

Spezifikation von Benutzeroberflächen

Ausgangssituation, Bedarf, Umfang, Sprachmittel und Praxiseinführung der formalen Oberflächenspezifikation für dialogintensive Anwendungen.

Wert und Kosten von Benutzeroberflächen

Trotz zahlreicher technologischer Verbesserungen und einschneidender Paradigmenwechsel¹ in der Geschichte der Softwareentwicklung, sind die Aufgaben von Benutzeroberflächen unverändert geblieben: Eine Mensch-Maschine-Schnittstelle (MMS/HCI) muss einerseits die Eingaben des Benutzers interpretieren und auf Basis der Interpretationsergebnisse Anwendungsfunktionen auslösen; andererseits muss sie die Daten und Signale der Anwendung interpretieren und auf Basis dieser Interpretationsergebnisse menschenwahrnehmbare Inhalte (Grafik, Sprache, Haptik) ausgeben.

Die Wechselwirkungen von Funktion und Bedienung beeinflussen die Entwicklungskosten einer Oberfläche beträchtlich, ebenso wie der Aufwand zur Beherrschung dieser Wechselwirkungen und zur Abstimmung der Anforderungen. Die Kosten hängen zudem davon ab, welcher Aufwand zur Umsetzung der Anforderungen aufgewandt werden muss und wieviele Änderungsanforderungen im Laufe des Entwicklungsprozesses hinzukommen.

Weil Benutzeroberflächen durch ihre Entwicklungskosten teuer sind, aber auch weil sie lange reifen und das Look & Feel des Softwareprodukts maßgeblich bestimmen, ist ihre Wiederverwendung naheliegend und aus ökonomischer Sicht sogar notwendig.

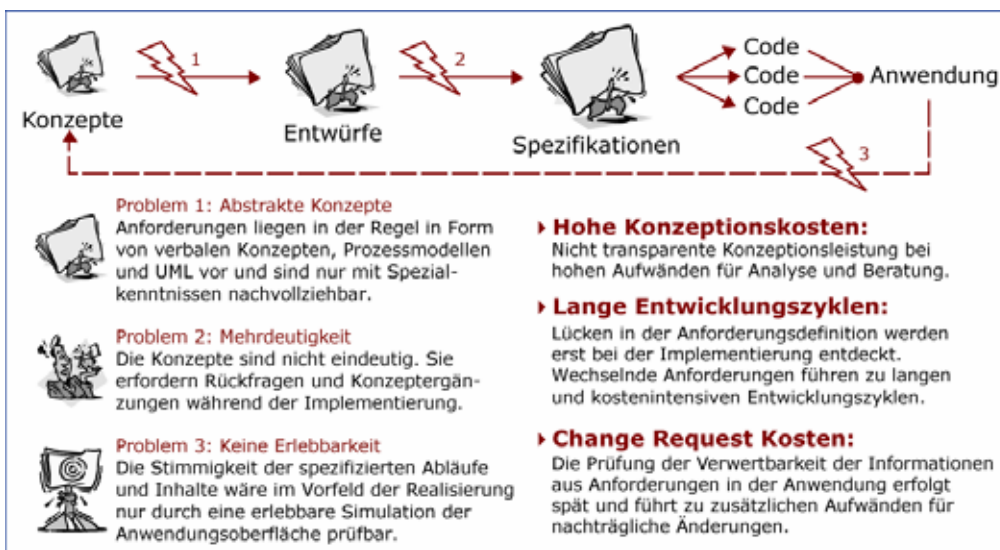


Abbildung 1: Klassisches Vorgehen bei der Entwicklung dialogintensiver Anwendungen

Dieses Papier zeigt Lösungsvorschläge auf, wie die Methoden der Benutzeroberflächenspezifikation verbessert werden können, um Konzeptionskosten und Change Request Kosten zu senken und Entwicklungszyklen zu verkürzen.

Die folgenden Ausführungen beleuchten hierzu, inwieweit Oberflächenspezifikationen (vertraglich und methodisch) notwendig sind, welchen inhaltlichen Rahmen sie haben und stellt Lösungsansätze zur Klassifizierung, Strukturierung und Formulierungen von Spezifikationsinhalten.

¹ Paul Chlebek: Benutzeroberflächen – ein Stiefkind der Softwareindustrie? HCI Entwicklungsgeschichte und Ausblick.

Der Überblick über den notwendigen Umfang von Oberflächenspezifikationen wird abgerundet, indem die Möglichkeiten der Unterstützung durch Methoden, Werkzeuge und Zuschnitt auf gängige Vorgehensmodelle skizziert werden. Hierzu gehört auch eine Kurzdarstellung weiterführender Themen, Chancen, Risiken und Auswirkungen des Einsatzes von Informationsmodellen und Sprachen zur Oberflächenspezifikation.

