

HAMVIS: Generierung von Visualisierungen in einem Rahmensystem zur systematischen Entwicklung von Benutzungsschnittstellen

Ralf Möller

Universität Hamburg, Fachbereich Informatik,
Vogt-Kölln-Str. 30, 22527 Hamburg

<http://kogs-www.informatik.uni-hamburg.de/~moeller/>

1 Einleitung

Im Bereich der Geschäftsweltmodellierung hat sich die explizite Modellierung von innerbetrieblichen Vorgängen und auch von Handlungen der an Arbeitsprozessen beteiligten Personen als notwendige Maßnahme zur Gewährleistung einer systematischen und kosteneffektiven Unterstützung der Arbeitsprozesse durch Computersysteme erwiesen (Stichwort „Workflow“). Auch im Bereich der ingenieurorientierten Konstruktionssysteme wird m.E. in naher Zukunft ein solches integriertes Vorgehen wichtig werden. Anstatt einzelne Konstrukteure mit speziellen CAD-Programmen arbeiten zu lassen, wird sich auch hier eine integrierte Produktdaten- und Aktivitätenmodellierung durchsetzen. Im Rahmen der Einbettung einer speziellen Konstruktionsaufgabe in einen Gesamtkontext bietet es sich an, einem Konstrukteur in einem CAD-System nicht eine leere „Werkbank“ zu präsentieren, sondern eine für die jeweilige Arbeitsaufgabe passende Interaktionsumgebung bereitzustellen, mit der eine bestimmte Teilaufgabe adäquat gelöst werden kann. Eine wichtige Rolle spielen in diesem Zusammenhang interaktive Visualisierungen, d.h. Graphiken oder Skizzen, mit denen durch graphische Interaktion eine bestimmte Aufgabe bearbeitet werden kann. Insbesondere die systematische Herleitung von bezüglich einer bestimmten Aufgabenstellung angemessenen Visualisierungsbestandteilen aus dem Objektmodell einer Anwendung ist in diesem Kontext ein wichtiger Gesichtspunkt.

Diese Arbeit definiert mit HAMVIS (HAMburger VISualisierungssystem) die Konzeption eines Rahmensystems zur methodischen Entwicklung von aufgabenorientierten, interaktiven Visualisierungen für Benutzungsoberflächen für Endanwender (und nicht zur Anwendungsmodellierung selbst [4]). Das Problem bei der Definition eines Rahmensystems dieser Art besteht darin, geeignete Teilprobleme zu definieren und so zu lösen, daß am Ende die Teillösungen und die jeweils zur Problemlösung verwendeten Verfahren mit ihren jeweiligen Begriffsbildungen nahtlos ineinandergreifen, so daß bei der Entwicklung einer Anwendung mit einer interaktiven Oberfläche vielfältige Entwurfsentscheidungen koordiniert werden können.

Entwurfsentscheidungen bei der Gestaltung einer Anwendung betreffen sowohl Implementierungs- als auch Präsentationsaspekte (vgl. die konzeptuellen Teilmodelle für das Design und die Nutzung von Artefakten aus Abbildung 1). In modernen graphischen Benutzungsschnittstellen sind meist direktmanipulative Interaktionsformen angemessen. Durch direktmanipulative, d.h. exekutive

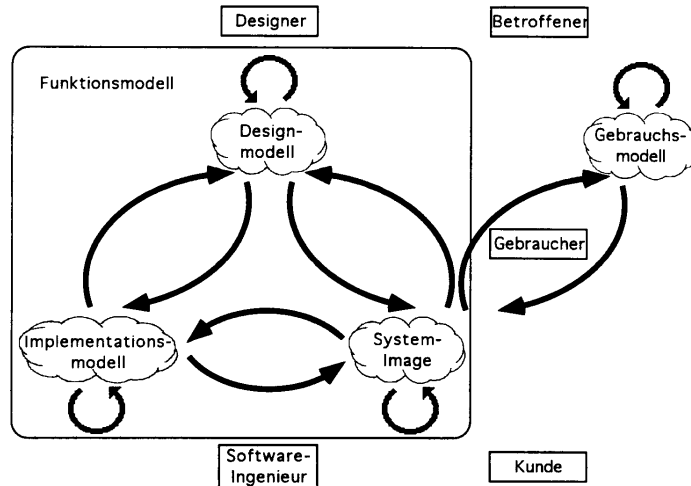


Abbildung 1. Konzeptuelle Modelle für Teilaspekte des Design und der Nutzung einer Anwendung (nach Holz [5], S. 85).

Interaktionstechniken tritt die Sicht einer „Kommunikation“ mit einem Rechensystem in den Hintergrund (im Gegensatz zu deskriptiven Interaktionsformen). Besonders im Hinblick auf eine automatisierte Inhaltszusammenstellung ist jedoch m.E. auch bei exekutiven Interaktionsformen eine kommunikationsorientierte Designperspektive zur Entwicklungszeit vorteilhaft. Obwohl einem Anwendungsnutzer eine kommunikationsorientierte Sicht auf die Designentscheidungen „hinter den Kulissen“ (siehe Abbildung 1) nicht bewußt zu sein braucht, bietet eine solche Sicht zur Entwicklungszeit vielfältigste Möglichkeiten zur methodischen, kosteneffektiven Entwicklung und Dokumentation einer graphischen Anwendung mit heutzutage zum Standard gehörenden direktmanipulativen Interaktionsformen.

Es ist das Ziel dieser Arbeit, ein Rahmensystem für die methodische Visualisierungsgenerierung auf der Basis von konzeptuellen Informationen zu erstellen, bei dem eine Trennung zwischen Entwurfszeit (bzw. Entwicklungszeit) und Laufzeit (oder Benutzungszeit) der Anwendung vorgenommen wird. Wesentliche Gestaltungsmerkmale einer Oberfläche und der darin präsentierten Visualisierungen sollen durch die beteiligten Personengruppen zur Entwicklungszeit bewertbar sein (vgl. hierzu die Arbeiten zu Intelligenten Multimedia-Präsentationssystemen (auch IMMPS genannt), die jedoch auf den Umgang mit konkreten Objekten zur Laufzeit ausgerichtet sind [1]). Ich möchte mit dieser Arbeit eine Basis schaffen für die Definition von wiederverwendbaren formalen „Grundbausteinen“, die der Design- bzw. Entwurfsebene zuzuordnen sind und nicht nur auf der Ebene der oberflächennahen Interaktionselemente anzusiedeln sind. Die in dieser Arbeit entwickelten Modellierungsmethoden und Repräsentationstechni-

ken zur modellbasierten, aufgabenorientierten Konstruktion von Visualisierungen werden anhand des interaktiven Rahmensystems HAMVIS demonstriert, das prototypisch implementiert wurde.

