

„ÄNDERUNGEN UNVERMEIDBAR“



Quelle: Nikolai Sorokin - Fotolia.com

Menschen sind unterschiedlich. Statt von allen das Gleiche zu erwarten, kann man auch von jedem das Beste erwarten. // People are different. Instead of expecting the same from everybody, one can now expect the best of everybody.

Autor // Author: Arnulf Christl, Metaspatial

Der Ursprung von „Agilem Projektmanagement“ geht auf das Ende der 1990er Jahre zurück und kommt jetzt zunehmend auch außerhalb der klassischen IT-Welt an. Agiles Projektmanagement (kurz: Agile) basiert auf der Annahme, dass Änderungen im Plan unvermeidbar sind. Statt zu versuchen, alles Unvorhergesehene voranzuplanen – was nachgewiesenermaßen unmöglich ist – wird der Plan flexibler gestaltet. Insbesondere bei starren Strukturen und Hierarchien braucht es eine Weile, um solche Methoden durchzusetzen.

// The origin of ‚agile project management‘ can be traced back to the end of the 1990s and can now also be found outside of the traditional IT world. Agile project management (or agile for short) is based on the assumption that changes in a plan are unavoidable. Instead of attempting to plan for anything unforeseeable in advance – which is by definition impossible – the plan is designed to be flexible. In rigid structures and hierarchies, in particular, time is needed until such methods are established.

// ,CHANGES ARE UNAVOIDABLE'

Es ist die Erkenntnis gereift, dass Menschen unterschiedlich sind. Statt von allen das Gleiche zu erwarten, kann man von jedem das Beste erwarten. Dieser Artikel erläutert die Hintergründe von Agile und stellt „Scrum“ vor, eine der bekannteren agilen Methoden. Der Artikel erhebt keinen Anspruch auf Vollständigkeit, sondern soll neugierig machen, wie man auf eine etwas modernere und andere Art und Weise Projekte steuert.

Scrum ist nur eine von vielen agilen Managementmethoden. Es ist besonders gut für die Entwicklung und Einführung von Softwareprojekten geeignet, kann aber auch auf ganze Organisationsstrukturen angewendet werden. Das Ordnance Survey in Großbritannien hat Scrum im gesamten Unternehmen eingeführt und damit eine erhebliche Steigerung der Produktivität bei gleichzeitig sinkenden Kosten erzielt. Mindestens genauso wichtig ist aber, dass sich bei der Belegschaft auch mehr Zufriedenheit eingestellt hat und es weniger Frust gibt. Denn erst das ist ein Nachweis für wirklich nachhaltiges Management.

URSPRÜNGE

Klassisches Projektmanagement basiert auf der sogenannten Wasserfall-Methode. Dabei wird vor dem Projektstart ein detaillierter Plan erstellt, der dann im Projekt abgearbeitet wird. Änderungen im Plan sind nicht vorgesehen und werden als Gefährdung eingestuft, wenn sie dennoch eintreten.

Vor allem in der IT ist dieser Ansatz zum Scheitern verurteilt, da kurzfristige Änderungen an der Tagesordnung sind. Jeder IT-Projektleiter und viele Kunden haben das selbst erlebt.

Eine Antwort aus der Software-Entwicklung war 1995 „Extreme Programming“ (XP). Dieser Ansatz aus der IT ist zwar nicht direkt auf einen Brückenbau oder die Umstrukturierung eines Konzerns übertragbar, hat aber wertvolle Impulse für das Projektmanagement geliefert: Regelmäßige kurze Iterationen ermöglichen frühzeitige Änderungen des Plans und führen effektiver zum Ziel.

