

# Multimodale Recherchezugänge: Neue Wege bei der Konzeption der integrierten Informationssysteme ELVIRA und GESINE

Maximilian Eibl, Maximilian Stempfhuber  
Informationszentrum Sozialwissenschaften, Berlin, Bonn

## Zusammenfassung

Im vorliegenden Artikel wird die Konzeption der Retrieval-Systeme ELVIRA und GESINE beschrieben. Hierbei liegt neben der Beschreibung der verschiedenen Recherchezugänge der Fokus auf der Konzeption nicht-traditioneller Zugänge. Ein solcher Zugang besteht in einer Visualisierung, die im Rahmen einer Kooperation von Softwareergonomie und Graphik Design am Informationszentrum Sozialwissenschaften, Bonn, geschaffen wurde und ergonomischen wie ästhetischen Ansprüchen genügt. Ein anderer Zugang ist ein tabellenbasierter Visual Formalism, der speziell für die gleichzeitige Recherche in heterogenen Datenbeständen entworfen wurde.

## 1 Einleitung

Die am Informationszentrum Sozialwissenschaften (Bonn) entwickelten Informationssysteme ELVIRA (Elektronisches Verbandsinformations-, Recherche- und Analysesystem) und GESINE (GESIS-Datenbestände integrierendes Informationssystem) stellen zwei komplexe Informationssysteme dar, die nach dem WOB-Modell (auf der Werkzeugmetapher basierenden strikt objektorientierten grafisch-direktmanipulative Benutzungsoberflächen) gestaltet sind. Aus informationswissenschaftlicher Sicht integrieren sie das Boolesche und quantitativ-statistische Retrieval-Modelle, aus softwareergonomischer Sicht Formularzugang, UND/ODER-Gitter und Freitextsuche. Der vorliegende Artikel beschreibt die Erweiterung der Systeme um eine Visualisierung für das Dokument Retrieval und einen Visual Formalism für heterogene Faktenbestände, sowie deren konzeptuelle Integration in das WOB-Modell.

## 2 Softwareergonomische Grundlage: Das WOB-Modell

Das WOB-Modell (Krause 1997) stellt ein Mittelmodell dar, das die Lücke zwischen den konkreten, im Einzelfall aber oft widersprüchlichen Gestaltungsanweisungen von Styleguides und den in der Regel schwer operationalisierbaren kognitionspsychologischen Erkenntnissen und Theorien schließt. Dynamische Anpassung, kontext-sensitive Durchlässigkeit, eine modifizierbare Zustandsanzeige und der gezielte Einsatz von Metaphern und Visual Formalisms stellen wichtige Elemente des WOB-Modells dar und sollen im Ergebnis zu »natürlichen« Benutzungsoberflächen führen.

### 2.1 Metaphern und Visual Formalisms

Mittlerweile untrennbar mit graphischen Benutzungsoberflächen verbunden sind *Metaphern*. Gerade der Siegeszug der Desktop-Metapher gab den Startschuss für ungehemmtes metaphernbasiertes Design. Vergleicht man die verschiedenen Gestaltungsmöglichkeiten für Benutzungsoberflächen, so erkennt man: Je bildlicher und weniger textuell die Gestaltung, desto eher wird auf die Erklärungsfähigkeit von Metaphern gesetzt. Ziel der Metapher ist, dem Anwender in

komplexen Situationen unauffällig Hilfestellung bei der Orientierung und Handlung zu geben, indem Analogien zu bereits bekannten Situationen hergestellt werden. Dieses Verfahren ist jedoch nicht ganz unproblematisch: »The direct consequence of this for teaching people computer systems is that the metaphors they invoke will drastically affect the success with which they are able to learn and use the system.« (Carroll&Thomas 1982, 110) Gerade die Übertragung der Semantik einer Domäne in eine andere funktioniert dabei kaum ohne Bruch der Metapher, was regelmäßig zu Interaktionsproblemen führt.

Als Alternative zu Metaphern hat sich mittlerweile der Gebrauch von *Visual Formalisms* etabliert. Johnson et al. 1993, 44 definieren sie wie folgt: »Visual Formalisms are diagrammatic displays with well-defined semantics for expressing relations. Examples of commonly used Visual Formalisms are tables, graphs, plots, panels, maps, and outlines.« Sie basieren also auf den jeweiligen Stärken der Interaktionspartner Mensch und Maschine und nutzen sowohl das visuelle Potential des Menschen als auch das analytische des Computers. Visual Formalisms setzen da ein, wo Metaphern zu Unzulänglichkeiten führen: bei der Darstellung komplexer Informationen und Beziehungen. Ihr Vorteil, eine eigene wohldefinierte Semantik aufzubauen, ist allerdings nicht so universal, wie etwa Nardi&Zarmer 1993 behaupten, sondern muss stets auf die spezielle Anwendungssituation hin untersucht werden. Generell muss gelten: »Gute Gestaltung kann sowohl metaphorische Elemente enthalten als auch photographische oder die *visual formalisms* im Sinne von Nardi/Zarmer 1993« (Krause 1996, 20).

