

Modellbasierte Methode zur Ableitung nicht-funktionaler Anforderungen im Kontext der Softwaremodernisierung

Lukas Strey¹, Christian Hein¹, Tom Ritter¹, Christian Knauer² und Angelika Ganz²

Abstract: Komplexe softwarebasierte Systeme müssen sowohl funktionale als auch nicht-funktionale Anforderungen (NFA) erfüllen, um von den Nutzern akzeptiert und als Unterstützung der Arbeitsaufgaben angenommen zu werden. Wir stellen in diesem Artikel eine modellbasierte Methode zur systematischen Ableitung nicht-funktionaler Anforderungen aus einem harmonisierten Katalog von Qualitätskriterien dar. Diese Methode wurde im Kontext der Modernisierung der Haushaltsverfahren des Bundes entwickelt und angewendet. Zunächst wurde ein Katalog von Eigenschaften erstellt, in dem generische nicht-funktionale Anforderungen aufgeführt sind. Der NFA-Katalog ist nach den Qualitätskriterien der ISO-Normenreihe 25000 strukturiert und erweiterbar. Dieser Katalog steht für die konkreten Entwicklungs- oder Modernisierungsvorhaben bspw. für die Erstellung von Ausschreibungen zur Verfügung. In einem methodischen Vorgehen wird mit Hilfe von Mustererkennung in Systemmodellen ein Softgoal-Modell aufgebaut, das als Filtermechanismus für die Auswahl der relevanten nicht-funktionalen Eigenschaften dient. Durch eine Anreicherung mit Systemkontextinformationen werden so konkrete und auf das System bezogene Anforderungen abgeleitet, welche sich für die Systementwicklung nutzen lassen.

Keywords: Nicht-funktionale Anforderungen; Modellierung; Modernisierung; Softgoal; BPMN

1 Einleitung

Bei der Entwicklung umfangreicher und verteilter IT-Systeme kommt es auf die korrekte Umsetzung der geforderten Funktionalität an. Bereits hierbei ergeben sich große Herausforderungen insbesondere in der Modellierung der funktionalen Anforderungen. Für den realen Einsatz solcher Systeme besitzen nicht-funktionale Anforderungen (NFA), welche auch als Qualitätsanforderungen bezeichnet werden, ebenfalls eine große Bedeutung. Ein funktional korrekt implementiertes System ist unbenutzbar, wenn bspw. das Antwortzeitverhalten nicht akzeptabel ist.

Im Rahmen der Modernisierung der Haushaltsverfahren des Bundes ist die Qualitätssicherung in allen Modernisierungsschritten von Anfang an fest eingeplant. Es wird ein einheitliches methodisches Vorgehen bereits in der Anforderungserhebung verfolgt und der Berücksichtigung von NFAs hohe Priorität gegeben. Das Ziel ist es, in Ausschreibungen eine hohe Abdeckung bzgl. NFAs und einen hohen Grad an Wiederverwendbarkeit für

¹ Fraunhofer FOKUS, SQC, Kaiserin-Augusta-Allee 31, 10589, Berlin, Deutschland {lukas.strey,christian.hein,tom.ritter}@fokus.fraunhofer.de

² Bundesministerium der Finanzen, Referat II A 6, Wilhelmstraße 97, 10117, Berlin, Deutschland

ähnliche Vorhaben zu erreichen. Zugleich sollen Fachexperten der Haushaltsverfahren intensiv in die Modernisierung eingebunden werden. Weiterhin wird in der Modernisierung konsequent auf möglichst vollständige Modellierung gesetzt, um auch auf diese Weise ein hohes Qualitätsniveau bspw. bzgl. der Konsistenz der Anforderungen zu erreichen.

Die Spezifikation nicht-funktionaler Anforderungen ist eine zur Spezifikation funktionaler Anforderungen vergleichbar komplexe und schwierige Aufgabe. So müssen NFAs auf den unterschiedlichen Abstraktionsebenen der logischen und technischen Architektur- und Prozessmodelle jeweils adäquat definiert werden. Orientierung bieten vorhandene Qualitätsnormen und -standards, die Qualitätskriterien in Form von Modellen beschreiben. Für ein konkretes System muss betrachtet werden, welches Qualitätskriterium in welcher Ausprägung für welches Element der funktionalen Architektur relevant ist, um dafür konkrete NFAs zu formulieren.

In der praktischen Anwendung ergeben sich dafür Erfahrungswerte und es lassen sich Muster erkennen, woraus wir eine Methode entwickelt und diese im Kontext der Modernisierung der IT-Systeme der Haushaltsverfahren angewendet haben. Mit dieser Methode lassen sich systematisch Qualitätskriterien für die Bestandteile der Architektur- und Prozessmodelle finden und daraus auf Basis eines umfassenden Katalogs nicht-funktionale Anforderungen ausformulieren. Zugleich ist diese Methode leichtgewichtig und lässt sich mit anderen Ansätzen kombinieren.

Überblicksartig stellen wir in diesem Beitrag diese modellbasierte Methode zur Ableitung nicht-funktionaler Anforderungen im Kontext der Softwaremodernisierung vor. Die Methode umfasst (1) den Aufbau und Pflege eines Qualitätskriterienkatalogs, in dem generische formulierte NFAs vorliegen, (2) der Aufbau eines Softgoal-Modells als essentieller Filtermechanismus, (3) die systematische Untersuchung von Systemmodellen bzgl. der Relevanz von NFAs mithilfe des Softgoal-Modells und (4) die Ableitung einer strukturierten Menge konkreter NFAs für ein gegebenes Systemmodell, mit der Systemanforderungen bspw. für Ausschreibungen eindeutig spezifiziert sind. Die Methode stellt eine Unterstützung der Fach- und Softwareexperten dar und wird als Ergänzung zu anderen etablierten Verfahren benutzt. So lassen sich die Zwischenergebnisse bspw. in Anforderungsanalyse- und Brainstorming-Phasen benutzen bzw. verfeinern.

