

# Vision und Wirklichkeit

## *Innovative Systementwicklung durch modernes Requirements Engineering*



www.photocase.com

*“I believe that this nation should commit itself to achieving the goal, before this decade is out, of landing a man on the moon and returning him safely to the earth.”* Als John F. Kennedy am 25. Mai 1961 diese technologische Vision formulierte, lag ihre Erfüllung noch acht Jahre in der Zukunft: Erst am 21. Juli 1969 sollte Neil Armstrongs legendärer Satz *„That’s one small step for man, one giant leap for mankind“* den ersten Schritt eines Menschen auf einen anderen Himmelskörper begleiten. Eine Vision beschreibt kurz und prägnant die gewünschte Veränderung der aktuellen Realität. Diese Veränderung kann nur durch innovative Lösungen erreicht werden. Im Bereich softwareintensiver Systeme – zum Beispiel beim Navigationssystem eines modernen Autos – werden Visionen zunehmend durch Software realisiert.

Dabei erhöht ein wachsender Verdrängungswettbewerb zunehmend die Kosten. Nur wer schnell und preisgünstig innovative Systeme anbietet, bleibt konkurrenzfähig. Der Wettbewerb verkürzt zudem die Entwicklungszyklen, weil innovative Systeme immer schneller am Markt verfügbar sein müssen.

Modernes Requirement Engineering definiert ausgehend von einer Vision innovative, lösungsorientierte Anforderungen an das zu entwickelnde System. Lösungsorientierte Anforderungen müssen so beschrieben sein, dass sie als verlässliche Grundlage für weitere Phasen der Systementwicklung, zum Beispiel die Implementierung oder den Test, dienen können.

### *Innovativer Ansatz*

Traditionell dokumentiert die Systemanalyse auf Grundlage eines bestehenden Systems oder Prozesses Anforderungen an ein neu zu entwickelndes System. Dabei stehen funktionale Ei-

genschaften im Vordergrund. Das sind zum Beispiel Funktionen („Das System speichert eine Adresse“), Daten („Die Adresse wird durch Vorname, Name, Straße, Hausnummer, Postleitzahl und Ort beschrieben“) und das Verhalten des Systems („Nach einer Authentifizierung wechselt der Geldautomat in den Zustand BE-REIT“). Diese Anforderungen nennt man lösungsorientiert, da sie die angestrebte Lösung beschreiben.

Für innovative Entwicklungen reicht eine reine Systemanalyse nicht aus, da sie immer von einem bereits bestehenden System ausgeht, das optimiert werden soll. Modernes Requirements Engineering beginnt dagegen mit einer Vision, die nur durch ein neues System bzw. neue Bestandteile des Systems verwirklicht werden kann. Visionen sind somit der Motor von Innovation.

Wie können aus einer Vision lösungsorientierte Anforderungen abgeleitet werden? Wie wird



zum Beispiel aus der Idee eines selbständig einparkenden Fahrzeuges die konkrete Anforderung an ein Steuergerät, das alle erforderlichen entsprechenden Sensordaten erfasst und die notwendigen Aktuatoren ansteuert? Ohne geeignete Techniken ist es kaum möglich, systematisch und möglichst vollständig alle notwendigen lösungsorientierten Anforderungen aus einer Vision abzuleiten.

Hier hat sich der Einsatz von Zielen und Szenarien bewährt. Ziele verfeinern Visionen. Sie initiieren die Definition von Szenarien. Szenarien konkretisieren Ziele und zeigen konkret, wie ein oder mehrere Ziele erreicht werden können. Mit Hilfe von Szenarien können Ziele validiert werden. Ziele und Szenarien dienen zur Ableitung von lösungsorientierten Anforderungen. Bei der Definition der lösungsorientierten Anforderungen kann es Rückkopplungen zu den Zielen und Szenarien geben (Abbildung 1).

### Von der Vision zum Ziel

Ziele dokumentieren die Absichten der Personen, die an einem System interessiert sind.



www.photocase.com

Früher spielte Software in Fahrzeugen keine Rolle. Heute werden in einem modernen Kraftfahrzeug über 2.500 Funktionen durch Software gesteuert.

Das sind zum Beispiel Kunden, Anwender, Marketingexperten, Entwickler oder Tester. Diese Gruppe wird als Stakeholder bezeichnet. Ziele definieren also weder die Nutzung noch die technische Realisierung eines Systems. Gefragt wird nicht nach den *Wie?*, sondern nach dem *Warum?*.