2 Prinzipien der Unterstützung des Visualisierungsentwurfs

Die Basisidee dieser Arbeit besteht darin, daß der Entwickler die gewünschten „Interaktionsdienste“ während der Entwicklungsphase einer Anwendung zunächst grob beschreibt. Die Beschreibungsperspektive bzw. das Beschreibungsvokabular liefert hierzu die aus der Mensch-Computer-Interaktion bekannte tiefe Modellierung von Benutzeraktionen. Allerdings muß die Beschreibung der Endbenutzeraktionen auf der Ebene des Umgangs mit Domänenobjekten erfolgen und nicht auf der Ebene der Manipulation von Graphikobjekten (mit graphischen Gesten). Bei der Aktionenmodellierung ist zu unterscheiden zwischen der konzeptuellen Beschreibung einer Aktion (mit entsprechenden Kasusrollen für die manipulierten Objekte oder die erzeugten Objekte) und der Beschreibung der Zerlegung einer Aktion bzw. deren Einbettung in einen größeren Kontext. Konzeptuelle Beschreibungen werden durch HAMVIS für eine Anwendungsklasse in einer „Bibliothek“ bereitgestellt. Eine Aktionenzerlegung hingegen wird anwendungsspezifisch durch den oder die Entwickler definiert. Die Anwendungsentwickler wählen jedoch zur Beschreibung der Einzelaktionen die konzeptuellen Beschreibungen aus der Aktionenbibliothek. Durch eine konzeptuelle Beschreibung der mit den Einzelaktionen in Zusammenhang stehenden Domänenobjekte lassen sich durch formale Schlußfolgerungen auch die zu den Vorgaben „passenden spezialisierten Aktionenkonzepte“ aus den vorgegebenen Bibliotheksmodellen bestimmen. Durch die im HAMVIS-Rahmensystem vordefinierten „generischen“ Aktionenkonzepte ist also der mögliche Raum der Beschreibungen strukturiert. Aktionenzerlegungen werden durch den Entwickler immer feiner beschrieben, so daß auf unterer Ebene eine automatische Zuordnung von Diensten des UIMS¹ bzw. der anwendungsklassenspezifischen UIMS-Bibliothek erfolgen kann.

Eine Aktionenzerlegung definiert die grobe Dialogstruktur einer Anwendung. Über die Aktionenmodellierung wird dabei auch der „Basisinhalt“ von Visualisierungen festgelegt. Durch die Konzepte der Aktionen werden Einschränkungen für die konkrete Ausgestaltung der Visualisierungen festgelegt. Die konzeptuelle Beschreibung von Interaktionsdiensten auf tiefer semantischer Ebene ermöglicht es, den Basisinhalt adäquat zu „vervollständigen“ (siehe die Komponenten zur Inhaltsplanung in IMMPS). Die Vervollständigung des Basisinhalts und die weiteren Schritte zur Strukturierung der Gesamtanwendung bis hin zur konkreten Präsentationsstruktur, in der alle Visualisierungskomponenten mit ihren Darstellungsfenstern, Zeichenattributen usw. festgelegt sind, müssen auf der Basis von konzeptuellen Informationen erfolgen, da konkrete Anwendungsobjekte zur Entwicklungszeit nicht notwendigerweise bekannt sind. Auch hier ist also eine besondere Modellbildung und eine entsprechende Formalisierung notwendig, die

¹ UIMS steht für „User Interface Management System“.

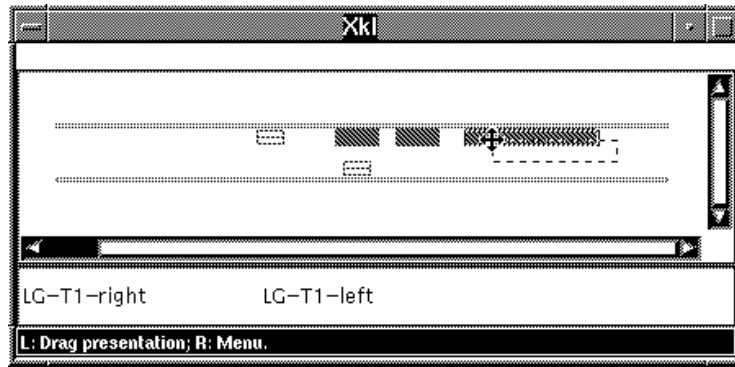


Abbildung 2. Schnappschuß aus einer fertige Anwendung, deren Oberfläche mit HAMVIS erzeugt wurde. Ein Einrichtungsgegenstand aus einer Palette (unten) wird auf einen automatisch berechneten Plazierungsbereich geschoben. In dem Beispiel wurden in vorhergehenden Konstruktionsschritten schon Objekte in der Kabine plaziert.

sich von Ansätzen, die im Rahmen von IMMP-Systemen entwickelt wurden, unterscheidet (vgl. z.B. [6]).

2.1 Ein Leitbespiel aus der Anwendungsklasse der Layoutprobleme

Eine Klasse von Anwendungen, bei denen Visualisierungen zur Realisierung von Benutzerhandlungen benötigt werden, sind interaktive Layoutsysteme für Einrichtungsgegenstände. Beispiele für Benutzerhandlungen in diesen Anwendungen sind „Auswählen zur weiteren Bearbeitung“, „Plazieren“, „Verschieben“ usw. In dieser Arbeit wird als Anwendungs- und Leitdomäne die Konzeption eines Programms zur interaktiven Gestaltung der Inneneinrichtung einer Flugzeugkabine betrachtet (genannt XKL). Die Inneneinrichtung einer Flugzeugkabine besteht für diese Anwendung aus räumlichen Objekten wie Passagiersitzen, Flugbegleitersitzen (cabin attendant seats), Küchen (galleys), Waschräumen (lavatories) etc. Abbildung 2 zeigt eine von HAMVIS erzeugte Oberfläche der XKL-Anwendung zur Plazierung von Einrichtungsgegenständen in einer Flugzeugkabine.