DAS „AGILE MANIFESTO“

Etwas zur gleichen Zeit wie Extreme Programming entstand das „Agile Manifesto“, eine Reaktion auf überstrukturierte Projektkorsetten, die jegliche Kreativität und Flexibilität (und oft die Menschen) im Keim ersticken. Die Kernaussage des „Agile Manifesto“ stellt deshalb fest:

- ▷ Menschen und deren Interaktion sind wichtiger als Prozesse und Werkzeuge
- ▷ Funktionierende Software ist wichtiger als ausgeklügelte Dokumentationen
- ▷ Zusammenarbeit mit dem Kunden ist wichtiger als Vertragsverhandlungen
- ▷ Auf Veränderungen reagieren ist wichtiger als den Plan durchzusetzen

Diese Grundlagen fördern eine auf den Menschen orientierte Herangehensweise. Das Erstaunliche dabei ist, dass die Einhaltung dieser Grundlagen zu mehr Produktivität bei

What's more, the realisation that people are different has matured. Instead of expecting the same from everybody, one can now expect the best from everybody. This article explains agile's background and introduces „Scrum“, one of the better known agile methods. The article does not claim to cover all aspects, but was written with the aim of awakening interest in how to control projects in a more modern way.

Scrum is only one of many agile management methods. It is especially well suited to the development and introduction of software projects, but can also be applied to entire organisation structures. The United Kingdom's Ordnance Survey has adopted Scrum across the entire organisation and achieved a substantial increase in productivity with a simultaneous lowering of costs. However, the fact that employee satisfaction is higher and frustrations lower is just as important, because this is the true indicator for real sustainable management.

ORIGINS

Traditional project management is based on what is known as the waterfall method. A detailed plan is compiled prior to the project start and is then worked through during the project. Changes in the plan are not envisaged and are classified as a hazard if they do occur.

In IT, in particular, this approach is destined for failure, because changes at short notice are the order of the day. Every IT project manager and numerous customers have experienced this in person.

In 1995, one reply from software development was „extreme programming“ (XP). This approach by IT cannot be directly applied to bridge engineering or the restructuring of a concern, but has provided valuable impulses for project management: regular, short iterations facilitate prompt changes to plans and lead more effectively to the objective.

THE ,AGILE MANIFESTO'

At around the same time as extreme programming, the ,Agile Manifesto' appeared, a reaction to overstructured project corsets, which nipped any kind of creativity and flexibility (and often people) in the bud. The core statements of the ,Agile Manifesto' are therefore:

- ▷ people and their interactions are more important than processes and tools;
- ▷ functioning software is more important than sophisticated documentation;
- ▷ cooperation with the customer is more important than contract negotiations;
- ▷ reacting to change is more important than sticking to a plan.

These fundamental principles demand people-oriented processes. The amazing thing is that adhering to these principles leads to greater productivity while simultaneously increasing employee satisfaction. The method is not new and is already well proven.

To better understand the background to the agile mo-

gleichzeitig größerer Zufriedenheit der Mitarbeiter führt. Die Methode ist nicht neu, der Nachweis bereits erbracht.

Um die Hintergründe der agilen Bewegung besser zu verstehen, hilft es, zunächst noch einmal verschiedene Persönlichkeitstypen zu betrachten.

EXKURS: PERSÖNLICHKEITSTYPEN

Eine der bekannteren Methoden, der Myers-Briggs-Typindikator, basiert auf der Psychologie von C. G. Jung. Bei Bewerbungen im höheren Management wird inzwischen davon ausgegangen, dass man eine gute Vorstellung davon hat, welchen Persönlichkeitstyp man selbst verkörpert.

Die Klassifizierung nimmt an, dass es grundsätzlich möglich ist, Persönlichkeiten durch vier Merkmale zu beschreiben. Das erste Merkmale ist Introversion oder Extroversion. Natürlich sind nur die wenigsten Menschen 100 Prozent introvertiert oder extrovertiert. Aber jeder sollte von sich selbst wissen, welcher Wesenszug stärker ausgeprägt ist. Das eine ist nicht besser oder schlechter als das andere, es ist nicht als Wertung zu sehen, sondern als Erkenntnis.

Das zweite Merkmal unterscheidet zwischen Intuition und „Sensorik“. „Sensoriker“ entscheiden aufgrund nachvollziehbarer Fakten. Intuitive Menschen entscheiden dagegen „aus dem Bauch heraus“. Selbst wenn solch eine „Eingebung“ zu einer richtigen Entscheidung geführt hat, ist dies für „Sensoriker“ aber oft nicht verständlich. Wichtig ist auch hier, dass die Art der Entscheidung nicht zwingend zum richtigen Ergebnis führt.

