

4. Berliner Werkstatt Mensch-Maschine-Systeme

Bedienen und Verstehen

Ein Ansatz zur kontextfreien Repräsentation multimodaler Dialoge unter Berücksichtigung der Dialogqualität.

B. Niedermaier⁺, M. Lang^{*}

⁺ BMW Group, Mensch-Maschine-Interaktion

^{*} Technische Universität München, Mensch-Maschine-Kommunikation

1 Abstract

Der Wettstreit verschiedener Automobilhersteller um das erste internetfähige Fahrzeug und das damit verbundene große öffentliche Interesse zeigen, welche hohe Beachtung neuen Informations- und Assistenzsystemen im Fahrzeug entgegengebracht wird. Dabei stellt die Integration zusätzlicher Wahrnehmungs- und Aktionsmodalitäten einen vielversprechenden Ansatz dar, um diese zusätzlich verfügbaren Funktionen in einer der Fahraufgabe angemessenen Weise benutzbar zu machen.

Da die hierbei entstehenden multimodalen Bedienkonzepte den Charakter eines Fahrzeuges sehr stark prägen, werden sie in Zukunft ein potentiell Unterscheidungsmerkmal für den Nutzer darstellen. Deshalb ist es bereits in einem sehr frühen Stadium des Entwicklungsprozesses notwendig, dass Nutzungskonzepte erlebbar und somit bewertbar gemacht werden. Die hierzu erforderlichen Werkzeuge müssen einem sehr breiten Spektrum an Anforderungen genügen. Diese Bandbreite reicht von einer geeigneten Dialogrepräsentation, über die funktionale Integration der einzelnen Modalitäten, bis hin zur Unterstützung des Entwicklers bei der nutzergerechten Dialoggestaltung.

In diesem Beitrag wird ein formales Verfahren zur Diskussion gestellt, das die strukturierte und kontextfreie Beschreibung multimodaler Bediendialoge erlaubt. Es wird dann gezeigt, wie diese formale Darstellung genutzt werden kann, um Teilaspekte der Dialogqualität bereits während der Dialogerstellung abzusichern.

2 Einleitung

Am Beispiel der aufeinanderfolgenden Modelle der BMW 7er Reihe lässt sich die Entwicklung der vom Fahrer bedienbaren Funktionen in den vergangenen 25 Jahren sehr gut darstellen (Bild 1).

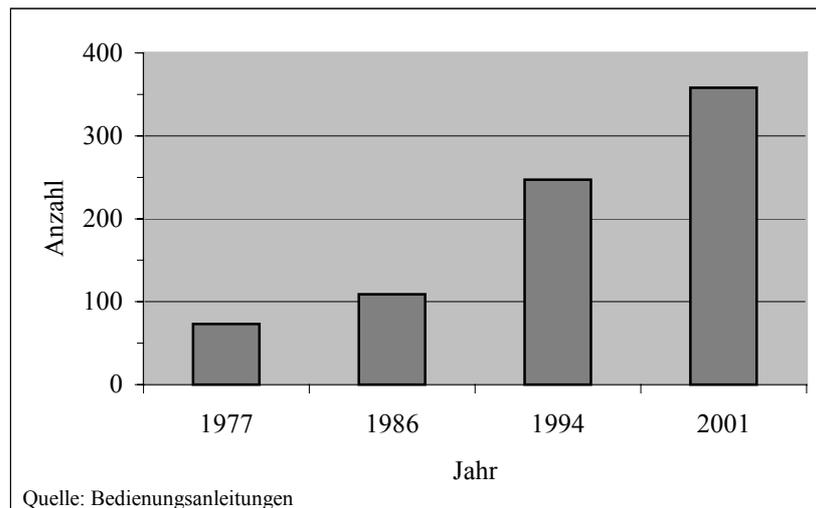


Bild 1 – Entwicklung der vom Fahrer bedienbaren Funktionen in der BMW 7er-Reihe

Ein großer Teil dieser neu hinzugekommenen Funktionen steht im Zusammenhang mit der Bedienung von Informations-, Kommunikations- und Unterhaltungssystemen. Die Fahrzeughersteller müssen deshalb diese Funktionen so zur Verfügung zu stellen, dass sie in einer für die Fahraufgabe geeigneten Art und Weise genutzt werden können.

Diesen gesteigerten Anforderungen in der Bedienung versucht man durch die Integration neuer Aktions- und Wahrnehmungsmodalitäten gerecht zu werden. Hierzu gehören möglicherweise Sprachausgaben und Spracherkennung oder auch Gestikerkennung. Eine sinnvolle Verknüpfung dieser verschiedenen Modalitäten ist erforderlich. Sie müssen so zusammengeführt werden, dass sie sich zu einem Gesamtbedienkonzept ergänzen und dadurch eine effiziente Bedienung ermöglichen [3].