Wozu Oberflächenspezifikationen?

Muss man eine Benutzeroberfläche überhaupt spezifizieren? Reicht es nicht, die Oberfläche agil-iterativ zu implementieren und damit das lästige Erstellen der Spezifikation einzusparen?

In kommerziellen Projekten sind Oberflächenspezifikationen oft unumgänglich, weil sie eine Grundlage für den Test und die Abnahme der beauftragten Anwendung sind. Die Beschreibung der Funktionen und Daten einer Anwendung seitens des Auftraggebers erfolgt meist durch Anwendervertreter, die das Ganze aus Sicht der Benutzeroberfläche sehen. Dabei beeinflussen die geforderten Oberflächeneigenschaften Funktionen und Daten. Es ist nicht unüblich, dass auf Basis von Oberflächen die Abstimmung zwischen den an der Anwendungserstellung beteiligten Rollen erfolgt und die beschriebenen Eigenschaften der Oberflächen zum Vertragsbestandteil werden. Somit wird der Entwurf der Benutzeroberfläche zum Dreh- und Angelpunkt bei den Gesprächen über die beauftragte Anwendung.

Die Randbedingungen, Anforderungen, Bezugnahmen und Wechselwirkungen, die in diesem Abstimmungsprozess festgelegt werden, gehen über das in einer Oberfläche oder ihrem Prototyp Sicht- und Erlebbares hinaus. Diese Abstimmungsergebnisse bestimmen maßgeblich den Entwurf der Benutzeroberfläche. Da diese Informationen über das im Code [etwa eines Prototypen] Formulierbare hinausgehen, ist damit ein weiterer zwingender Grund für Oberflächenspezifikationen gegeben.

Der Erfolg der Anwendung ist demnach auch abhängig von eindeutigen und erlebbaren Spezifikationensinhalten zu den Oberflächen, der Robustheit der Spezifikation und der Implementierung bei wechselnden Anforderungen und der Qualität der technischen Umsetzung der Anforderungen.

In der Praxis kommerzieller Projekte sind Oberflächenspezifikationen aus Sicht der Autoren also unverzichtbar.

Informationsstrukturen für Oberflächenspezifikationen

Was genau soll eine User Interface Spezifikation enthalten und mit welchen Formulierungen können diese Inhalte ausgedrückt werden?

Da Oberflächenspezifikationen nachweisbar notwendig sind, lohnt es sich zu klären, welche **Gesamtmenge von Informationen** zum Bau einer Oberfläche benötigt wird und was dem zufolge der informationsstrukturelle Rahmen einer UI-Spezifikation ist.

Die für eine informationelle Bestimmung einer Oberflächenspezifikation zentralen Fragestellungen sind:

- Welchen Informationsgehalt muss die Spezifikation haben, um eine Oberfläche in allen Entwicklungsphasen hinreichend zu beschreiben?
- Welche Informationsstrukturen und Formulierungsmöglichkeiten braucht man hierzu?

Um die Qualität und die Kosten von Oberflächen (mittels geeigneter Spezifikationen) optimieren zu können, benötigen wir geeignete Ausdrucksmittel zur eindeutigen Beschreibung von Inhalten und dem Verhalten der Oberfläche, der Abhängigkeiten und Wechselwirkungen Kontext der Anwendungsfunktionen, der Darstellungsformen und der zugrunde liegenden Anforderungen.

Durch eine maschinelle Umformung (Umwandlung) der UI-Spezifikation zu Netzgraphen, Aufstellungen über Daten- und Funktionsanforderungen, Mengengerüste, sowie Darstellungs- und Ablaufsimulationen kann der Abstimmungsgrad zwischen den Prozessbeteiligten frühzeitig erhöht werden. Diese Abstimmung kann zudem durch Werkzeuge unterstützt werden, welche die Spezifikationsinhalte maschinell in die Implementierungsplattform übernehmen, beispielsweise in Form von Codegenerierung aus den formalen Anteilen der UI-Spezifikation. Ergänzende Werkzeuge können die maschinelle Erzeugung von Onlinehilfe- und Handbuchinhalten (Gerüstbau) aus den nichtformalen Anteilen der Spezifikation bewerkstelligen.

Anhand der am Anwendungsbau beteiligten Rollen und deren Sichten (Perspektiven) auf eine UI-Spezifikation lassen sich Rückschlüsse auf die Gliederung der Spezifikationsinhalte (Informationsstrukturen) und die benötigten Formulierungsmittel (Sprachkonstrukte) treffen.