Als drittes Merkmal wird zwischen analytischem Denken oder Einfühlung unterschieden. Auch diese zwei Eigenschaften schließen sich zwar nicht vollständig aus, eine Tendenz zum einen oder anderen hat jedoch erheblichen Einfluss darauf, wie ein Mensch Entscheidungen trifft. Eine analytisch „richtige“ Entscheidung kann ganz anders sein als eine einfühlsame, die auf den Menschen eingeht. Für Ingenieure sind eher analytische Fähigkeiten wichtig, während für Sozialarbeiter auch einfühlsame Fähigkeiten wichtig sind. Eine Brücke ist nicht einfühlsam, Menschen tun dagegen manchmal Dinge, die nicht durch analytische Logik nachvollziehbar sind.

vement it helps to study the various personality types more closely.

EXCURSION: PERSONALITY TYPES

One of the better known methods, the Myers-Briggs type indicator, is based on the psychology teachings of C. G. Jung. When applying for positions in higher management, it is now assumed that you have a good idea of the personality types you yourself personify.

The classification assumes that it is fundamentally possible to describe personalities using four characteristics. The first characteristic is introversion or extroversion. Of course, only very few people are truly 100 percent introvert or extrovert. But everybody should know which of their personality traits is dominant. Because one is neither better nor worse than the other, it should not be regarded as a judgement, but as an insight.

The second characteristic differentiates between intuition and sensing. Sensing people decide based on verifiable facts. Intuitive people, on the other hand, decide based on ‚gut feeling‘. Even if such ‚inspiration‘ leads to a correct decision, it often cannot be understood by sensing people. What is also important here is that the type of decision does not necessarily lead to the correct result.

The third characteristic differentiates between analytical thinking and feeling. However, these two properties are not entirely mutually exclusive either, but a tendency in one or the other direction has a considerable influence on how a person makes decisions. An analytically ‚correct‘ decision can be completely different to a sensitive one that responds to people. Analytical capabilities are more important for engineers, while a capacity for feeling is more important for social workers. A bridge has no feelings; people, on the other hand, occasionally do things that cannot be explained by analytical logic.

The fourth characteristic describes planning skills versus flexibility, i.e. whether decisions are put off as long as possible or are preferentially made as earlier as possible. Both options have their advantages and disadvantages and are mutually ex-



Gunter Dueck unterscheidet richtige, natürliche und wahre Menschen. Keiner ist schlechter oder besser, aber alle sind anders. // Gunter Dueck differentiates between correct, natural and true people. No one is better or worse, but all are different.

Quelle: Vladimir Ovchinnikov - Fotolia.com

Das vierte Merkmal beschreibt Planungsfertigkeit versus Flexibilität, also ob man Entscheidungen möglichst lange hinauszögert oder am liebsten so früh wie möglich trifft. Beide Varianten haben Vor- und Nachteile und schließen sich gegenseitig aus. Wer Entscheidungen früh trifft, kann besser langfristig planen. Wer Entscheidungen bis zum letzten Augenblick hinauszögert, kann dagegen flexibel reagieren. Sie können sich vorstellen, dass es dann schwierig wird, wenn ein Frühplaner (nach Myers-Briggs ein „Judge“ oder „Richter“) und ein „Abwarter“ („Perceiver“) etwa gemeinsam eine Reise planen. Wenn sie um die jeweiligen Persönlichkeitstypen des anderen nicht wissen, dann kommt es mit hoher Wahrscheinlichkeit zu Konflikten. Wenn sie aber die Fähigkeiten des anderen richtig einschätzen können, dann kann der „Judge“ zum Beispiel früh planen, während der „Abwarter“ vor Ort die „Führung“ übernimmt und flexibel, kurzfristige „ungeplante“ Entscheidungen treffen kann.

Dies nur in aller Kürze, das Thema kann man noch beliebig vertiefen und der Autor kann nur empfehlen, sich eingehend mit dem Thema auseinanderzusetzen, weil es die Zusammenarbeit mit anderen in einem Team unwahrscheinlich verbessert.