## 2.2 Dynamische Anpassung und kontextsensitive Durchlässigkeit

Dynamische Anpassung und kontextsensitive Durchlässigkeit haben die Verringerung redundanter und falscher Eingaben und eine effiziente Verwendung von Bildschirmplatz zum Ziel. Die Differenzierung beider Prinzipien, zusammen mit der modifizierbaren Zustandsanzeige (siehe unten), erlaubt im Gegensatz zur allgemeineren »engen Kopplung« (tight coupling) von Oberflächenelementen (Ahlberg&Shneiderman 1994) eine formale Beschreibung durch Softwareentwurfsmuster, die Entscheidungskriterien für die Wahl der Prinzipien und Richtlinien für deren konsistente Umsetzung an die Hand geben (Stempfhuber 2001).

Bei an mehreren Stellen der Benutzungsoberfläche benötigten Eingaben oder Informationen sorgt die *kontextsensitive Durchlässigkeit* durch eine *enge Kopplung* zwischen Information und Visualisierung für Konsistenz zwischen den unterschiedlichen Sichten auf die Informationseinheit. Die Darstellung der Information kann dabei durch deduktive Prozesse beeinflusst werden (z.B. alphanumerische vs. grafische Darstellung) und sich unterschiedlicher Interaktionsmodi bedienen (z.B. Tastatur vs. Direktmanipulation).

Durch die *dynamische Anpassung* werden Oberflächenelemente zu einem *losen Verbund* zusammengefasst, in dem Zustandsänderungen eines Elements (z.B. Inhalt oder Größe) zwar kommuniziert werden, die übrigen Elemente jedoch nicht zwangsläufig darauf reagieren müssen. Typische Anwendungsbeispiele sind die Koordinierung des Inhalts von Auswahllisten, das kontext-abhängige Verdecken oder Anzeigen von Oberflächenelementen oder die dynamische Änderung des Bildschirmlayouts zur optimierten Platzausnutzung. Eine Nicht-Reaktion auf Zustandsänderungen darf das mentale Modell des Benutzers nicht stören, sondern muss innerhalb des generellen Prinzips erklärbar und damit kompatibel sein. Die Größenbeschränkung eines Fensters (Maximalgröße) kann zum Beispiel ein nachvollziehbarer Grund sein, warum dieses ab einem bestimmten Zeitpunkt nicht mehr wächst.

In ELVIRA (Krause&Stempfhuber 2001) dient die dynamische Anpassung zur Optimierung der Platzausnutzung bei der Anfrageformulierung und zur Vermeidung von Null-Antworten bei der Suche. Die kontext-sensitive Durchlässigkeit bewirkt eine Entlastung des Kurzzeitgedächtnisses während der Anfrageformulierung und der Analyse des Suchergebnisses.

Der Suchbildschirm besteht im Grundzustand (Abb. 1a) aus drei Zustandsanzeigen (oben) und damit verknüpften Auswahllisten (unten). Jede der drei Facetten »Themen«, »Branchen / Pro-

dukte« und »Länder« enthält zwei Oberflächenelemente, die interagieren (die Auswahlliste für »Länder« ist aus Platzgründen verdeckt). Die Selektion eines Eintrags in der Facette »Themen« wirkt sich modifizierend auf die Inhalte der übrigen Facetten aus, so dass in der Facette »Branchen / Produkte« nur noch die Nomenklaturen sichtbar sind, die mit dem selektierten Eintrag aus »Themen« kombiniert werden können (Abb. 1b). Gleichzeitig wurden alle selektierten Einträge in die entsprechenden Zustandsanzeigen übernommen. Der dafür benötigte Bildschirmplatz wurde durch Verkleinerung der Vorlagelisten gewonnen. Vorlageleistung und Zustandsanzeige werden *dynamisch* gegeneinander getauscht, so daß der Anfänger (recall vs. recognition) und der Fortgeschrittene (Entlastung des Kurzzeitgedächtnisses bei umfangreichen Anfragen) optimal unterstützt werden.

Damit Zustandsanzeige und zugehörige Auswahlliste zu jedem Zeitpunkt einen konsistenten Inhalt aufweisen, werden sie *eng* miteinander verbunden. Der Selektionszustand in der Auswahlliste wirkt sich zwangsläufig sofort auf die Zustandsanzeige aus. Beide Oberflächenelemente bilden den internen Systemzustand (formulierte Anfrage bzw. selektierte Suchbegriffe) konsistent ab und können in Kombination (Suchbildschirm) oder alleine (Ergebnisbildschirm ersetzt Auswahllisten durch Ergebnisliste) verwendet werden.