Ziele verfeinern Visionen. Das Ziel „Mehr Komfort durch personalisierte Einstellungsmöglichkeiten“ ist eine Verfeinerung von „Ein komfortables, sicheres und wirtschaftliches

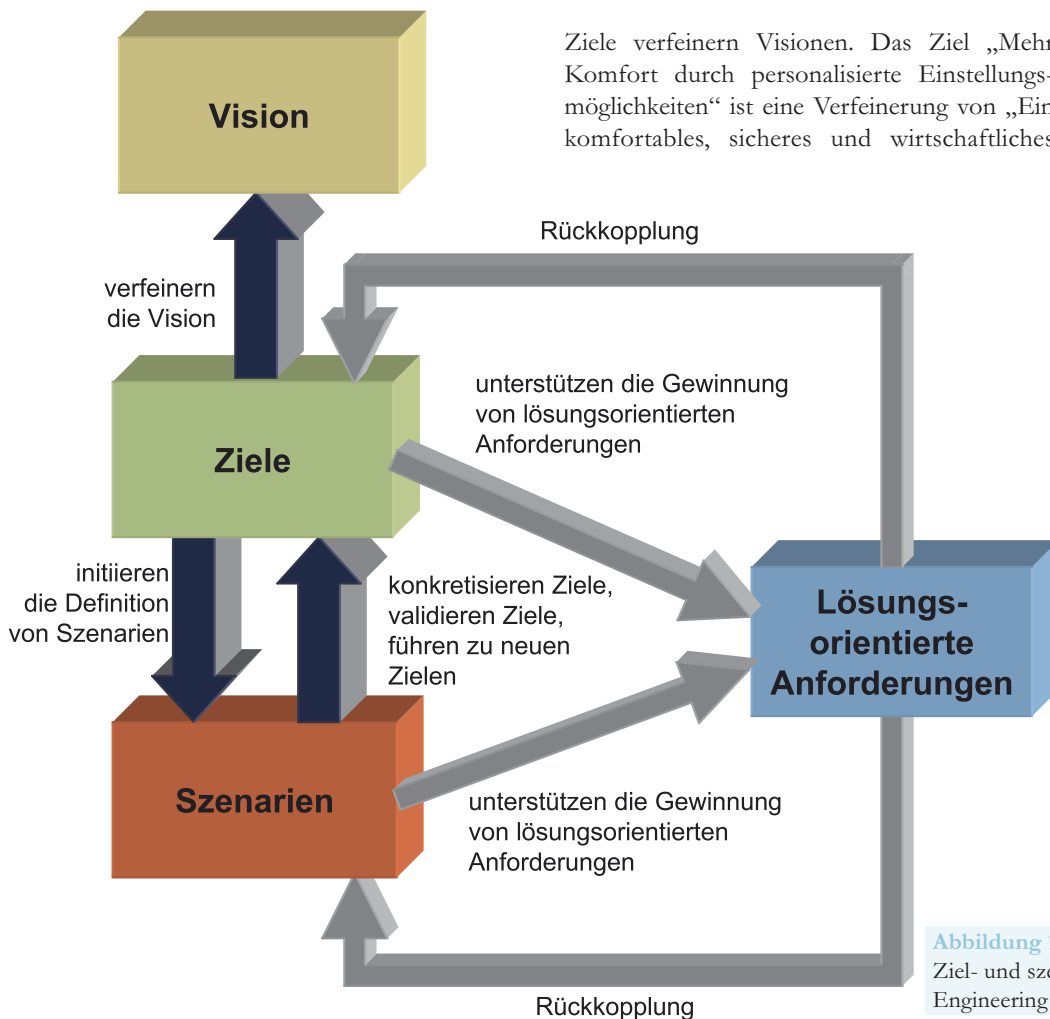


Abbildung 1: Ziel- und szenariobasiertes Requirements Engineering



Kraftfahrzeug, welches den Fahrer in jeder Fahrsituation unterstützt“.