RICHTIGE, NATÜRLICHE UND WAHRE MENSCHEN

Eine etwas andere Nomenklatur wurde in drei Büchern und vielen Vorträgen von Gunter Dueck behandelt. Er unterscheidet zwischen „richtigen“, „natürlichen“ und „wahren“ Menschen. Auch hier gilt, dass die Richtigen nicht besser sind als die Natürlichen oder Wahren, sondern eben anders. Sie haben unterschiedliche Fähigkeiten: Richtige Menschen halten sich an Regeln, natürliche Menschen folgen mit hoher Energie ihren Impulsen, während wahre Menschen ständig auf der Suche nach dem tieferen Sinn sind.

In der Automobilproduktion beispielsweise braucht man jede dieser Eigenschaften. Die wahren Menschen haben Ideen und können diese visionär beschreiben. Viele dieser Ideen sind gar nicht umsetzbar, deshalb sorgen die richtigen Menschen dafür, dass die realisierbaren Ideen umgesetzt werden. Die natürlichen Menschen setzen die Idee dann in Prototypen um. Aber ein natürlicher Mensch wird nie einen optimierten Organisationsprozess entwickeln, der sich nach einem Plan richtet. Denn dafür sind wieder die richtigen Menschen da. Ganz wichtig ist, dass ein natürlicher Mensch nicht in die Rolle eines richtigen Menschen gepresst werden sollte und umgekehrt gilt das für richtige und wahre Menschen genauso. Denn dabei kommen unweigerlich „falsche“ Menschen raus, also solche, die eigentlich hätten gute Prototypen bauen können, aber in einer optimierten Prozesskette innerlich verhungern. Umgekehrt wird man richtige Menschen nicht einer vollkommen unplanbaren Situationen aussetzen, bei der sie kreativ und flexibel auf eine sich ständig verändernde Umgebung reagieren müssen.

SCRUM

In seinem sehr schönen Vortrag „Der Mensch in artgerechter Haltung“ hat Christoph Baudson beim Bonn Agile Meetup im Mai 2013 den Ursprung von Scrum zusammengefasst:

- ▷ XP, Software Craftsmanship → natürlich

clusive. People who make decisions early can plan better for the long-term. Those who delay decisions until the very last moment, in contrast, can react flexibly. You can imagine the difficulties if an early planner (a judge according to Myers-Briggs) and a ‚waiter‘ (‚perceiver‘) plan a journey together, for example. If one has no knowledge of the personality type of the other, the most probable result is conflict. But if each is well aware of the capabilities of the other, the ‚judge‘ can plan ahead, for example, while the ‚waiter‘ assumes local ‚leadership‘ and can make flexible, fast, ‚unplanned‘ decisions.

This serves merely as a very short introduction. The topic can be deepened at will and the author can only recommend a rigorous study of the subject, because it enormously improves cooperation with the other members of a team.

CORRECT, NATURAL AND TRUE PEOPLE

Three books and numerous presentations by Gunter Dueck deal with a somewhat different nomenclature. It differentiates between ‚correct‘, ‚natural‘ and ‚true‘ people. Here, too, what counts is that correct people are not better than natural or true people, just different. They have different capabilities: correct people stick to the rules, natural people follow their impulses with a great deal of energy, while true people search for a deeper meaning.

In automobile production, for example, all of these characteristics are required. True people have ideas and can describe them as visionaries. Many of these ideas are impossible to implement, which is why the correct people ensure that the realistic ideas are implemented. Natural people then implement the ideas in prototypes. But a natural person will never develop an optimised organisation process that complies with a plan, because this is where correct people come in again. It is very important that a natural person is not pressed into the role of a correct person and this applies vice versa for ‚correct‘ and ‚true‘ people. The unavoidable result of this would be ‚wrong‘ people, i.e. people who would have been able to build good prototypes, but who starve in an optimised process chain. On the other hand, one cannot subject ‚correct‘ people to a completely unplanable situations, where they would need to react creatively and flexibly to a constantly changing environment.