2 Der NFA-Katalog

Das in diesem Artikel beschriebene Vorgehen stützt sich auf die Verwendung eines NFA-Katalogs, der einen Wissensspeicher von vordefinierten bzw. zu verfeinernden NFAs darstellt, die jeweils bestimmten Qualitätskriterien zugeordnet sind. Der NFA-Katalog enthält in allgemeiner Form generisch beschriebene NFAs, die für ein konkretes System mit Kontextinformationen angereichert werden. Als Grundlage für die Struktur des Katalogs wurde die ISO-Normenreihe 250xx herangezogen. Dennoch ist die Struktur flexibel erweiterbar und kann mit neuem Wissen iterativ angereichert werden.

2.1 ISO-Normenreihe 250xx

Die ISO-Norm 25000 bildet den Einstieg in die Normenreihe für "Qualitätskriterien und Bewertung von Softwareprodukten (SQuaRE)" [IS14]. ISO-Norm 25010 beschreibt die Qualitätsmodelle für *Product Quality* (Produktqualität) und *Quality in Use* (Anwendungsqualität) [IS11]. Außerdem beschreibt die ISO-Norm 25012 ein Modell für *Data Quality* (Datenqualität) [IS08]. Diese Qualitätsmodelle nennen einzelne Charakteristika, welche den jeweiligen Qualitätsaspekt des Modells betreffen. Für *Product Quality* und *Quality in Use* ist jedes Charakteristikum in einzelne Sub-Charakteristika unterteilt. Zusammengefasst beschreiben die Charakteristika der drei Modelle umfassend die Qualität eines Softwareprodukts. ISO-Normen liefern außerdem Messgrößen für quantifizierbare Charakteristika der drei Qualitätsmodelle, ISO-Norm 25022 für *Quality in Use* [IS16a], Norm 25023 für *Product Quality* [IS16b] und Norm 25024 für *Data Quality* [IS15].

2.2 Vorgehen zum Aufbau des Katalogs

Abbildung 1 zeigt die Struktur des NFA-Katalogs. Erkennbar ist die Unterteilung entsprechend der drei Qualitätsmodelle in einzelne Teilbereiche, sowie die Struktur aus Charakteristika und Sub-Charakteristika. Die vollständige Auflistung der Charakteristika ist in den entsprechenden ISO-Normen zu finden. Als Grundlage bei der Formulierung einer

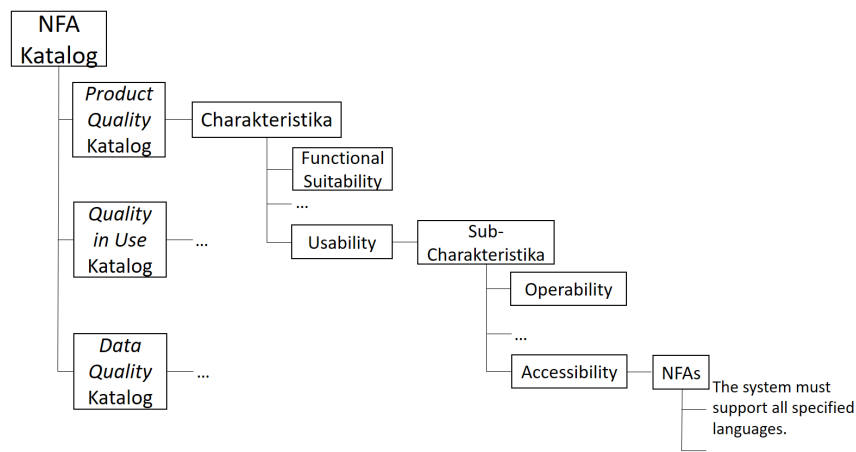


Abb. 1: Struktur des NFA-Katalogs entsprechend der ISO-Normenreihe 25000

ersten Fassung generischer NFAs dienen die genannten Messgrößen. Aus jeder Messgröße wird mindestens eine NFA abgeleitet, welche die Anforderung bzgl. der Charakteristik quantifiziert beschreibt. Die so gewonnenen NFAs werden entsprechend in die Struktur des NFA-Katalogs eingeordnet. In der somit erstellten Version des Katalogs sind 191 generische

NFAs enthalten. In der ersten Fassung des Katalogs sind die Begriffe der englischen ISO-Normen übernommen und zur Wahrung der Konsistenz die NFAs ebenfalls auf Englisch formuliert. Für den Einsatz im Bundesministerium ist eine deutsche Übersetzung geplant, um eine Einbindung der Fachexpertise zu ermöglichen.

Jede generische NFA im Katalog bezieht sich auf ein Subjekt wie System, Service oder Database. Außerdem kann mithilfe des Verbs die Priorität definiert werden (can, shall, must). Eine Beispiel-NFA aus dem Katalog lautet: „The system must support all specified languages“ (Product Quality - Usability - Accessibility); wobei sowohl Subjekt als auch Verb durch den Anwender kontextspezifisch definiert werden, bspw.: „The user interface shall support all specified languages“. Weiterhin muss der Anwender in einigen quantifizierbaren NFAs konkrete Werte festlegen: „The system must have a test coverage of at least [X]%“ (Product Quality - Reliability - Maturity).

2.3 Erweiterbarkeit des Katalogs

Der initiale Katalog ist in zweierlei Hinsicht erweiterbar. Neue NFAs können in die bestehende Struktur eingeordnet werden, entsprechend der Funktionalität des Katalogs als Wissensspeicher. Dieser kontinuierliche Aufbau des Katalogs muss planvoll durchgeführt werden, so sollte vor dem Hinzufügen neuer NFAs eine Dublettenprüfung stattfinden und der Detailgrad im Katalog konstant gehalten werden. Außerdem kann die Struktur des Katalogs erweitert werden, indem neue Unterkataloge mit entsprechenden Charakteristika und Sub-Charakteristika definiert werden. Hierdurch lassen sich wiederkehrende, projektspezifische Kontexte abbilden. Bei der Modernisierung der Haushaltsverfahren ist dies für spezielle Reglementierungen relevant.