Insbesondere bei Verwendung eines Displays als zentrale und frei programmierbare Anzeigefläche ist es möglich, aus den einzelnen Modalitäten weitgehend orthogonale, also in der Bedienphilosophie grundsätzlich verschiedene Konzepte zu realisieren. Deshalb müssen bereits in einem sehr frühen Stadium des Entwicklungsprozesses mögliche Ansätze bewertet werden können. Dabei ist eine fundierte Beurteilung in Usability-Untersuchungen vor allem anhand funktionsfähiger Prototypen möglich. Weil es aber an geeigneten Werkzeugen fehlt, ist der Prozess, in dem solche multimodalen Bedienkonzepte erstellt und geändert werden, bisher sehr zeit- und kostenaufwändig. Im folgenden werden nun Anforderungen an ein geeignetes Werkzeug herausgearbeitet und ein Lösungsvorschlag dargestellt.

3 Anforderungen an ein Werkzeug zur Realisierung multimodaler Dialoge im Fahrzeug

Zur Bestimmung der Erfordernisse müssen zunächst die Randbedingungen klar sein, die bei der prototypischen Realisierung nutzergerechter multimodaler Nutzungskonzepte für das Fahrzeug zu beachten sind.

3.1 Unterstützte Modalitäten

Unter dem Begriff „Modalität“ versteht man im Bereich der User-Interface-Entwicklung eine im System bereitgestellte Möglichkeit, über die der Mensch mit der Maschine interagieren kann [14].

Was ist aber nun solche eine Möglichkeit? Wie in [8] dargestellt ist, erfolgt die Interaktion hauptsächlich über den auditiven, den visuellen und den taktilen Sinneskanal. Da der Mensch seine rezeptorischen und aktuatorischen Fähigkeiten in einer sehr vielfältigen und komplexen Weise einsetzen kann, ist eine gesamtheitliche technische Realisierung der Schnittstelle auf der Ebene der Sinneskanäle nicht möglich. Vielmehr muss jede gewünschte Ausprägung der Interaktion wie z.B. Spracherkennung, Sprachausgabe, Gestikerkennung, graphische Anzeigen, manuelle Eingabeelemente oder kraftrückkoppelnde Elemente separat realisiert werden. Jede dieser Interaktionsmöglichkeiten wird daher im Sinne des Mensch-Maschine-Dialogs als Modalität bezeichnet.

Um nun werkzeuggestützt Bediendialoge für verschiedene Modalitätskonfigurationen erstellen zu können, sollten möglichst wenige Einschränkungen vom Werkzeug bezüglich der Modalitäten ausgehen:

- Für jede Modalität existieren meist mehrere technische Realisierungen, die im allgemeinen nicht schnittstellenkompatibel sind. Oft besteht die Aufgabe gerade darin, diese verschiedenen technischen Lösungen – z.B. in Form verschiedener Spracherkennung – zu vergleichen.

Das Werkzeug darf also nicht auf eine einzige technische Implementierung einer Modalität festgelegt sein. Mit verträglichem Aufwand müssen Modalitäten ersetzt oder getauscht werden können.

- Manche Modalitäten, wie z.B. neue Eingabeelemente sind a priori nicht bekannt und müssen deshalb nachträglich im Werkzeug bekannt gemacht werden können. Das Werkzeug darf also nicht nur auf ein bestimmtes Set an Modalitäten festgelegt sein, sondern muss um zusätzliche Modalitäten erweiterbar sein.

Diese Randbedingungen implizieren, dass die Realisierungen der einzelnen Modalitäten nicht funktional in das Werkzeug integriert werden dürfen. Vielmehr muss eine kontextfreie Beschreibung bereitgestellt werden, mit deren Hilfe sich ihr funktionales Verhalten unabhängig vom Mensch-Maschine-Dialog darstellen lässt.

3.2 Multimodalitätstyp

Es gibt eine große Zahl multimodal bedienbarer technischer Systeme. Allerdings weisen die Realisierungen große Unterschiede bezüglich des Zusammenspiels der einzelnen Modalitäten und des damit verbundenen technologischen Aufwands auf. Um das Werkzeug entsprechend gestalten zu können, muss deshalb zunächst geklärt werden, wie die verschiedenen Modalitäten im Fahrzeug zusammenwirken, das heißt, welcher Typ von Multimodalität zu erwarten ist. Die in [11] vorgestellte systemorientierte Klassifikation unterteilt multimodale Systeme anhand des möglichen zeitlichen Gebrauchs der einzelnen Modalitäten zueinander und der Vor-

gehensweise, wie Ein- und Ausgaben über die verschiedenen Modalitäten zur Gesamtinformation fusioniert werden (Bild 2).