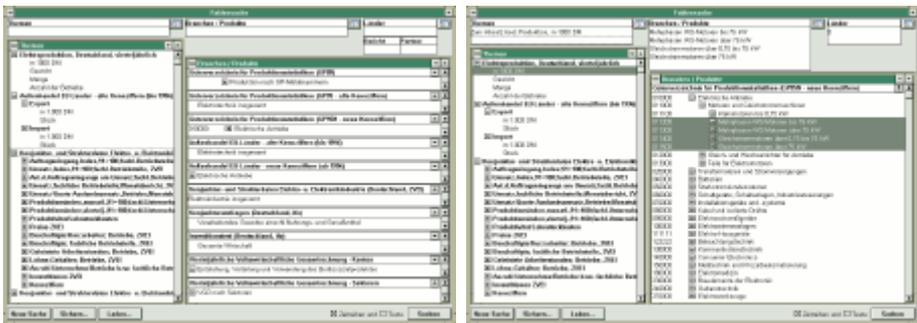


Abb. 1a&b: ELVIRA Faktenrecherche im Grundzustand und dynamisch angepasst

### 2.3 Modifizierbare Zustandsanzeige

Die Zustandsanzeige im WOB-Modell dient nicht nur der Entlastung des Kurzzeitgedächtnisses, sondern sie kompensiert ihren Platzbedarf auf der Oberfläche durch zusätzliche Funktionalität. Ähnlich den Hypertext Links, die gleichzeitig Ausgabe- (lesbarer Text) und Eingabecharakter (Klick verzweigt auf eine neue Seite) besitzen, stellt die Zustandsanzeige Informationen dar und erlaubt deren Löschung und Neueingabe (output-is-input-Prinzip). Hierzu enthält sie eine Leerzeile, die Eingaben über die Tastatur ermöglicht und diese (eventuell mit zusätzlichen Hilfsmitteln wie einem Thesaurus oder einer Rechtschreibprüfung) auf die zugehörige Auswahlliste abbildet. In ELVIRA können über die Zustandsanzeige auch sogenannte Warennummern oder Kennziffern eingegeben werden, die es dem fortgeschrittenen Benutzer sehr effizient ermöglichen, häufig benötigte Einträge in den teilweise über 1.800 Produkte umfassenden Nomenklaturen zu selektieren.

### 2.4 Zusammenfassung

Das WOB-Modell reagiert durch dynamische Anpassung, kontext-sensitive Durchlässigkeit, modifizierbarer Zustandsanzeige und der Verwendung von Metaphern und Visual Formalisms auf die Heterogenität der Anwender. Während Anfänger ein System mit hoher Selbsterklärungs-

fähigkeit benötigen, fordern fortgeschrittene Anwender eine konzise Darstellung, die schnelle und effiziente Arbeit zulässt. Die Differenzierung zwischen Anfängern und Fortgeschrittenen ist jedoch fließend, so dass eine Umsetzung in zwei strikt unterschiedliche Systemoberflächen wenig sinnvoll ist. Dieser Konflikt wird im WOB-Modell gelöst und kann zum Beispiel durch konsequente Ausnutzung von Domänenrestriktionen in vielen Anwendungsfällen zu benutzerfreundlichen Ein-Bildschirm-Systemen führen.

### 3 Integration von Softwareergonomie und Medien Design

Während Softwareergonomie den Fokus auf eine effektive und effiziente Benutzung von Benutzungsoberflächen legt, stehen beim Graphik Design ästhetische Aspekte im Vordergrund. Resultat sind stark unterschiedliche Benutzungsoberflächen, die entweder einfach zu bedienen oder ästhetisch ansprechend sind - selten aber beides. Ästhetik fand in der Softwareergonomie bislang wenig Resonanz, obwohl ästhetische Benutzungsoberflächen aus ergonomischer Sicht eine erwünschte psychologische Folge haben: Ein Anwender, für den der Computer nicht mehr graue technifizierte Langeweile ist, lernt in der Regel schneller und besser mit einem Programm arbeiten. Die Zusammenführung beider Ansätze auf theoretischer und praktischer Ebene wird in Eibl 2000 versucht. Hier wird vorgeschlagen, Elemente der Theorie der Produktsprache in den Katalog des WOB-Modells aufzunehmen.