Durch detaillierte Analyse der Anwendungsfälle von Oberflächen lassen sich Informationsstrukturen und Konstrukte ableiten, mit denen sich Eigenschaften der Oberfläche und ihr Zusammenspiel beschreiben lassen.

Zunächst stellen sich aus Sicht des UI-Spezifikateurs folgende Leitfragen:

- Was beinhalten die Abläufe und Inhalte der Anwendung aus fachlicher Sicht?
- Wie hängen die Aktivitäten und ihre Inhalte strukturell in der Anwendung zusammen?
- Wie wird eine konsistente Darstellung der Anwendungsinhalte erreicht?
- Wie wird der Anwender durch den Arbeitsfluss und durch die Arbeitsschritte geführt?
- Wie werden verschiedene Sichten auf den gleichen Anwendungsinhalt abgebildet?
- Nach welchen Regeln erfolgt die Steuerung der Anwendungsabläufe?

Aus den Leitfragen ergibt sich eine fachliche Gliederung der Oberflächenspezifikation in folgende Perspektiven:

- Anforderungsperspektive
- Ablaufstrukturperspektive
- Darstellungsperspektive
- Kontextperspektive
- Interaktionsperspektive

Die Perspektiven einer Oberflächenspezifikation sind miteinander verflochten und voneinander abhängig.

- Die **Anforderungen** bilden die Grundlage für logische Schritte, Dialogelemente und Kontextelemente.
- Die **Struktur** der logischen Schritte folgt den Anforderungen und benutzt die Darstellungsperspektive und den Kontext.

- Die **IO-Elemente** realisieren die Kommunikation (Darstellung, Eingaben) der logischen Schritte.
- Die **Kontextelemente** werden von logischen Schritten und den Dialogelementen herangezogen. Sie folgen den Anforderungen.
- Die **Interaktionen** beschreiben Ereignisse und Reaktionen mit Auswirkung auf Ablaufstruktur und Dialogelemente unter Einbeziehung der Kontextelemente.

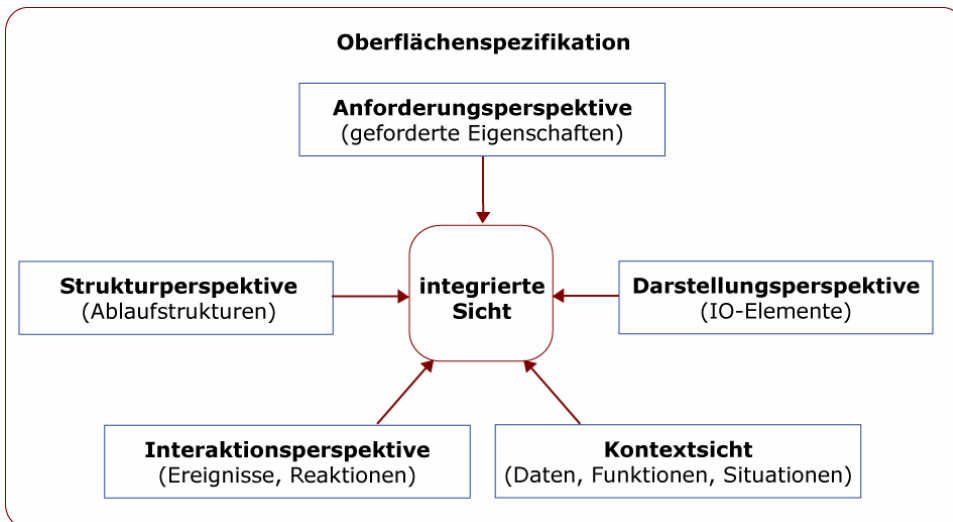


Abbildung 2: Inhaltliche Perspektiven einer Oberflächenspezifikation

Die Strukturen, Ausdrücke und Eigenschaftsfestlegungen der Oberflächenspezifikation müssen in einer **von Menschen leicht verständlichen Form** das Beziehungsgeflecht zwischen Ablaufschritten, Bildschirmhalten, Ein- und Ausgabeformaten, Darstellungselementen, Eingaben, Signalen, Interaktionen, Funktionen, Abfragen, Situationen, Ereignissen und Empfängern, Konfigurationen, Varianzen, Invarianzen und zugrunde liegende Anforderungen abbilden können.