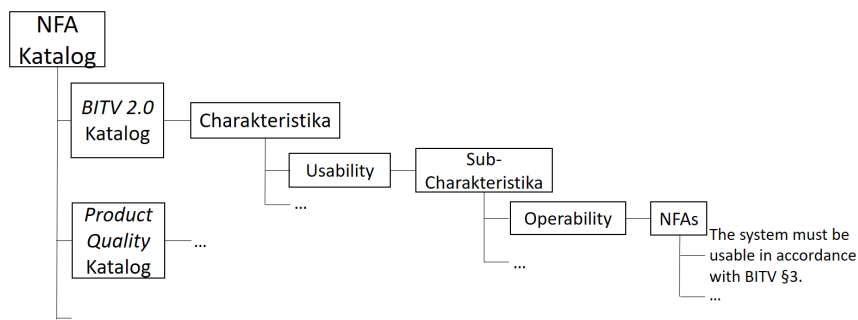


Abb. 2: Erweiterung des NFA-Katalogs um BITV 2.0

Die erste durchgeführte Erweiterung des Katalogs ist die Hinzunahme der Barrierefreie-Informationstechnik-Verordnung (BITV 2.0) [BIT11]. Die BITV 2.0 regelt die Barrierefreiheit, unter anderem für elektronische Dienstleistungen öffentlicher Stellen und elektronische Verwaltungsabläufe, in Deutschland. Aus der Analyse der BITV 2.0 gewonnene NFAs sind,

aufgrund des speziellen Detailgrads, in einem eigenen BITV 2.0 Katalog gesammelt. Abbildung 2 verdeutlicht die Erweiterung des NFA-Katalogs um einen speziellen Unterkatalog für BITV 2.0 NFAs. Einschließlich dieser Erweiterung umfasst die aktuell genutzte Version des Katalogs insgesamt 207 generische NFAs.

3 Modellbasierte Methode zur Anwendung des NFA-Katalogs

Der nächste Schritt ist die Abbildung des Katalogs in einem gegebenen Kontext, um konkrete NFAs für diesen Kontext zu erzeugen. In Anbetracht der geplanten kontinuierlichen Erweiterung des Katalogs ist eine Methode zur Filterung notwendig. Dabei wird der Gesamtsatz an NFAs im Katalog sinnvoll gefiltert und dem Nutzer zur Auswahl angeboten. Die Filterung berücksichtigt die Struktur des Katalogs und erlaubt eine grobe Vorauswahl nach relevanten Sub-Charakteristika, wodurch die Anzahl der zu betrachteten NFAs reduziert wird. Als Input für die Filterung des Katalogs können sowohl Architektur- als auch Prozessmodelle verwendet werden. Besonders bei der Automatisierung von Abläufen und der gemeinsamen Ausrichtung von Geschäftsprozessen und IT ist der Input einer Prozessbeschreibung relevant. Die Beschreibung und Darstellung der Filterung erfolgt mithilfe von Softgoal-Modellierung.

3.1 Softgoal-Modellierung

Softgoal Interdependency Graphen (SIG) dienen zur Modellierung von Softgoals, also Zielen welche nicht eindeutig erfüllbar sind. SIG zeigen Softgoals mit ihren gegenseitigen

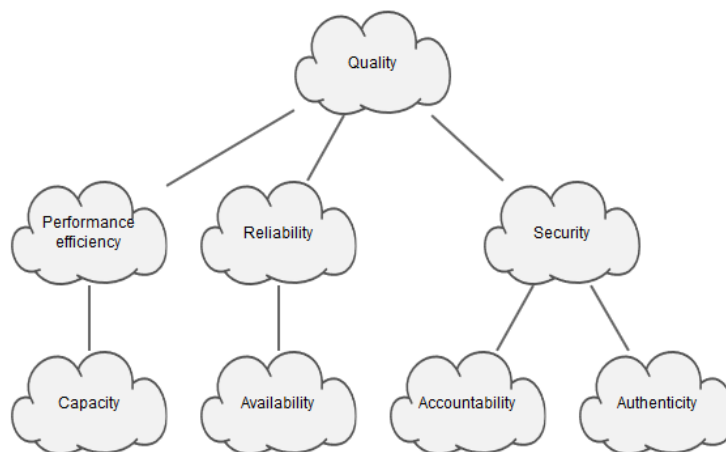


Abb. 3: Beispiel eines Softgoal-Baums

Einflüssen an, bspw. ob ein Softgoal andere Softgoals behindert oder unterstützt [CL09]. Bei der Anwendung des NFA-Katalogs werden Softgoals zur Modellierung der Filterkriterien verwendet. Dafür wird eine Baumstruktur aus Softgoals gebildet. Dieser Softgoal-Baum bildet auf seiner untersten Ebene die Sub-Charakteristika ab, welche im Zuge einer methodischen Vorgehensweise als relevant identifiziert werden. Die Sub-Charakteristika sind die Filterkriterien für den Katalog. Auf der obersten Ebene des Softgoal-Baums steht das Softgoal "Quality", welchem die ISO-Charakteristika entsprechend ihrer Struktur untergeordnet sind. Abbildung 3 zeigt ein Beispiel einer solchen Baumstruktur. Basierend auf dem finalen Softgoal-Baum können NFAs aus dem dadurch gefilterten Katalog ausgewählt werden. Die Modellierung anhand des Softgoal-Baums erlaubt es, die Einflüsse der aus

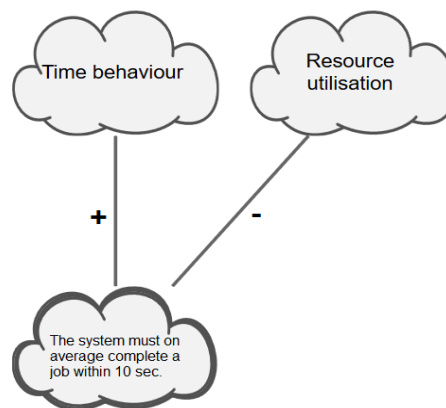


Abb. 4: Einflüsse einer NFA auf die Softgoals

dem Katalog gewählten NFAs auf die Softgoals darzustellen. Somit können potenziell negative Einflüsse zwischen Elementen des Softgoal-Baums modelliert werden, falls die Erfüllung eines ISO Sub-Charakteristikums ein anderes negativ beeinflusst. Dies ist in Abbildung 4 beispielhaft dargestellt, welche zwei Softgoals auf der untersten Ebene der Baumstruktur mit einer aus dem Katalog ausgewählten NFA zeigt. Diese gegenseitigen Einflüsse werden aktuell manuell modelliert, geplant ist allerdings diese zukünftig im NFA-Katalog zu hinterlegen.

