vorgehen als Freibrief für Chaos und Unordnung. Das Team und das Projekt scheinen nicht mehr kontrollierbar. Das Bedürfnis des Managements nach Sicherheit, Kontrolle und Überwachung ist in Frage gestellt.

Es war einmal ein so genanntes agiles Team. Das hatte den Auftrag, ein neues Produkt zu entwickeln und das so schnell wie möglich. Der Geschäftsführer hatte davon gehört, dass agile Teams bis zu einem Mehrfachen schneller Software entwickeln können und dies auch noch exakt nach den Wünschen des Kunden. Deshalb ließ er seinen Entwicklungsleiter dieses Team zusammenstellen. Die Leute waren gute Entwickler, sie hatten bereits

in anderen Projekten zusammengearbeitet. Allerdings sind sie dabei noch nie iterativ inkrementell vorgegangen.

Eine Woche nach dem Projektstart kam der CEO beim Teamleiter, der die Rolle des ScrumMasters innehatte, vorbei, und fragte, wie es den so gehe. „Gut“, war die Antwort des Teamleiters, „ich habe die Planung für die erste Monatsiteration gemacht und jedem der Entwickler einige der Tasks nach seiner Kapazität und Fähigkeit zugewiesen. So wie ich den Berichten der Entwickler an den Standups entnehme, werden schon die ersten Features implementiert.“

Es kam der Tag, an dem das Ergebnis der Arbeiten dem Kunden gezeigt werden sollte. Der Kunde war verärgert. Die Qualität entsprach so gar nicht dem, was er sich vorgestellt hatte. Der Geschäftsführer war perplex. Was war passiert? Hat hier der Teamleiter versagt? Oder das Team? Wo ist die Selbstorganisation geblieben? Hat sie im Chaos geendet?

Bevor ich die Geschichte zu Ende erzähle, wollen wir zuerst ein wenig theoretischen Grund zum Thema Selbstorganisation legen.

Von Maschinen ...

Die Zeit, als eine Organisation als Maschine [2] verstanden wurde, die man in klar definierte Einzelteile zerlegen und

genau programmieren kann, liegt bereits eine gewisse Zeit zurück. Stattdessen hat man erkannt, dass eine Organisation (so mit auch ein Team) eher einem lebenden, sich kontinuierlich verändernden Organismus ähnelt.

Damit ein Team jedoch nicht einfach nur aus einem Haufen Individualisten besteht, die mehr zufällig als gezielt interagieren, braucht es mehr, als nur einige Entwickler einem Projekt zuzuteilen. Was braucht man genau, damit der Prozess in Gang kommt, der das Team eine Einheit werden lässt? Was kann man tun, um das Team dahingehend zu beeinflussen, dass es agile Verhaltensweisen annimmt, ohne mit der Peitsche zu drohen, womit auf alle Fälle das Gegenteil erreicht wird? Dazu müssen wir zuerst kurz die Bedingungen betrachten, die gegeben sein müssen, damit es überhaupt zur Selbstorganisation kommt. In Abbildung 1 sind diese kurz schematisch zusammengestellt, gestützt auf [3]. Drei Faktoren müssen wirken:

- **Der Container:** Er legt die Grenzen des selbst organisierenden Systems fest. In einem Team wird dadurch das „Wir“ bestimmt. Wer sind wir, und wer sind die anderen. Diese Grenzen können physischer Natur (z.B. geographische Nähe), organisatorischer Natur (z.B. eine Abteilung) oder auch konzeptueller

Der dornige Weg zum agilen Team – die Serie

- Warum agile Teams so schwer zu schaffen und am Leben zu erhalten sind
- Was ist kein agiles Team?
- Warum müssen Builds immer brechen?
- **Selbstorganisation – Chaos oder Wunderwaffe?**
- Was soll heißen: „Ich bin fertig“?
- Metriken: Ballast oder Notwendigkeit?
- Abnahmen: Verhinderer des Projekterfolges?
- Brauchen agile Teams agile Kunden?
- Gedeihen agile Teams nur in agilen Unternehmen?
- Agiles Vorgehen als rotes Tuch
- Nur die Besten sind für das agile Team gut genug
- Cross-funktionale Teams – eine Illusion?
- Wo bleibt der Manager im agilen Team?
- Agile Teams skalieren nicht!
- Wie erhalte ich ein agiles Team am Leben?

Natur (z.B. Identität oder Zweck der Gruppe) sein. Solche Grenzen sind in der Regel durchlässig und erlauben dadurch eine gewisse Interaktion mit der Welt außerhalb des Teams.

- **Bedeutende Unterschiede:** Diese beeinflussen primär die Verhaltensmuster, die während der Selbstorganisation entstehen. Unterschiede zwischen einzelnen Teilnehmern (Agenten) im System (dargestellt durch die verschiedenen Formen und Farben der Team-Mitglieder) werden von anderen Teilnehmern reflektiert oder verstärkt. Beispiele für signifikante Bereiche, in denen Unterschiede bestehen können, sind: Einfluss, Erfahrung, Qualität, Geschlecht, Rasse, Ausbildung usw. Je größer die Unterschiede, desto dynamischer (aber eventuell auch chaotischer) das System und desto mehr Veränderung ist möglich.
- **Transformierender Austausch:** Er bildet die Verbindungen (schwarze Pfeile) zwischen den einzelnen Team-Mitgliedern. Informationen, finanzielle Mittel und andere Ressourcen sind das Medium für diesen transformierenden Austausch. Während die Ressource von einem Teilnehmer zum anderen fließt, werden die beiden jeweils auf irgendeine Art verändert (transformiert). Diese Muster der individuellen Veränderungen machen schließlich die Anpassungsfähigkeit des ganzen Systems (sprich Teams) aus. Ein gemeinsames Verständnis dieser Muster (eine Art mentales Modell bzw. kulturelle Identität) kann sich herausbilden. Transportmittel für solchen Austausch bilden z.B. Gespräche, E-Mails, finanzielle Transaktionen usw.

