



□ Dr. Ernst Sikora

(ernst.sikora@autosafety.de)

ist Teamleiter für die Entwicklung vorausschauender Sicherheitsfunktionen bei der Automotive Safety Technologies GmbH. Zur Gestaltung wirksamer und leistungsfähiger Prozesse in der Fahrzeug-Serienentwicklung greift er auf die gesamte Palette verfügbarer Methoden zurück: Systemanalyse ebenso wie Szenarien und Use Cases, textuelle Spezifikation genauso wie Anforderungsmodellierung, Traceability ebenso wie agiles Projektmanagement.

## objektspektrum themenspecial: requirements engineering

### „Du musst viel, viel genauer wünschen Papa“, sagte das Sams

Wer ist Ihr Lieblingsautor, wenn es um Requirements Engineering geht? Ist es vielleicht Ian Alexander, Alister Cockburn, Alan Davis oder Tom DeMarco? Oder sind es McMenamin und Palmer, möglicherweise auch Michael Jackson – und damit meine ich nicht den King of Pop. An den Kinderbuchautor Paul Maar haben Sie vermutlich nicht gedacht. Dabei können wir aus seinen Büchern einiges über Requirements Engineering lernen.

#### Eine Wunschmaschine mit Tücken

Stellen Sie sich vor, Sie haben eine wunderbare Maschine, die Ihnen jeden Wunsch erfüllen kann. Sie müssen ihn nur aussprechen. Allerdings hat diese Wunschmaschine eine störende Eigenschaft: Die Maschine nutzt jeden Interpretationsspielraum in der Wunschformulierung auf eine für Sie kaum vorhersehbare Weise.

In [Maa80] hat der Wunsch, ein Papagei solle sprechen, zur Folge, dass dieser Vogel eine ununterbrochene Folge von sinnlos aneinandergereihten Satzfragmenten von sich gibt. Der Wunsch ist damit zwar dem Wortlaut nach erfüllt, doch der Wünschende ist mit dem Resultat nicht zufrieden.

Er hatte es versäumt, seine Vorstellung vom „Sprechen“ genauer zu formulieren. Für ihn war klar, was Sprechen bedeutet, und so ist ihm gar nicht in den Sinn gekommen, dass die Wunschmaschine seinen

Wunsch auf eine ganz andere Art und Weise umsetzen könnte.

In der Systementwicklung ist es ganz ähnlich: Ob wir am Ende das richtige

System bekommen, hängt entscheidend davon ab, ob wir unsere Anforderungen ausreichend präzise formulieren. Dabei wissen wir zunächst nicht, wann unsere

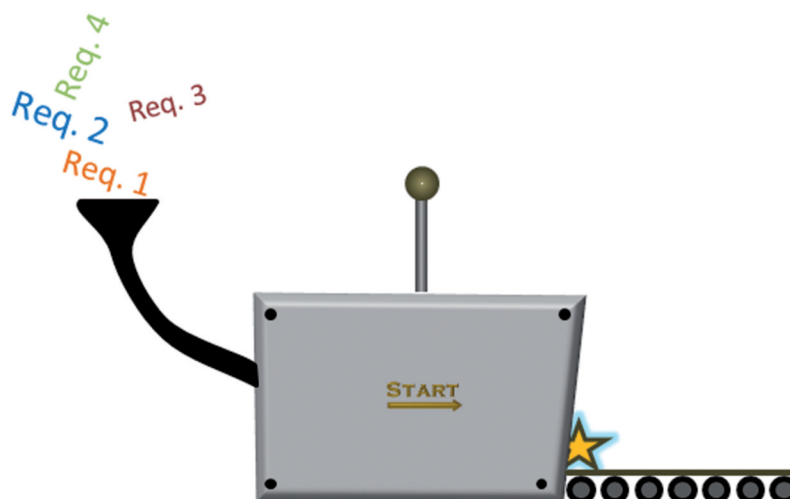


Abb.: Die Wunschmaschine

Anforderungen wirklich ausreichend präzise sind.

Begriffe und Konzepte, von denen wir eine klare Vorstellung haben, können durch andere Personen auch ganz anders verstanden werden. Erst wenn wir zurückgespiegelt bekommen, wie ein Entwickler unsere Anforderung interpretiert, wissen wir, ob wir ausreichend genau gewünscht haben.

Mit der Zeit lernen wir, die Ungenauigkeiten in unseren Anforderungen bereits im Vorfeld zu erkennen. Dann benötigen wir weniger Versuche, um das System zu bekommen, das wir uns vorstellen.

In den Büchern von Paul Maar finden sich auch noch weitere Analogien zum Requirements Engineering, zum Beispiel:

- Auch wenn wir eine Wunschmaschine haben, stehen wir bisweilen einem Problem ratlos gegenüber. Uns fällt dann einfach nicht der richtige Wunsch ein, der das Problem beheben würde.
- Wünschen wir uns zu viel in zu kurzer Folge, überhitzt die Maschine. Dann werden eine Zeit lang keine Wünsche mehr erfüllt. Wir sollten daher besser auf unnötige Wünsche verzichten.
- Eine defekte Wunschmaschine liefert grundsätzlich nicht das, was wir wollen. Egal, wie wir unseren Wunsch formulieren, es kommt das Falsche heraus. In einem solchen Fall, kann uns nur ein Experte helfen, der die Maschine repariert.