Die Formulierung von Zielen bietet für die Entwicklung eines Systems folgende Vorteile:

- ➔ Akzeptanz und Systemverständnis der Stakeholder werden erhöht, da Ziele den Mehrwert des Systems erkennbar machen.
- ➔ Ziele unterstützen die Gewinnung von lösungsorientierten Anforderungen, da für jedes Ziel geklärt werden muss, durch welche Anforderung(en) es erfüllt werden kann. Umgekehrt dienen Ziele zur Überprüfung, ob eine gewonnene lösungsorientierte Anforderung tatsächlich für das System benötigt wird.
- ➔ Durch die Betrachtung von Zielen und Verfeinerungen können Alternativen erkannt werden. Das Ziel „Zugangsberechtigung zu einem Gebäude“ kann zum Beispiel durch eine Schlüsselkarte oder durch Eingabe einer PIN oder durch beide Verfahren erfüllt werden.
- ➔ Durch die Definition von Zielen können bereits frühzeitig Konflikte aufgedeckt werden. So kann das Ziel, alle Daten eines Fahrers in einem Kraftfahrzeug zu sammeln und zu speichern, mit Datenschutzbestimmungen kollidieren. Im Requirements Engineering muss dieser Konflikt spätestens bei der Ableitung der lösungsorientierten Anforderungen aufgelöst werden, da diese widerspruchsfrei sein müssen.

Im gezeigten Modell wird das Ziel „Sicheres, komfortables Kraftfahrzeug“ unter anderem durch die Unterziele „Schutz vor Diebstahl“ und „Unterstützung persönlicher Fahrzeugeinstellungen“ verfeinert. Neben anderen Zielen (im Graphen durch „[...]“ gekennzeichnet) müssen diese beiden Bedingungen erfüllt sein, um im Ergebnis ein sicheres und komfortables Kraftfahrzeug zu erhalten. Das Ziel „Schutz vor Diebstahl“ wird durch eine Und-Zerlegung in die beiden Unterziele „Diebstahlschutz durch Alarmsystem“ und „Lokalisierbarkeit des Fahrzeugstandorts“ verfeinert. Der Diebstahlschutz kann durch zwei alternative Ziele, nämlich durch einen akustischen oder einen visuellen Alarm realisiert werden. Das Ziel, personalisierte Fahrzeugeinstellungen vornehmen zu können, ist durch eine Und-Zerlegung verfeinert worden: Um die gewünschten Einstellungen vornehmen zu können, wird es unter anderem notwendig sein, sich zu identifizieren.

Mit dem Und-Oder-Graphen lässt sich die Vision durch Ziele weiter verfeinern und konkretisieren. Vielfach sind diese aber immer noch so abstrakt, dass es einer weiteren Konkretisierung bedarf. Dies geschieht durch die Definition von Szenarien.

## Vom Ziel zum Szenario

Im Requirements Engineering beschreibt ein Szenario eine Folge von Interaktionsschritten, die zur Erfüllung eines oder mehrerer Ziele führt.

Das folgende Szenario konkretisiert das Ziel, sich in einem Fahrzeug durch die Eingabe einer PIN zu identifizieren: Der Fahrer öffnet die Tür mit einem Schlüssel, die Elektronik wird aktiviert, der Steuerrechner fragt nach der PIN, der Fahrer gibt sie ein. Anschließend kann er personalisierte Einstellungen etwa in Bezug auf die Sitzeinstellungen abrufen.

Die Beschreibung von Szenarien bietet für die Entwicklung eines Systems unter anderem folgende Vorteile:

- ➔ Die Stakeholder können sich eine wesentlich konkretere Vorstellung über die Abläufe zur Zielerfüllung machen.
- ➔ Szenarien können in unterschiedlichen Bereichen eingesetzt werden. Ein Manager kann in Form einer kurzen Geschichte erläutern, auf welche Weise das zukünftige Auto automatisch durch eine Stadt geführt werden soll. Ein Szenario kann aber auch genutzt werden, um die Interaktionen zweier Softwarekomponenten zu beschreiben.

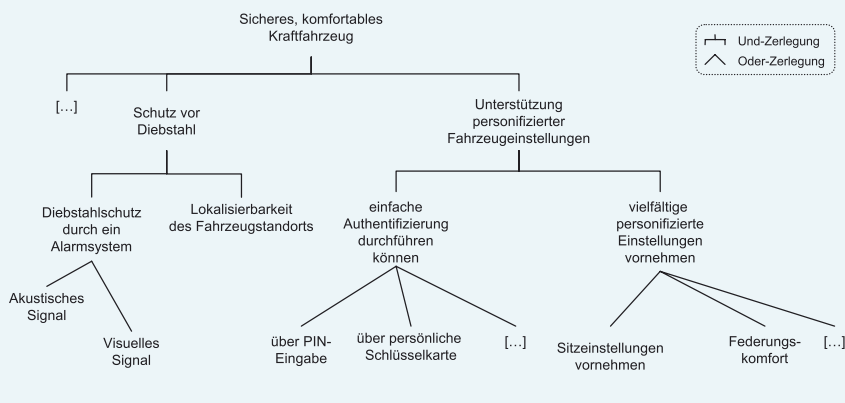


Abbildung 2:

Beispiel für die Zielmodellierung mit Und-Oder-Graphen

Die Verfeinerung von Zielen wird in so genannten Und-Oder-Graphen dargestellt. Bei einer Und-Beziehung müssen zur Erfüllung des Oberziels *alle* entsprechenden Unterziele erreicht werden. Eine Oder-Beziehung gibt dagegen alternative Verfeinerungen eines Oberziels an. Und-Oder-Graphen haben den Vorteil, dass Zusammenhänge zwischen einzelnen Zielen leicht erkennbar sind und dadurch mit den Stakeholdern diskutiert werden können (Abbildung 2).

## Anforderungsbasiertes Testen

- ➔ Szenarien können unterschiedlich beschrieben werden: Informell als Fließtext, in strukturierter Form als Schablonen oder auch formal in so genannten „Message Sequence Charts“. Dadurch kann man die Präsentation den jeweiligen Erfordernissen anpassen.
- ➔ Die im Requirements Engineering dokumentierten Szenarien eignen sich sehr gut als Ausgangspunkt zur Qualitätssicherung, insbesondere für den Test des entwickelten Systems. Aus den Szenarien können Testfälle abgeleitet werden, um Fehler in der Implementierung/Realisierung des Systems aufzudecken. Darüber hinaus können bereits durch die Ableitung der Testfälle aus den Szenarien Fehler in den Anforderungen gefunden werden (vgl. Kasten „Anforderungsbasiertes Testen“).

Im Requirements Engineering unterscheidet man verschiedene Arten von Szenarien. Das *Haupt-szenario* beschreibt den normalen Handlungsablauf zwischen Nutzer und System. *Normal* bedeutet hier, dass keine Ausnahmesituation – zum Beispiel eine Störung - eintritt. Abbildung 3 zeigt das Hauptszenario für die personalisierte Einstellung eines Fahrzeugsitzes. Die Spaltenüberschriften „Fahrer“ und „Steuerrechner“ bezeichnen die Akteure des Szenarios. Akteure eines Szenarios sind Personen oder Systeme, die z. B. einzelne Szenarioschritte anstoßen bzw. durchführen.

In *Alternativszenarien* können die gesamte Abfolge oder einzelne Interaktionsschritte als Alternative zu einem Hauptszenario beschrieben werden. Ein Ausnahmeszenario wiederum beschreibt Situationen, in denen eine Störung oder ein Fehler auftritt und somit das verfolgte Ziel nicht erreicht werden kann.

Die Vorteile des Einsatzes von Zielen und Szenarien im Requirements Engineering liegen insbesondere auch in der Wechselwirkung zwischen Zielen, Szenarien und lösungsorientierten Anforderungen.

### Positive Wechselwirkungen

Noch einmal zurück zum Ziel „Sitzstellungen vornehmen können“ aus den Abbildungen 2 und 3. In einem Workshop zur Abstimmung des Szenarios taucht die Frage auf, wie die Position der Außen- und Rückspiegel berücksichtigt werden könne, da diese bei veränderter Sitzposition oftmals ebenfalls angepasst werden muss. Es entsteht das weitere Unterziel „vielfältige personalisierte Einstellungen vornehmen“, welches die Einstellung von Spiegeln im Zusammenhang

Die Anforderungen an ein softwareintensives System beeinflussen alle Schritte der Systementwicklung. So werden ausgehend von den Anforderungen zum Beispiel die Software-Architektur entworfen und die einzelnen Software-Bausteine implementiert. Die Qualität der Anforderungen hat daher signifikante Auswirkungen auf die Qualität des endgültigen Produkts, da Fehler in den Anforderungen zu Folgefehlern in der Architektur und der Implementierung führen.