Die Theorie der Produktsprache wurde in den siebziger Jahren an der Hochschule für Gestaltung (Offenbach) als Antwort auf die Funktionalismuskritik entwickelt. Der Funktionalismus

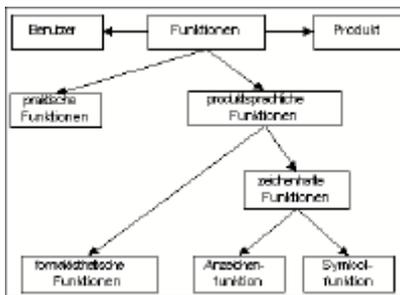


Abb. 2: Produktsprache (Gros 1983, Abb. 6)

war in den sechziger Jahren vor allem aufgrund seiner Negierung von Zeichenhaftem im Design in Kritik geraten. Die Theorie der Produktsprache versucht die Zeichenhaftigkeit von Design auszudrücken und unterscheidet zwei Funktionen (Abb. 2). *Praktische Funktionen* beziehen sich auf die physische Produktwirkung, Produktsprache die psychische. So ist die praktische Funktion einer Türklinke die, mit ihr eine Tür öffnen zu können. Ihre *produktsprachliche Funktion* ist erkennbar zu machen, dass und wie mit ihr eine Tür geöffnet werden kann.

Analog zur natürlichen Sprache wird auch bei der Produktsprache zwischen Syntax und Semantik unterschieden. Der Syntax entsprechen dabei formalästhetische Funktionen, die weitgehend durch die Gesetze der Gestalttheorie beschrieben werden. Diese zeigen rein formal und ohne jegliche Bedeutungsbildung, wie einzelne Elemente durch ihre Positionierung, Form- und Farbgebung wirken können.

Daneben wird die Semantik durch die zeichenhaften Funktionen ausgedrückt. Diese wiederum lassen sich in *Anzeichenfunktionen* und *Symbolfunktionen* unterteilen. Anzeichenfunktionen beziehen sich dabei immer auf den Gegenstand selbst, verdeutlichen beispielsweise seine Funktionsweise, geben Auskunft über seine Stabilität, zeigen seine Bedienungsmöglichkeiten, verhindern Fehlbedienung. In oben genannten Klinkenbeispiel verdeutlichen sie die Funktionsweise der Klinke.

Klinken können aber noch mehr Auskunft geben als nur über ihre Funktionalität. So werden in Büroräumen andere Klinkentypen zu finden sein als in Luxushotels. Eine goldverzierte geschwungene Klinke an der Bürotüre im Arbeitsamt würde sofort als unpassend ins Auge stechen. Hier kommen die *Symbolfunktionen* zu tragen. Sie erlauben es Produkten, weitergehende Ideen und Zusammenhänge zu transportieren. Symbolfunktionen übernehmen die Einordnung des Gegenstandes in das kulturelle Umfeld. Sie liefern Hintergrundinformationen zu einem Produkt.

Anhand der Symbolfunktionen ist ein Produkt beispielsweise dem asiatischen oder europäischen Kulturraum zuordenbar. Sie geben dem Produkt eine beruflich-nüchterne oder freizeitlich-spielerische Anmutung. Auch weisen sie ein Produkt als Statussymbol aus oder ordnen es einer bestimmten Stilrichtung zu.

Abbildung 3 zeigt Schaltflächen, welche die Auswirkungen der Produktsprache verdeutlichen. Die rechte Schaltfläche wirkt im Sinne der *Formalästhetik* linkslastig. Sie drängt dem Anwender die Erwartung auf, dass rechts neben dem Schriftzug noch etwas geschieht. Die Platzierung des Schriftzuges im Zentrum hingegen gibt ihm etwas Abgeschlossenes und Schweres. Andere formalästhetische Gesichtspunkte könnten die Gruppierung von mehreren Buttons sein, um sie von andern Buttons unterscheidbar zu machen.



Abb. 3: Produktsprachliche Aspekte

Die dreidimensionale Erscheinung ist im Sinne der *Anzeichenfunktionen* - in Anlehnung an reale Drucktasten - ein Anzeichen für die Drückbarkeit der Schaltflächen, die damit über zwei Zustände Auskunft geben: gedrückt und nicht gedrückt. Ihre Formgebung ist also gleichzeitig

Anzeichen für Bedienung und Zustand. Bezüglich der *Symbolfunktion* dieser Schaltflächen läßt sich zumindest sagen, dass es sich hier nicht um ein extravagantes Multimediaprogramm handeln dürfte.

## 4 Erweiterung der Recherchemöglichkeiten

### 4.1 Visualisierung für das Dokument Retrieval

Eine Erweiterung der Recherchemöglichkeiten ist der Einsatz einer Visualisierung. Diese wurde im Rahmen einer Kooperation zwischen dem Informationszentrum Sozialwissenschaften (Bonn), der Universität Koblenz-Landau und der Hochschule für Gestaltung (Offenbach) erarbeitet (Bürdek et al. 1999). Die Grundidee ist die Darstellung der möglichen Kombinationen von Suchbegriffen und der entsprechenden Anzahl von gefundenen Dokumenten. Abbildung 4 zeigt die Suchbegriffe »Mann« und »Frau« (links) zu denen 232 bzw. 2857 Dokumente gefunden wurden. Rechts davon wird die Kombination »Mann UND Frau« dargestellt, die 199 Dokumente enthält. Unter den Suchbegriffen befindet sich ein weiteres Eingabefeld zur Erweiterung der Anfrage.