In den folgenden Abschnitten möchte ich einen Überblick über die HAMVIS-Architektur geben (siehe Abbildung 3) und die Teilmodelle und Begrifflichkeiten zur Anwendungs- und Visualisierungsmodellierung motivieren.

2.2 Grundmodell und Domänenmodelle

Für die Anwendungsklasse der Layoutprobleme stellt HAMVIS ein Grundmodell bereit, mit dem die Konzepte von relevanten Benutzerhandlungen aus der Bibliothek formal beschrieben sind. Sowohl das Grundmodell als auch die Aktionsmodelle sind durch beschreibungslogische Repräsentationsstrukturen repräsentiert [3, 2]. Im HAMVIS-Ansatz werden anwendungsspezifische Domänenmodelle

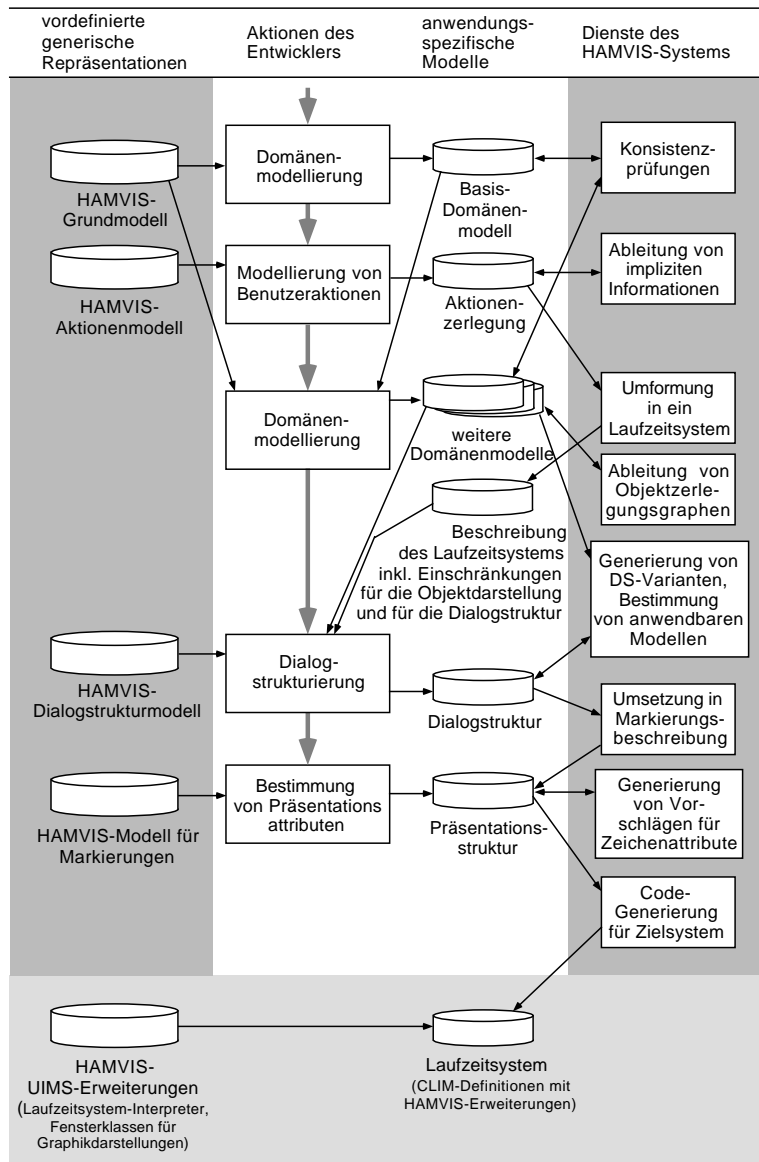


Abbildung3. Überblick über das Gesamtsystem. Vordefinierte Bausteine sind in dunklem Grau dargestellt. Das Ergebnis ist ein Laufzeitsystem für ein UIMS (unten in hellerem Grau).

als Erweiterung des HAMVIS-Grundmodells ebenfalls durch beschreibungslogische Repräsentationsformen definiert (siehe den mittleren Teil der Abbildung 3). Domänenmodelle werden durch das Systementwicklungsteam der Anwendung erstellt.

Im Grundmodell ist z.B. ein Konzept „implizit definierter räumlicher Bereich“ enthalten. Dieses Konzept sieht eine Relation „definiert durch“ zu einem „physikalischen Objekt“ vor, das die Ausdehnung des räumlichen Bereichs bestimmt. Der Bereich ist also in gewisser Weise nur vorhanden, wenn auch das definierende Objekt vorhanden ist. Wenn ein solcher Bereich zu präsentieren ist, so sollte das definierende Objekt ebenfalls dargestellt werden. Ein solcher Bereich könnte in einem Domänenmodell z.B. der Schwenkbereich eines Krans sein. Wenn ein Kranschwenkbereich in einer Visualisierung benötigt wird, so sollte auch der diesen Bereich definierende Kran mit eingezeichnet werden. Für die Krandarstellung selbst ergeben sich Einschränkungen aus der rhetorischen Stellung des Krans als „Rechtfertigung“ für den Schwenkbereich. Der Kran könnte durch blasser Farben abgeschwächt dargestellt werden, um die rhetorische Funktion graphisch umzusetzen.

Weiterhin kann ein räumlicher Bereich scharf abgegrenzt sein oder als räumlicher Bereich mit fließendem Übergang in andere Bereiche modelliert werden. Im ersteren Fall würde bei der Darstellung eine scharfe Abgrenzung etwa durch eine schwarze Linie gewählt werden, während dieses im zweiten Fall nicht erfolgen sollte. Durch Hinzunahme des Konzeptes „scharf abgegrenzter Bereich“ wird ein Schwenkbereich eines Krans entsprechend visualisiert.