3.2 BPMN-NFA Methode

Nachfolgend wird die BPMN-NFA Methode zur Erhebung nicht-funktionaler Anforderungen unter Verwendung von Prozessmodellen vorgestellt. Geläufig wird zum Modellieren von Geschäftsprozessen die „Business Process Model and Notation“ (BPMN) der Object Management Group (OMG) verwendet [OM11]. Unter anderem die REMO Methode von Vieira et al. benutzt BPMN-Diagramme als Input zur Anforderungserhebung [Vi12b]. NFAs werden in REMO, mithilfe von Brainstorming, basierend auf Aktivitäten und Message-Events erhoben [Vi12a]. Die hier vorgestellte Methode benutzt hingegen im

BPMN-Diagramm vorhandene Elemente und Muster zum Aufbau des Softgoal-Baums und für die anschließende Filterung des NFA-Katalogs. Abbildung 5 stellt den Ablauf der Methode dar.

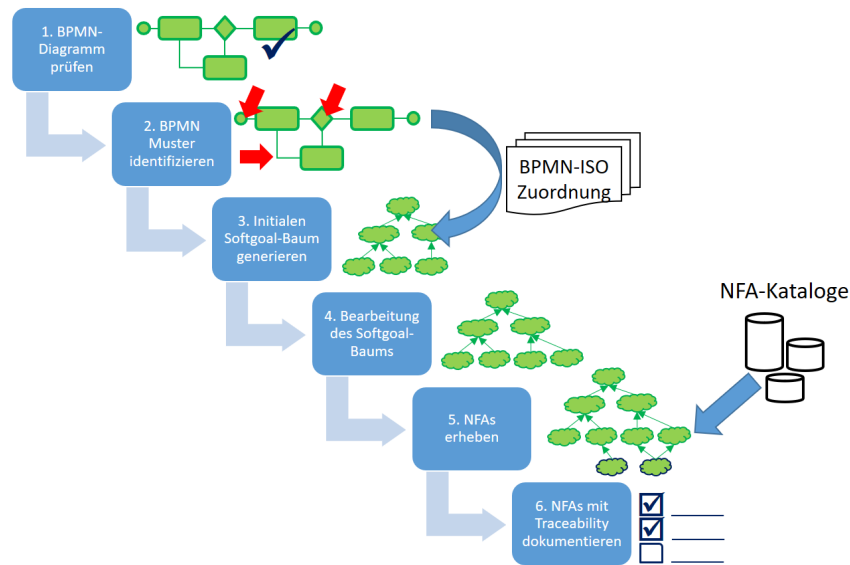


Abb. 5: Vorgehensweise der BPMN-NFA Methode

Zusammengefasst besteht die Methode aus den sechs aufeinanderfolgenden Arbeitsschritten:

1. Prüfung des BPMN-Diagramms
2. Identifikation vorhandener BPMN-Elemente/-Muster
3. Aufstellen des initialen Softgoal-Baums
4. Bearbeitung des Softgoal-Baums
5. Formulierung der NFAs
6. Dokumentation der NFAs

Grundvoraussetzung ist ein inhaltlich korrektes sowie aktuelles BPMN-Diagramm, was im ersten Arbeitsschritt überprüft wird. Daraufhin werden im Diagramm vordefinierte BPMN-Muster und -Elemente identifiziert und mit Stereotypen markiert. Jedem Muster/Element sind ausgewählte ISO Sub-Charakteristika zugeordnet. Diese allgemeine Zuordnung ist ein Grundbestandteil der BPMN-NFA Methode und für die erfolgreiche Durchführung der NFA-Erhebung von besonderer Bedeutung. Entsprechend wird die Zuordnung von BPMN zu ISO im nachfolgenden Unterkapitel detailliert erläutert. Mithilfe der fest definierten

Zuordnung wird automatisiert ein initialer Softgoal-Baum generiert. Bei der Verwendung einer Werkzeugumsetzung der Methode ist der dritte Arbeitsschritt somit ohne manuellen Aufwand verbunden.

Nach der automatisierten Generierung wird der Softgoal-Baum durch einen Fachexperten unter Nutzung der vorhandenen Kontextinformationen bearbeitet. Hierbei können einzelne Softgoals hinzugefügt oder entfernt werden. Die manuelle Bearbeitung ist Arbeitsschritt 4 der Methode, entsprechend des Ablaufs in Abbildung 5. Die automatische Generierung des initialen Softgoal-Baums begründet sich in allgemeinen Annahmen und ist somit notwendigerweise nicht für jeden Projektkontext vollständig übertragbar. Der manuelle Bearbeitungsschritt ermöglicht eine Umsetzung kontextspezifischer Aspekte. Es werden einerseits im initialen Softgoal-Baum vorhandene Elemente begründet verworfen, falls diese im konkreten Anwendungsfall keine Relevanz besitzen. Andererseits werden manuell Softgoals hinzugefügt, deren Notwendigkeit sich bspw. aus weiterführenden Projektinformationen ergeben kann, welche nicht konkret in BPMN modelliert sind.

Daraufhin wird mithilfe des Softgoal-Baums der NFA-Katalog nach potenziell relevanten NFAs gefiltert. Die finale Auswahl der NFAs aus dem Katalog erfolgt unter Anwendung der vorhandenen Kontextinformationen, also dem System- und Expertenwissen, welches zum Prozess vorliegt. Abschließend müssen die aus dem Katalog gewählten NFAs dokumentiert werden. Aufgrund des modellbasierten Ansatzes liegen hierfür ausführliche Informationen zur Nachverfolgbarkeit vor.