Durch das Prinzip, dem Anwender die zu den Suchbegriffen möglichen Booleschen Kombinationen zusammen mit den Trefferzahlen anzubieten, wird die Last der Rechercheformulierung vom Anwender zum System hin verlagert. Der Benutzer selbst wird nicht mehr mit der oft mühsamen und fehleranfälligen Rechercheformulierung belastet, sondern wählt bequem eine vorgefertigte Anfrage aus. Diese Methode hat sich in Nutzertests gegenüber herkömmlichen textuellen Rechercheumgebungen als deutlich vorteilhaft herausgestellt. Daneben bietet die Visualisierung auch probabilistisches Ranking an, um die Ergebnisse nach Relevanz zu ordnen. Vages Retrieval liefert zu einer Dokumentmenge ähnliche Dokumente,

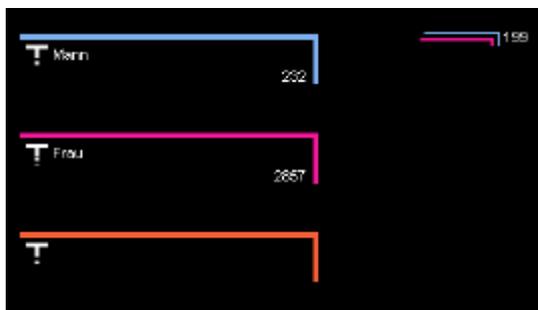


Abb. 4: Suchanfrage mit zwei Suchkriterien

ohne dass die Anfrage modifiziert werden muss, wodurch in Tests die Retrieval-Leistung nochmals verbessert werden konnte. Die Möglichkeiten der Visualisierung werden ausführlich in

Eibl 2000 beschrieben. Abbildung 5 zeigt die Visualisierung integriert in das Informationssystem GESINE.

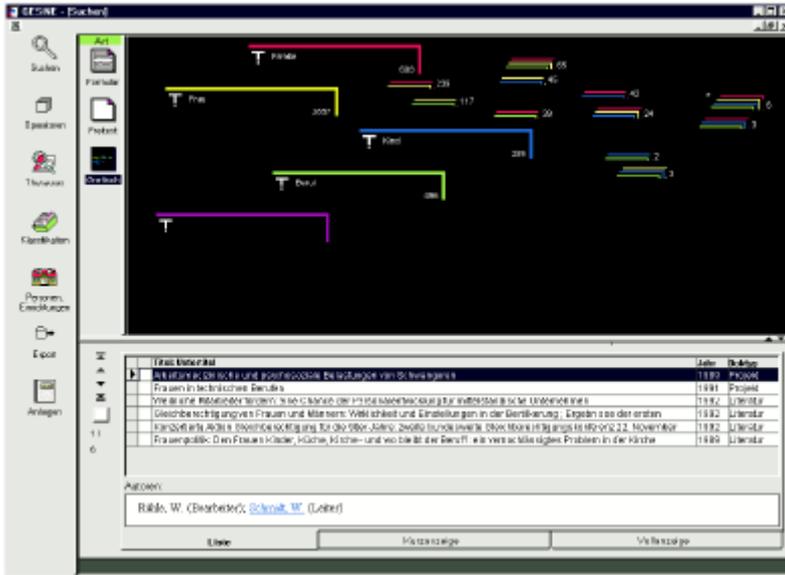


Abb. 5: GESINE mit integrierter Visualisierung

Das *gestalterische Grundelement* der Visualisierung ist der Winkel. Er wurde in einer für das Graphik Design typischen und die Softwareergonomie eher fremden hochgradig assoziativen Weise gefunden. Ziel war es, ein Gestaltungselement zu finden, das der Aufgabe, ein Retrieval-System zu formen gerecht werden kann, d.h. zum einen als durchgängiges einheitliches Gestaltungsmittel taugt und gleichzeitig den Typus des Retrieval-Systems zum Ausdruck bringen kann. Im wesentlichen beruht die Idee des Winkels auf Karteikarten, wie sie in Bibliotheken zum Teil heute noch Verwendung finden. Abbildung 6a zeigt, wie mehrere Karteikarten hintereinander gelegt erscheinen. Für die Visualisierung wurde entsprechend Abbildung 6c das verwendet, was von den hinteren Karten zu sehen ist, wobei die Kanten unten abgeschnitten wurden, um die Gestaltung optisch zu glätten. In Abbildung 6 ist eine Kombination zu sehen. Sie wird durch



Abb. 6a,b&c: Karteikartenassoziation

Karteikarten ausschließlich dazu diente ein grundlegendes Gestaltungselement zu finden. Keinesfalls soll umgekehrt diese Assoziation beim Anwender hervorgerufen werden.