Ein Beispiel aus der XKL-Anwendung sind Gangbereiche innerhalb des Flugzeugrumpfes. Es bestünde die Möglichkeit, Freiräume wie z.B. Gänge indirekt durch die Objekte, die die Form definieren (Sitze, Küchen, Waschräume), darzustellen, sofern es die Diskursstellung erlaubt. Für diese Darstellungsform wird im Grundmodell ein Konzept „indirekt darstellbarer Bereich“ vorgesehen. Die indirekte Darstellung von Objekten ist insbesondere dann attraktiv, wenn die Objekte, die das indirekt dargestellte Objekt definieren, aus anderen Gründen sowieso präsentiert werden müssen. Gänge werden ohne die formbestimmenden Objekte allerdings eventuell nicht als Gänge erkannt und müssen ggf. durch Beschriftungen als solche ausgewiesen werden. Diese und andere Darstellungsanforderungen lassen sich durch Verwendung eines entsprechenden Oberkonzeptes aus dem Grundmodell erben.

Zu beachten ist, daß in diesem Entwicklungsszenario wesentliche Gestaltungsmerkmale der Oberfläche und der gezeigten visuellen Darstellungen festgelegt werden, bevor konkrete Domänenobjekte durch Berechnungsfunktionen erzeugt werden. Für die Realisierung der hier betrachteten Lokalisierungshandlung (Abbildung 2) ist es z.B. wichtig, daß die präsentierten Plazierungsbereiche nicht übereinander liegen, da sie in diesem Fall nur sehr schwer ausgewählt werden können. Wenn nun die Planung der Perspektive der Darstellung erst zur Laufzeit durchgeführt wird, so liegt vielleicht zunächst eine Menge von Bereichen vor, die sich auch in der Seitenansicht nicht überlappen. Falls keine anderen Einschränkungen die Seitenansicht ausschließen, so könnte sie gewählt werden.

Wenn sich später doch eine Menge von Plazierungsbereichen ergibt, die sich in der Seitenansicht überlagern, müßte die Darstellungsperspektive geändert werden. Ein Perspektivenwechsel ist jedoch eine entscheidende Änderung der Darstellung und sollte vermieden werden. Es muß zur Entwicklungszeit eine Sicht festgelegt werden, die auch für den ungünstigsten Fall geeignet ist.

HAMVIS erweitert das beschreibungslogische Repräsentationssystem um die Möglichkeit, verschiedene „Sichten“ auf die Objekte der Anwendungswelt zu deklarieren. Eine „Sicht“ umfaßt sowohl eine geometrische Perspektive auf die dreidimensionale Raummodellierung im Grundmodell als auch eine logische Sicht. Mit logischen Repräsentationskonstrukten können in einer Sicht bestimmte Objekte explizit gemacht werden, die in einer anderen Sicht auf die Welt als nicht vorhanden gelten. Eine solche Sicht wird als „Modell“ bezeichnet. Jedem Ausgabefenster der fertigen Anwendung wird ein bestimmtes Modell und damit eine bestimmte „Sicht“ zugeordnet. Mögliche Modelle sind durch die mit einer Darstellung zu unterstützenden Benutzerhandlungen und die jeweils involvierten Domänenobjekte eingeschränkt.

HAMVIS führt explizite Schlußfolgerungen über die zu verwendenden Modelle durch. Durch diese Schlüsse kann dem Entwickler einer Anwendung vermittelt werden, daß die Domänenmodellierung im Laufe der Anwendungsentwicklung aus Gründen der Darstellungsgenerierung noch erweitert werden muß. Dieses kann die Einführung eines neuen Domänenmodells oder die Anpassung eines bestehenden Modells bedeuten. Im Gegensatz zu IMMP-Systemen ([1, 6]) wird also im HAMVIS-Kontext nicht von einem fertigen Domänenmodell ausgegangen.

Im Entwicklungsteam wird die Modellierung der Domäne und die Strukturierung der Anwendung festgelegt. Aktionen, die durch den Benutzer durchgeführt werden sollen, müssen durch automatische Berechnungs- und Speicherfunktionen unterstützt werden, die parallel zur Oberflächenentwicklung durch den HAMVIS-Benutzer von einem Anwendungsentwickler auf der Basis der Vorgaben aus dem Aktionenmodell implementiert werden. In der XKL-Anwendung sollen beispielsweise mögliche Plazierungsbereiche durch Berechnungsfunktionen automatisch bestimmt werden. Für die Definition dieser Funktionen und für die Gestaltung der Oberfläche werden Datenmodelle entworfen und Daten bereitgestellt (z.B. CAD-Daten für die Geometrie des Flugzeugs und der Einrichtungsgegenstände). Für die Implementation von Berechnungs- und Speicherfunktionen können komplexe Werkzeuge eingesetzt werden. Für HAMVIS sind jedoch nur die Typen der Parameter und Werte von Anwendungs- und Berechnungsfunktionen relevant, ansonsten werden sie in dieser Arbeit als Blackbox betrachtet. Das Endprodukt (Laufzeitsystem) wird auf der Basis der anwendungsspezifischen Modelle und einiger anwendungsübergreifender UIMS-Erweiterungen (Bibliothek von Interaktionswerkzeugen für eine Anwendungsklasse) automatisch generiert.

2.3 Modellierung der Aktionenzerlegung für eine Anwendung

Die Aufgabe des Entwicklungsteams besteht insbesondere darin, zur Entwicklungszeit der Anwendung durch Komposition von elementaren Benutzeraktio-

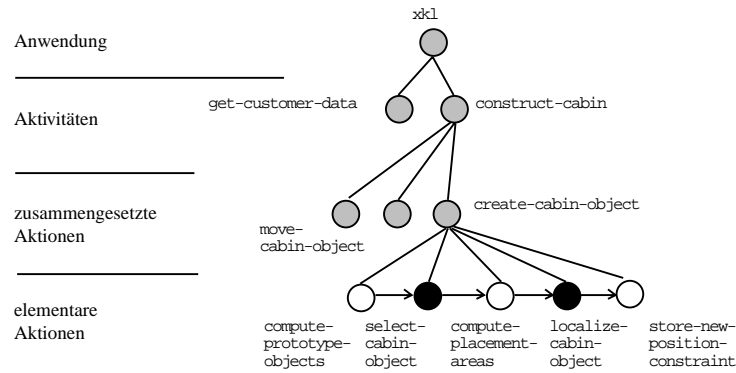


Abbildung 4. Skizze der Aktionenzerlegung zur Modellierung der Struktur der XKL-Anwendung. Benutzeraktionen sind mit schwarzen Kreisen gekennzeichnet. Weiße Kreise markieren automatische Berechnungsfunktionen. Die Abhängigkeiten zwischen Benutzeraktionen und Berechnungsfunktionen sind symbolisch mit Pfeilen angedeutet.