SCRUM

In a very good presentation, ‚Der Mensch in artgerechter Haltung‘ (People kept in humane conditions), Christoph Baudson summarised the origin of Scrum at the Agile Meetup in Bonn in May 2013:

- ▷ XP, software craftsmanship → natural;
- ▷ Scrum certified → correct;
- ▷ Agile Manifest → true.

The Agile Manifest (authored by true people) and extreme programming (developed by natural people) was optimised in Scrum by correct people. This method allows all people and personality types to be catered for and optimally controlled in a work process. The technique used to learn this method is relatively simple. Implementing it in an organisation is incomparably more difficult.

OVERVIEW

Scrum is an iteration of what are known as sprints. These

- ▷ Scrum-zertifiziert → richtig
- ▷ Agiles Manifest → wahr

Das Agile Manifest (das wahre Menschen verfasst haben) und Extreme Programming (von natürlichen Menschen entwickelt) haben richtige Menschen in „Scrum“ optimiert. Diese Methode ermöglicht es, allen Menschen und Persönlichkeits-typen gerecht zu werden und optimal in einem Arbeitsprozess zu steuern. Die Technik dieser Methode zu erlernen ist relativ einfach. Sie in einer Organisation umzusetzen ist dagegen ungleich schwerer.

ÜBERSICHT

Scrum ist eine Iteration von sogenannten Sprints. Das sind zwischen zwei und sechs Wochen lange, abgeschlossene Entwicklungsphasen. Diese werden so oft wiederholt, bis das Ziel erreicht ist (oder die Zeit oder das Budget zu Ende sind). Nach jedem Sprint wird überprüft, ob der Sprint das Projekt dem Ziel näher gebracht hat. Wenn nicht, können Gegenmaßnahmen ergriffen werden, wie das im dem klassischen Projektmanagement bekannt ist. Dann wird ein neuer Plan für den nächsten Sprint erstellt und wieder abgearbeitet. Dadurch reduziert sich das Risiko, in die falsche Richtung zu entwickeln auf maximal zwei bis sechs Wochen. Durch die Notwendigkeit, nach jedem Sprint ein abgeschlossenes Produkt abzuliefern, reduziert sich das Risiko, dass wichtige Anforderungen liegen bleiben. Die Teamleistung steigt nach einer Lernphase meist um ein Vielfaches an.

DAS STAND-UP-MEETING

Das zentrale Ereignis im Sprint ist das tägliche Stand-Up, eine auf 15 Minuten begrenzte Kurzbesprechung des Teams (bei dem alle stehen). In der Besprechung berichten alle im Team, was sie am Vortag gemacht haben, welche Probleme dabei aufgetreten sind, welche Aufgaben heute anstehen und ob sie dabei Probleme erwarten. Der Scrum-Master notiert alles und kümmert sich darum, dass alle Probleme gelöst werden, damit das Team wieder optimal arbeiten kann. Es ist dabei von ganz zentraler Bedeutung, dass das Team nicht dem Management oder einem Vorgesetzten berichtet, sondern den anderen Team-Mitgliedern. Jeder im Team ist für alle Arbeiten verantwortlich. Da alle Aufgaben in einem Projekt immer zusammenhängen, kann es auch nicht sein, dass eine Aufgabe an einer einzelnen Person „hängen“ bleibt. Diese starke Team-Orientierung führt im Idealfall dazu, dass das Team über sich selbst hinaus wächst und mehr wird als die Summe seiner Teile.

ROLLEN

Scrum legt großen Wert auf eine klare Trennung der Zuständigkeiten in einem Projekt. Ein Mensch kann in einem Projekt immer nur eine Rolle bekleiden. Konflikte können dann in einem geregelten Prozess mit klar definierten Zuständigkeiten geklärt werden. Scrum definiert drei Kernfunktionen: den Product-Owner, das Team und den Scrum-Master.

