

Universität Konstanz
FB Informatik und Informationswissenschaft
Bachelor-Studiengang Information Engineering

Bachelorarbeit

Visualisierungs- und Interaktionskonzept zur Integration hierarchischer Daten in die HyperGrid

*zur Erlangung des akademischen Grades eines
Bachelor of Science (B.Sc.)*

Studienfach: Information Engineering
Schwerpunkt: Mensch-Computer Interaktion
Themengebiet: Informationswissenschaften

von

Dominic Winkler

Matr.-Nr.: 01/556121
Erstgutachter: Prof. Dr. Harald Reiterer
Zweitgutachter: Prof. Dr. Marc H. Scholl
Einreichung: 14. März 2007

Zusammenfassung

Hierarchien sind nicht nur ein Werkzeug zur Strukturierung von Informationen, sondern sie sind auch Träger von Informationen. Zur Veranschaulichung der Informationen, die durch eine Hierarchie präsentiert werden, benötigt der Betrachter effektive Visualisierungsmethoden. Um mit einer großen Menge an Informationen effizient umgehen zu können, werden visuelle Suchsysteme eingesetzt, die zugleich das Browsen, aber auch analytische Methoden wie Suchen, Sortieren und Filtern unterstützen. MedioVis, ein Forschungsprojekt der AG Mensch-Computer Interaktion der Universität Konstanz, ist ein innovatives visuelles Suchsystem, dessen Hauptvisualisierung eine zoombare Tabelle namens HyperGrid ist. In der HyperGrid können bisher keine hierarchischen Zusammenhänge dargestellt werden. Da Hierarchien aber ein immens wichtiger Teil großer Informationsräume sind, wurde im Rahmen dieser Arbeit die HyperGridXGL, ein Visualisierungs- und Interaktionskonzept zur Integration hierarchischer Daten in die HyperGrid, entwickelt. Zur Illustration der konzeptionellen und technischen Machbarkeit wurde die HyperGridXGL praktisch umgesetzt. Der daraus entstandene Prototyp wurde anschließend in einer Evaluation auf seine Benutzerfreundlichkeit getestet. Das Konzept, die Umsetzung sowie die Ergebnisse der Evaluation und die daraus resultierenden Redesign-Vorschläge werden in der vorliegenden Arbeit eingehend betrachtet und diskutiert. Die Arbeit schließt mit einer Zusammenfassung und gibt einen Ausblick auf mögliche Weiterentwicklungen der HyperGridXGL.

Abstract

Hierarchies are not only used to structure information but also to deliver information. Therefore effective methods to display the information presented by hierarchies are necessary. Furthermore, to handle such a huge amount of information visual seeking systems have to be used. Innovative seeking systems support both an exploring and analytical approach. MedioVis, a research project of the Human-Computer Interaction workgroup of the University of Konstanz, is such a system. The main visualization of MedioVis is a zoomable table called HyperGrid. As of yet the HyperGrid cannot visualize hierarchical relations. Since hierarchies are a really important part of complex information spaces, they need to be displayed inside the HyperGrid. Therefore within this thesis the HyperGridXGL, a new visualization and interaction concept to integrate hierarchical data into the HyperGrid, was developed. In order to demonstrate its conceptual and technological feasibility the concept is applied and implemented as a prototype. In the following evaluation the developed prototype was tested. In the context of this thesis, the concept, implementation and the results of the evaluation are observed and discussed in detail. The thesis is summed up with a résumé which includes an outlook towards the possible further development of the HyperGridXGL.

Inhaltsverzeichnis

1	Einleitung	5
1.1	Motivation	5
1.2	Aufbau der Arbeit	6
2	Hierarchische Visualisierungen	8
2.1	Hierarchien	8
2.1.1	Was sind hierarchische Daten?	8
2.1.2	Wozu dienen Hierarchien?	9
2.1.3	Hierarchien in MedioVis	10
2.2	State-of-the-Art-Analyse	11
2.2.1	Baum	12
2.2.2	Liste	13
2.2.3	TreeBrowser	14
2.2.4	Hyperbolic Browser	16
2.2.5	Treemap	18
2.2.6	Cone Tree	22
2.2.7	SpaceTree	23
3	MedioVis	25
3.1	Projekt MedioVis	25
3.2	Motivation	26
3.3	Anwendungskontext	28
3.4	Zoomable User Interface	29
3.5	Die HyperGrid	31
3.6	Hierarchien in MedioVis	37
3.7	Zusammenfassung und Problemstellung	41
4	HyperGridXGL	43
4.1	Anforderungen und Ziele	43
4.2	Konzeptideen	44
4.2.1	classical	44
4.2.2	finder	46
4.2.3	cube	48
4.3	Wieso HyperGridXGL?	50
4.4	Anwendungsszenario	51
4.5	Das Visualisierung- und Interaktionskonzept HyperGridXGL	54
4.5.1	Hierarchischer Zoom	55
4.5.2	Visualisierung	55
4.5.3	Interaktion	60
4.6	Umsetzung	66
4.6.1	Datenmodell	67

4.6.2	Visualisierung	69
5	Evaluation	71
5.1	Ziele und Absichten des Tests	71
5.2	Testmethode	72
5.2.1	Versuchspersonen	72
5.2.2	Testumgebung	73
5.2.3	Hard- und Software Equipment der Testumgebung	74
5.2.4	Testwerkzeuge	74
5.2.5	Testaufgaben	74
5.2.6	Test-Design	75
5.2.7	Ablauf des Tests	75
5.3	Ergebnisse	78
5.3.1	Usability-Schwächen	78
5.3.2	Redesign-Vorschläge	86
5.3.3	Positive Aspekte	92
5.3.4	AttrakDiff-Ergebnisse	94
6	Zusammenfassung und Ausblick	95
A	Aufgaben	97
B	Interview-Leitfaden	97
C	Pre-Test-Fragebogen	100
D	Post-Test-Fragebogen	103
E	Anhang	104
	Abbildungsverzeichnis	105
	Literatur	107

1 Einleitung

1.1 Motivation

„The information presented by a hierarchy is of no use without an effective method to display and browse the information.“ [Beaudoin u. a., 1996]

Hierarchien sind Mechanismen, um Informationen eines großen Informationsraumes zu organisieren. Die Darstellung hierarchischer Zusammenhänge muss dabei so erfolgen, dass der Betrachter die sich ihm präsentierenden Informationen erfassen, analysieren und verarbeiten kann. Zu diesem Zweck wurden unterschiedliche Visualisierungsmethoden entwickelt. Dabei ist die Darstellung einer Hierarchie als Baum am weitesten verbreitet. Die Gründe für die Favorisierung der Baumvisualisierung sind naheliegend: anschaulich, verständlich und der menschlichen Denkstruktur angepasst [Roth, 2003].

Allerdings zeigt die originäre Baumdarstellung bei großen Hierarchien Mängel. Diese Erkenntnis führte zur Entwicklung neuer Visualisierungsmethoden. Die gebräuchlichsten werden im Verlauf der Arbeit vorgestellt und analysiert. Bemerkenswert ist dabei, dass alle Visualisierungen - ob in 2D- oder gar in 3D-Ausprägung - zwei Gemeinsamkeiten besitzen:

- die Baumstruktur ist weiterhin die Basis der jeweiligen Gestaltungsidee,
- alle verfolgen das Prinzip der Darstellung von „Fokus und Kontext“ in einer Ansicht.

Die heutige Zeit ist u.a. geprägt von einer sich explosionsartig potenzierenden Informationsflut. Es ist nicht nur wichtig, die vielen Informationen effektiv darzustellen und so für den Benutzer durch explorative Ansätze „browsebar“ zu machen, sondern auch analytische Methoden, wie effizientes Suchen, Filtern und Sortieren, anzubieten [Shneiderman, 1996]. In diesem Zusammenhang sind gute visuelle Suchsysteme der Schlüssel zum Erfolg. Einen sehr innovativen Ansatz verfolgt dabei das visuelle Suchsystem MedioVis der Arbeitsgruppe Mensch-Computer Interaktion der Universität Konstanz [Grün u. a., 2005]. MedioVis kann auf ein großes Visualisierungsrepertoire zurückgreifen, wobei die HyperGrid [Reiterer u. a., 2005][Jetter u. a., 2005] die Hauptvisualisierung darstellt.

Hierarchien stellen einen wichtigen Bestandteil zur Strukturierung großer Informationsräume dar. Es ist daher immens wichtig, Hierarchien innerhalb eines Suchsystems zu visualisieren. Die HyperGrid kann diese Anforderung bisher nicht erfüllen. Demzufolge sollen hierarchische Daten in das Visualisierungs- und Interaktionskonzept

der HyperGrid integriert werden. Die HyperGrid basiert auf einer zoombaren Tabellenvisualisierung.

Bisherige Visualisierungen hierarchischer Daten erfolgen zumeist auf der Grundlage der Baumstruktur. Die HyperGrid hingegen basiert auf einer Tabellenstruktur. Da sich bei bisherigen Versuchen eine Baumstruktur nur mit Vorbehalt bzw. einer Reihe von Nachteilen in eine Tabellenstruktur integrieren lässt, soll hier die Visualisierung von Hierarchien ohne Baumstruktur auskommen. Folglich musste ein neuartiges, innovatives Konzept zur Integration hierarchischer Daten in die HyperGrid entwickelt werden. Die hierfür erforderlichen Überlegungen, Maßnahmen und Aktivitäten sind zentraler Bestandteil der vorliegenden Arbeit, in der die Entwicklung der HyperGridXGL und die daraus folgenden Möglichkeiten und Chancen beschrieben und diskutiert werden.