Auch bei der Durchführung von Entwurfs- und Implementierungsaktivitäten lassen sich Fehler nie vollständig ausschließen. Es können daher Fehler in der Architektur und der Implementierung des Systems vorhanden sein, selbst wenn die Anforderungen fehlerfrei sind. Der Software-Test hat eine sehr hohe praktische Bedeutung für die Aufdeckung solcher Fehler. So wird z. B. bei einem Systemtest ein softwareintensives System gezielt ausgeführt, um Abweichungen von den Anforderungen aufzudecken. Jede gezielte Ausführung des Systems wird dabei durch einen Testfall beschrieben, der definiert, welche Eingaben in das System gemacht und welche Ausgaben erwartet werden.

Die Durchführung eines Testfalls bedeutet, dass die im Testfall definierten Eingaben an das System übermittelt und die Ergebnisse, die vom System geliefert werden, abgelesen werden. Die Durchführung eines Testfalls beinhaltet daher eine Folge von Interaktionen zwischen dem Benutzer und dem System. Hierbei handelt es sich also um Szenarien (siehe Hauptbeitrag). Die Szenarien aus dem Requirements Engineering bieten daher einen ausgezeichneten Startpunkt zur Definition von Testfällen. Man wählt zunächst die zu prüfenden Szenarien aus und bestimmt dann die für den Ablauf des Szenarios zu tätigen Eingaben und die erwarteten Ausgaben.

Testaktivitäten können in die Definition und in die Ausführung von Testfällen unterteilt werden. Anders als die Ausführung von Testfällen ist deren Definition auch ohne die Existenz eines lauffähigen Systems möglich. Ein Teil der Testaktivitäten lässt sich daher parallel zum Requirements Engineering durchführen. Die Definition von Testfällen auf Basis der Anforderungen, d. h. das anforderungsbasierte Testen, hat dabei einen positiven Effekt auf die Qualität der Anforderungen. Da ein Tester die Anforderungen verstehen muss, um Testfälle zu definieren, können viele Fehler in den Anforderungen bereits frühzeitig aufgedeckt werden. Zu den häufigsten Fehlern zählen dabei Unvollständigkeits- und Widersprüche.

Am Institut für Informatik und Wirtschaftsinformatik (ICB), Arbeitsgruppe Software Systems Engineering (SSE), wird ein weiterführender Ansatz für das anforderungsbasierte Testen untersucht. Bei diesem Ansatz wird bei der Testfallableitung in einem Zwischenschritt ein Testmodell konstruiert. Aus diesem Testmodell werden dann – mit Unterstützung von Software-Werkzeugen – automatisch Testfälle abgeleitet. Die hierzu in der Arbeitsgruppe SSE entwickelte ScenTED-Technik ermöglicht es, ausgehend von den im Requirements Engineering identifizierten Szenarien das Gesamtverhalten des Systems in einem Testmodell (einem Aktivitätsdiagramm der Unified Modeling Language UML) zu beschreiben. Der Einsatz der ScenTED-Technik in der Industrie hat eine deutliche Unterstützung bei der anforderungsbasierten Testfallableitung gezeigt. Aktuell arbeitet die Arbeitsgruppe SSE im Rahmen zweier Forschungsprojekte an der Erweiterung der ScenTED-Technik zur Durchführung risikobasierter Tests (BMBF-Verbundprojekt ranTEST) und der Unterstützung von Testaktivitäten in der Software-Produktlinienentwicklung (DFG-Projekt IST-SPL).

Abbildung 3:  
Hauptszenario „Personifizierte  
Sitzeinstellung“ in einer tabella-  
rischen Beschreibung

Szenario „Personifizierte Sitzeinstellung“	
Fahrer	Steuerrechner
1. Aktivierung des Kraftfahrzeugs durch Schlüssel	
	2. Abfrage der Authentifizierung
3. Authentifizieren	
	4. Prüfen der Authentifizierung auf Korrektheit
	5. Abfrage, ob die personalifizierte Sitzeinstellung vorgenommen werden soll
6. Bestätigen der Funktion	
	7. Sitz in die gespeicherte Position fahren

mit einer bestimmten Sitzposition zum Ziel hat. Abbildung 4 zeigt den angepassten Ausschnitt aus dem Und-Oder-Graphen.

Das Beispiel zeigt eine Wechselwirkung bei der Entwicklung von Zielen und Szenarien. Durch die Beschreibung von Szenarien werden weitere Ziele identifiziert, da die Stakeholder eine konkretere Vorstellung über die Abläufe und Interaktionen mit dem System und somit ein besseres Problemverständnis gewinnen.