Der Product-Owner ist dafür zuständig, das Produkt zu beschreiben, Arbeitspakete oder „Tasks“ zu definieren und so zu priorisieren, dass das Ziel erreicht wird. Der Product-Owner kann die Prioritäten der Arbeitspakete zwischen den

are self-contained development phases between two and six weeks long. They are repeated until the objective has been achieved (or time or the budget run out). Checks are performed following every sprint to examine whether the sprint has moved the project closer to the target. If not, countermeasures can be initiated, similar to classical project management. A new plan is then compiled for the next sprint and worked



through once again. This reduces the risk of moving in the wrong direction to a maximum period of two to six weeks. The necessity of delivering a self-contained product after every sprint means that the risk of ignoring important demands is reduced. Following an initial learning phase, team performance is generally multiplied.

THE STAND-UP MEETING

In a sprint, the central event is the daily stand-up meeting, a short team meeting limited to 15 minutes (where everybody stands). In the meeting all team members report on the previous day's activities, the problems that occurred, today's tasks and any problems anticipated. The Scrum Master notes down everything and ensures that all problems are solved so that the team can work again to their optimum.

One of the central points here is that the team does not report to management or a superior, but to the other team members. Everybody in the team is responsible for all the work. Because all the tasks in a project are always related, it is not correct that a task gets ‚stuck‘ on a single person. Ideally, this strong team orientation means that the team rises above itself and becomes greater than the sum of its parts.

ROLES

Scrum greatly emphasises a clear delineation of responsibilities within a project. One person can only assume one role in a project. Conflicts can then be resolved in a controlled process with clearly defined responsibilities. Scrum defines three core functions: the product owner, the team and the Scrum Master.

The product owner is responsible for describing the product, defining work packages or tasks, and prioritising them such that the objective is achieved. The product owner can continuously adapt the priorities of the work packages to reality between the sprints. The product owner will never personally interfere in the production process, because this is the task of the team.

The team decides which work packages can be dealt with in a sprint and work through it autonomously, following the sequence of priorities. At the end of each sprint, the finished product is presented and handed over to the product owner in the review meeting. The quality of the team's assessment is then examined retrospectively, together with the problems that arose during implementation and the aspects that can be improved. In this way the team continuously learns to better assess its own capabilities. The product owner gets to know his team better and can adapt his plan following each sprint (the product backlog), present management a status report

Sprints kontinuierlich an die Realitäten anpassen. Der Product-Owner wird niemals selbst in den Herstellungsprozess eingreifen, weil das die Aufgabe des Teams ist.

Das Team entscheidet selbst, welche Arbeitspakete in einem Sprint erledigt werden können und bearbeitet diese dann selbstverantwortlich in der Reihenfolge der Priorität. Am Ende eines jeden Sprints wird dem Product-Owner im Review-Meeting das fertig Produkt vorgestellt und übergeben. Danach wird in der „Retrospektive“ überprüft, wie gut die Einschätzung des Teams war, welche Probleme bei der Umsetzung aufgetreten sind und was verbessert werden kann. Das Team lernt so kontinuierlich die eigenen Fähigkeiten besser einzuschätzen. Der Product-Owner lernt sein Team besser kennen und kann nach jedem Sprint seinen Plan (das Product-Backlog) anpassen, dem Management einen Statusbericht geben und gegebenenfalls zusätzliche Ressourcen anfordern, um das Ziel in der vorgegebenen Zeit zu erreichen.

Die dritte Kernrolle ist der „Scrum Master“. Er ist dafür zuständig, dass alle Scrum-Methoden eingehalten werden und das Team ungestört arbeiten kann. Auf diese Weise wird die Rolle des klassischen Projektmanagers, der oft zwischen Management und Team zerrieben wird, durch zwei Personen mit weitreichenden Kompetenzen und ein selbstverantwortliches Team ersetzt. Für viele Teams, den Product-Owner, ganz besonders aber auch für das Management, ist es oft überraschend, zu welchen Höchstleistungen ein Team fähig ist.