1.2 Aufbau der Arbeit

In Kapitel 2 werden relevante Informationen zu Hierarchien zusammengestellt, um anschließend grundlegende Visualisierungs- und Interaktionskonzepte zur Darstellung hierarchischer Daten anhand von wissenschaftlichen Prototypen und konkreten Systemen zu erläutern.

Das visuelle Suchsystem MedioVis, ein Forschungsprojekt der AG Mensch-Computer Interaktion der Universität Konstanz, wird in Kapitel 3 vorgestellt. Im weiteren Verlauf dieses Kapitels wird die HyperGrid, eine zoombare Tabellenvisualisierung von MedioVis, welche Interaktionskonzepte des zuvor beschriebenen Zoomable User Interface (ZUI) verwendet, erläutert. Ferner wird die TreeTable, das bisher einzige Konzept im Visualisierungsrepertoire von MedioVis zur Darstellung hierarchischer Daten, vorgestellt und deren kritische Faktoren hinsichtlich der Darstellung von Hierarchien diskutiert. Den Schluss dieses Kapitels bildet ein zusammenfassendes Zwischenergebnis, aus dem sich gleichzeitig die Problemstellung für die vorliegende Arbeit ableiten lässt.

In Kapitel 4 wird die HyperGridXGL, das neue Visualisierungs- und Interaktionskonzept zur Integration hierarchischer Daten in die HyperGrid, beschrieben. Hierbei werden die sich aus der Problemstellung ergebenden Ziele und Anforderungen des neuen Konzepts sowie dessen Rahmenbedingungen diskutiert. Daran schließt sich die Vorstellung verschiedener Konzeptideen und deren Einfluss in das in dieser Arbeit präsentierte Konzept an. In einem nächsten Schritt wird die HyperGridXGL anhand eines Anwendungsszenarios vorgestellt und einzelne Komponenten, wie der hierarchische Zoom, die Visualisierung und deren Interaktionen sowie die praktische

Umsetzung im Detail beleuchtet.

In Kapitel 5 wird die zum Test der Gebrauchstauglichkeit der HyperGridXGL durchgeführte Evaluation ausführlich geschildert. Zu Beginn des Kapitels werden verfolgte Absichten und Ziele des Tests erörtert. Dann wird die Testmethodik erläutert. Zum Schluss dieses Kapitels werden die aus dem Test gewonnenen Ergebnisse vorgestellt, wobei insbesondere auf die erkannten Usability-Schwächen sowie auf die in der Konsequenz dazu entwickelten Vorschläge für Redesign-Maßnahmen eingegangen wird.

Diese Arbeit schließt in Kapitel 6 mit einer Zusammenfassung und gibt einen Ausblick auf mögliche Weiterentwicklungen der HyperGridXGL.

2 Hierarchische Visualisierungen

2.1 Hierarchien

2.1.1 Was sind hierarchische Daten?

Systeme von einander über- bzw. untergeordneten Elementen werden in Hierarchien¹ eingeteilt. Ein anschauliches Beispiel dafür ist das in der Biologie verwendete Ordnungssystem der Organismen. Hier werden die Lebewesen systematisch und detailliert hierarchisch aufgegliedert und zugeordnet. Die Ordnungssystematik erfolgt nach dem Prinzip: Reich, Stamm, Klasse, Ordnung, Familie, Gattung, Art, Rasse.

Dabei ist zumindest bei Hierarchien, die im Rahmen dieser Arbeit erörtert werden, jedem Element höchstens ein anderes Element unmittelbar über- oder untergeordnet.

Ein weiteres klassisches Beispiel für tiefgegliederte Hierarchien ist die militärische Rangordnung. Schon im Römischen Heer gab es eine gut geordnete Hierarchie der Dienstgrade unter den Soldaten, die sich im Umfang der Befehlsgewalt der einzelnen Personen widerspiegelte. Die römische Militärhierarchie konnte selbst an der Formation des Heeres auf dem Schlachtfeld erkannt werden.

Hierarchische Daten sind aus unserem täglichen Leben nicht mehr wegzudenken. So sind zum Beispiel Gesetzestexte in einer strengen Hierarchie angeordnet, ebenso Produktverzeichnisse in Versandhauskatalogen oder Inhaltsverzeichnisse wissenschaftlicher Bücher. In allen Fällen sind die Daten durch eine klare Strukturierung gekennzeichnet. Die Struktur dient aber nicht nur der Übersicht, sondern sie hat zusätzlich eine semantische Bedeutung.

Das menschliche Gehirn benutzt ebenfalls hierarchische Strukturen zum Speichern von Informationen. Die Systematik des menschlichen Denkens, sozusagen das mentale Modell, das man sich zum Verstehen einer Hierarchie konstruiert, wird häufig mit dem Aufbau eines Baumes veranschaulicht [Roth, 2003]:

Alle hierarchischen Daten lassen sich in Baumstrukturen abbilden, wobei die einzelnen Knoten des Baumes als Objekt betrachtet werden können. Jedem Knoten können wiederum weitere Kind-Knoten zugewiesen werden. Ausgangspunkt eines solchen Baumes ist die Wurzel (in der Abbildung 1 oben: ABC). Die Knoten auf der untersten Ebene jedes Astes sind jene Knoten, die keine weiteren Kind-Knoten besitzen. Sie werden als „Blätter“ bezeichnet. Da der Baum in der Abbildung 1 nicht ausbalanciert ist, somit die Tiefe der Blätter nicht gleich groß ist, sind sowohl Knoten „a“ (von „A“), „b“ (von „C“) als auch „B“ und alle Knoten mit Zahlen Blätter. Bei einer solchen Baumstruktur sind also nicht nur die Knoten selbst von

¹Hierarchie - ein Kompositum aus „hieré“ - heilige und „arché“ - Herrschaft, Ordnung, Prinzip

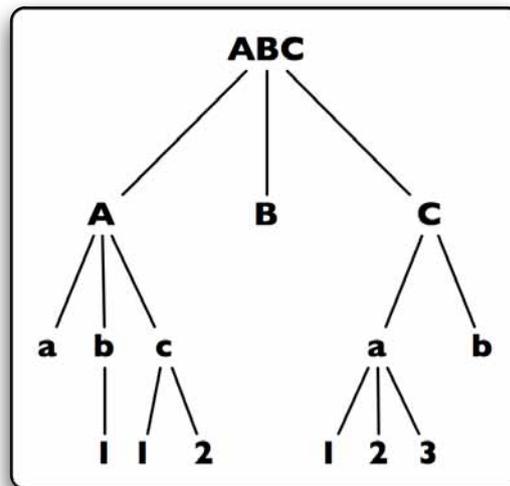


Abbildung 1: Baum

Bedeutung, sondern auch die Topologie des Baumes, also die Position des jeweiligen Knotens in der Baumstruktur. Die Position gibt Auskunft über die Vor- und Nachfahren des jeweiligen Knotens.

Ein treffendes Beispiel für diese Art der Darstellung ist beispielsweise der Stammbaum eines alten Adelsgeschlechts.

2.1.2 Wozu dienen Hierarchien?

Hierarchische Daten werden aufgrund ihrer Struktur durch die Hierarchie kategorisch geordnet. Diese Ordnung dient der Organisation innerhalb von Datenmengen. Anhand von Dateisystemen können Vorteile eines hierarchischen Aufbaus für nahezu jedermann verständlich erläutert werden. Da Computer aus dem Alltag nicht mehr wegzudenken sind, beschäftigt sich nahezu jeder Nutzer Tag für Tag mit Datenmengen, wie sie vor wenigen Jahren selbst für einen Fachmann nicht möglich erschienen. Er kann somit die aufgezeigten Vorteile aus eigener Erfahrung leicht nachvollziehen.

Die Wurzel hierarchischer Dateisysteme, wie beispielsweise das UNIX-Dateisystem², ist das Root-Verzeichnis „/“. Ausgehend von dem Root-Verzeichnis enthalten die Kind-Knoten entweder weitere Verzeichnisse, d.h. weitere Kinder, denen wieder weitere Kind-Knoten untergeordnet sein können, oder aber Dateien, die die Blätter jeder Hierarchie-Ebene bilden.

²UNIX ist ein Datei- und Betriebssystem, auf dem Mac OS X und Linux basieren. <http://www.unix.org/>

Um sich der Bedeutung der Strukturierung von Dateien in Dateisystemen bewusst zu werden, muss man sich lediglich vergegenwärtigen, wie das Arbeiten mit solchen Dateisystemen aussehen würde, wenn sie immer noch wie eine flache Tabelle organisiert wären. Bei der heutigen Größe von Datenmengen würden sich Unmengen von Dateien auf einer einzigen Ebene befinden, dementsprechend ineffizient und damit „teuer“³ würde das gezielte Suchen nach speziellen Informationen sein. Das heißt: Eine gute Strukturierung beim Einordnen von Objekten erleichtert bzw. ermöglicht späteres effizientes und effektives Auffinden von Objekten innerhalb einer solchen Verwaltungsstruktur. Dies gilt selbst dann, wenn man zum Auffinden von speziellen Informationen ein Suchsystem verwendet. Denn auch die Algorithmen von Suchsystemen benutzen „am liebsten“ eine Struktur, um die gesuchte Information effizient zu finden. Ein gängiges Beispiel für eine solche Suche ist ein Telefonbuch. Wären die Daten dieses Telefonbuchs in einer flachen Tabelle gespeichert, müsste der Such-Algorithmus durch die komplette Tabelle gehen, um die gesuchte Person oder Telefonnummer zu finden. Sind die Daten allerdings in einer hierarchischen Baumstruktur geordnet, kann sich der Algorithmus an den Knoten entlang hangeln und so schneller und damit effizienter zu einem Ergebnis kommen.