Die optische *Integration* der Erweiterungen richtet sich nach den formalästhetischen Gestaltungskonzepten additiv, integrativ und integral (Bürdek 1994: 191). Das probabilistische Retrieval greift in die Recherche selbst ein. Es wird der bestehenden Recherche nicht aufgesetzt, sondern wird integraler Bestandteil. Daher wurde für die Realisierung des probabilistischen Modells die integrale Gestaltungskonzeption angewandt, indem die einzelnen Elemente der Darstellung selbst unberührt bleiben und sich nur ihre Positionierung verändert. Die so erzielte symbol-

funktionale Wirkung stellt die strikte Interpretation von Relevanz des Booleschen Retrievals der eher fließenden des probabilistischen auch optisch gegenüber.

Das vage Retrieval hingegen wird in der Konzeption der Visualisierung den bereits bestehenden Dokumentmengen aufgesetzt. Es wird nicht in die Recherche selbst eingegriffen, sondern das Ergebnis der Recherche zusätzlich bearbeitet und erweitert. Die Funktionalität wird gestalterisch - daher additiv - umgesetzt: Auf die bereits bestehenden Ergebnismengen wird zusätzlich ein grauer Winkel aufgesetzt. Die Positionierung der Menge bleibt jedoch unverändert. Durch die additive Formgebung wird ein zusätzlicher Retrieval-Schritt symbolisiert.

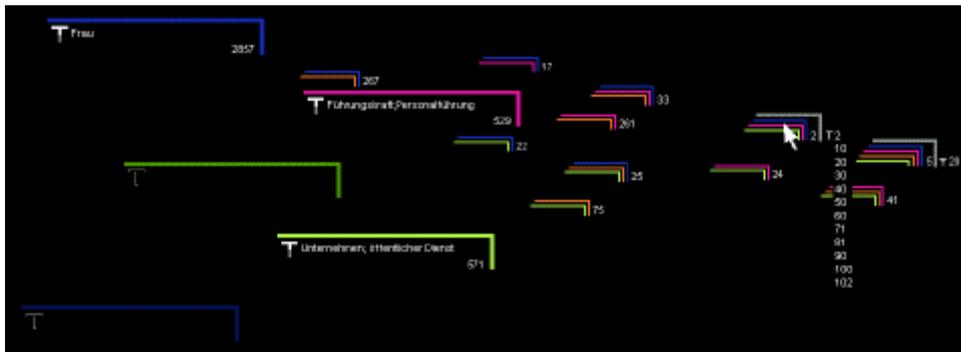


Abb. 7: Probabilistisches und vages Retrieval

In einem formellen Anwendertest wurde die Visualisierung mit den Recherchesystemen Messenger und freeWAIS verglichen. Sowohl bezüglich Precision (Visualisierung: 0.66; Messenger: 0.616; freeWAIS: 0.518) also auch bezüglich Recall (Visualisierung: 0.156; Messenger: 0.127; freeWAIS: 0.105) konnte die Visualisierung in der Booleschen Grundversion bessere Ergebnisse erzielen. Diese Werte konnten in einem weiteren Test durch den Einsatz des vagen Retrievals und probabilistischen Rankings zusätzlich deutlich optimiert werden (vgl. Eibl 2000).

## 4.2 Visual Formalism zur Heterogenitätsbehandlung

Genau wie die Visualisierung verlassen auch Visual Formalisms den Rahmen der durch Style Guides und Normen vorgegebenen Gestaltungsrichtlinien. Obwohl für ihre Anwendung keine vergleichbaren Regelwerke existieren, sondern zum Beispiel auf allgemeinen Grundlagen der Informationsgrafik (Tuft 1990) zurückgegriffen werden muss, zeigen sie interessante Wege auf, das »Kästchendenken« heutiger Standardoberflächen zu verlassen. Oberstes Ziel muss es aber auch hier sein, Lösungen zu finden, welche die Gebrauchstauglichkeit der Software erhöhen, vom Benutzer akzeptiert werden und in vorhandenen Lösungen integrierbar sind. Die im Abschnitt 2 bereits eingeführte Zeitreihenrecherche in ELVIRA kann als Beispiel dafür dienen, wie durch Visual Formalisms Beziehungen innerhalb der Daten einer Anwendung sichtbar gemacht werden können, die ohne ihren Einsatz zu falschen Interpretationen oder ineffizienter Bedienung führen würden.