3.3 BPMN Muster

Die Generierung des initialen Softgoal-Baums basiert auf der Identifizierung von vordefinierten Mustern im Input. Jedem spezifizierten Muster sind potentiell relevante NFA-Kategorien zugeordnet. Wird während der Analyse ein bestimmtes Muster im Input gefunden, werden die entsprechenden Kategorien in den Softgoal-Baum eingefügt. Die Workflow Pattern nach

Workflow Muster	Bedeutung in BPMN
Parallel Split	Aufteilung des Sequence Flows in mehrere parallele Flows
Synchronisation	Zusammenführen paralleler Sequence Flows zu einem einzelnen Flow
Exclusive Choice	Auswahl einer Verzweigung im Sequence Flow aus mehreren Optionen
Multiple Choice	Selektive Aufteilung des Sequence Flows in parallele Flows
Discriminator	Zusammenführen mehrerer Sequence Flows, wobei der zuerst abgeschlossene Flow den Prozess weiterführt und nachfolgende ignoriert werden
Arbitrary Cycles	Sich zyklisch wiederholende Aktivitäten mit Verzweigungsmöglichkeiten
Multiple Instances	Mehrfache, gleichzeitige Ausführung einer Aktivität
Deferred Choice	Auswahl einer Verzweigung basierend auf Informationen zur Laufzeit
Cancel Activity	Abbruch einer Aktivität bei Eintritt eines bestimmten Umstands
Cancel Case	Abbruch eines kompletten Prozesses bei Eintritt eines bestimmten Umstands

Tab. 1: Beschreibung der Workflow Pattern

van der Aalst sind eine Sammlung von Mustern, welche in Ablaufdiagrammen vorkommen können [Aa03]. In Tabelle 1 sind die nachfolgend verwendeten Workflow Muster beschrieben. Wohed et al. analysierten, inwieweit die Workflow Pattern in BPMN abbildbar sind [Wo06].

Unter Anbetracht dieser Analyse werden die Muster, sowie die einzelnen BPMN-Elemente, auf ihre Relevanz für die Erhebung von NFAs untersucht. Die dabei erstellte Zuordnung basiert auf den ISO-Messgrößen der Qualitätsmodelle. In den ISO-Normen sind für jedes einzelne Sub-Charakteristika Messgrößen definiert. Zur Generierung einer Zuordnung wird

Workflow Muster	ISO Sub-Charakteristika
Parallel Split	Capacity; Co-existence; Modularity
Synchronisation	Resource utilisation; Interoperability
Exclusive Choice	Maturity; Modularity
Multiple Choice	Capacity; Co-existence; Maturity; Modularity
Discriminator	Resource utilisation; Interoperability; Fault tolerance
Arbitrary Cycles	Resource utilisation; Maturity
Multiple Instances	Time behaviour; Resource utilisation; Capacity
Deferred Choice	Time behaviour; Maturity; Modularity
Cancel Activity	User error protection; Maturity; Recoverability
Cancel Case	Maturity; Recoverability; Accountability; Analysability

Tab. 2: Relevanz von Product Quality Sub-Charakteristika für Workflow Pattern

für alle Workflow Pattern untersucht, ob durch eine Implementierung des Musters theoretisch Einfluss auf die jeweilige Messgröße ausgeübt wird. So konnten für einige Workflow Pattern jeweils zwei bis vier potentiell relevante ISO Sub-Charakteristika identifiziert werden.

Diese Zuordnung wurde in mehreren Anwendungsbeispielen überprüft und verfeinert. Für jeden Anwendungsfall wurde die BPMN-NFA Methode durchgeführt. Falls bei der Anwendung in Arbeitsschritt 4 weitgehende manuelle Änderungen notwendig waren, so war der initial generierte Softgoal-Baum potentiell nicht ausreichend. Die notwendigen Änderungen der verschiedenen Anwendungsfälle wurden zusammengetragen und verglichen, wodurch Rückschlüsse auf die Qualität der initialen Generierung entstanden. Hierdurch konnten notwendige Anpassungen an der Zuordnung identifiziert werden, wodurch diese iterativ verbessert wurde. In Tabelle 2 ist die Zuordnung für das Qualitätsmodell der *Product Quality* aufgeführt.

Außerdem werden bei der Generierung des initialen Softgoal-Baums die im BPMN-Diagramm vorhandenen Elemente betrachtet. zur Muehlen und Recker kommen in ihrer Analyse des BPMN-Gebrauchs zu der Erkenntnis, dass nur 20% der BPMN-Elemente regelmäßig in Diagrammen verwendet werden [zMR08]. Entsprechend wird für eine Auswahl relevanter BPMN-Elemente ebenfalls eine Zuordnung zu den ISO Sub-Charakteristika vorgenommen. Dabei wird das gleiche Vorgehen wie bei den Workflow Pattern verwendet. Die Relevanz eines Sub-Charakteristika ist mithilfe der ISO-Messgrößen abgeschätzt worden und jedem BPMN-Element wurden zwei bis fünf Sub-Charakteristika zugeordnet. Die

BPMN-Element	ISO Sub-Charakteristika
Start-Event	Capacity; Availability; Accountability; Authenticity
Error Event Definition	Fault tolerance; Recoverability; Integrity; Accountability; Analysability
Message Event Definition & Receive Task	Capacity; Interoperability; Availability; Confidentiality
User Task	Time behaviour; Appropriateness recognizability; Learnability; Operability; User error protection; User interface aesthetics; Accessibility
Service Task, Script Task & Business Rule Task	Time behaviour; Resource utilisation; Maturity; Availability
Sub Process	Interoperability; Availability; Integrity; Modularity
Call Activity	Time behaviour; Co-existence; Interoperability; Availability; Modularity
Message Flow	Resource utilisation; Interoperability; Confidentiality; Integrity
Data Object	Resource utilisation; Confidentiality; Integrity
Data Store	Resource utilisation; Interoperability; Availability; Confidentiality; Integrity
Lane (Akteure)	Capacity; Co-existence; Confidentiality; Authenticity
Sequence Flow mit Lane Wechsel	Interoperability; Availability; Confidentiality

Tab. 3: Relevanz von Product Quality Sub-Charakteristika für BPMN-Elemente

Ausnahme bildet hierbei das Element *User Task*, welchem aufgrund der unterschiedlichen Usability-Aspekte insgesamt sieben Sub-Charakteristika zugeordnet sind. Die Zuordnung wurde ebenfalls anhand der Anwendungsbeispiele überprüft und ist in Tabelle 3 aufgelistet.

3.4 Werkzeugunterstützung

Die BPMN-NFA Methode lässt sich grundsätzlich mit gängigen Modellierungswerkzeugen umsetzen. Viele BPMN- und UML-basierte Werkzeuge bieten die Möglichkeit zur Erweiterung der Modellierungskonzepte. So lassen sich Softgoal-Bäume bspw. auch mithilfe eines UML-Profiles darstellen. Diese Nutzung eines UML-Profiles ist allerdings im Anwendungskontext als weniger intuitiv empfunden worden als ein spezialisiertes Werkzeug.