		Gebrauch der Modalitäten	
		sequentiell	parallel
Fusion	verbunden	alternierend	synergetisch
	unabhängig	exklusiv	simultan

Bild 2 – Klassifikationsschema für multimodale Systeme [11]

Das klassische „Put-that-there“-System [4] ist z.B. in dieser Klassifikation als synergetisch-multimodal einzuordnen, weil die parallel möglichen Eingaben über Sprache und Zeigegesten im Verbund interpretiert werden.

Aktuell im Fahrzeug erhältliche Systeme (‘A’ in Bild 3) sind dadurch gekennzeichnet, dass Informationen aus den einzelnen Modalitäten unabhängig voneinander verarbeitet werden. Es findet also keine Informationsfusion statt. Technisch ist zwar eine parallele Bedienung verschiedener Funktionen mit unterschiedlichen Modalitäten, wie z.B. der Eingabe einer Telefonnummer per Sprache und der parallelen manuellen Eingabe eines Navigationsziels, möglich. Aufgrund der hierbei entstehenden hohen kognitiven Beanspruchung des Fahrers sind solche simultan-multimodalen Bedienvorgänge selten, weshalb diese Systeme stärker exklusiv-multimodal einzuordnen sind.

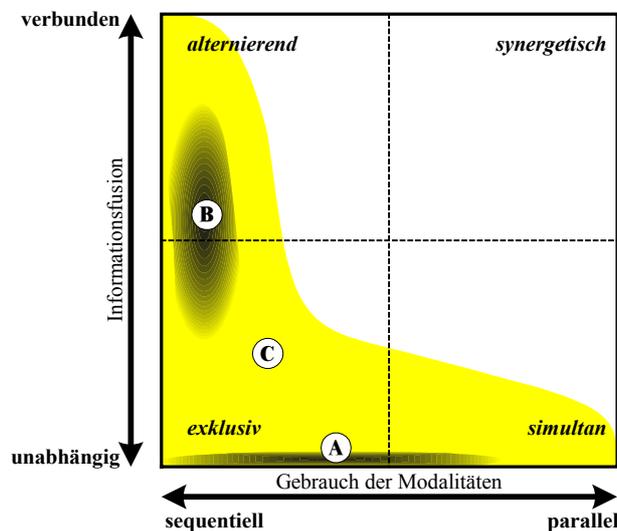


Bild 3 – Multimodalitätstypen aktueller (A) und experimenteller (B) Bediensysteme. Vom Werkzeug abzudeckender Bereich (C).

Bei aus dem Forschungsbereich stammenden experimentellen Systemen (‘B’ in Bild 3) können die einzelnen Modalitäten im Verbund eingesetzt werden [2] [10]. Die verschiedenen Modalitäten werden dabei so zur Verfügung gestellt, dass sie sich zu einem Gesamtbediendialog zusammenfügen. Es wird also dem Faktum, dass der Fahrer komplexe Funktionen im allgemeinen nacheinander ausführt, dadurch Rechnung getragen, dass die verschiedenen Modalitäten dazu benutzt werden können, um **eine** Funktion effizient auszuführen und nicht mehre-

re gleichzeitig. Der Benutzer wird dabei vom System durch die jeweiligen Teilschritte der Bedienung geleitet. Dies führt zu einem sequentiellen Gebrauch der Modalitäten, wobei in jedem Teilschritt ein Wechsel der Modalität erfolgen kann.

Diese Entwicklung in Richtung alternierend-multimodaler Bedienung erscheint auch durchaus sinnvoll, weil verschiedene Modalitäten im Verbund die Anpassung der Interaktion an die aktuelle Fahrsituation deutlich erhöhen können. Bediendialoge diesen Typs sind deshalb ebenfalls im vom Werkzeug abzudeckenden Bereich ('C' in Bild 3) enthalten.

Dies macht einen Fusionsmechanismus zur Verbindung der Informationen aus den einzelnen Modalitäten erforderlich. Da der Gebrauch der Modalitäten sequentiell erfolgt, ist ein weniger aufwändiges Verfahren als beispielsweise für synergetisch-multimodale Systeme [12] ausreichend.