2.1.3 Hierarchien in MedioVis

Im Rahmen des Projekts MedioVis (siehe Projekt MedioVis Kapitel 3.1) wird hauptsächlich der Datenbestand der Universitätsbibliothek Konstanz⁴ bzw. dessen Untermenge „Mediothek“ verwendet. Zusätzlich werden die Daten der Mediothek mit gespiegelten Film- und Schauspielerdaten der Internet Movie Database IMDb⁵ sowie gespiegelten Daten von Wikipedia⁶ und mit GoogleMaps⁷ angereichert. Die Bibliothek stellt einen Datenbestand von über zwei Millionen Titeln zur Verfügung. Unter diesen Titeln bilden identische Exemplare oder auch mehrteilige Werke logische Hierarchien. Diese logischen Hierarchien sind höchstens in einer Tiefe von bis zu drei Ebenen⁸ gegliedert. Im Redesign des Frameworks [König, 2005] von MedioVis wurden diese hierarchischen Strukturen in das Datenmodell integriert.

Unter den Datenbeständen der Bibliothek befinden sich auch so genannte mehrteilige Werke. Diese haben einen Gesamttitel⁹, der sozusagen die Wurzel des hierarchischen

³Als „teuer“ werden in der Informatik rechenintensive Prozesse bezeichnet.

⁴Bibliothek der Universität Konstanz, <http://www.ub.uni-konstanz.de/>

⁵IMDb: Internet Movie Database, mit Metadaten zu mehr als 500.000 verschiedenen Filmen und über 1,8 Millionen Personen, <http://www.imdb.com>

⁶Wikipedia, <http://de.wikipedia.org>

⁷GoogleMaps, <http://maps.google.de/>

⁸zumindestens zum Zeitpunkt der bisher visualisierten Daten, Dezember 2006

⁹Gesamttitel ist die Bezeichnung für das mehrteilige Werk im Gesamten. Dies entspricht der Bezeichnung, die in der Bibliothek verwendet wird. Aus technischer Sicht wird ein Gesamttitel auch mit der Bezeichnung Container versehen.

Datensatzes bildet. Bezogen auf den Beispieldatensatz (siehe Abbildung 2) wäre dies „Der Herr der Ringe“. Jedes mehrteilige Werk besteht aus mehreren untergeordneten Titeln¹⁰. Angewandt auf den Beispieldatensatz sind dies

- Der Herr der Ringe: Die Gefährten
- Der Herr der Ringe: Die Zwei Türme
- Der Herr der Ringe: Die Rückkehr des Königs.

Jeder dieser Titel kann auch noch in mehrfacher, aber identischer Ausführung vorhanden sein. Diese Ausführungen werden „Exemplare“ genannt. Sie unterscheiden sich lediglich in der Signatur und eventuell im Standort in der Mediothek. Somit gibt es in den Bibliotheksdaten hierarchische Datensätze mit maximal drei Ebenen. Einzelne Werke, die mehrfach vorhanden sind, haben maximal zwei Ebenen, nämlich die Titel-Ebene und die Ebene der Exemplare.

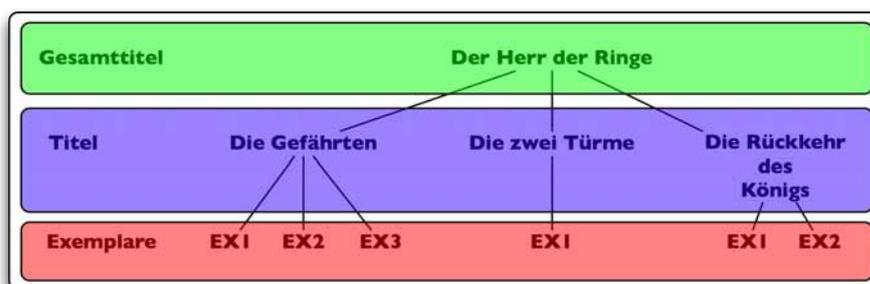


Abbildung 2: Baum mit Beispieldatensatz „Der Herr der Ringe“

2.2 State-of-the-Art-Analyse

Hierarchien kommen sowohl in der Wissenschaft als auch im Alltag vor. Wie schon erörtert, dienen Hierarchien insbesondere der Strukturierung einer großen Menge von Daten. Um diese Struktur nicht nur auf der datentechnischen Ebene nutzen zu können, ist es wichtig, Hierarchien visuell aufbereitet darzustellen. Über die Jahre hinweg wurde eine beachtliche Vielzahl unterschiedlicher Darstellungsmöglichkeiten von hierarchischen Daten entwickelt. Die meisten dieser Techniken basieren jedoch auf einer kleineren Anzahl von Kernvisualisierungstechniken. Diese werden im Folgenden näher erläutert.

¹⁰Titel ist die Bezeichnung für untergeordnete Titel eines mehrteiligen Werks. Dies entspricht der Bezeichnung, die in der Bibliothek verwendet wird. Aus technischer Sicht wird ein Titel auch mit der Bezeichnung Unterdokument versehen.

2.2.1 Baum

Die Visualisierung „Baum“ ist mit Sicherheit die grundlegendste aller Visualisierungstechniken. Über 90% aller hierarchischen Visualisierungsformen greifen auf diese Darstellungsweise in verschiedenen Varianten zurück. Die meisten Lehrbücher, die Informatik-Know-how vermitteln, greifen zur Veranschaulichung von hierarchischen Zusammenhängen gern zu dieser Visualisierungstechnik. Daher gilt der „Baum“ insbesondere in der Informatik als Standarddarstellungsweise für hierarchische Daten.

In der Gehirnforschung ging man sehr lange Zeit davon aus, dass die Abspeicherung von Daten im Gehirn dem Aufbau von Baumstrukturen folgt. Im letzten Jahrzehnt kam man zu der Erkenntnis, dass die Informationsspeicherung eher netzartigen Verknüpfungen von Bildern, Worten, Gefühlen und Geschmackserfahrungen entspricht. Für die meisten Menschen wird dennoch die Denk- und Speicherfähigkeit im Zusammenhang mit der Erfassung von Daten durch Visualisierung hierarchischer Daten in Baumform verbessert [Roth, 2003].

Die Baumsystematik wird unterschiedlich angewandt:

Die Wurzel ist zentral, in der Regel oben, platziert. Von dort aus wächst dann der Baum - entgegen seinem Auftreten in der Natur - nach unten. Die seitliche Platzierung der Wurzel oder eine Anordnung von unten nach oben ist in der Literatur allerdings gleichermaßen zu finden.

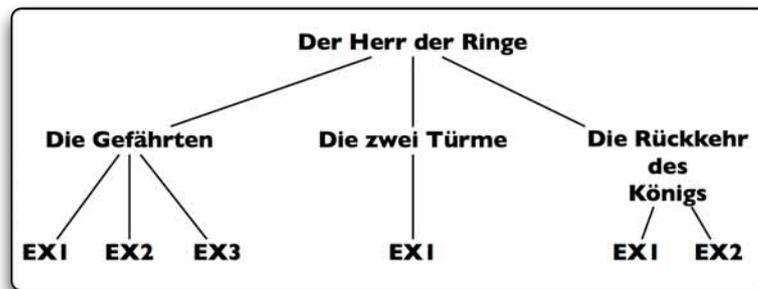


Abbildung 3: Baum-Visualisierung mit der Hierarchie von „Der Herr der Ringe“

Obwohl diese Visualisierung in vielen Lehrbüchern gebräuchlich ist und auch als sehr eingängig empfunden wird, kämpft sie gegen eine Vielzahl praktischer Probleme an.

Zunächst ist eine Darstellung dieser Art enorm platzintensiv, da man in der Praxis mit weitaus größeren Hierarchien arbeitet als mit den zur Illustration eines Zusammenhangs in Lehrbüchern verwendeten Bäumen. Wenn man sich allein das Systemverzeichnis eines Windows-Rechners¹¹, visualisiert in Form eines Baumes, vorstellt,

¹¹Ein Windows-Rechner ist ein Computer, auf dem das Betriebssystem Microsoft Windows installiert ist

wird einem klar, dass ein Knoten mehrere Tausend Kind-Knoten besitzen kann. Um nun eine überschneidungsfreie Darstellung aller Knoten, Kanten und Beschriftungen zu erreichen, müssen entweder sehr große Abstände zwischen den einzelnen Elementen gewählt werden oder man setzt aufwendige Algorithmen ein, um eine kompakte aber dennoch korrekte Darstellung zu erzielen.

Eine geringe Knotendichte führt dazu, dass eine großflächige Hierarchie sich dem Betrachter entweder mit allen Details, aber nur in einem Ausschnitt präsentiert, oder dass die komplette Hierarchie als Overview mit geringem Detaillierungsgrad gezeigt werden kann. Daher sind intelligente Interaktionsmechanismen, wie Zooming und Panning, erforderlich. Zooming bedeutet: man fährt sozusagen in einen Ausschnitt des Baumes hinein, um alle Details zu sehen. Um nun einen anderen Ausschnitt zu sehen, muss man Panning betreiben, das heißt, man verschiebt den Ausschnitt des Baumes, der momentan angezeigt wird, innerhalb des gesamten Baumes an eine andere Stelle. Dies führt zu einem schnellen Verlust der Orientierung in der Hierarchie, da nicht mehr nachvollziehbar ist, in welchem Kontext der aktuelle Ausschnitt liegt.