Für den funktionalen und technischen Bereich der Anwendungsentwicklung hat mit der UML (unified modelling language) bereits eine Standardisierung der Ausdrucksmittel stattgefunden. Für den Bereich der UI-Spezifikation besteht lediglich weitgehender Konsens darüber, dass hierfür State-Charts (Harel 1984) in der von D. Harel vorgeschlagenen Form die Grundlage bilden. Die State-Charts sind inzwischen Teil der UML 2.0, damit gibt es in der UML einen Diagrammtyp, mit dem sich Spezifikationen von UI-Abläufen (theoretisch) beschreiben lassen.

Die Komplexität und die IO-Vielfalt zeitgemäßer Oberflächen führt jedoch schnell zu unübersichtlichen und chaotischen Diagrammen. Die State-Charts sind darüber hinaus vor allem zur Beschreibung von Ablauflogik erdacht und eingeführt worden und lassen viel von dem in den Spezifikationsperspektiven umrissenen Beziehungsgeflecht unberücksichtigt. Weiterhin genügen State-Charts nicht dem Kriterium, von Personen überblickt und verstanden werden zu können, die nicht in State-Machines ausgebildet sind und im Lesen solcher Diagramme geübt sind.

Umfang einer UI-Spezifikationssprache

Wir schlagen einen Ansatz vor, der die Ideen und Ansätze der State-Charts aufgreift und weiterführt, sich jedoch bei der Wahl der Notation an der natürlichen Sprache orientiert. State-Charts werden hierzu um oberflächenspezifi-

sche Strukturierungs- und Formulierungsmittel zur Abbildung der oben genannten Beziehungsgeflechte erweitert.

Die Grundkonstrukte einer so hergeleiteten HCI-Spezifikationssprache sind:

- Ablaufschritte
- Screens und andere Dialogkanäle (mit Dialogelementen und Dialogelementgruppen)
- Kontextressourcen (Situationen, Funktionen, Daten, Darstellungselemente)
- Interaktionen (Auslöser, Bedingungen, Reaktionen)
- Anforderungen

Aus den oben vorgestellten Perspektiven ergibt sich, dass eine Sprache zur Oberflächenspezifikation das folgende Beziehungsgeflecht abbilden muss.

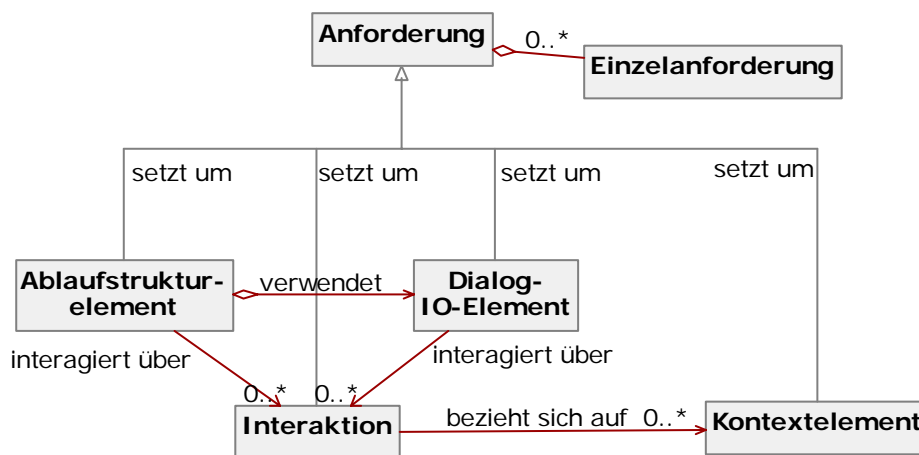


Abbildung 3: Informationsmodell für eine Oberflächenspezifikation

Die Detaillierung der hier skizzierten Informationsstruktur ergibt den Symbol- und Regelvorrat zur integrierten Beschreibung aller Informationen, die zum Bau einer Benutzeroberfläche benötigt werden.

Auf der ontologischen Basis dieses Symbolvorrats können konkrete Grammatiken definiert werden. Hervorzuheben ist dabei, dass die obige Informationsstruktur die abstrakte Struktur der Sprache definiert. Konkrete Notationen sind Erscheinungsformen zur Formulierung des oben skizzierten Informationsgehalts. Sie können beliebig variieren und dennoch auf die gleiche Informationsstruktur abbildbar sein.

Bei der Suche nach einer für Oberflächenspezifikationen geeigneten Notation ist daher zum einen die Abbildbarkeit auf die Informationsstruktur, zum anderen die Lesbarkeit und Flexibilität der konkreten Notation entscheidend.

Das Resultat der Notationssuche des Autors ist eine an umgangssprachliches Englisch angelehnte Notation, die sich in erweiterte State-Charts auf Basis der oben skizzierten Informationsstruktur zurück übersetzen lässt. Wir knüpfen damit einerseits an die in der Industrie etablierte formale Grundlage der State-Charts an, räumen jedoch durch die umgangssprachliche Orientierung einem erweiterten Personenkreis die Mitgestaltungsmöglichkeit beim Spezifizieren der UI ein.