... und Menschen

Wie kann man auf so ein dynamisches, komplexes System Einfluss nehmen? Von außen sind nur die drei erwähnten Bedingungen beeinflussbar. Den Rest muss das Team von innen heraus tun.

So ist es gut, einen überschaubaren Container festzulegen. Ein kleines – jedoch nicht zu kleines – Team ist ebenso wichtig, wie klar definierte, einfache Regeln, die im Team festgelegt werden müssen, z.B. bezüglich der Iterationslänge, der Synchronisierung mit der Außenwelt (mit Kunden, mit anderen Teams und mit

der Firma) sowie der Definition, wann ein Stück Applikationsfunktionalität als wirklich „fertig“ gilt (siehe auch den nächsten Artikel in dieser Reihe). Dazu gehört auch, eine Atmosphäre des Vertrauens zu schaffen, in der es problemlos möglich ist, von einer althergebrachten Meinung abzuweichen, wenn dies dem Fortschritt des Projektes zuträglich ist.

Zudem kann man die Zusammensetzung des Teams beeinflussen, indem man ein möglichst breites Spektrum an Personen einbezieht, mit möglichst vielfältiger Expertise und breitem Erfahrungshorizont. Eine gute Mischung von verantwortlich handelnden und innovativen Team-Mitgliedern, die bereit sind, stets Neues zu lernen und ihr Wissen miteinander zu teilen, wird dazu beitragen, altbewährte Vorgehensweisen und neue – vielleicht in der Organisation revolutionäre Ansätze – zu mischen.

Um den Gedankenaustausch zu fördern, ist es unabdingbar, Barrieren niederzureißen! Dies gilt sowohl für die Kommunikation innerhalb des Teams als auch mit der Außenwelt. Je breitbandiger die Kommunikation und je mehr Arbeiten die Teammitglieder miteinander erledigen, desto größer die Chance, dass das Team sich erfolgreich autonom selbst organisieren wird.

Einige Testfragen können helfen festzustellen, ob ein Team wirklich agil selbst organisierend arbeitet.

- Weiß jeder Entwickler wenigstens im Groben über alle vom Team programmierten Teile Bescheid oder gibt es gewisse private „Gärtchen“?
- Was passiert, wenn der ScrumMaster oder der Teamleiter für ein paar Wochen – wegen Urlaub oder Krankheit – ausfällt? Ist das Team dann hilflos und weiß nicht, wie es weiterfahren soll?
- Wenn Not am Mann beim Testen ist, bestimmt das Team selbstverständlich jemanden von den Entwicklern, damit er beim Testen hilft, oder wartet es auf einen Management-Entscheid?

... und das Ende der Geschichte

Wie ging unser Geschichte aus? Nachdem der Teamleiter in sich gegangen war und einen befreundeten, erfahrenen Scrum-

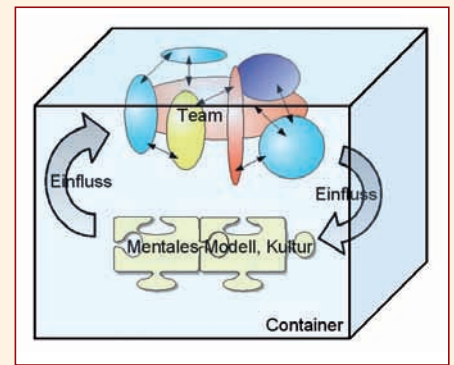


Abb. 1: Dynamik der Selbstorganisation

Master zu Rate gezogen hatte, entschloss er sich, viel weniger allein zu tun [4], wie z.B. das Planen und die Task-Zuteilung. Stattdessen plante das Team gemeinsam die nächsten Iterationen. Es legte Maßnahmen fest, um die Qualität der gelieferten Inkremente zu steigern. So wurde im Team bestimmt, welche Feature wann implementiert wurden, und man einigte sich auf konsequente Paar-Programmierung, um auch die Qualität der weniger erfahrenen Entwickler durch kontinuierliche Code Reviews zu steigern. Das Team übernahm zunehmend als Ganzes die Verantwortung für das gelieferte Produkt und bekam schließlich in der vierten Iteration die Qualität in den Griff. Sollten nicht alle Geschichten mit einem Happy End ausklingen?



Jiri Lundak ist mit mehr als 20 Jahren Erfahrung ein Urgestein der Softwareentwicklung. Er arbeitet gegenwärtig als Entwicklungsleiter für Löwenfels Partner in der Schweiz. Er entwickelt als Architekt immer noch aktiv Software, hat aber erkannt, dass die meisten Probleme in Entwicklungsprojekten nicht technischer, sondern vielmehr menschlich-sozialer Natur sind. Als praktizierender Certified ScrumMaster lebt er die Rolle eines Vorkämpfers für agiles Verhalten innerhalb der Firmenorganisation. Kontakt: jiri.lundak@loewenfels.ch.

■ Links & Literatur

- [1] Ken Schwaber: Agile Processes and Self Organisation: controlchaos.com/download/SelfOrganization-n.pdf
- [2] Gareth Morgan: Images of Organization, Sage Publications, 1996
- [3] Edwin E. Olson, Glenda H. Eoyang: Facilitating Organization Change: Lessons from Complexity Science (S. 11–15), Jossey-Bass/Pfeiffer, 2001
- [4] Peter Block: Stewardship: Choosing Service Over Self-Interest, Berrett-Koehler Publishers, 1996