Eine kompakte Darstellung kann diesen Nachteil zwar teilweise ausgleichen, jedoch sind die Algorithmen, die dafür nötig sind, extrem rechenintensiv. Dies kann sich sehr störend auf die direkte Manipulation durch Zooming und Panning auswirken, da das System mit einem enormen Rechenaufwand belastet wird.

2.2.2 Liste

Die Darstellung als Liste gehört neben der Baum-Visualisierung zu den ältesten Visualisierungsformen von hierarchischen Daten. Trotzdem stellt diese Art der visuellen Aufbereitung mit ihrer sehr einfachen Grundidee die Basis für viele andere populäre Varianten von Hierarchiendarstellungen dar.

Bei der Darstellung in Form einer Liste wird jeder Knoten durch eine Zeile repräsentiert. Die Zeile enthält jegliche Knoteneigenschaften, wobei hier auch die Position in der Hierarchie über den Pfad wiedergegeben werden kann. Im Prinzip kann man die Darstellung hierarchischer Daten als Liste mit einem „plattgedrückten Baum“ vergleichen. Ein in der Informatik allgemein bekanntes Beispiel ist der UNIX-Befehl „ls“ (siehe Abbildung 4).

```

total 2496
drwx----- 36 dom dom 1224 Jan 9 18:45 .
drwxr-xr-x 31 dom dom 1054 Jan 12 09:31 ..
-rw----- 1 dom dom 12292 Jan 9 18:45 .DS_Store
-rw-r--r-- 1 dom dom 0 Jul 27 13:01 .localized
-rwxrwxrwx 1 dom dom 146543 Feb 15 2005 05_msnyder_01.jpg
-rwxrwxrwx 1 dom dom 189042 Feb 15 2005 05_msnyder_02.jpg
-rwxrwxrwx 1 dom dom 192981 Feb 15 2005 05_msnyder_03.jpg
-rwxrwxrwx 1 dom dom 283981 Feb 15 2005 05_msnyder_04.jpg
-rwxrwxrwx 1 dom dom 135820 Feb 15 2005 05_msnyder_05.jpg
drwxr-xr-x 166 dom dom 5644 Oct 9 16:57 Diverses
drwxr-xr-x 10 dom dom 340 Nov 29 15:17 IMGgal
drwxr-xr-x 43 dom dom 1462 Nov 28 22:58 LastDayInBeijing
drwxr-xr-x 73 dom dom 2482 Sep 21 04:03 Olas Fotos
drwxr-xr-x 73 dom dom 2482 Dec 25 19:20 SkifahrnSuedTirol2005_Casio
drwxr-xr-x 28 dom dom 952 Dec 25 17:59 SkifahrnSuedTirol2005_Samsung
drwxr-xr-x 26 dom dom 884 Sep 4 04:11 autobahn
drwxr-xr-x 85 dom dom 2890 Nov 17 09:54 campus
drwxr-xr-x 16 dom dom 544 Oct 8 17:38 charloe_birthday
drwxr-xr-x 3 dom dom 182 Nov 28 12:08 deniseSH
drwxr-xr-x 17 dom dom 578 Oct 8 17:34 fabianfarevell
drwxr-xr-x 31 dom dom 1054 Oct 11 13:56 fotos
lrwxr-xr-x 1 root wheel 47 Jul 27 13:01 iChat Icons -> /Library/Application
Support/Apple/iChat Icons/
drwxr-xr-x 8 dom dom 272 Nov 29 15:07 iPhoX
drwx----- 19 dom dom 646 Jan 13 18:36 iPhoto Library
drwxr-xr-x 12 dom dom 488 Sep 1 07:39 jinsoo
drwxr-xr-x 3 dom dom 182 Oct 19 04:15 johannes Bilder
drwxr-xr-x 220 dom dom 7480 Sep 4 04:22 markt
-rwxrwxrwx 1 dom dom 377211 Jul 4 2005 mb_newfashion_jun05.jpg
drwxr-xr-x 60 dom dom 2040 Sep 19 18:25 morgentraining
drwxr-xr-x 229 dom dom 7786 Oct 8 17:28 nanjing
drwxr-xr-x 38 dom dom 1292 Sep 1 07:39 newoffice
drwxr-xr-x 32 dom dom 1088 Aug 19 04:16 office
drwxr-xr-x 14 dom dom 476 Oct 8 17:35 oldshanghai
drwxr-xr-x 24 dom dom 816 Oct 12 10:20 picsMatt
drwxr-xr-x 148 dom dom 5932 Sep 6 15:55 pudongatnight
drwxr-xr-x 13 dom dom 442 Jan 9 10:46 wallpaper
dombok:~/Pictures dom$

```

Abbildung 4: Listendarstellung des UNIX-Befehls „ls“.

Erfahrene Benutzer können damit ein hierarchisches Dateisystem explorieren, darin navigieren und schnell und gezielt Inhalte auffinden. Jedoch ist es ein großes Manko, dass die Topologie des Baumes bzw. die Struktur der Hierarchie und der Kontext, in dem die dargestellten Knoten stehen, so gut wie gar nicht visualisiert werden. Vor diesem Hintergrund hat der User beim Interpretieren der Zeilen in Bezug auf den Aufbau des gesamten hierarchischen Systems einen erheblichen kognitiven Aufwand zu leisten. Eine weitere Schwierigkeit ergibt sich bei der notwendig werdenden Orientierung über mehrere Hierarchie-Ebenen.

Trotz der oben beschriebenen Schwäche in der Visualisierung ist die Listendarstellung eine der gebräuchlichsten Veranschaulichungsformen für hierarchische Strukturen. Fragt man nach dem Grund für die Beliebtheit listenartiger Darstellungen, so erfährt man u.a., dass es eine der platzsparendsten Darstellungen ist.

2.2.3 TreeBrowser

Der TreeBrowser ist heute eine der populärsten Darstellungsformen für Hierarchien. Er ist bekannt aus dem Windows Explorer¹², der tagtäglich von Millionen

¹²Der Windows Explorer ist ein Dateimanager, der von Microsoft für Windows entwickelt wurde. <http://www.microsoft.com/germany/windows/>

von Windows-Usern benutzt wird. Der TreeBrowser ist eine Kombination aus einer listenartigen Darstellung eines Knoteninhalts mit einer grafischen Darstellung der Baumtopologie in einem 2-teiligen Fenster. Im Prinzip ist er eine Synthese aus den Ansätzen 2.2.1 (Baum) und 2.2.2 (Liste).

Bei dem bekanntesten TreeBrowser „Windows Explorer“ wird in der linken Hälfte des 2-teiligen Fensters die Baumtopologie in einer sehr kompakten Treeview eingeblendet, mit der das gezielte Navigieren in einer Hierarchie durch Anklicken von Verzeichnissen möglich ist. Der Inhalt der Verzeichnisse wird im rechten Teil des Fensters in einer Listendarstellung angezeigt (vergleiche Abbildung 5).

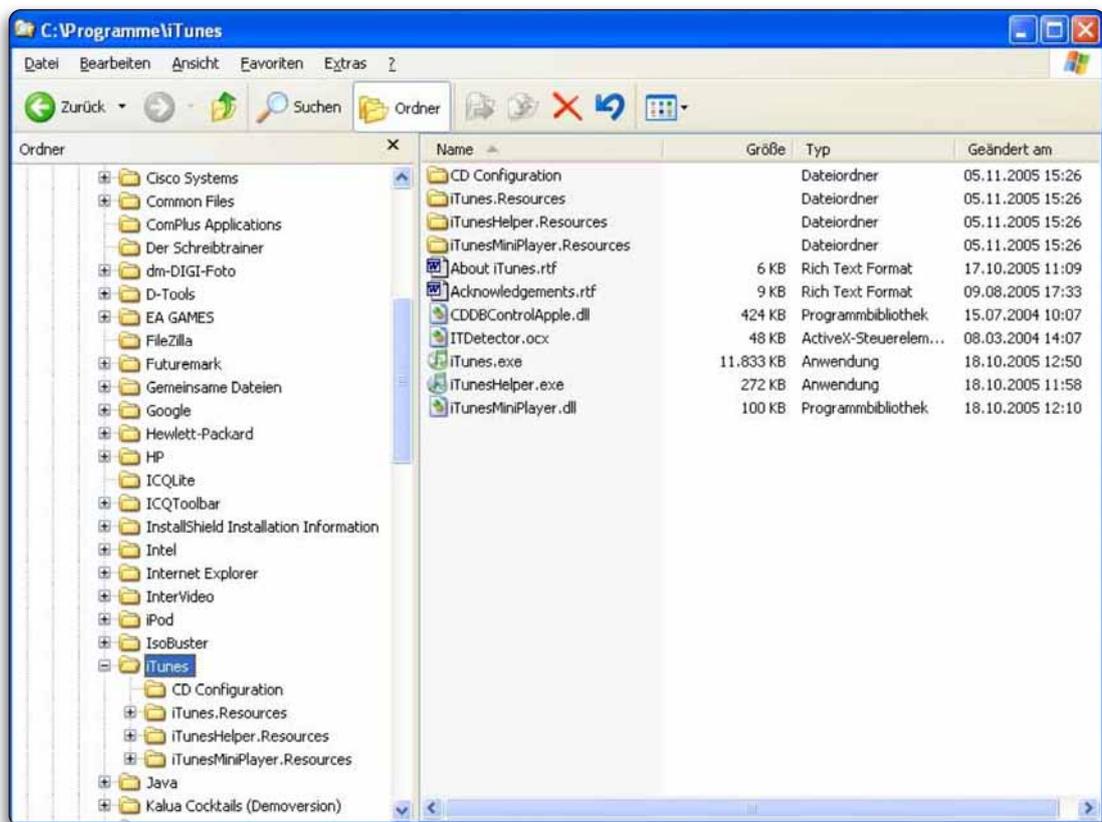


Abbildung 5: Windows Explorer

Die kompakte Darstellung der Topologie in der Treeview wird im Prinzip erst durch drei besondere Eigenschaften möglich:

Erstens werden in der Treeview nur Verzeichnisse eingeblendet, also Knoten mit Kindern, aber keine Blätter. Zweitens werden die Verzeichnisse nur zeilenweise eingeblendet, so wird der platzsparende Effekt einer Liste genutzt. Drittens wird die Topologie des Baumes in der Liste durch orthogonale Kanten und die Einrückung der Verzeichnisse dargestellt. Mit den Knoten bzw. den Verzeichnissen kann der User interagieren. Sind die Kind-Knoten von Interesse, so kann man durch Anklicken des

Knotens die Kind-Knoten entweder ein- oder ausklappen.