3.3 Dialogqualität

Wie in 3.2 bereits angedeutet, liegt ein wichtiger Aspekt bei der Entwicklung multimodaler Dialoge für das Fahrzeug in der für die Fahraufgabe geeigneten Benutzbarkeit. In [5] wird hierzu deutlich gemacht, dass zunächst Bedarf an Methoden besteht, um dieser Forderung gerecht werden zu können. Die entsprechenden Anforderungen wurden von der ISO TC22/SC 13/ WG8 im Normentwurf ISO/DIS 15005 [7] erarbeitet. Von der Kommission der europäischen Gemeinschaft wurde eine entsprechende Richtlinie [1] erstellt, um die ergonomische und verkehrssichere Gestaltung von Mensch-Maschine-Schnittstellen in Fahrzeugen zu beschreiben. Die Inhalte der Dokumente sind dabei sehr breit gefächert, wobei auf Fragen des Designs und der Positionierung von Anzeigen und Bedienelementen, der Präsentation von Informationen und auf eine adäquate Mensch-Maschine-Interaktion eingegangen wird.

Entsprechend diesem breiten inhaltlichen Rahmen, müssen verschiedenste Methoden angewandt werden, um die Einhaltung dieser Anforderungen überprüfen zu können. Hierzu zählen die Überprüfung objektiv messbarer Größen, wie Erreichbarkeit von Eingabeelementen und Verdeckungen von Anzeigen. Des Weiteren werden Methoden entwickelt, die mit Hilfe von Benutzermodellen [9] oder Prüflisten [13] die Bedienbarkeit und Tauglichkeit des jeweiligen Systems abschätzen. Diese Aussagen zur Gebrauchstauglichkeit können mit Hilfe von Nutzertests weiter detailliert werden. Allen diesen Verfahren ist dabei gemeinsam, dass zumindest eine Spezifikation oder oft sogar ein funktionierendes System vorhanden sein muss, um sie anwenden zu können.

Deshalb sollte das Werkzeug den Dialogentwickler schon bei der Erstellung dieser Spezifikationen bzw. Prototypen so weit unterstützen, dass bereits in diesem frühen Stadium – also bevor obige Verfahren angewandt werden können – Maßnahmen zur Sicherung der Dialogqualität getroffen werden.

4 Entwicklungsumgebung

Im Folgenden wird nun ein Rahmenwerk vorgestellt, mit dem an das Fahrzeug angepasste multimodale Bediendialoge prototypisch realisiert werden können.

4.1 Kontextfreie Repräsentation der Modalitäten

Wie bereits in 3.1 angedeutet, soll die Entwicklungsumgebung mit beliebigen Modalitäten umgehen können. Jede technische Realisierung einer Modalität, wie beispielsweise ein bestimmter Spracherkennung oder die Anbindung an ein spezielles Bedienelement, wird als Dialogkomponente bezeichnet. Dabei handelt es sich jeweils um ein eigenständiges Softwaremodul, das während der Dialogausführung mit dem Dialogmanager kommuniziert (Bild 4). Der Dialogmanager ist Teil des Werkzeugs und für die Kontrolle des Dialogablaufs verantwortlich. Er steuert entsprechend der Dialogbeschreibung (vgl. 4.2) zentral die verschiedenen Dialogkomponenten.

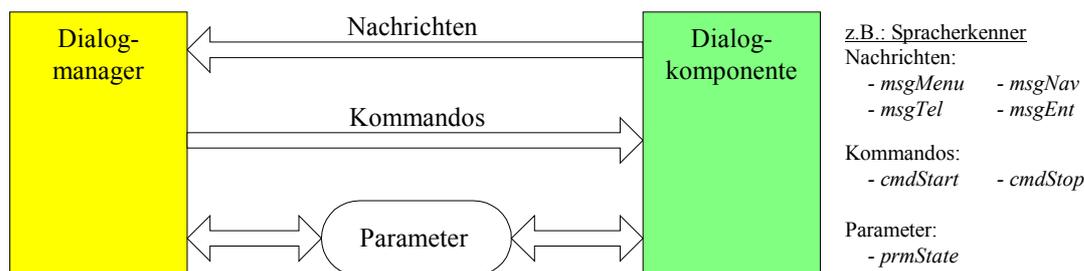


Bild 4 – Interaktion Dialogmanager und Dialogkomponente

Um nun beispielsweise einen Spracherkennung in die Dialogbeschreibung zu integrieren, ohne dabei Veränderung an der Entwicklungsumgebung, oder der Dialogrepräsentation vorzunehmen, wird dessen Funktionsumfang kontextfrei beschrieben. Diese Schnittstellendefinition wird in der Entwicklungsumgebung bekannt gemacht und kann dann zur Dialogbeschreibung verwendet werden.