Die Bearbeitung eines Szenarios kann auch dazu führen, dass bestehende Ziele verändert oder vollends aufgegeben werden müssen. Diese Wechselwirkung ist besonders für die Validierung von Zielmodellen von Bedeutung.

Die Entwicklung von Zielen und Szenarien beeinflusst sich gegenseitig. Der schrittweise Prozess dieser Entwicklung führt zu immer feiner definierten Zielen sowie zu konkreteren Szenarien. Auf diese Weise nimmt das Problemverständnis für das System zu. Abschließend müssen aus den Zielen und Szenarien die lösungsorientierten Anforderungen abgeleitet werden (Abbildung 1),

damit die Entwicklung in der Lage ist, das System zu realisieren.

### Lösungsorientierten Anforderungen

In lösungsorientierten Anforderungen muss die geforderte Funktionalität und Qualität eines Systems weiter präzisiert werden. Ziele und Szenarien bieten dazu eine sehr gute Grundlage. Beispielsweise lassen sich aus einem Szenarioschritt lösungsorientierte Anforderungen an das System formulieren.

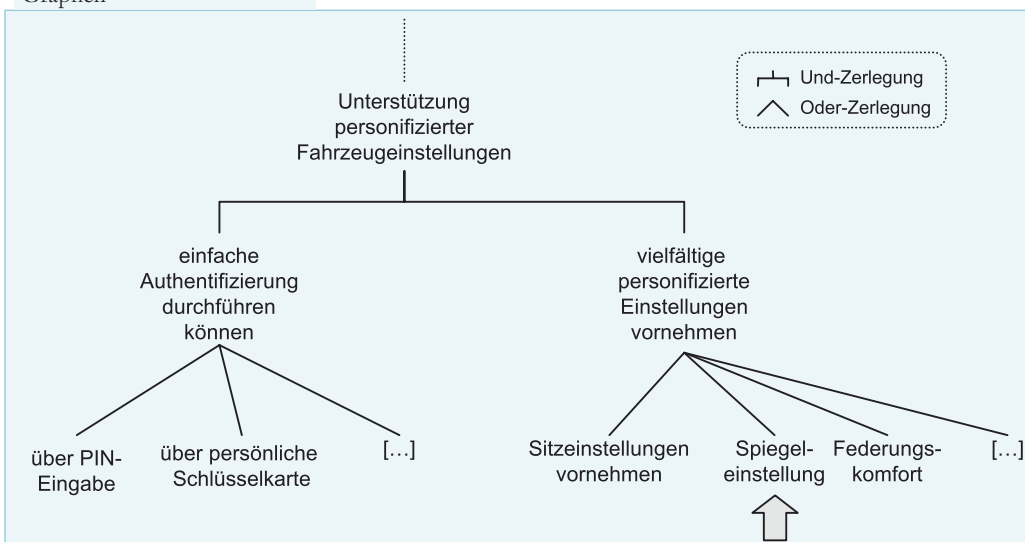
Dies soll anhand von Schritt 7 aus dem Hauptszenario (Abb. 3) demonstriert werden. Das Szenario sieht an diesem Punkt vor, dass der Steuerrechner den Sitz in die personalifizierte Sitzposition bringt. Aus diesem Schritt lassen sich unter anderem folgende konkrete und lösungsorientierte Anforderungen ableiten:

- A.1: Das System muss die gespeicherte Sitzposition abrufen
- A.2: Das System muss die Koordinaten für die gewünschte Sitzposition an die Steuergeräte X und Y weitergeben
- A.3: Das System muss eine Rückmeldung erhalten, wenn die geforderte Sitzposition erreicht ist
- A.4: ...

Auf diese Weise lässt sich für jeden Schritt analysieren, welche Funktionen und Qualitätseigenschaften das System haben muss, damit der betrachtete Schritt ausgeführt werden kann.

Die schnelle Entwicklung von innovativen Systemen ist in einem Markt mit starkem Verdrängungswettbewerb eine notwendige Voraussetzung, um als

Abbildung 4:  
Erweiterung des Und-Oder-  
Graphen





## TÜV Rheinland Kraftfahrt GmbH

Personalrecruiting und -entwicklung    Tel: ++49 221- 806 2440  
 Am Grauen Stein    Mail: karriere@de.tuv.com  
 51105 Köln    Homepage: www.tuv.com



### Ihr neuer Arbeitgeber:

Die TÜV Rheinland Group mit Sitz in Köln zählt zu den international führenden Dienstleistungskonzernen für dokumentierte Sicherheit und Qualität.