nen und Berechnungs- und Speicherfunktionen Schemata für zusammengesetzte Handlungen zu modellieren. Genauer gesagt, es werden die zur Laufzeit möglichen Aktionen des Benutzers modelliert. Wenn also im HAMVIS-Kontext zur Entwicklungszeit von einer Benutzeraktion gesprochen wird, ist immer eine Handlungsmöglichkeit bzw. ein Aktionenschema gemeint. Damit Ableitungen über den Inhalt von Visualisierungen möglich werden, sind die durch HAMVIS im Grundmodell bereitgestellten Aktionenkonzepte auf einer „tiefen“ Ebene modelliert, z.B.: „Auswahl zur weiteren Bearbeitung“, „Lokalisierung eines räumlichen Objekts“ usw. Die Aktionenkonzepte von HAMVIS modellieren in der Aktionenzerlegung sog. elementare Benutzeraktionen. Abbildung 4 vermittelt ein Beispiel für eine Aktionenzerlegung der hier als Leitbeispiel betrachteten XKL-Anwendung.

Der XKL-Benutzer wählt ein Prototypobjekt aus. Die möglichen Plazierungsbereiche für ein Prototypobjekt werden daraufhin automatisch berechnet. Der Benutzer wiederum lokalisiert anschließend das neue Kabinenobjekt durch Plazierung in einem der möglichen Plazierungsbereiche. Die Plazierungsinformationen müssen dann im Datenmodell abgespeichert werden. Die Typen der Parameter und Werte der Berechnungs- und Speicherfunktionen werden durch die vorgesehenen elementaren Benutzeraktionen eingeschränkt. HAMVIS führt hierzu Typableitungen durch.

Für zusammengesetzte Handlungen versucht HAMVIS, aufgrund der Konzepte der Benutzerhandlungen eine oberflächennahe Realisierung zu bestimmen. In dem Beispiel aus Abbildung 4 ist eine Ziehen-und-Fallenlassen-Geste adäquat (siehe Abbildung 2). Durch eine solche Realisierung einer zusammengesetzten Handlung mit einer einzigen Mausgeste sind für den Benutzer die Berechnungsfunktionen transparent. Man kann also auf höherer Ebene auch bei einer zusam-

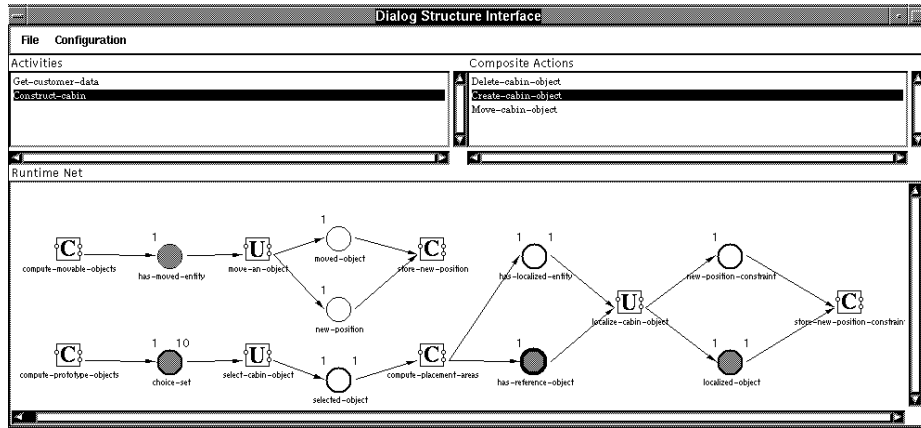


Abbildung 5. Darstellung der Abhängigkeiten zwischen Benutzeraktionen (U-Transitionen) und assoziierten Berechnungs- und Speicherfunktionen (C-Transitionen), die jedoch aus Sicht des Benutzers der endgültigen Anwendung transparent sind.

mengesetzten Handlung von einer Benutzeraktion sprechen. In dem Beispiel aus Abbildung 2 wird die zur Laufzeit verwendete Oberfläche für eine zusammengesetzte Handlung zur Erzeugung eines neuen Einrichtungsgegenstandes gezeigt.

Zusammengesetzte Handlungen stellen Handlungsalternativen dar. Im XKL-Beispiel könnte z.B. eine andere zusammengesetzte Handlung die Möglichkeit zur Verschiebung eines Kabinenobjekts durch den XKL-Benutzer modellieren. Mehrere Handlungsalternativen können durch das Entwicklungsteam zu einer sogenannten Aktivität zusammengefaßt werden, wobei für jede Aktivität durch HAMVIS ein separates Interaktionsfenster vorgesehen wird. Mehrere Aktivitäten bilden eine Applikation. Durch Definition einer Aktionenzerlegung legt das Entwicklungsteam die Struktur der Anwendung fest. HAMVIS stellt eine graphische Oberfläche bereit, mit der eine Aktionenmodellierung und -zerlegung durchgeführt werden kann (siehe Abbildung 5). Wichtig ist, daß die Zerlegung nicht willkürlich vorgenommen wird. Die Bedeutung jeder der vier Ebenen in Hinblick auf die Gestaltung der Anwendung bleibt klar.

Durch eine anwendungsspezifische Aktionenzerlegung wird durch den Entwickler festgelegt, wann welche Berechnungs- oder Speicherfunktion evaluiert werden muß und wie das, was zur Laufzeit darzustellen ist, ermittelt wird. Die Aktionenzerlegung wird zur Modellierung des Laufzeitsystems als Petri-Netz repräsentiert (siehe Abbildung 5). Transitionen stehen für Berechnungs- und Speicherfunktionen sowie auch für Benutzeraktionen. Die Stellen in diesem Netz repräsentieren „Platzhalter“ für die zur Laufzeit im jeweiligen Interaktionszyklus (Durchgang durch das Petri-Netz) berechneten Domänenobjekte. Die im Beispiel aus Abbildung 5 eingezeichneten grauen Stellen enthalten zu kommunizierende Objekte.