Hierzu stehen drei semantische Mittel zur Verfügung (Bild 4):

1.) Nachrichten

Nachrichten sind im allgemeinen asynchrone Ereignisse, die in den Dialogkomponenten beispielsweise durch Nutzereingaben erzeugt werden. Bei einem Spracherkennung werden sie deshalb bei der Erkennung des entsprechenden Sprachkommandos generiert. Nach ihrem Eintreffen im Dialogmanager löst dieser dann entsprechend der Dialogdefinition bestimmte Aktionen aus (vgl. 4.2).

Bsp.: Nutzer: „Hauptmenü“ => Nachricht: *msgMenu*
Nutzer: „Navigation“ => Nachricht: *msgNav*

2.) Kommandos

Umgekehrt kann der Dialogmanager mit Hilfe von Kommandos auf das Verhalten der Dialogkomponenten Einfluss nehmen. Beim Beispiel des Spracherkennungers kann damit die Erkennung gestartet (*cmdStart*) oder beendet (*cmdStop*) werden.

3.) Parameter

Mit Hilfe von Parametern werden zum einen Informationen über den internen Zustand der Dialogkomponenten bereitgestellt. Zum anderen ermöglichen sie die Einflussnahme auf das Verhalten der Dialogkomponenten. Sie bilden einen Informationspool, der

zu jedem Zeitpunkt der Dialogausführung den jeweils aktuellen Zustand der Dialogkomponenten widerspiegelt. Im Beispiel wird der aktuelle Aktivitätszustand des Spracherkenners verfügbar gemacht (*prmState*).

Weitere Wahrnehmungs- und Aktionsmodalitäten, wie beispielsweise ein anderer Spracherkennung, oder graphische Anzeigen, haptische Eingabelemente etc. werden analog hierzu als Dialogkomponenten ebenfalls kontextfrei in das Werkzeug integriert.

4.2 Formale Dialogrepräsentation

Um aus diesen zunächst unverbunden nebeneinander stehenden Funktionalitäten der Dialogkomponenten einen Bediendialog erstellen zu können, ist eine informationstechnische Repräsentation des Dialogs erforderlich. Hierzu wird in Anlehnung an [6] ein endlicher hierarchischer Zustandsautomat als formales Beschreibungsverfahren verwendet.

Der Bediendialog wird dazu in einzelne Zustände unterteilt. Transitionen, also Übergänge zwischen den Zuständen, werden durch Nachrichten aus den Dialogkomponenten ausgelöst. In Bild 5 wird beispielsweise der Übergang vom Zustand *Hauptmenü* zum Zustand *Navigation* durch die Nachricht *msgNav* ausgelöst.

Das funktionale Verhalten der Dialogkomponenten innerhalb eines Zustandes wird mittels Reaktionen gesteuert. Zur Auslösung einer Reaktion ist wie bei den Transitionen eine Nachricht erforderlich, die aber anstatt des Zielzustandes Kommandos zugeordnet werden, die in den jeweiligen Dialogkomponenten das gewünschte Verhalten erzeugen. Um z.B. beim Drücken des Push-to-Talk-Knopfes (PTT) den Spracherkennung zu starten, was durch die Nachricht *msgPTT* signalisiert werden soll, muss folgende Reaktion definiert werden:

Nachricht: *msgPTT* => Kommando: *cmdStart*

Somit kann für jeden dieser Dialogzustände das Verhalten der Dialogkomponenten separat konfiguriert werden, wodurch eine getrennte Beschreibung des Systemverhaltens für die unterschiedlichen Dialogsituationen erreicht wird.

Dies bedeutet aber auch, dass Reaktionen und Transitionen, die in mehreren Zuständen identisch sind, für jeden dieser Zustände einzeln definiert werden müssen. Da dadurch mit steigender Zahl an Zuständen die Komplexität sehr stark zunehmen würde, wird der Entwickler durch eine hierarchische Gliederungsmöglichkeit unterstützt. Hierzu muss für jede Transition und Reaktion ein Gültigkeitsbereich – Scope – angegeben werden. Dieser kann lokal für den jeweiligen Zustand, global für alle oder für eine Teilmenge der Dialogzustände sein. Definiert man beispielsweise in Bild 5 die Transition zum Zustand *Hauptmenü* als global, werden 2 Transitionen eingespart. Die zur Aktivierung des Spracherkenners notwendigen Reaktionen können auf gleiche Weise reduziert werden.