Wir haben daher die beschriebene BPMN-NFA Methode in einer Werkzeugkette basierend auf den Bestandteilen ModelBus [Fra19] und ModelICE [Ing19] realisiert. Die Werkzeugkette ermöglicht eine Bearbeitung sämtlicher Modelle im Webbrowser und über Adaptoren sind gängige Modellierungswerkzeuge und -bibliotheken (wie bspw. Enterprise Architect, Eclipse Modelling, Papyrus, Diagram.js) integriert und anwendbar. Im Zuge des modellbasierten Vorgehens sind Informationen zur Nachverfolgbarkeit (Traceability) besonders zu berücksichtigen. Eine Nachverfolgbarkeit muss über Modellgrenzen hin-

weg erfolgen, weil Abhängigkeiten zwischen den Prozessbeschreibungen (in BPMN), den Architekturbeschreibungen (vorwiegend in SysML) und den Softgoal-Bäumen bestehen.

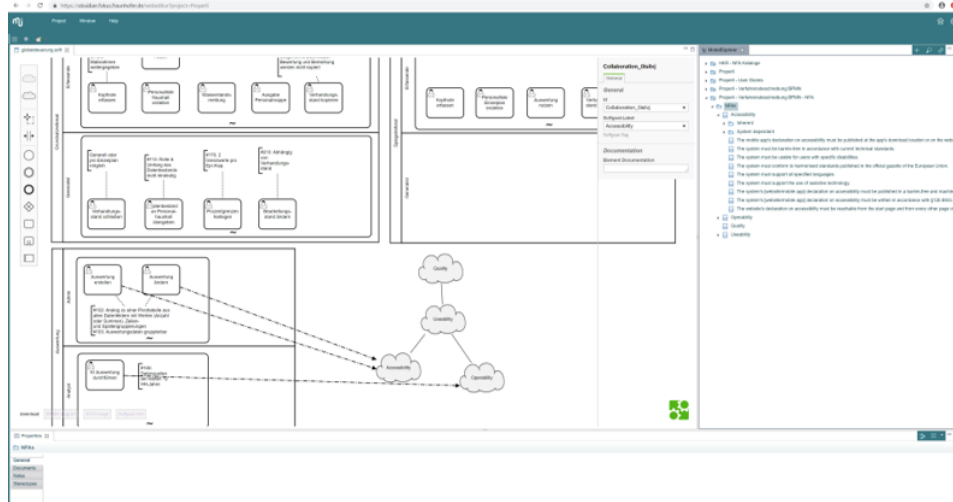


Abb. 6: Bearbeitung von BPMN und Softgoals mit ModelICE

Für die Umsetzung der Methode haben wir die Beschreibung der Prozesse und deren Beziehung zu Softgoal-Bäumen mittels einer Erweiterung von bpmn-js realisiert und in das Werkzeug ModelICE integriert. In Abbildung 6 ist die Umsetzung dargestellt. Zusätzlich zu der Darstellung der Softgoal-Bäume haben wir eine Erweiterung für ModelICE entwickelt, welche den NFA-Katalog mithilfe der Softgoals nach konkreten NFAs filtert. Die Verwendung von ModelICE erlaubt es, diese Funktionalität als Dienst in der Werkzeugkette zur Verfügung zu stellen und so bspw. im Enterprise Architect zu nutzen.

4 Anwendung der BPMN-NFA Methode

Die entwickelte BPMN-NFA Methode wird im Kontext der Modernisierung der Haushaltsverfahren des Bundes zur Erhebung von NFAs eingesetzt. Hierfür wird die beschriebene Werkzeugkette genutzt, um die Erkennung vorhandener BPMN-Elemente und -Muster im Diagramm zu unterstützen und teilautomatisiert durchzuführen. Nachfolgend ist die Anwendung der BPMN-NFA Methode auf den Beispielfall ProPerLi beschrieben. Es werden BPMN-Modelle des zu implementierenden Prozesses und die im vorherigen Kapitel beschriebene Zuordnung von BPMN zu den *Product Quality* Sub-Charakteristika genutzt.

4.1 ProPerLi Anwendungsfall

Mit ProPerLi (Produktion Personallisten) soll ein Softwaresystem zur Unterstützung der Aufstellung des Personalhaushalts des Bundes (Planstellen der Beamtinnen und Beamten, Stellen der Tarifbeschäftigten) geschaffen werden. Ziel ist es, ein durchgehendes, IT-gestütztes Verfahren zu entwickeln, dass den gesamten Prozess umfassen soll. Es sind die Prozessabläufe von der Erfassung aller Personalforderungen über die Beurteilung dieser Forderungen, die Darstellung von Verhandlungsergebnissen innerhalb der Bundesregierung und von Änderungen aus dem parlamentarischen Verfahren bis zur Produktion der Personallisten und zur Übergabe des Datenbestandes in eine bestehende Datenbank umzusetzen. Der Prozess wird aktuell manuell, unterstützt von Office-Software, durchgeführt und ist von Medienbrüchen geprägt und äußerst prüfungsintensiv, weil keine automatisierte Fehler- und Plausibilitätsprüfung existiert. Die Abläufe des Verfahrens liegen strukturiert als BPMN-Diagramme vor.

Mithilfe der modellierten BPMN-Diagramme werden die NFAs für ProPerLi erhoben. Die einzelnen Ablaufschritte der BPMN-NFA Methode sind in Abbildung 5 dargelegt. Die Erhebung der NFAs erfolgt durch Nutzung der NFA-Kataloge. Insgesamt wurden mithilfe der BPMN-NFA Methode 29 prozessbezogene NFAs aus dem *Product Quality* Katalog für ProPerLi erhoben, welche von einem Prozessexperten als korrekt eingestuft wurden. Dafür wurde ein Softgoal-Baum mit insgesamt 18 Sub-Charakteristika genutzt. Einige NFAs sind beispielhaft in Tabelle 4 aufgelistet.