Somit hat man beim TreeBrowser eine sehr gute Navigation in der Breite des Baumes; das Explorieren von Dateien und Verzeichnissen auf einer Hierarchie-Ebene ist damit sehr anschaulich gelöst. Trotzdem kann auch hier der Überblick sehr schnell verloren gehen. Die Treeview kann bei einer Vielzahl von Kanten und Einrückungen „sehr entarten“ und damit ist die Baumtopologie nicht mehr auf einen Blick zu erkennen. Hierdurch wird die Orientierung in der gesamten Hierarchie erschwert.

Darüber hinaus ist es sehr interessant, dass bei Microsoft Windows häufig der „Arbeitsplatz“, bei dem keine Treeview zur Verfügung steht, zum Navigieren im Dateisystem verwendet wird. Hierauf wird in der Evaluation in Kapitel 5 genauer eingegangen.

2.2.4 Hyperbolic Browser

Beim Hyperbolischen Browser¹³ [Xiang u. a., 2005][Jetter, 2002] hat man versucht eine Visualisierung zu finden, bei der sowohl der Gesamtkontext, in dem sich ein Knoten befindet, als auch seine unmittelbaren Nachbarn, kombiniert mit Details, dargestellt werden können (siehe Abbildung 6).

Wie bei vielen anderen Visualisierungen ist auch hier der „Baum“ die „Mutter“ des Ansatzes. Um nun der Aufgabe von Overview und Zoom in einer Darstellung gerecht zu werden, wird hier eine sogenannte hyperbolische Ebene verwendet. Diese Ebene hat im Zentrum einen Fokusbereich, in dem einzelne Knoten sehr detailliert dargestellt sind. Die Vergrößerung in der Mitte der Ebene wirkt wie ein Zoom. Nach außen hin wird der Detailgrad zunehmend niedriger. Die Anzahl der dargestellten Knoten nimmt hingegen mit wachsendem Abstand zum Zentrum zu. So behält man bei der Benutzung des Hyperbolischen Browsers stets den Überblick über die gesamte Hierarchie.

Dieser Fisheye-Effekt¹⁴ führt zu einer Integration von Overview und Zoom in einer konstanten Darstellung.

In Usability Tests, in denen große Hierarchien exploriert wurden, fand der Hyperbolische Browser regelmäßig hohe Zustimmungswerte im Hinblick auf effizientes Auffinden der zu suchenden Daten. Jedoch wurde ebenso festgestellt, dass kleine Hier-

¹³Der hyperbolische Browser wird unter der Produktbezeichnung StarTree bei INXIGHT verwendet. <http://www.inxight.com/products/sdks/st/>

¹⁴Fisheye-Effekt, <http://wwwcs.uni-paderborn.de/cs/ag-szwillus/lehre/vw/seminare/fisheye/1Einfuehrung.html>

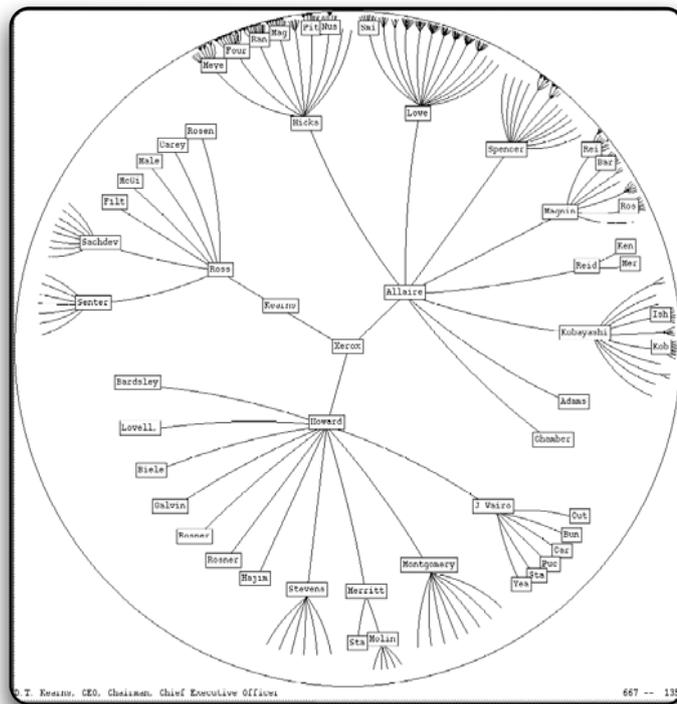


Abbildung 6: Hyperbolic Browser

archien mit einem TreeBrowser schneller zugänglich sind [Lamping u. a., 1995].

Eines der Keyfeatures im Hyperbolischen Browser ist die gleichzeitige Darstellung eines Fokusbereichs mit der Overview, um so den Kontext, in dem die Knoten des Fokusbereichs stehen, zu bewahren. Durch die Einführung eines zweiten Fokusbereichs im Bifokalen Hyperbolischen Browser erhoffte man sich eine weitere Verbesserung dieser Eigenschaft.

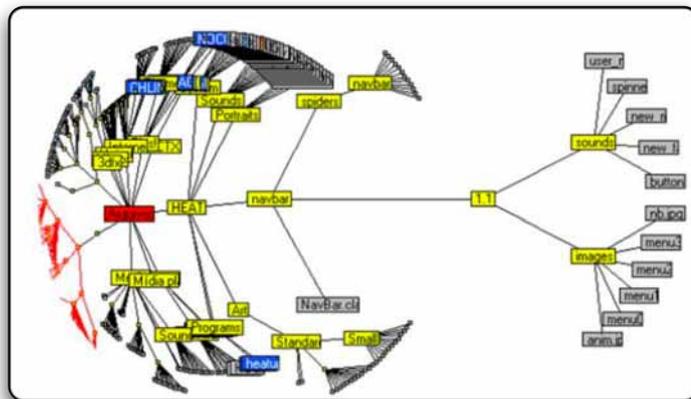


Abbildung 7: Bifocale Hyperbolic Browser

Der Bifokale Hyperbolische Browser in der Abbildung 7 hat ebenfalls eine hyperbolische Ebene mit dem bekannten Fisheye-Effekt im linken Bereich, der mit einer sehr detaillierten Darstellung im rechten Bereich gekoppelt ist. Der linke Bereich wird dabei als „Context Focus“, der rechte als „Detailed Focus“ bezeichnet. Wird der Fokus im linken Bereich aus der Gesamthierarchie neu gewählt, so wird gleichzeitig im rechten Bereich eine detaillierte Darstellung der im Fokus liegenden Knoten und deren Kontext angezeigt.

2.2.5 Treemap

Fast jedem PC-User ist der Windows Explorer bekannt. Wie schon in Kapitel 2.2.3 erörtert, ist der Windows Explorer ein TreeBrowser, der zum Browsen durch die Dateihierarchien eines Computer-Systems verwendet wird. Mit Hilfe dieses Tree-Browsers können die User immer nur ein Verzeichnis auf einmal explorieren, was in der täglichen Nutzung bei der Mehrzahl der Benutzer gut funktioniert. Jedoch hat der TreeBrowser einen großen Nachteil: Es ist nicht möglich, den gesamten Inhalt der Festplatte in einer globalen Übersicht darzustellen, da die Anzahl an Dateien und Verzeichnissen, die gleichzeitig angezeigt werden können, aufgrund des vorhandenen Bildschirmplatzes begrenzt ist. Aber ohne eine kompakte Übersicht über das gesamte Dateisystem kann man Fragen wie „welches Verzeichnis nimmt den meisten Platz auf meiner Festplatte ein?“ oder generell „warum ist meine Festplatte voll?“ schwer beantworten.

Anfang der 90er Jahre entwickelten Shneiderman und Johnson die sogenannten Treemaps [Shneiderman, 2006]. Diese Treemaps sind eine elegante Lösung zu dem oben beschriebenen Problem, indem sie den vorhandenen Bildschirmplatz sehr effizient nutzen. Das heißt, Treemaps werden zum Darstellen großer hierarchischer Strukturen eingesetzt. Um den verfügbaren Platz effizient auszunutzen, wird dieser in insich-geschachtelte Rechtecke zerteilt. Diese Rechtecke bilden eine Visualisierung für 2-dimensionale Datasets, wobei eine Dimension auf die Größe der Rechtecke und die andere auf die Farbe der Rechtecke abgebildet wird. Große Rechtecke beinhalten zusätzlich noch Label.

Ursprünglich wurden Treemaps für die Visualisierung von Daten auf einer Festplatte konzipiert, heute werden Treemaps aber in nahezu allen denkbaren Erscheinungsfeldern von großen Datenmengen, angefangen bei Finanzanlagen bis hin zu Sportnachrichten, eingesetzt.