Aus einer Aktionenmodellierung werden durch formale Verfahren Einschränkungen für die Darstellung von Objekten abgeleitet. Die dem Entwicklungsteam von HAMVIS zur Verfügung gestellten Werkzeuge sind so gestaltet, daß durch HAMVIS aus einer Aktionendekomposition unter Zuhilfenahme eines UIMS automatisch ein Laufzeitsystem erstellt werden kann, wobei jedoch die Modellierung von Aktionen auf einem hohen Abstraktionsniveau erfolgen kann. Durch das Laufzeitsystem wird zur Benutzungszeit der Anwendung die Verwaltung und Darstellung der Daten übernommen. Das Wissen zur Abbildung auf UIMS-Dienste, so daß Benutzeraktionen zur Laufzeit durchführbar sind, wird durch die Konzepte der elementaren Benutzeraktionen repräsentiert. Mögliche Interaktionsformen zur Umsetzung einer Aktion legen weitere Einschränkungen für die Gestaltung der hierfür benötigten Visualisierungen fest.

2.4 Dialogstrukturierung

Aktionen und deren Konzepte definieren, was für die Ausführung der Aktionen in einer Visualisierung mindestens dargestellt werden muß. Es muß noch bestimmt werden, in welchem Teilfenster einer Anwendung und in welchem Kontext die (zur Laufzeit in den Stellen auftretenden) Objekte gezeichnet werden sollen. Damit für den Anwendungsbenutzer zur Laufzeit eine adäquate Interaktionsform erzielt werden kann, müssen ggf. noch weitere Objekte kommuniziert werden. HAMVIS stellt Modelle für die in der Anwendungsklasse benötigten „Kommunikationsschemata“ und Dialogstrukturen bereit. In dem betrachteten Beispiel aus Abbildung 2 wird z.B. für die Lokalisierungshandlung als räumliches Referenzsystem der Flugzeugrumpf mit Landmarken wie Türen, Cockpitfenster usw. benötigt. Die Aufgabe von HAMVIS besteht auch darin, zur Entwicklungszeit durch formale Schlußfolgerungsprozesse sicherzustellen, daß die erforderlichen Informationen zur Laufzeit aus den Domänenmodellen, die durch das Systementwicklungsteam bereitgestellt werden, abgeleitet werden können.

HAMVIS gestattet dem Entwickler, die Oberflächengestaltung schon zur Entwicklungszeit aus der Kommunikationsperspektive zu betrachten. Damit der Anwendungsbenutzer zur Laufzeit agieren kann, werden ihm die hierzu notwendigen Informationen präsentiert. Wissen über kohärente Kommunikationsformen in der hier betrachteten Anwendungsklasse wird durch HAMVIS repräsentiert, so daß eine adäquate Dialogstruktur aufgebaut werden kann. In einer Dialogstruktur werden die Komponenten der Oberfläche, die Beziehungen zwischen Komponenten sowie die für die Erstellung eines Laufzeitsystems notwendigen Informationen repräsentiert und verwaltet.

Zum Aufbau eines Dialogmodells stellt HAMVIS für den Designer eine weitere interaktive Oberfläche bereit. Mit dieser Oberfläche werden dem Oberflächenentwickler die zur Laufzeit darzustellenden „Informationseinheiten“ (Mengen von Objekten in den Stellen des Petri-Netzes) der Anwendung präsentiert, die aus der vorher erstellten Aktionendekomposition abgeleitet wurden. Nach und nach kann er die Einheiten in das Dialogmodell integrieren (siehe die Stelle „has-reference-object“ bei der Transition „localize-cabin-object“ in Abbildung 5). Ein Beispiel für eine solche Einheit ist die Menge der Plazierungsbereiche, die zur Lo-

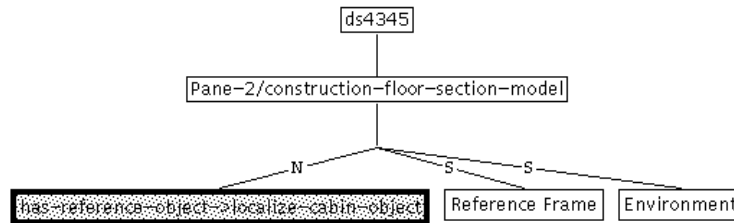


Abbildung6. Schnappschuß der Dialogstruktur für die XKL-Anwendung. Für die Darstellung von Referenzobjekten – in diesem Beispiel handelt es sich um die Platzierungsbereiche – wird sowohl eine Darstellung des Referenzsystems als auch eine Darstellung der Objekte in der Umgebung der Referenzobjekte benötigt.

kalisierung eines Kabinenobjektes möglich sind. Bei der Integration einer Informationseinheit bestimmt HAMVIS, welche Zusatzinformationen (Referenzsysteme, Landmarken) für die Darstellung benötigt werden bzw. wünschenswert sind und welche Präsentationseinschränkungen hierfür „in Kauf genommen“ werden müssen. In unserem XKL-Beispiel werden die für die Platzierung notwendigen Bestandteile in Abbildung 6 verdeutlicht.

Jeder Bestandteil der Dialogstruktur erfüllt innerhalb des Interaktionszyklus⁷ einen bestimmten Zweck in der Gesamtdarstellung (Diskurszweck). HAMVIS modelliert diese Diskurszwecke explizit durch entsprechende Konzepte und leitet aus diesen Angaben später Darstellungsformen und -attribute ab. Je nach Diskurszweck werden Hauptobjekte (Nuklei) von zusätzlichen Objekten (Satelliten) unterschieden. Zur Sicherstellung einer kohärenten Darstellung wird jeder Dialogstruktureinheit noch eine sog. Diskursposition zugeordnet. Durch die Diskursposition wird repräsentiert, ob durch das Laufzeitsystem ein Objekt vor der Ausführung von Aktionen gezeichnet werden muß (z.B. das Referenzsystem) oder ob ein Objekt, nachdem es einmal gezeichnet wurde, sofort nach der Aktion wieder gelöscht werden muß (z.B. die Platzierungsbereiche bei der Lokalisierungsaktion) oder permanent gezeichnet werden muß (z.B. das lokalisierte Objekt).