ID	Sub-Charakteristika	NFA
1	Capacity	The system must allow at least 5 users to use the system simultaneously.
14	Accountability	The system must audit all user access to the system or data.
18	Availability	The database must be available for 95% of the scheduled system operational time.
22	Operability	The system must support all appropriate input devices.
29	Analyseability	The system must implement all specified diagnosis functions.

Tab. 4: Beispiel-NFAs für ProPerLi aus dem *Product Quality* Katalog

4.2 Bewertung der Methode

Die vorgestellte BPMN-NFA Methode und allgemein die Verwendung eines NFA-Katalogs zur Anforderungserhebung werden nachfolgend bewertet. Als Vergleich dienen die Ergebnisse eines Vorläuferprojekts von ProPerLi, in welchem NFAs für den Prozess der Personallisten durch eine externe Beratergesellschaft erhoben wurden. Hierbei erstellte die Beratergesellschaft eine NFA-Sammlung basierend auf bestehenden, unstrukturiert erhobenen Originalanforderungen im Kassen- und Rechnungswesen des Bundes. Diese

NFA-Sammlung war für die Anwendung im Kontext der Modernisierung der Haushaltsverfahren des Bundes vorbereitet und nach den Kategorien der ISO-Norm 9126 strukturiert. Die NFA-Sammlung der Beratergesellschaft wird nachfolgend als Vergleichs-NFAs bezeichnet.

In dieser Arbeit wurde ein strukturiert aufgebauter NFA-Katalog präsentiert, basierend auf den Messgrößen der ISO-Normenreihe 250xx. Der Katalog schließt inhaltliche Lücken, welche bei den Vergleichs-NFAs bestanden. Weiterhin erfolgte die Erstellung des NFA-Katalogs mithilfe eines geringeren Ressourcenaufwands als bei den Vergleichs-NFAs. Somit erfüllt der NFA-Katalog die Anforderungen, welche seitens der Prozessbeteiligten an eine NFA-Sammlung gestellt und durch die externe Beratergesellschaft nicht ausreichend abgedeckt wurden.

Die strukturierte Anwendung, mittels Softgoal-Modellierung basierend auf BPMN, ermöglichte im ProPerLi-Prozess eine für alle Stakeholder nachvollziehbare Erhebung von NFAs. Durch die Strukturierung des Vorgehens ist eine Abdeckung des Problemraums, konkret der Prozessbeschreibungen, sichergestellt. Außerdem ermöglicht die Implementierung der Methode eine Teilautomatisierung der Anwendung. Neben der erweiterten Abdeckung des Problemraums stellt die Methode mithilfe der BPMN-Modellierung ein Verständnis der Prozessabläufe bei allen Stakeholdern sicher.

Schlussendlich soll die vorgestellte Methode quantitativ bewertet werden. Die Vergleichs-NFAs umfassen 19 als allgemeingültig definierte NFAs. Ausgehend von der Kategorisierung nach ISO-Norm 9126 bietet sich ein Vergleich mit dem *Product Quality* Unterkatalog an. Dieser beinhaltet insgesamt 91 generische NFAs auf einer detaillierteren Abstraktionsstufe, wovon mithilfe der BPMN-NFA Methode 29 für den ProPerLi-Prozess erhoben wurden. Im Zuge eines inhaltlichen Vergleichs ist erkennbar, dass die Anforderungen des *Product Quality*-Katalogs konkreter sind, als die Vergleichs-NFAs. Insgesamt 17 der 19 Vergleichs-NFAs werden durch die 29 erhobenen ProPerLi-NFAs abgedeckt und durch diese detaillierter beschrieben. Die beiden nicht abgedeckten Vergleichs-NFAs behandeln die Anonymisierung von Testdaten und das Wartungskonzept. Diese beiden Aspekte werden von der Prozesssicht nicht abgedeckt, entsprechend könnte die Erhebung um bspw. eine Architekturanalyse erweitert werden, um den Systemkontext auf weitere fehlende NFAs zu überprüfen. Ein Rückvergleich unter Betrachtung der Kategorien der ISO-Norm 9126 ergibt, dass sieben BPMN-NFAs in den Vergleichs-NFAs überhaupt nicht und fünf weitere nicht ausreichend abgebildet sind. Somit sind 24% der ermittelten BPMN-NFAs komplett neu für den Projektkontext erhoben und insgesamt 41% stellen eine Erweiterung der vormals vorhandenen NFAs dar.

5 Zusammenfassung

Mit Hilfe des erweiterbaren Katalogs der Qualitätskriterien und dem Vorgehen zum Aufbau des Softgoal-Baums lassen sich für Architektur- und Prozessmodelle konkrete NFAs ableiten. In diesem Beitrag wurde die Anwendung der Methode für Prozessmodelle

dargestellt und untersucht. Aufgrund des systematischen Vorgehens wird ein umfassender und zugleich passgenauer Satz von NFAs erstellt. Das von uns implementierte webbasierte Werkzeug zu Erstellung von BPMN und Softgoal-Modellen ermöglicht ein interaktives teamübergreifendes Arbeiten, was insbesondere in den Schritten der Überarbeitung der Modelle mithilfe von Experten sehr hilfreich ist.

In der bisherigen Anwendung der Methode konnten vielversprechende Ergebnisse erreicht werden. Insbesondere, wenn mit der Erhebung von nicht-funktionalen Anforderungen komplett neu begonnen wird, lässt sich relativ leichtgewichtig ein umfassender Entwurf von NFAs für die verschiedenen Teile eines Systems erzeugen, um so frühzeitig in die Diskussion mit den Stakeholdern zu kommen. Durch die strukturierte Zuordnung der erzeugten NFAs zu den im Katalog benutzten ISO-Charakteristiken erleichtert sich auch die weitere Verarbeitung der Anforderungen in Ausschreibungen bzw. dann in der technischen Umsetzung.

5.1 Related Work

Die Wiederverwendung bekannter Anforderungen in neuen Softwareprojekten kann die Effizienz der Anforderungserhebung erhöhen. Toval et al. schlagen hierfür die SIREN Methode vor. SIREN verwendet ein spiralförmiges Vorgehen, bei dem Anforderungskataloge genutzt werden, bspw. ein Katalog für Security Profile [To08]. Das NFR Framework stellt Kataloge in Form von Softgoal Interdependency Graphen (SIG) zur Verfügung [CL09]. Cysneiros liefert weitere Kataloge, ebenfalls in SIG abgebildet, und das i*-System zur systematischen Nutzung von Katalogen [Cy07]. In der Literatur genannt sind unter anderem SIG für Usability [CWK05], Performance [Yu03] und Invisibility [Ma18]. Der in diesem Artikel beschriebene NFA-Katalog zeichnet sich durch eine Strukturierung basierend auf ISO-Normen und eine unkomplizierte Erweiterbarkeit aus.