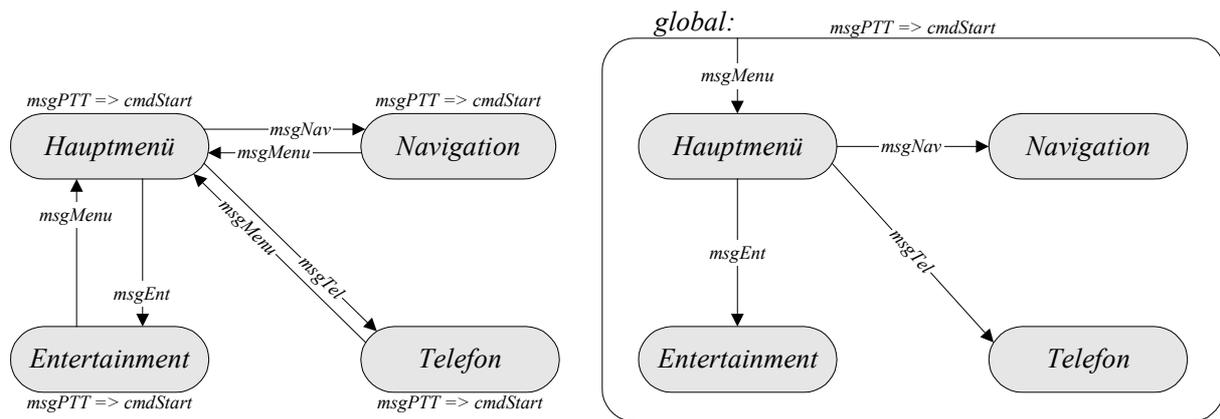


Bild 5 – Dialogbeschreibung ohne (links) und mit (rechts) hierarchischer Gliederung

4.3 Dialogqualitätssicherung

Da die formalen Beschreibungen sowohl des Dialogs als auch der Dialogkomponenten keine Informationen über den jeweiligen Kontext – also was und wie bedient wird – enthalten, können qualitätssichernde Maßnahmen ebenfalls nur weitgehend ohne inhaltlichen Bezug erfolgen. Dies sind Methoden zur formalen Verifikation und Konsistenzsicherung des Dialogs. Die meisten der stark inhaltlich geprägten Forderungen der Statement of Principles [1], wie beispielsweise den Fahrer „nicht visuell zu unterhalten“, können daher nicht sichergestellt werden. Da das Anwendungsfeld des Werkzeugs und die Verwendung der semantischen Elemente eindeutig festgelegt sind, ist zumindest eine Unterstützung des Entwicklers bei der Einhaltung bestimmter inhaltlicher Qualitätsmerkmale möglich (vgl. 4.3.3).

4.3.1 Dialogverifikation

Um eine sinnvolle Ausführung des Dialogs zu ermöglichen, muss die Dialogdefinition formale Kriterien erfüllen. Diese werden während der Erstellung überprüft und im Falle eines Verstoßes dem Entwickler angezeigt. Die vollständige und eindeutige Dialogbeschreibung ist hierbei eine grundlegende Voraussetzung, um die in [1] genannten „Prinzipien zur Interaktion mit den Anzeigen und Eingabeelementen“ überhaupt erfüllen zu können.

- **Vollständigkeit**
Zustände, die aufgrund fehlender Transitionen nicht erreicht oder nicht verlassen werden können, erhalten eine Kennzeichnung, so dass unvollständige Dialogdefinitionen vermieden werden.
- **Eindeutigkeit**
In einem Dialogzustand darf eine Nachricht nur dann zwei verschiedene Transitionen auslösen, wenn sich diese durch eine Kondition – also eine zusätzliche Bedingung – unterscheiden. Es wird deshalb sichergestellt, dass die Transitionsdefinitionen für den jeweiligen Dialogzustand eindeutig sind.
- **Erreichbarkeit und Entfernung zweier Zustände**
Insbesondere bei größeren Modellen ist es häufig schwierig zu entscheiden, ob ein Zustand von einem anderen Zustand aus unter Verwendung bestimmter Dialogkomponenten erreichbar ist. Der Distance-Counter ermöglicht die Bestimmung der hierzu

notwendigen Anzahl an Transitionen und erlaubt dadurch eine zustandsübergreifende Verifikation des Dialogablaufs.