Für jede Dialogstruktureinheit ist bekannt, für welche Aktion sie benötigt wird. Wenn für die eine Aktion eine Seitenansicht angemessen ist und für eine andere Aktion eine Sicht von oben benötigt wird, so werden hierfür zur Entwicklungszeit zwei verschiedene Teilvisualisierungen im Dialogmodell „eingeplant“. Zur Laufzeit werden in getrennten Unterfenstern jeweils Darstellungen mit verschiedener Perspektive gezeigt. Es kann allerdings auch der Fall sein, daß eine einzige Visualisierung geeignet ist, mehrere Aktionen zu unterstützen (z.B. die Lokalisierung von Kabinenobjekten und auch deren Verschiebung). Es ergeben sich eventuell mehrere Varianten zur Dialogstrukturierung, die von HAMVIS bestimmt und verwaltet werden (siehe Abbildung 7 für ein Beispiel aus der XKL-Anwendung). Die von HAMVIS bereitgestellte Entwicklungsoberfläche für die Organisation der Dialogstruktur dient dem Oberflächenentwickler dazu, mögliche Varianten zu bewerten und ggf. zu verwerfen. Es wird durch das Dialogmo-

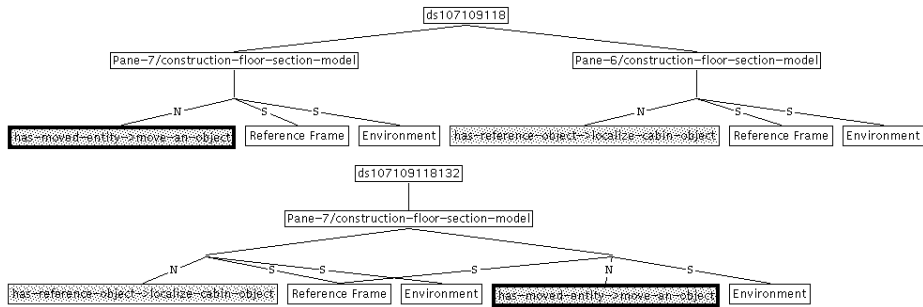


Abbildung 7. Darstellung von Varianten in der Dialogstruktur. In der unteren Variante wird in einem Fenster (Pane-7) sowohl eine Platzierungs- als auch eine Verschiebeaktion durchgeführt. Für die Oberfläche aus Abbildung 2 wurde diese Variante gewählt.

dell weiterhin festgelegt, welches Domänenwissen zu Darstellungszwecken vom Entwicklungsteam zu repräsentieren ist und wie Domänenobjekte strukturiert werden müssen, damit die für die Durchführung der Aktionen benötigten Visualisierungen generiert werden können.

2.5 Präsentationsstrukturierung und Codegenerierung

Die in der Dialogstruktur vermerkten Einheiten haben jeweils eine bestimmte kommunikative Funktion (Diskurszweck) und stehen in bestimmten Beziehungen zueinander (Hauptobjekte, Zusatzobjekte, manipulierbare Objekte). Durch diese Charakterisierungen ergeben sich (weitere) Einschränkungen für die Darstellungsform und die zur Darstellung verwendeten Zeichenattribute wie z.B. Farbe, Strichdicke usw. Verschiedene Informationseinheiten in der Dialogstruktur, die durch gleiche Darstellungscharakterisierungen ausgezeichnet sind, sollten mit gleichen Zeichenattributen dargestellt werden. HAMVIS überträgt hier die Idee der Markierungen (markups), die im Bereich der Textsatzsysteme für die Beschreibung bzw. die Charakterisierung von Textteilen entwickelt wurde, auf den Graphikbereich. Ähnlich wie in Textverarbeitungssystemen eine Markierung z.B. „Section“ vergeben wird, sieht HAMVIS eine Markierung „Referenzrahmen“ für bestimmte Objekte vor. Zu beachten ist, daß der Markierungstyp für eine Informationseinheit nicht vom Konzept der beschriebenen Objekte abhängt, sondern sich aus der kommunikativen Stellung in der Dialogstruktur ergibt.

Für jeden Markierungstyp müssen geeignete Zeichenattribute bestimmt werden. Die Zeichenattribute für verschiedene Markierungen sind jedoch nicht voneinander unabhängig. Wenn der Hintergrund weiß ist, so können zur Darstellung von wichtigen Objekten dunkle Farben, von Zusatzobjekten hellere Farben gewählt werden. Ist der Hintergrund jedoch schwarz, lassen sich dunkle Farben kaum noch verwenden usw. Aus den im Dialogmodell eingetragenen Informationen über Darstellungseinschränkungen und kommunikative Funktionen von Dialogstruktureinheiten berechnet HAMVIS geeignete Standard-Zeichenattribute

(Defaults). Die Standardwerte können durch den HAMVIS-Benutzer aber noch nach seinen Wünschen angepaßt werden. Durch den Mechanismus der Markierung braucht aber die Änderung nur an einer Stelle vorgenommen zu werden. Über das Konzept der Markierung wird also – ähnlich wie bei Textsatzsystemen – auch in Graphiken eine konsistente Verwendung von Zeichenattributen in der gesamten Anwendung gefördert.

Zusammen mit den auf der Basis der Spezifikation in der Aktionenzerlegung implementierten Berechnungs- und Speicherfunktionen und einem UIMS kann aus der Präsentation ein Laufzeitsystem generiert werden, so daß sich der Oberflächenentwickler nicht mit den Details der Programmierung von Fensterschnittstellen beschäftigen muß.

3 Zusammenfassung

Die wissenschaftlichen Ergebnisse möchte ich wie folgt zusammenfassen. Die wesentlichen Teilprobleme bei der Entwicklung eines Unterstützungssystems zur Visualisierungsgenerierung für die im Rahmen von HAMVIS Lösungsmöglichkeiten erarbeitet wurden, sind:

- Repräsentation von Wissen über grundlegende Konzepte und Relationen für eine Anwendungsklasse in einer Form, die es erlaubt, die Erzeugung und Ausgestaltung von Visualisierungen als formale Inferenzschritte deklarativ zu beschreiben,
- Bereitstellung von Möglichkeiten zur Strukturierung einer Applikation in Teilphasen oder „Aktivitäten“, die jeweils zur Ausführung von mehreren zusammengehörenden Benutzeraktionen gedacht sind und an der Benutzungsoberfläche des Rechners jeweils durch ein Fenster repräsentiert werden.
- Schaffung von Repräsentationsformen zur Definition von Beziehungen und Abhängigkeiten zwischen Benutzeraktionen und automatischen Berechnungs- und Speicherfunktionen („Aktionenzerlegung“), so daß an der Oberfläche als Kombination von exekutiven und impliziten deskriptiven Interaktionsformen die Illusion einer direkten Manipulation entstehen kann,
- Formalisierung von Schlußfolgerungsprozessen, so daß zur Entwicklungszeit einer Anwendung Visualisierungen hergeleitet und Visualisierungskombinationen vorgeschlagen werden können, mit denen Benutzeraktionen interaktiv zur Laufzeit durchführbar sind (notwendige Perspektive, adäquater Detailreichtum, usw.),
- Bereitstellung von Repräsentationsformen zur Bestimmung von Zeichenattributen und Darstellungsformen für graphische Objekte (z.B. Techniken zur Hervorhebung),
- Analyse der Möglichkeiten zur Komposition von Visualisierungen mit Standardinteraktionsbausteinen zur Einbettung in ein Fenster der Benutzungsschnittstelle und automatische Codegenerierung für ein Wirtssystem.