Eine strukturierte Erhebung von NFAs ermöglicht bspw. die SENoR Methode, welche NFA-Vorlagen basierend auf den Charakteristika der ISO-Norm 25010 nutzt. Dieser Katalog an Vorlagen wird während der Methode in Brainstorming-Sitzungen verwendet, um die Stakeholder bei der NFA-Erhebung zu unterstützen [KN14]. Das PABRE-System liefert eine Sammlung an Anforderungsmustern für einige der ISO-Norm 25010 Charakteristika [PQF14]. Die BPMN-NFA Methode erhebt strukturiert NFAs unter Verwendung des vorgestellten NFA-Katalogs. Ein gleichsam strukturiertes Vorgehen ist auch für Architekturmodelle möglich. Mithilfe teilautomatisierter Arbeitsschritte wird manueller Aufwand abgenommen. Diese Automatisierung kann im Rahmen der Werkzeugunterstützung noch erweitert werden.

5.2 Ausblick

Die ersten Ergebnissen der bisherigen Anwendungen der Methode im Zuge der Modernisierung der Haushaltsverfahren und speziell im Submodul ProPerLi (Produktion Personallisten) liegen inzwischen vor. Die dabei gewonnenen Erkenntnisse werden zur Erweiterung des Katalogs und zur Verbesserung der Methode genutzt. Aktuell werden die Muster teilweise manuell markiert, daher ist die weitgehende Automatisierung der Mustererkennung durch verbesserte Werkzeugunterstützung unser nächster Schritt. Außerdem soll die Methode zukünftig in unterschiedlichen Anwendungsbereichen eingesetzt werden, wodurch die aktuell benutzten Muster überprüft und weiter verfeinert werden können. Ergänzend zur Anwendung im Submodul ProPerLi wird die Methode zukünftig auch bei der Modernisierung weiterer Haushaltsverfahren des Bundes eingesetzt.

Literaturverzeichnis

- [Aa03] van der Aalst, W.M.P.; ter Hofstede, A.H.M.; Kiepuszewski, B.; Barros, A.P.: Workflow Patterns. Distributed and Parallel Databases, S. 5–51, 2003.
- [BIT11] Barrierefreie-Informationstechnik-Verordnung - BITV 2.0, 09 2011.
- [CL09] Chung, L.; Leite, Julio: On Non-Functional Requirements in Software Engineering. S. 363–379, 01 2009.
- [CWK05] Cysneiros, Luiz; Werneck, Vera; Kushniruk, Andre: Reusable knowledge for satisficing usability requirements. S. 463–464, 01 2005.
- [Cy07] Cysneiros, Luiz Marcio: Evaluating the Effectiveness of Using Catalogues to Elicit Non-Functional Requirements. 2007.
- [Fra19] Fraunhofer FOKUS: ModelBus, <https://www.modelbus.org>, 2019. Online, zuletzt aufgerufen: 13.10.2019.
- [Ing19] Ingrano Solutions: ModelICE, <https://www.modelice.io>, 2019. Online, zuletzt aufgerufen: 13.10.2019.
- [IS08] ISO: ISO/IEC 25012:2008 Data quality model. International Organization for Standardization, 2008.
- [IS11] ISO: ISO/IEC 25010:2011 System and software quality models. International Organization for Standardization, 2011.
- [IS14] ISO: ISO/IEC 25000:2014 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Guide to SQuaRE. International Organization for Standardization, 2014.
- [IS15] ISO: ISO/IEC 25024:2015 Measurement of data quality. International Organization for Standardization, 2015.
- [IS16a] ISO: ISO/IEC 25022:2016 Measurement of quality in use. International Organization for Standardization, 2016.

- [IS16b] ISO: ISO/IEC 25023:2016 Measurement of system and software product quality. International Organization for Standardization, 2016.
- [KN14] Kopczynska, Sylwia; Nawrocki, Jerzy R.: Using non-functional requirements templates for elicitation: A case study. 2014 IEEE 4th International Workshop on Requirements Patterns (RePa), S. 47–54, 2014.
- [Ma18] Maia, Rainara; Andrade, Rossana; Oliveira, Káthia; Kolski, Christophe: Catalog of Invisibility Requirements for UbiComp and IoT Applications. 08 2018.
- [OM11] OMG: Business Process Model and Notation Specification Version 2.0. Object Management Group, 2011.
- [PQF14] Palomares, Cristina; Quer, Carme; Franch, Xavier: Requirements Reuse with the PABRE Framework. Requirements Engineering Magazine, 01 2014.
- [To08] Toval, Ambrosio; Moros Valle, Begoña; Nicolás, Joaquín; Lasheras, Joaquín: Eight key issues for an effective reuse-based requirements process. Computer Systems Science and Engineering, 23:373–385, 11 2008.
- [Vi12a] Vieira, Sergio; Viana, Davi; Nascimento, Rogerio; Conte, Tayana: Evaluating a technique for requirements extraction from business process diagrams through empirical studies. S. 1–10, 10 2012.
- [Vi12b] Vieira, Sergio; Viana, Davi; Nascimento, Rogerio; Conte, Tayana: Using Empirical Studies to evaluate the REMO Requirement Elicitation Technique. 01 2012.
- [Wo06] Wohed, Petia; Aalst, Wil M. P.; Dumas, Marlon; Ter, Arthur; Russell, Nick: On the Suitability of BPMN for Business Process Modelling. 10 2006.
- [Yu03] Yu, Yijun; Mylopoulos, John; Yu, Eric; Leite, Julio; Linda, Leite; Liu, Lin: Software Refactoring Guided By Multiple Soft-Goals. 12 2003.
- [zMR08] zur Muehlen, Michael; Recker, Jan: How Much Language Is Enough? Theoretical and Practical Use of the Business Process Modeling Notation. 03 2008.