Zur Faktenrecherche werden in ELVIRA Nomenklaturen eingesetzt, mit denen die einzelnen Zeitreihen analog zu Textdokumenten verschlagwortet sind. Die Nomenklaturen entsprechen dabei Thesauri, die spezifisch für einzelne Datenbestände sind. Da in beliebig vielen Datenbeständen gleichzeitig recherchiert werden kann, kommen mehrere Nomenklaturen gleichzeitig zum Einsatz, die in ihrer Begrifflichkeit ähnlich, stellenweise sogar gleich sein können. Die Abbildung 8 zeigt eine typische Nutzungssituation, bei der die in beiden Nomenklaturen selektierten Suchbegriffe in die dynamisch wachsende Zustandsanzeige übertragen wurden. Damit ein-

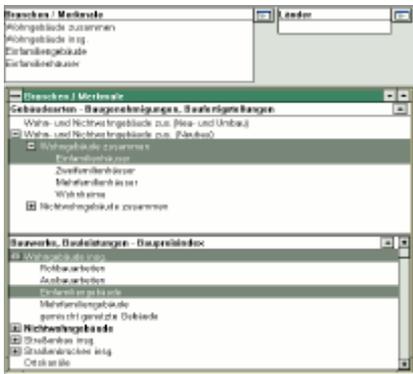


Abb. 8: Auswahllisten und Zustandsanzeige

her geht ein Verlust an Kontext: alleine aus der Zustandsanzeige erschließt sich nicht mehr, in welcher Nomenklatur die Suchbegriffe jeweils gewählt wurden.

Im Rahmen von Styleguides, die meist eine Verwendung vorhandener Oberflächenelemente nahelegen und in der Regel keine Hinweise auf konzeptuelle Erweiterungen der Oberflächensyntax geben, bieten sich nur wenig befriedigende Lösungen an, wie zum Beispiel die farbliche bzw. textuelle Codierung oder eine Gruppierung innerhalb der Zustandsanzeige nach Nomenklaturen. Diese Lösungen weisen Nachteile aufgrund der schlechten Diskriminierbarkeit beim Einsatz von Farben (es würden 20 verschiedene Farben benötigt) oder dem zusätzlichen Platzbedarf für Zwischenüberschriften in der Zustandsanzeige auf. Sie bieten eine schlechte visuelle Kopplung zwischen der Zustandsanzeige und den einzelnen Nomenklaturen. Parallel zur Zahl an Nomenklaturen wächst zusätzlich auch die Zahl der zur Diskriminierung benötigten Farben oder Symbole.

Das zu lösende Problem besteht daher in der Visualisierung der Beziehungen zwischen den zwei Attributen »Nomenklaturname« und »Nomenklatureintrag«. Auch mit zunehmender Komplexität (mehrere Nomenklaturen oder Einträge) bleibt die grundlegende Struktur gleich, lediglich die Zahl der darzustellenden Attributausprägungen erhöht sich. Der variierenden Zahl an Einträgen trägt die Zustandsanzeige bereits dadurch Rechnung, daß sich ihre Größe in vertikaler Richtung dynamisch anpasst. Es liegt daher nahe, die Horizontale in ähnlicher Weise zur Visualisierung der zweiten Dimension (Nomenklaturnamen) zu nutzen.

Die Abbildung 9 zeigt eine auf dem Visual Formalism »Tabelle« basierende Lösung, die gleichzeitig dynamische Anpassung, kontext-sensitive Durchlässigkeit und eine modifizierbare Zustandsanzeige integriert. Die ursprünglich einspaltige Zustandsanzeige wird dabei um eine zusätzliche Spalte für jede Nomenklatur erweitert, die mit der aktuellen Anfrage in Beziehung steht. Eine Beziehung besteht dann, wenn ein Eintrag dieser Nomenklatur oder ein gleichlautender Eintrag einer anderen Nomenklatur in der Zustandsanzeige enthalten ist. Im ersten Fall muss aufgrund der kontext-sensitiven Durchlässigkeit die Auswahlliste (Nomenklatur) mit der Zustandsanzeige synchronisiert werden. Im zweiten Fall wirkt die dynamische Anpassung, wodurch sich die Oberfläche im Sinne eines intelligenten Systemverhaltens adaptiert. Auf der Basis eines einfachen Vergleichs von Zeichenketten oder - falls vorhanden - unter Zuhilfenahme einer Cross-Konkordanz ermittelt das System selbständig, welche weiteren Nomenklaturen das bisher formulierte Informationsbedürfnis (z.B. ausgewählte Produkte) befriedigen könnten.