Experten, wie die Autoren dieses Themenspecials, können Ihnen dabei helfen, die Wunschmaschine mit den richtigen Anforderungen zu füttern sowie Fehler in der Umsetzung abzustellen.

### Spezifikationswerkzeuge mit eingebauten Prüffunktionen

Welche Unterstützung bieten heutige Entwicklungsumgebungen dem Programmierer? Und auf welche Unterstützung kann ein Spezifizierer im Normalfall zurückgreifen?

- Auf Vorschläge von Satzstrukturen gängiger Anforderungstypen?
- Auf eine Autovervollständigung zuvor definierter Bezeichner von Komponenten, Funktionen und Ein- und Ausgängen?
- Vielleicht auf eine Überprüfung der erstellten Spezifikation auf Konsistenz und Vollständigkeit?

Was davon bietet Ihre Werkzeugkette für das Requirements Engineering?

**Markus Fockel, Jörg Holtmann** und **Dr. Matthias Meyer** stellen den aktuellen Stand einer Methodik und Werkzeugunterstützung für textuelle Anforderungsspezifikationen vor. Der Ansatz bietet eine konstruktive Unterstützung beim Formulieren von Anforderungen ebenso wie automatische Qualitätsprüfungen. Zudem wird der Übergang zwischen textueller Spezifikation und modellbasierter Analyse erleichtert.

Woran denken Sie, wenn Sie den Begriff „formale Methoden“ hören? An kryptische Vorlesungsinhalte und bedienerunfreundliche Werkzeuge mit akademischem Anstrich? Vielleicht ist es nun an der Zeit, umzudenken. **Dr. Sebastian Siegl** und **Thomas Wöhr** haben zwei industrietaugliche Werkzeuge zur formalen Spezifikation und Verifikation in der Praxis evaluiert. In ihrem Beitrag beschreiben sie ihre Erfahrungen und Erkenntnisse.

### Von archaischer Verwaltung zu modernem Management

RE-Techniken wie z. B. User Stories und Anforderungsmodellierung haben sich in den letzten Jahren von skeptisch beäugten Ideen zu anerkannten Best Practices entwickelt. Gleichzeitig gelten Prozess- und Projektanforderungen, wie Nachvollziehbarkeit, systematische Wiederverwendung, Variantenmanagement, kollaboratives Arbeiten und offene Schnittstellen, nicht mehr als überzogen und unerreichbar.

Mit dieser Entwicklung konnten Anforderungsmanagement-Werkzeuge nur mühsam Schritt halten. **Dominik Jergus** und **Hubert Spieß** geben einen Überblick über ein neu konzipiertes Werkzeug, mit dem sich die modernen RE-Techniken in den Entwicklungsalltag integrieren lassen.

**Dr. Horst Kargl** und **Dr. Alexandra Mazak** beschreiben einen umfassenden Ansatz zum Management der Nachvollziehbarkeit zwischen Anforderungen und Entwurfs-elementen. Sie erläutern detailliert die Aufgaben im Projekt, die sich mit dem Ansatz bewerkstelligen lassen und gehen dabei auch auf die Umsetzung mittels eines Modellierungswerkzeugs ein.

### Adieu Anforderungsingenieur?

**Bernd Stehr** und **Artur Strasser** legen dar, weshalb der Product Owner in einer agilen Teamstruktur durch einen Requirements

Engineer unterstützt werden sollte. Der Requirements Engineer trägt in einer solchen Konstellation dank seines RE- und Domänenwissens maßgeblich zum Projekterfolg bei.

### Tester mit Qualitäten

**Rudolf Groetz** und **Dr. Frank Simon** sprechen sich in ihrem Beitrag für das frühzeitige Einbeziehen von Testern aus, damit diese zur Qualitätssicherung der Anforderungen beitragen. Die Autoren empfehlen dafür eine Schulung und Zertifizierung sowohl zum Requirements Engineer als auch zum Tester.

Tester sollten sich zudem durch Spezialkurse auf Branchen- oder technikspezifische Aufgabenstellungen vorbereiten. Als Beispiel führen die Autoren den „Certified Agile Tester“ an, das jüngste Erweiterungsmodul des International Software Testing Qualification Board (ISTQB).

### Suchen Sie noch nach dem Schuldigen oder entwickeln Sie schon?

Erinnern Sie sich noch an Ihr erstes Software-Entwicklungsprojekt? Vermutlich haben Sie es im Verlauf Ihres Studiums absolviert und haben dabei kaum einen Stolperstein ausgelassen. **Stefan Mieth** wirft einen Blick in seine Vergangenheit und nennt die Konzepte, mit denen sich die häufigsten Stolpersteine in Entwicklungsprojekten vermeiden lassen.

Sein Plädoyer: Anforderungen sollten nicht mit dem Ziel geschrieben werden, sich für den Fall des Scheiterns abzuschern. Denn damit wird eine Chance vertan. Richtig genutzt, können Anforderungen einen wichtigen Beitrag zur erfolgreichen Produktentwicklung leisten.

Und vergessen Sie vor allem nicht, genau genug zu wünschen! ■

*Dr. Ernst Sikora*

### Referenz

[Maa80] Paul Maar: Am Samstag kam das Sams zurück. Oetinger, 1980.