Zum Erstellen der Treemap, das heißt für das Zeichnen der Rechtecke, ist ein Algorithmus zuständig. Dieser berechnet die Größe und den Ort der Rechtecke. Daher ist dieser Algorithmus das Hauptmerkmal einer jeden Treemap.

Zwei bedeutende Faktoren trieben über die Jahre hinweg die Evolution und Entwicklung des Treemap-Algorithmus voran:

- Zum einen gibt es das *Aspect Ratio*¹⁵, was soviel heißt wie „das Maximum von Höhe/Breite und Breite/Höhe“. Ein Rechteck mit einer quadratähnlichen Form hat ein sehr niedriges Aspect Ratio, da ein Quadrat ein Aspect Ratio von 1 hat, was der niedrigste mögliche Wert ist. Ein langes dünnes Rechteck hingegen hat einen sehr hohen Aspect Ratio-Wert. Im Gegensatz zu dem gut auszumachenden quadratischen Rechteck, ist ein langes dünnes Rechteck weniger gut zu sehen, zu selektieren, in seiner Größe zu anderen Rechtecken zu vergleichen und zu beschriften.
- Zum anderen ist da die *Vorhersehbarkeit der Platzierung eines Rechtecks*, da bei einigen Datensätzen die Reihenfolge der einzelnen Items eine Rolle spielt. Hierdurch können zum Beispiel hilfreiche Patterns erkannt oder aber einfach nur bestimmte Objekte wieder gefunden werden.

Der bekannteste Algorithmus ist der **slice-and-dice Algorithmus** [Shneiderman, 1992].

Dieser Algorithmus verwendet parallele Linien, um ein Rechteck, das einen Knoten repräsentiert, in kleinere Rechtecke, die die Kind-Knoten repräsentieren, zu zerteilen. Bei jedem Level der Hierarchie wird zwischen horizontalen und vertikalen Linien abgewechselt, um so die Ebenen voneinander unterschieden zu können. Dieser Algorithmus kriert vielleicht gerade wegen seiner Einfachheit oft Layouts, die sehr viele Rechtecke mit einem hohen Aspect Ratio beinhalten.

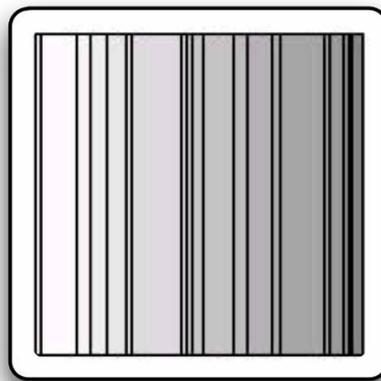


Abbildung 8: slice-and-dice Layout, Schattierung kennzeichnet Reihenfolge, die beibehalten wird [Bederson u. a., 2002]

Aufgrund dieses Nachteils des slice-and-dice Algorithmus wurden einige andere Algorithmen entwickelt, welche versuchen, das Problem des hohen Aspect Ratio in den

¹⁵aspect ratio = das Seitenverhältnis der Kanten eines Rechtecks zueinander

Griff zu bekommen.

Der SmartMoney Map of the Market [Wattenberg, 1999] zum Beispiel verwendet **Clustered Treemaps**, welche einen simplen rekursiven Algorithmus verwenden, der den gesamten Aspect Ratio reduziert. Bruls, Huizing, und van Wijk [Bruls u. a., 2000] benutzen **Squarified Treemaps**, mit deren Hilfe das gleiche Ziel erreicht werden kann.

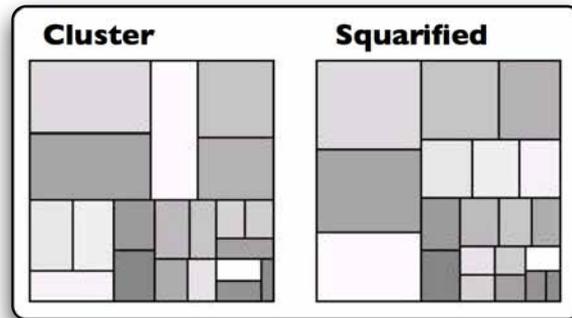


Abbildung 9: Layouts mit geringem Aspect Ratio, die Schattierung kennzeichnet die Reihenfolge, die aber nicht beibehalten wird [Bederson u. a., 2002]

Diese Algorithmen haben jedoch ein paar andere Nachteile. Beispielsweise verursachen Datenänderungen dramatische Veränderungen im Layout. Falls nun die Daten sekundenweise aktualisiert werden, wie bei einem Aktien-Bestands-Monitor, machen es Layoutveränderungen schwer, bestimmte Objekte zu verfolgen oder zu selektieren. Genauso können durch die Layoutveränderungen die schon erwähnten Patterns nicht mehr erkannt werden.

Grundsätzlich kann man sagen, dass die bisher vorgestellten Konzepte entweder einen niedrigen Aspect Ratio haben oder aber die Reihenfolge der Objekte beibehalten. Demnach haben diese zwei Faktoren ein invertiertes Verhältnis zueinander.

Um die Problematik des invertierten Verhältnisses zu mindern, entwickelten Shneiderman und Bederson die **Ordered und Striped Treemaps** [Bederson u. a., 2002], die einen Kompromiss aus den bisherigen Lösungsansätzen bilden und damit ein relativ niedriges Aspect Ratio haben und zugleich die Anordnung bei einer Datenaktualisierung größtenteils beibehalten. Zusätzlich wurde die **Quantum Treemap** [Bederson u. a., 2002] vorgestellt. Hier kann den Rechtecken eine vorgelegte Größe und Form gegeben werden, um Objekte wie Thumbnails, Photos oder Papers in den Rechtecken einer Treemap darzustellen. Shneiderman und Bederson benutzen die Quantum Treemap, um einen Photo Browser zu erstellen. Dieser Browser, namens PhotoMesa¹⁶ [Bederson, 2001], stellt eine große Anzahl an Thumbnails mit Metadaten, geclustert zu kleinen Treemaps, dar. PhotoMesa ist ein Zoomable User

¹⁶PhotoMesa kann man unter <http://www.cs.umd.edu/hcil/photomesa> downloaden

Interface (ZUI) [Perlin u. Fox, 1993], siehe Kapitel 3.4, mit dessen simplen Interaktionen die gewünschten Bilder schnell durch Zoomen, das heißt, viele Bilder in geringer Auflösung oder ein Bild in hoher Auflösung, gefunden werden können.

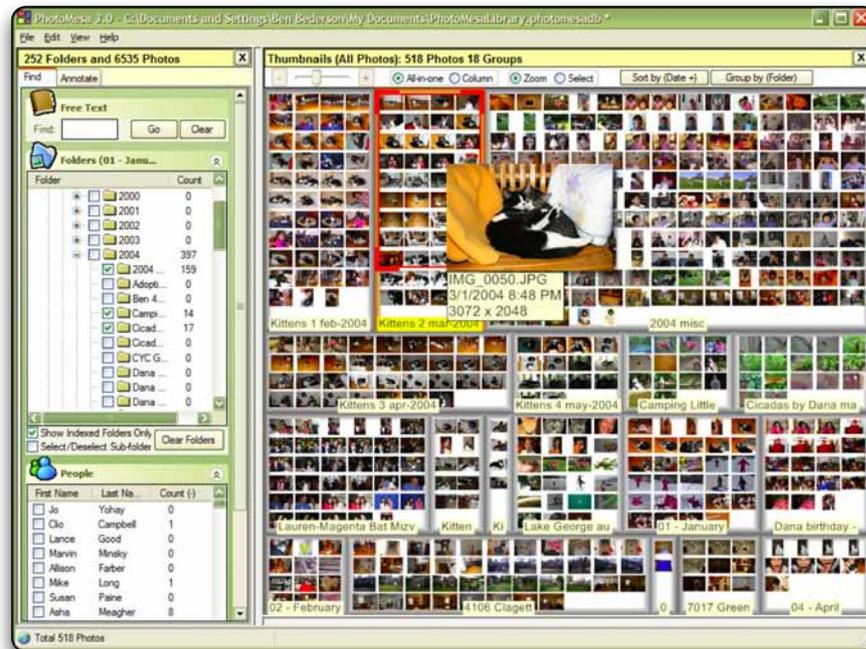


Abbildung 10: PhotoMesa benutzt Quantum Strip Treemaps, um Bilder in einem ZUI darzustellen

Blanch und Lecolinet [Blanch u. Lecolinet, 2006] stellen das Konzept der Zoomable Treemaps (ZTM) vor. ZTMs erweitern traditionelle Treemaps durch das ZUI Paradigma. Dementsprechend sind diese ZTMs multi-skalierbar, das heißt, verschiedene Detailgrade können definiert werden, und daher ist es möglich, sehr große Baumstrukturen zu explorieren. Traditionelle ZUIs lassen den Benutzer direkt und fortlaufend mit dem Informationsraum durch Interaktionstechniken wie Zooming und Panning interagieren. Für die ZTM wurden eigene Multi-Scale Navigationstechniken kreiert, die durch die Struktur der Treemaps unterstützt werden.