In einem gesonderten Papier¹ stellen wir eine Beispielspezifikation in dieser semiformalen Sprache vor und erläutern die verwendeten Konstrukte.

Ausblick

Durch den Einsatz der vorgestellten HCI-Spezifikationsprache ergeben sich vielfältige Möglichkeiten - aber auch Risiken.

Eine durchgängige Optimierung der Oberflächen-Entwicklung ist an einige Voraussetzungen geknüpft.

Es müssen geeignete Spezifikationsformalismen geschaffen werden, welche für die verschiedenen Phasen, Perspektiven und Abstimmungsmedien in der Oberflächen-Entwicklung übergreifend genutzt werden können. Dies erfordert auch die Einführung von werkzeuggestützten Spezifikationsmethoden, wobei abhängig vom Verwendungszweck auch eine Werkzeugunterstützung für die Ableitung in Abstimmungsmedien hilfreich ist. Außerdem kann der Einsatz von MDA-Werkzeugen zur maschinellen Verwertung der Spezifikationsinhalte in der Implementierungsplattform erheblichen Programmieraufwand einsparen und ist daher ein logischer Baustein einer durchgängig optimierten Oberflächen-Entwicklung.

Die notwendigen Prozessveränderungen in der Oberflächenentwicklung bergen jedoch auch Risiken. So kann eine unbeabsichtigte Überformalisierung der Spezifikationsform dazu führen, dass der Aufwand für die Erstellung und Abstimmung des Spezifikationsinhalts in keinem Verhältnis zu den Kosteneinsparungen und der Qualitätssteigerungen bei der tatsächlichen Implementierung der Oberflächen steht. Man sollte sich auch bewusst sein, dass eine massive Werkzeugunterstützung im Entwicklungsprozess die Beteiligten mit einem weiteren Aspekt konfrontiert, der uns bereits von anderen Entwicklungswerkzeugen bekannt ist: der gegenseitigen Abhängigkeit von Inhalten und Werkzeugen. Die Prozessveränderungen erfordern eine gründliche Vorbereitung und auch ein Umdenken in den verantwortlichen Köpfen. Auch hier gilt, dass ohne ein aktives Management rollenübergreifender Aufgaben der Ausgang der Optimierungsstrategie zum Glücksspiel gerät.

Welche typischen Arbeitsschritte gilt es beim Einführen von UI-Spezifikationsmethoden durchzuführen?

Sicher kann diese Frage nicht universell beantwortet werden, es lassen sich jedoch einige typische Arbeitsschritte aufzählen, die bei der praktischen Umsetzung von Prozessverbesserungen in der Oberflächenentwicklung nützlich sind:

- Zuschnitt eines anwendungsfeldspezifischen Spezifikationsformats und Unterstützung beim iterativen Aufbau der Spezifikation.
- Abbildung der Informationsstrukturen für die Spezifikationserfassung auf das Erfassungswerkzeug, Unterstützung bei der Bestimmung der Systemgrenzen und Strukturierung des Oberflächenumfangs, Moderation des iterativen Vertiefens der Spezifikation von der Inception bis zur Implementierungsreife.
- Anbindung der Spezifikation an einen UI-Styleguide und ableiten von Bildschirmen, Ablaufsimulationen und Strukturberichten.
- Überleitung der Spezifikation in die Umsetzungsplattform, Unterstützung bei der maschinellen Umwandlung der Spezifikation in zielplattformenspezifischen Code

Da die neuen Möglichkeiten formaler Spezifikationen die bisherigen Arbeitsweisen und Zuständigkeitsgrenzen aufbrechen, sind Veränderungen der Entwicklungsprozesse wahrscheinlich.

Der interdisziplinäre Charakter einer durchgängigen Oberflächenspezifikation kann zu Spannungen zwischen den klassischen Rollen des Entwicklungsteams

führen und erfordert eine integrierende Begleitung durch Teammitglieder, die auf entsprechende Erfahrungen zurückgreifen können.

pch, 02/2002
last revised 01/2006

Paul Chlebek
HCI • GUI • UML • OOD • MDA • XSLT

mailto: pc@benutzeroberflaeche.de
phone: +49 173 106 1830

<http://www.projectpeople.net/chlebek/>
<http://www.benutzeroberflaeche.de>

Links:

State-Charts

<http://www.wisdom.weizmann.ac.il/~dharel/SCANNED.PAPERS/Statecharts.pdf>