Branchen / Produkte	OPPEN	OPPE	OPPE
Elektromotoren bis 37,5 W	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Gleichstrommotoren u. -generatoren bis 0,0375 kW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Abb. 9: Zustandsanzeige als Visual Formalism

Zur Darstellung der Einzelbeziehungen zwischen Einträgen (Zeilen) und Nomenklaturen (Spalten) wird am jeweiligen Schnittpunkt ein leeres Feld (keine Beziehung) oder ein Auswahlfeld (Beziehung besteht) angezeigt. Das Auswahlfeld erlaubt dem Benutzer die Feinparametrisierung der Suche, indem Suchbegriffe in einzelnen Nomenklaturen aktiviert oder deaktiviert werden können. Das Prinzip der modifizierbaren Zustandsanzeige wurde daher konsequent auf den Visual Formalism »Tabelle« übertragen und erweitert diesen.

## 5 Ausblick

Sowohl die Visualisierung als auch der Visual Formalism stellen in den aktuellen Versionen von ELVIRA und GESINE Subsysteme dar, die durch weitergehende Funktionalitäten ergänzt werden. Sie erweitern gleichzeitig das WOB-Modell um die ästhetischen Aspekte des Grafik Design und um Möglichkeiten zur Heterogenitätsbehandlung in Informationssystemen. Durch die erfolgreiche Übertragung des Visual Formalism von der Fakten- auf die Textrecherche (Stempfhuber 2001) wurde die Grundlage für eine Neugestaltung von GESINE auf der Basis der hier vorgestellten Ergebnisse gelegt, aus deren Synergie eine weitere Verbesserung von Benutzerfreundlichkeit und Retrieval-Qualität zu erwarten ist.

## Literatur

- Ahlberg, Christopher; Shneiderman, Ben (1994). Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays. In: CHI'94 Human Factors in Computing Systems, Boston, Massachusetts, United States, April 24-28, 1994. S. 313-317 u. 479-480.
- Bürdek, Bernhard E. (1994). Design. Geschichte, Theorie und Praxis der Produktgestaltung. 2.Auflage, Köln.
- Bürdek, Bernhard E.; Eibl, Maximilian; Krause, Jürgen (1999). Visualization in Document Retrieval. In: Proceedings of the 8th HCI International'99, München, 22.-27.8.1999, Vol.2, S.102-106.
- Carroll, John M.; Thomas, John, C. (1982). Metaphor and the Cognitive Representation of Computing Systems. In: IEEE Transactions on Systems, Man, and Cybernetics, Vol.SMC-12, Nr.2, März/April1982, S.107-116.
- Eibl, Maximilian (2000). Visualisierung im Document Retrieval: Theoretische und praktische Zusammenführung von Softwareergonomie und Graphik Design. Bonn. [Forschungsberichte des IZ Sozialwissenschaften Band 3]
- Gros, Jochen (1983). Einführung in die Grundlagen einer Theorie der Produktsprache. Hochschule für Gestaltung, Offenbach (Hrsg.). Grundlagen einer Theorie der Produktsprache, Heft 1.
- Johnson, Jeff A.; Nardi, Bonnie A.; Zarmer, Craig L.; Miller, James R. (1993) ACE: Building Interactive Graphical Applications. In: Communications of the ACM, April 1993, Vol.36, Nr.4, S.41-55.
- Krause, Jürgen (1996). Visualisierung und graphische Benutzungsoberflächen. IZ-Arbeitsbericht Nr. 3, Informationszentrum Sozialwissenschaften, Bonn.
- Krause, Jürgen (1997). Das WOB-Modell. In: Krause, Jürgen; Womser-Hacker, Christa (1997). Vages Information Retrieval und graphische Benutzungsoberflächen: Beispiel Werkstoffinformation. Konstanz. [Schriften zur Informationswissenschaft Bd.28], S. 59-88.
- Krause, Jürgen; Stempfhuber, Maximilian (Hrsg.) (2001). Integriertes Retrieval in heterogenen Daten. Text-Fakten-Integration am Beispiel des Verbandinformationssystems ELVIRA. [Forschungsberichte des IZ Sozialwissenschaften Band 4] (im Druck).
- Nardi, Bonnie; Zarmer, Craig (1993). Beyond Models and Metaphors: Visual Formalisms in User Interface Design. In: Journal of Visual Languages and Computing, 1993, 4, S.5-33.
- Stempfhuber, Maximilian (2001). ODIN: Objektorientierte dynamische Benutzungsoberflächen für die Recherche in heterogenen Datenbeständen. Dissertation im Fachbereich für Informatik an der Universität Koblenz-Landau.
- Tufte, Edward (1990). Envisioning Information. Ceshire (Conn.).

## Adressen der Autoren

Maximilian Eibl  
 Informationszentrum Sozialwissenschaften  
 Schiffbauerdamm 19  
 10117 Berlin  
 ei@berlin.iz-soz.de

Maximilian Stempfhuber  
 Informationszentrum Sozialwissenschaften  
 Lennéstr. 30  
 53113 Bonn  
 st@bonn.iz-soz.de