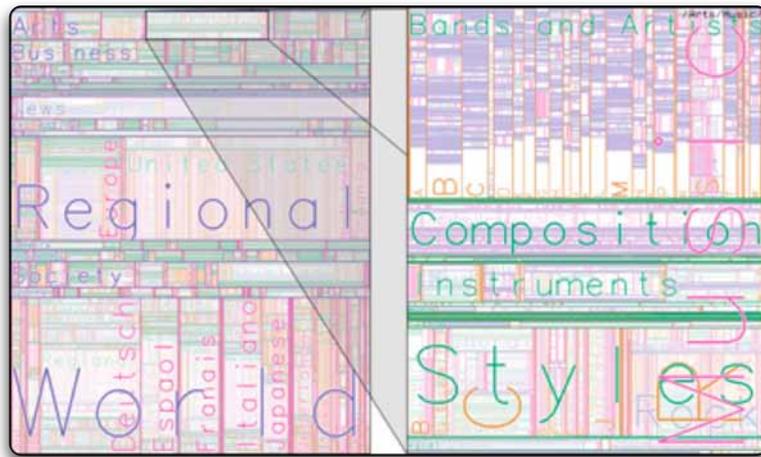


Abbildung 11: Baum in einer Zoomable Treemap dargestellt, wobei links der Baum in der Overview und rechts ein detaillierter Ausschnitt des Baums zu sehen ist [Blanch u. Lecolinet, 2006]

2.2.6 Cone Tree

Als eine weitere Visualisierungsvariante zur Darstellung von Hierarchien bietet sich der Cone Tree [Robertson u. a., 1991] an. Auch hier wurde die Grundidee „Baum“ weiterentwickelt: In einer dreidimensionalen Ansicht, wobei sich die Baumwurzel an der Decke des Raumes befindet, soll der User die gesuchten Daten schnell und insbesondere unter ständiger Wahrung des Überblicks finden.

Jeder Teilbaum ist ein Kegel. Daher auch der Name Cone Tree, was soviel wie Kegel-Baum heißt. Die Spitze des Kegels ist die Wurzel der Hierarchie. Auf dem Umkreis der Kreisfläche ist die Knotenebene, auf der die Kind-Knoten angebracht sind. Alle Kegel sind genau gleich hoch, das heißt, die Gesamthöhe des Raums wird durch die Anzahl der Ebenen geteilt, um die Höhe eines einzelnen Kegels zu ermitteln.

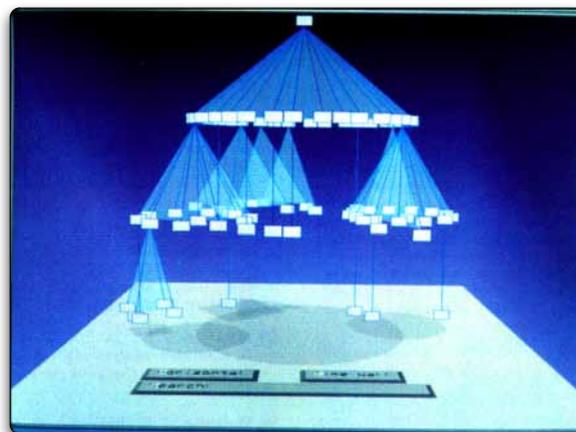


Abbildung 12: Cone Tree [Robertson u. a., 1991]

Wird ein Knoten im Cone Tree durch Klick selektiert, so rotiert der komplette 3D-Baum, so dass jeder Knoten vom Selektionspunkt bis hin zur Wurzel „gehighlightet“ wird und nach vorn kommt.

So kann die komplette Struktur einer Hierarchie auf einmal visuell dargestellt werden. Durch die Animation und deren visuellen Effekt versteht man auch sehr komplexe Strukturen einer Hierarchie relativ schnell und einfach.

Allerdings muss festgestellt werden, dass der ConeTree eher für „nichtausbalancierte“ Hierarchien geeignet ist. Bei ausbalancierten Hierarchien ist die Struktur nicht immer sehr deutlich festzustellen. Zudem sollte die Hierarchie nicht mehr als tausend Knoten und zehn Ebenen umfassen, andernfalls besteht die Gefahr, dass der Überblick verloren geht.

2.2.7 SpaceTree

Wie die meisten der beschriebenen Visualisierungen basiert auch der SpaceTree [Plaisant u. Grosjean, 2002] auf dem Darstellungsprinzip des Baums. Durch das Erweitern des klassischen Baums in ein Zoomable User Interface und das Hinzufügen von verbesserten Animationsprinzipien wurde ein neuer interaktiver TreeBrowser auf Basis der traditionellen Visualisierung geschaffen.

Durch die Zooming Umgebung werden die Äste des Baums dynamisch anhand des zur Verfügung stehenden Bildschirmplatzes ausgebreitet. Das heißt, wenn ein Ast nicht vollkommen angezeigt werden kann, da zu wenig Platz vorhanden ist, wird dieser durch ein „Preview Icon“ repräsentiert. Das Icon entspricht der Form eines gleichschenkligen Dreiecks. Die Schattierung des Dreiecks ist proportional zur Gesamtanzahl der Knoten im Subtree. Während die Höhe des Dreiecks die Tiefe des Subtrees verkörpert, kann man anhand der Länge der Basis die durchschnittliche Breite des Subtrees ablesen. Dementsprechend ist durch die Form und Farbgebung des Dreiecks eine Vorschau des jeweiligen Subtrees gegeben.

Der Benutzer kann seinen „Focus of Interest“ durch schrittweises Öffnen der Äste verfeinern. Dieses Navigieren kann der Benutzer entweder durch Klicken auf die einzelnen Knoten erreichen oder er benutzt die Pfeiltasten, um unter den Geschwistern sowie Vor- und Nachfahren eines Knotens hin- und her zu wechseln.

Mit Hilfe einiger Layout-Optionen kann der Abstand zwischen den Knoten, sowie die Anordnung der Knoten und auch das „Preview Icon“ dem Use-Case angepasst werden.

Der SpaceTree besitzt ebenfalls eine Such- und Filter-Funktion. Sobald der Benutzer einen Query eingibt, werden im Baum die Stellen des Ergebnisses markiert. So kann

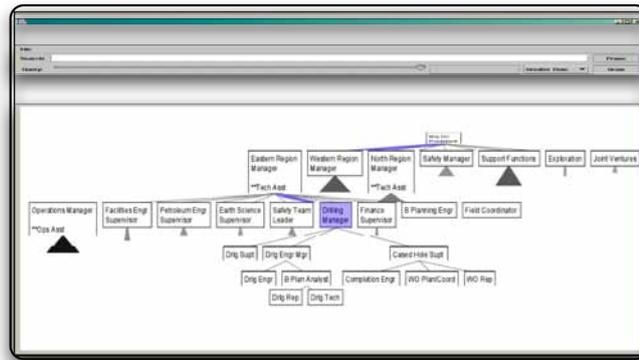


Abbildung 13: SpaceTree [Plaisant u. Grosjean, 2002]

der Benutzer entweder direkt zu diesen Stellen navigieren oder er kann sich durch einen Klick auf den „Prune“ Button den Baum in einer gefilterten Ansicht ansehen, die nur die Pfade zu den entsprechenden Knoten anzeigt.

3 MedioVis

3.1 Projekt MedioVis

MedioVis ist ein Forschungsprojekt der Arbeitsgruppe Mensch-Computer Interaktion¹⁷ des Fachbereichs Informatik und Informationswissenschaften der Universität Konstanz, das in enger Kooperation mit der Bibliothek der Universität Konstanz durchgeführt wird [Grün u. a., 2005]. Ziel dieses Projekts ist es, neue Interaktions- und Visualisierungstechniken für den Umgang mit komplexen heterogenen Informationsräumen zu entwickeln. Die Techniken werden iterativ in ein für andere Bibliotheken oder wissenschaftliche Einrichtungen frei verfügbares und nachnutzbares System integriert.

Das Projekt MedioVis ist aus den drei Vorgängerprojekten INSYDER [Mußler u. Reiterer, 2000], INVISIP [Haist u. Göbel, 2002] und VisMeB [Reiterer u. Klein, 2003] entstanden.

INSYDER¹⁸ war ein Teilprojekt des EU-Programms ESPRIT und wurde mit dem Ziel entwickelt, kleinen und mittelständischen Unternehmen das Suchen, Visualisieren und Analysieren von wirtschaftlich relevanten Daten aus dem Internet zu erleichtern. So wurde mit Hilfe von linguistischen Recherche-Verfahren sowie verschiedenen Ergebnisdarstellungen, wie Result Table, Scatterplot, Bar Chart und Segment View eine intuitive, visuelle Umgebung geschaffen.

Im darauf folgenden EU-Projekt INVISIP¹⁹ wurde ein Metadaten-Browser, der im wesentlichen auf dem Framework von INSYDER basierte, zur Suche und Analyse von Geo-Metadaten entwickelt und evaluiert. INVISIP sollte Standortentscheidungen und Folge-Prozesse begleiten und unterstützen.

Anschließend entstand ein aus verschiedenen Visualisierungen kombinierbarer Visueller Metadaten Browser mit dem Projektnamen VisMeB²⁰. Ursprünglich wurde VisMeB als Experten-Tool konzipiert. Erst durch die enge Kooperation mit der Universitätsbibliothek wurde VisMeB an die Bedürfnisse von Gelegenheitsnutzern bzw. Bibliotheksnutzern angepasst.