Es werden in Scrum weitere Rollen berücksichtigt, die nicht im eigentlichen Arbeitsablauf auftreten, aber wichtige Rahmenbedingungen bereitstellen. Hierzu zählt als eine zentrale Nebenrolle das Management. Es setzt die Ziele für den Product-Owner, stellt die Arbeitskräfte und das Budget bereit. Als Ergebnis bekommt das Management vom Product-Owner zur versprochenen Zeit das funktionierende fertige Produkt. Eine weitere wichtige Nebenrolle bekleiden die Stakeholder. Das können Kunden des Endprodukts sein, Auftraggeber und Zulieferer. Sie haben Einfluss auf Budget und Zeitrahmen, können und sollen aber explizit nicht in die Sprints eingreifen.

DAS PRODUCT-BACKLOG

Um die verschiedenen Prozesse steuern zu können, gibt es sogenannte Artefakte. Das sind Listen, die Arbeitspakete beschreiben und Protokolle der Besprechungen (Ereignisse). Das wichtigste Artefakt ist das Product-Backlog.

Der Product-Owner bekommt vom Management den Auftrag, ein Produkt zu erstellen. Dafür muss er einen Plan erstellen. Dieser Plan wird nun aber nicht komplett durchgeplant, da man aus der IT weiß, dass dies sowieso nicht klappt. Deshalb wird als übergreifender Plan lediglich das Ziel definiert. Daraufhin identifiziert der Product-Owner die Arbeiten, die zuerst erledigt werden müssen, um dem Ziel näher zu kommen und zerlegt sie in kleine „Produkte“, also Ergebnisse, die das Team in einem Sprint erledigen, und – ganz wichtig – auch abschließen kann. Diese Liste wird als „Product-Backlog“ bezeichnet. Das Product-Backlog sollte immer Arbeitspakete für einige Sprints im Voraus planen und ständig verfeinert werden.

SPRINT-BACKLOG

Das Sprint-Backlog wird vom Team selbst erstellt. Der Product-Owner wählt Arbeitspakete aus seinem Product-Backlog

and, if necessary, demand resources in order to achieve the objective in the specified time.

The third core role is that of the Scrum Master. He is responsible for ensuring that all Scrum methods are adhered to and that the team can work uninterrupted. In this way the role of the classical project manager, who is often ground down between management and team, is replaced by two people with extensive responsibilities and an autonomous team.

For many teams and the product owner, but especially for management, the extreme performance a team is capable of is often surprising.

Additional roles are available in Scrum that do not form part of the actual work process, but which provide an important framework. A central secondary role is given here by management. Management defines the objective for the product owner, and provides the workforce and the budget. The end result for management is the functioning, finished product, at the promised time from the product owner.

Another important secondary role is assumed by the stakeholders. These can be the end-product customers, clients and suppliers. They can influence the budget and the time frame, but explicitly cannot and must not interfere in the sprints.

THE PRODUCT BACKLOG

So-called artefacts are available to control the various processes. These are lists describing the work packages and meeting protocols (events). The most important artefact is the product backlog.

The product owner receives the order to create a product from management. He must then compile a plan. However, this plan is not completely fixed, because we know from IT that this won't work anyway. This is why the overriding plan only defines the objective. The product owner then identifies the tasks that require attention first in order to move closer to the objective and dissects them into small 'products', that is results which can be produced and - this is important - concluded by the team in a sprint. This list is referred to as the product backlog. The product backlog should always plan work packages for a number of sprints in advance and be continuously refined.

SPRINT BACKLOG

The sprint backlog is compiled by the team itself. The product owner selects work packages from the product backlog and allocates priorities. However, he cannot prescribe to the team how long a task should take. This is usual practice in traditional project work, but is totally illogical. Even those busy on a work package often do not know which uncertainties are waiting in any particular problem.

The team itself therefore assesses how long a task will take and notes this in the sprint backlog. As soon as the team hits an unforeseen problem, it informs the Scrum Master in the morning stand-up meeting, who then initiates measures to eradicate the problem. The product owner, who can escalate problems to management if necessary, is also informed, so that additional resources can be ordered or the scope of the project modified.