Mit 10 000 Mitarbeiterinnen und Mitarbeitern erwirtschaftet sie einen Gesamtjahresumsatz von rund 850 Millionen Euro (vor Konsolidierung). Das Unternehmen ist mit einem Netzwerk von 100 Tochtergesellschaften an 300 Standorten in über 50 Ländern in nahezu allen wichtigen Wirtschaftszentren der Welt vertreten.

### Was wir Ihnen bieten....

Beste Entwicklungsmöglichkeiten, hervorragende Weiterbildungsprogramme, sowie eigenverantwortliches Arbeiten. Weiter bieten wir Ihnen ein attraktives und leistungsgerechtes Gehalt. Zusätzlich werden Entwicklungsprogramme weltweit, sowohl für Fach- und Führungskräfte angeboten.

### Wir suchen....

#### Diplomingenieure zur Ausbildung zum Prüflingenieur (w/m)

*Fachrichtung:* Kraftfahrzeugtechnik/ Maschinenbau/ Elektrotechnik  
*Regionen:* Mönchengladbach, Krefeld, Köln, Aachen, Dortmund, Duisburg, Hamm, Oberhausen, Düsseldorf, Wuppertal, Essen, Siegen, Bonn, Gelsenkirchen, Recklinghausen, Olpe, Lüdenscheid, Betzdorf

*Kennziffer:* PIA 061

#### Eisenbahningenieure (w/m)

*Standort:* Köln  
*Kennziffer:* ISA 061

#### Verkehrs- und Telematikingenieure (w/m)

*Standort:* Köln  
*Kennziffer:* TIT 061

#### Mechatroniker (w/m)

*Standort:* Köln  
*Kennziffer:* TVSS 061

#### Fahrzeugtechniker (w/m)

*Standort:* Köln  
*Kennziffer:* TVS 061

**Sind Sie interessiert? Oder haben Sie Fragen? Wir freuen uns auf Ihre Unterlagen, Ihre e-mail oder Ihren Anruf.**

Erforderliche Unterlagen: Bewerbung, Lebenslauf und Zeugnisse, sowie die Angabe von Gehaltsvorstellung und bevorzugtem Standort

Anbieter und Entwickler von softwareintensiven Systemen bestehen zu können. Ein ziel- und szenariobasiertes Requirements Engineering unterstützt die Entwicklung innovativer Systeme, in dem eine Vision durch Ziele immer weiter verfeinert wird und diese Ziele durch Szenarien konkretisiert werden, die beispielhaft Lösungswege aufzeigen. Ziele und Szenarien bilden so eine sehr gute Basis für die Definition lösungsorientierter Anforderungen, welche als weitere Grundlage für die Entwicklung dienen (vgl. Kasten „Anforderungsbasiertes Testen“). Weiterführende Informationen über ziel- und szenariobasiertes Requirements Engineering finden Sie in dem Buch „Requirements Engineering, Grundlagen, Prinzipien, Techniken“ von Klaus Pohl beim dpunkt.verlag.

Aktuell beschäftigt sich der Bereich Software Systems Engineering des Instituts für Informatik und Wirtschaftsinformatik (ICB) mit dem vom BMBF geförderten Forschungsprojekt SoftWiki. Im Zentrum steht die Frage, wie eine ziel- und szenariobasierte Vorgehensweise im Requirements Engineering für die verteilte Entwicklung mit einer großen Anzahl von Stakeholdern ein-

gesetzt werden kann. Darüber hinaus wird in dem ebenfalls vom BMBF geförderten Forschungsprojekt REMsES der spannenden Frage nachgegangen, wie Ziele und Szenarien das Requirements Engineering in der Entwicklung eingebetteter Systeme unterstützen und die Verknüpfung von Hard- und Software geeignet repräsentieren können.

## Kontakt

**Prof. Dr. Klaus Pohl**

**Günter Halmans**

**Dr. Andreas Metzger**

Software Systems Engineering

Tel.: 02 01 / 1 83 - 46 60

klaus.pohl@sse.uni-due.de

guenter.halmans@sse.uni-due.de

andreas.metzger@sse.uni-due.de

<http://www.sse.uni-due.de>