HAMVIS verbindet bzw. erweitert Arbeiten aus den Bereichen Wissensrepräsentation/KI, KI-Softwaretechnik, Mensch-Computer-Interaktion, Intelligen-

te Multimedia-Präsentationssysteme sowie Entwicklungsumgebungen für Benutzungsschnittstellen und zeigt die Querbezüge auf. Mit dieser Arbeit wird zum ersten Male deutlich herausgearbeitet, daß wesentliche Gestaltungsmerkmale der Anwendung schon zur Entwicklungszeit einer Anwendung unter Einbeziehung der verschiedenen Einschränkungen (Abbildung 1) systematisch festgelegt werden können, so daß ein effizientes Laufzeitsystem generiert werden kann. Da zur Entwicklungszeit konkrete Objekte nicht in jedem Fall verfügbar sind, werden konzeptuelle Informationen verwaltet und im Rahmen von Inferenzverfahren verarbeitet. Die Arbeit zeigt, daß mit Hilfe von Beschreibungslogiken und einiger Erweiterungen die notwendigen Ableitungen zur Visualisierungsgestaltung formalisiert werden können.

Die praktische Bedeutung dieser eher theoretischen Perspektive der Modellierung mit Beschreibungslogiken sollte nicht unterschätzt werden. Auch im Bereich der Produktmodellierung ist in den nächsten Jahren eine Formalisierung der Repräsentationstrukturen zu erwarten (z.B. bei STEP). Wenn also wie beim wissensbasierten Ansatz bei entsprechender ontologischer Modellbildung wesentliche Teilprobleme durch automatische, und damit getestete und theoretisch abgesicherte Inferenzverfahren durchgeführt werden, lassen sich sicherere Softwaresysteme mit weniger Kosten erstellen. Obwohl also die beschreibungslogische Modellierung zur Zeit noch wenig Verbreitung findet, sind doch die Erkenntnisse des HAMVIS-Systems auf zukünftige Modellierungssysteme mit formaler Semantik übertragbar. Insbesondere in großen Anwendungssystemen wird durch HAMVIS die Entwicklung von kohärenten Visualisierungen für Benutzungsoberflächen unterstützt. Die Gestaltung der Oberflächen kann aus der Design- und Nutzungsperspektive zur Entwicklungszeit bewertet werden. Eine Evaluierung des Ansatzes erfolgte durch eine Prototypimplementation, die sich schon in vielen Demonstrationen bewährt hat.

Die Flexibilität, die adaptive Systeme wie z.B. IMMPS bereitstellen, wird für interaktive Oberflächen nicht in jedem Fall benötigt. Die für Anwendungen wie z.B. XKL zu entwerfenden Oberflächen und Visualisierungen sind für ganz bestimmte Aufgaben vorgesehen. Die zu unterstützenden Benutzeraktionen sind durch die Aktionenzerlegung a priori bekannt. Daher ist es sinnvoll, die in der Oberfläche zu präsentierenden Visualisierungen in einer vorangehenden Planungsphase genau auf die Erfordernisse der Anwendung abzustimmen. Hierdurch kann auch der Berechnungsaufwand zur Benutzungszeit der Oberfläche minimal gehalten werden. Eine A-priori-Bewertung einer Oberfläche wird dadurch ebenfalls ermöglicht. So scheint es z.B. für Leitstände in Kraftwerken oder Kontrolleinrichtungen für technische Prozesse günstig zu sein, bei Störfällen die Informationen möglichst gut auf den jeweiligen Systemzustand abzustimmen (Redundanzvermeidung) und weiterhin möglichst schnell Informationen auch unvollständig zu präsentieren (Reaktivität). Andererseits ist die situative Informationsdarstellung vielleicht für den Operateur ungewohnt, und in einer Gefahrensituation können Darstellungen von Teilinformationen zu vorschneellen Entscheidungen führen. Für sicherheitsrelevante Anwendungen scheinen mir IMMPS-Ansätze nicht unproblematisch. Eine Überprüfungsmöglichkeit der Ge-

samtapplikation für einen menschlichen Systemdesigner ist hier notwendig. In diesem Kontext ist die von HAMVIS vorgenommene Trennung von Entwicklungszeit und Laufzeit sehr vorteilhaft.

Literatur

1. E. André. Ein planbasierter Ansatz zur Generierung multimodaler Präsentationen. Dissertation, Universität Saarbrücken, Infix-Verlag, 1995.
2. A. Borgida, P.F. Patel-Schneider. A Semantics and Complete Algorithm for Subsumption in the CLASSIC Description Logic. *Journal of Artificial Intelligence Research*, No. 1, Morgan-Kaufmann Publ., 1994, pp. 277–308.
3. R.J. Brachman, D.L. MacGuinness, P.F. Patel-Schneider, L.A. Resnick, A. Borgida. Living with CLASSIC: When and How to Use a KL-ONE-Like Language. In: *Principles of Semantic Networks*, J. Sowa (ed.), Morgan-Kaufmann Publ., pp. 401–456.
4. U. Gappa. Graphische Wissensakquisitionssysteme und ihre Generierung. Dissertation, Universität Karlsruhe, Infix-Verlag, 1995.
5. D. Holz. Über das Entwerfen von Gebrauchssoftware. Dissertation, Universität Basel, 1995.
6. Th. Rist. Wissensbasierte Verfahren für den automatischen Entwurf von Gebrauchsgraphiken in der technischen Dokumentation. Dissertation, Universität Saarbrücken, Infix-Verlag, 1996.