4.3.2 Dialogkonsistenz

Konsistenz ist eine generelle Anforderung an benutzungsgerechte Mensch-Maschine-Dialoge, die insbesondere bei der Domäne Fahrzeug zu beachten ist, weil hier der Nutzer zur Auflösung von Uneinheitlichkeiten unnötig Aufmerksamkeit von der Fahraufgabe abwenden muss. Unter Einsatz folgender Methoden unterstützt das vorgestellte Werkzeug den Entwickler bei der Realisierung konsistenter Dialoge:

- **Dialogverhalten**
Mit Hilfe des Scoping-Mechanismus, also der Möglichkeit zur hierarchischen Gliederung des Dialogs durch die Definition verschiedener Gültigkeitsbereiche, wird sichergestellt, dass die Dialogkomponenten in verschiedenen Dialogzuständen identisches Verhalten zeigen. Hierzu erweitert man den Scope einer Transition, Reaktion oder eines Parameterwertes auf mehrere Zustände. Mit einer einzigen Definition kann somit in allen Zuständen eines Gültigkeitsbereichs gleiches Verhalten sichergestellt werden. Insbesondere Modifikationen können dadurch in einer konsistenten Art und Weise durchgeführt werden, da sie lediglich eine einzige Änderung der Dialogdefinition erfordern. Fehleranfällige und unter Umständen inkonsistente Mehrfachdefinitionen werden somit vermieden.
- **Konfiguration der Dialogkomponenten**
In multimodalen Systemen beinhaltet Konsistenz eine weitere Dimension, da die verschiedenen Dialogkomponenten aufeinander abgestimmt werden müssen, damit sie in einem sinnvollen Verbund zusammenwirken können. Hierzu erlauben spezielle Mechanismen in der kontextfreien Beschreibung, die zentrale Konfiguration der Dialogkomponenten im Werkzeug. So können beispielsweise das Vokabular des Spracherkenners mit den Sprachausgaben und bestimmte Displayanzeigen innerhalb einer gemeinsamen Oberfläche aufeinander abgestimmt und bei Bedarf modifiziert werden. Hierdurch werden fehleranfällige dezentrale Verfahren vermieden, die beispielsweise auf separaten Konfigurationsdateien für jede Dialogkomponente basieren. Dies wird durch ein Verfahren unterstützt, bei dem jedem Dialogzustand charakteristische Schlüsselwörter zugeordnet werden, auf die dann bei der Konfiguration der Dialogkomponenten zugegriffen werden kann. Dadurch wird eine konsistente Verwendung der Begriffe und Bezeichner zwischen den Dialogkomponenten erleichtert.

4.3.3 Usability – Benutzbarkeit

Die oben genannten Methoden stellen auf formaler Ebene den notwendigen Rahmen zur Verfügung, um im Sinne eines Rapid Prototypings multimodale Dialoge zu implementieren. Bei einer sinnvollen Verwendung können mit ihnen auch inhaltliche Qualitätskriterien berücksichtigt und dadurch die Benutzbarkeit des Dialogs erhöht werden:

- **Interaktionslänge**
Eine Forderung an Bediendialoge im Fahrzeug sind unterbrechbare Interaktionssequenzen von angemessener Länge [1]. Da ein Interaktionsschritt normalerweise von

einer Transition begleitet wird, kann die vom Distance-Counter errechnete Entfernung als Abschätzung für die Interaktionslänge eingesetzt werden. Somit steht eine Kennzahl zur Verfügung, die als Diskussionsgrundlage benutzt werden kann.

- **Einfachheit und Verständlichkeit der Ausgaben**
Ein weiteres Erfordernis sind einfache und verständliche Systemausgaben. Das Werkzeug bietet hierzu in Form der Schlüsselwörter bzw. der Möglichkeit zur Angabe einer maximalen Zeichenlänge für kritische Parameter eine sinnvolle Hilfestellung an.
- **Ablenkung von der Fahraufgabe**
Zur Vermeidung von Inkompatibilitäten zwischen Bedien- und Fahraufgabe („Allgemeine Gestaltungsprinzipien“ [1]) aufgrund von Inkonsistenzen in der Dialogdefinition stehen dem Dialogentwickler mit dem Scoping-Mechanismus und der zentralen Konfigurationsmöglichkeit der Dialogkomponenten verschiedene Werkzeuge zur Sicherung der Dialogkonsistenz zur Verfügung.

5 Zusammenfassung und Ausblick

In diesem Beitrag wird basierend auf notwendigen Anforderungen und vorhandenen Randbedingungen ein Rahmenwerk vorgestellt, mit dem multimodale Bedienkonzepte prototypisch realisiert werden können. Der Entwickler wird durch den Prozess der Dialogerstellung geführt. Die konzeptuelle Verwendung der semantischen Mittel zur Modellierung des Bediendialogs und die graphische Gestaltung des Werkzeugs sind deshalb für diesen speziellen Anwendungsfall optimiert. Dadurch werden Probleme vermieden, die bei der Adaption von Werkzeugen entstehen, die ein anderes Anwendungsgebiet oder einen zu allgemeinen Fokus haben.