The sprint backlog serves targeted working. If a team has amassed too much work for a spring (and this is almost al-

Quelle: Nikolai Sorokin - Fotolia.com



„Slack time“, vom Management oft als nutzlose Zeit betrachtet, ist die Zeit, die das Team braucht, um sich zwischen zwei Sprints zu erholen. // ‚Slack time‘ often regarded by management as wasted time, is the time between two sprints that the team needs to breath.

aus und versieht sie mit Prioritäten. Er kann dem Team aber nicht vorschreiben, wie lange eine Arbeit zu dauern hat. Das ist zwar übliche Praxis in der herkömmlichen Projektarbeit, aber völlig unlogisch. Selbst die Bearbeiter eines Arbeitspakets wissen oft nicht, welche Unwägbarkeiten in einem speziellen Problem stecken.

Das Team schätzt deshalb zunächst selbst, wie lange eine Arbeit dauern wird und schreibt das in den „Sprint-Backlog“. Sobald das Team auf ein unvorhergesehenes Problem stößt, informiert es im morgendlichen Stand-up den Scrum-Master, der Maßnahmen ergreift, um das Problem zu beseitigen. Dabei wird auch der Product-Owner informiert, der Probleme gegebenenfalls bis zum Management eskalieren kann, um weitere Ressourcen anzufordern oder den Umfang des Projekts zu verändern. Das Sprint-Backlog dient der zielorientierten Arbeit. Wenn sich ein Team zu viel für einen Sprint vorgenommen hat (und das ist anfangs fast immer so), wird es zunächst frustriert sein. Nach und nach lernt das Team aber hinzu und kommt zu immer realistischeren Einschätzungen, was dem Projekt zugute kommt.

SLACK-TIME

Zwischen den Sprints gibt es sogenannte „Slack-Time“. Vom Management oft argwöhnisch als „nutzlose“ Zeit betrachtet, ist es die Zeit, die ein Team braucht, um zwischen zwei Sprints Luft zu holen. Hier werden neue Ideen entwickelt, die eigenen Fähigkeiten erweitert und neue Energie für den nächsten Sprint gesammelt. Im Grunde genommen ist diese Phase selbstverständlich, niemand kann ohne Unterbrechung sprinten. Dennoch wird gerade diese wichtige Phase oft aus Zeit- und Kostengründen wegoptimiert, was zur Folge hat, dass das Team ausbrennt. Das Ergebnis ist dann eine verspätete Lieferung, schlechtere Qualität und wesentlich höhere Kosten, bis ein neues Team aufgebaut werden konnte und das alte ersetzt. ◀

Weitere Informationen unter:

- Kent Beck: *Extreme Programming Explained. Embrace Change. 1st Edition, Addison Wesley, 2000*
- Stephen, Page, Myers, Brown, Watson, Sir Magee: *Fixing the flaws in government IT, Institute for Government, 2011*
- ☞ <http://agilemanifesto.org/>
- ☞ <http://www.myersbriggs.org/>
- ☞ <http://www.omnisophie.com/>
- ☞ <http://de.wikipedia.org/wiki/Scrum>

ways the case initially), it will be frustrated. But gradually the team learns and arrives with time at more realistic assessments, which benefits the project.

SLACK TIME

Between sprints there is what is known as slack time. Often suspiciously regarded by management as ‚wasted‘ time, it is the time between two sprints that a team needs to breath. Here, new ideas are developed, member’s capabilities expanded and energy replenished ready for the next sprint. In principle, this phase should be taken for granted, nobody can sprint constantly without a break. Yet it is just this important phase that is often optimised out for time and cost reasons, with the consequence that the team burns out. The result is then overdue delivery, poor quality and considerably higher costs, until a new team can be put together and the old one replaced. ◀

More information at:

- Kent Beck: *Extreme Programming Explained. Embrace Change. 1st Edition, Addison Wesley, 2000*
- Stephen, Page, Myers, Brown, Watson, Sir Magee: *Fixing the flaws in government IT, Institute for Government, 2011*
- ☞ <http://agilemanifesto.org/>
- ☞ <http://www.myersbriggs.org/>
- ☞ <http://www.omnisophie.com/>
- ☞ <http://de.wikipedia.org/wiki/Scrum>