Die kontextfreie Beschreibung der einzelnen Dialogkomponenten ermöglicht die Integration neuer Aktions- und Wahrnehmungsmodalitäten in das Werkzeug.

Auf Grundlage der formalen Repräsentation des Dialogs und der Dialogkomponenten sowie der engen Aufgabenfokussierung des Werkzeugs werden dann Maßnahmen zur Sicherstellung der Dialogqualität vorgestellt. Dabei wird gezeigt, wie diese formalen Kriterien verwendet werden können, um den Entwickler bereits in diesem frühen Stadium des Entwicklungsprozesses bei der Einhaltung bestimmter inhaltlicher Gestaltungsprinzipien im Sinne der Statement of Principles [1] zu unterstützen.

Im nächsten Schritt erfolgt die Untersuchung der praktischen Tauglichkeit des Werkzeugs. Hierzu wird mit Hilfe von Experten aus der MMI-Entwicklung ein Nutzertest durchgeführt.

6 Literatur

- [1] 2000/53/EC: *Commission Recommendation of 21 December 1999 on safe and efficient in-vehicle information and communication systems: A European statement of principles on human machine interface*. In: Official Journal of the European Communities

- L19, Vol. 43 (25.01.2000), S.64-68
- [2] Bengler, K.; Geutner, P.; Niedermaier, B. & Steffens, F.: „*Eyes free – Hands free*“ oder „*Zeit der Stille*“. *Ein Demonstrator zur multimodalen Bedienung im Fahrzeug*. In: Multimodale Interaktion im Bereich der Fahrzeug- und Prozessführung. DGLR-Bericht 2000-02 (2000), S.299-307
 - [3] Bengler, K.: *Aspekte der multimodalen Bedienung und Anzeige im Automobil*. In: Fahrzeugführung. Springer-Verlag (im Druck)
 - [4] Bolt, R. A.: *Put that there: Voice and gesture at the graphics interface*. In: ACM Computer Graphics, Bd. 14 (1980), Nr. 3, S.262-270
 - [5] Haller, R.: *HMI – New Technologies and Safety Aspects*. In: Proceedings of 10th International Conference Traffic Safety on Two Continents. VTI konferens 13A part 5 (2000), S.7-15
 - [6] Harel, D.: *Statecharts: A visual formalism for complex systems*. In: Science of Computer Programming, Vol. 8 (1987), Nr. 3, S.231-274
 - [7] ISO/DIS 15005: *Straßenfahrzeuge – Ergonomische Aspekte von Fahrerinformations- und -assistenzsystemen – Grundsätze und Prüfverfahren des Dialogmanagements*. Hrsg. International Organization for Standardization. Genf: 2000
 - [8] Lang, M.: *Mensch-Maschine-Kommunikation 1 – Kurzmanuskript zur Vorlesung*. München, Technische Universität, Lehrstuhl für Mensch-Maschine-Kommunikation. Vorlesungsskript, 1994
 - [9] Marrenbach, J.; Maaßen, D. & Kraiss, K.-F.: *Formale Methode zur Evaluierung der Gebrauchstauglichkeit von technischen Systemen*. In: 3. Berliner Werkstatt Mensch-Maschine-Systeme „Bewertung von Mensch-Maschine-Systemen“. ZMMS-Spektrum, Bd. 11 (1999), S.283-296
 - [10] Neuss, R.: *Usability Engineering als Ansatz zum Multimodalen Mensch-Maschine-Dialog*. München, Technische Universität, Lehrstuhl für Mensch-Maschine-Kommunikation. Dissertation, 2000
 - [11] Nigay, L. & Coutaz, J.: *A Design Space For Multimodal Systems: Concurrent Processing and Data Fusion*. In: Proceedings of INTERCHI'93 (1993), S.172-178
 - [12] Nigay, L. & Coutaz, J.: *A Generic Platform for Adressing the multimodal Challenge*. In: Proceedings of CHI'95 (1995), S.98-105
 - [13] Nirschl, G. & Blum, E.-J.: *MMI-Prüfliste – Verfahren und Werkzeug zur Bewertung von Mensch-Maschine-Systemen im Kraftfahrzeug*. In: Informations- und Assistenzsysteme im Auto benutzergerecht gestalten. Berichte der Bundesanstalt für Straßenwesen, Heft M116 (2000), S.42-50
 - [14] Timpe, K.-P.; Jürgensohn, T. & Kolrep, H.: *Mensch-Maschine-Systemtechnik. Konzepte, Modellierung, Gestaltung, Evaluation*. Düsseldorf: Symposion Publishing 2000