Unter dem Projektnamen MedioVis wurde die angepasste Version zum Testbetrieb in der Universitätsbibliothek Konstanz freigegeben. MedioVis sollte dabei den Bibliotheksbenutzern eine Alternative bei der Suche und Exploration des Mediothekbe-

¹⁷AG Mensch-Computer Interaktion: <http://hci.uni-konstanz.de>

¹⁸INSYDER: INternet SYstème DE Recherche

¹⁹INVISIP: INformation VISualization in Site Planning

²⁰VisMeB: Visueller Metadaten Browser

standes zum bestehenden Online-Suchsystem KOALA²¹ bieten. Dieser Live-Einsatz von MedioVis in der Bibliothek ist extrem wichtig. So wird MedioVis tagein tagaus unter Realbedingungen getestet. Mit Hilfe von DROID²², einem Protokollierungssystem für Interaktionsvorgänge, werden Nutzungsdaten gesammelt. Um eine hohe Gebrauchstauglichkeit für den Endnutzer zu erzielen, wird MedioVis ständig verbessert, neue Interaktions- und Visualisierungstechniken werden entwickelt und integriert. Die während des operativen Testbetriebs gesammelten Live-Daten und regelmäßige Evaluationen helfen dann dabei, die neu-integrierten Konzepte hinsichtlich ihrer Benutzerfreundlichkeit zu überprüfen.

MedioVis wird dabei von der Deutschen Forschungsgemeinschaft²³ im Rahmen des Förderprogramms für Wissenschaftliche Literaturversorgungs- und Informationssysteme (LIS 4) gefördert.

3.2 Motivation

Die Bibliothek der Universität Konstanz verfügt über eine große Menge an intellektuell gepflegten Daten mit einer beachtlichen Medienvielfalt. Im Bestand der Mediothek, die eine Untermenge der Universitätsbibliothek ist, befinden sich eine Vielzahl elektronischer und multimedialer Titel, wie DVDs, CDs, Videomitschnitte, Tonträger, Mikrofiches, etc.. Diese multimedialen Titel der Mediothek stellen ein beutendes Serviceangebot der Bibliothek dar. Studenten, Lehrpersonal und Wissenschaftler sowohl im Bereich der Theater-, Film- und Medienwissenschaften als auch in der Fremdsprachenausbildung nutzen dieses Serviceangebot für wissenschaftliche Zwecke, aber auch zur Unterhaltung.

Aus Gesprächen mit Benutzern und deren Beobachtungen bei der Verwendung des traditionellen bibliographischen Webkatalogs bzw. OPACs „KOALA“ der Bibliothek der Universität Konstanz erwuchs die Motivation, ein neues System zu schaffen, das durch spezielle Visualisierungen und neuartige Interaktionskonzepte ein effizientes Suchen und Browsen ermöglicht. Als Hauptproblem kristallisierte sich aus den Gesprächen die mangelnde Unterstützung bei der Wahl eines Titels mit Hilfe des traditionellen bibliographischen Katalogs heraus. So fehlen beispielsweise zu Filmen entscheidungsrelevante Informationen, die das Bibliothekswesen nicht berücksichtigt und die daher im KOALA-System nicht dargestellt werden. Deshalb ist der Benutzer gezwungen, eine umfangreiche Recherche im Internet unter Einbeziehung verschie-

²¹KOALA: Konstanzer Ausleih- und Anfrage-System <http://ub.uni-konstanz.de/koala>, basiert seit Januar 2005 auf dem Library Management System LIBERO <http://www.libero.com.au>

²²DROID: Dynamic Remote Operation Incident Detection <http://hci.uni-konstanz.de/index.php?a=research&b=projects&c=15851850&lang=de>

²³DFG: <http://www.dfg.de>

denster verstreuter Informationsquellen, wie IMDb²⁴, Wikipedia²⁵ oder Amazon²⁶, durchzuführen.

MedioVis verfolgt das Ziel, den traditionellen bibliographischen Katalog mit umfassenden textuellen und multimedialen Informationen, wie Inhaltsangaben, Postern, Porträts, Videosequenzen und Biografien, aus verschiedenen Quellen anzureichern [Jetter, 2007]. So entsteht ein komplexer und heterogener Informationsraum, der in einer sinnvollen Darstellung präsentiert werden soll.

Suchergebnisse im KOALA-System, aber auch bei vielen Suchanwendungen im Internet, wie zum Beispiel Google²⁷, werden typischerweise in wenig strukturierten Listendarstellungen visualisiert. Vor allem bei der Orientierung im Informationsraum, beim gezielten Auffinden von Inhalten und deren Gegenüberstellung, sind Tabellen aufgrund ihrer Interaktivität und ihres flexiblen Eingehens auf die Informationsbedürfnisse des Benutzers der Listendarstellung nachweislich überlegen [Gerken, 2004]. Ferner kann eine interessante Analogie zwischen dem Browsen in einer realen Bibliothek und dem Browsen in einer Tabelle einer digitalen Bibliothek festgestellt werden [Wake u. Fox, 1995]: Titel in einer realen Bibliothek sind semantisch in verschiedene Regale eingeordnet und sind damit räumlich voneinander unterteilt. Auch in einer Tabelle werden Titel räumlich voneinander getrennt dargestellt, indem sie nach einem Attribut sortiert werden. Das Browsen geschieht in der realen Bibliothek durch Ablaufen der Regale, in der Tabelle hingegen durch Überfliegen der einzelnen Zeilen. So können die räumlich voneinander getrennten Titel analog in der realen Bibliothek als auch in der Tabelle verglichen werden. Die Tabelle besitzt jedoch einen entscheidenden Vorteil: Der Benutzer kann die Tabelle nach für ihn wichtigen Kriterien sortieren.

Die Inhalte, optionale Zusatzinformationen oder multimediale Daten wie Videos oder PDFs, werden in gängigen Webanwendungen als „details on demand“ [Shneiderman, 1996] in räumlich weit entfernten Bildschirmbereichen, überlappenden Fenstern oder auf isolierten Webseiten angeboten. Bei deren Abruf geht der visuelle Kontext verloren und zudem findet ständig ein kognitiv belastender Wechsel der Modalität statt [Müller u. Klein, 2002]. Navigationswege und -ziele müssen vom Benutzer bei der Suche kontinuierlich über viele Medienbrüche und Wechsel im Layout hinweg mental vergegenwärtigt werden, was die Effektivität, Effizienz und die Benutzerzufriedenheit beeinträchtigt.

²⁴IMDb: Internet Movie Database, <http://www.imdb.com>

²⁵Wikipedia: <http://de.wikipedia.org/>

²⁶Amazon: <http://www.amazon.de>

²⁷Googel: www.google.de

Vor diesem Hintergrund scheint eine interaktive Visualisierung optimal zu sein, die heterogene Informationen ohne Medienbrüche, Modalitätswechsel und den Verlust des visuellen Kontexts darstellt: Das Zoomable User Interface (ZUI), das näher in Kapitel 3.4 beleuchtet wird, entspricht diesen Anforderungen.

Daher wurde im Rahmen von MedioVis die HyperGrid, die in Kapitel 3.5 vorgestellt wird, entwickelt. Die HyperGrid stellt dabei eine Kombination aus Tabelle und ZUI dar.

3.3 Anwendungskontext

MedioVis ist ein Framework, bei dessen Entwicklung viel Wert auf eine möglichst flexible Anbindung neuer Anwendungskontexte gelegt worden ist. Damit soll eine hohe Kompatibilität über verschiedene Projekte, Domänen und Versionen hinweg gewährleistet werden. So ist es möglich, verschiedene Ausprägungen einer Visualisierung vergleichen oder den Einsatz einer Visualisierung in verschiedenen Anwendungsdomänen evaluieren zu können.

Im Laufe der Zeit wurde MedioVis in den unterschiedlichsten Bereichen verwendet. So wurden geographische Metadaten im Projekt INVISIP, über bibliographische Sammlungen wissenschaftlicher Publikationen, wie DBLP²⁸ und DELOS²⁹, bis hin zu sehr dynamischen Online-Diensten, wie IMDb³⁰, Yahoo³¹, Ebay³² oder die P2P-Tauschbörse³³ mldonkey³⁴ angekoppelt. Der Hauptanwendungsbereich des Projekts MedioVis ist jedoch die Universitätsbibliothek Konstanz³⁵ bzw. deren Untermenge „Mediothek“. Die Bibliotheks- bzw. Mediotheksdaten, die mit dem Library Management System LIBERO im MAB2-Format³⁶ verwaltet werden, werden zusätzlich mit gespiegelten Film- und Schauspielerdaten der Internet Movie Database IMDb, sowie gespiegelten Daten von Wikipedia³⁷ und mit GoogleMaps³⁸ angereichert.

²⁸DBLP: Digital Bibliography & Library Project, mit über 600000 indexten Artikeln. <http://dblp.uni-trier.de>

²⁹DELOS: <http://www.delos.info>

³⁰IMDb: Internet Movie Database, mit Metadaten zu mehr als 500.000 verschiedenen Filmen und über 1,8 Millionen Personen, <http://www.imdb.com>

³¹Yahoo: <http://www.yahoo.de>

³²Ebay: <http://www.ebay.de>

³³P2P: Peer-to-Peer Netzwerke haben eine gleichgestellte, dezentrale Kommunikation ohne die klassische Client-Server Rollenverteilung.

³⁴mldonkey: <http://mldonkey.sourceforge.net/>

³⁵Bibliothek der Universität Konstanz, <http://www.ub.uni-konstanz.de/>

³⁶MAB2: Maschinelles Austauschformat für Bibliotheken in Version 2

³⁷Wikipedia, <http://de.wikipedia.org>

³⁸GoogleMaps, <http://maps.google.de/>

3.4 Zoomable User Interface

Ein Zoomable User Interface (ZUI) ist eine spezielle zoombare Benutzeroberfläche, die durch das Darstellen von Informationsobjekten in einer räumlichen Umgebung eine analoge Navigation zur physischen Umwelt schafft. Die im Rahmen von MedioVis entwickelte HyperGrid greift Prinzipien des Konzeptansatzes eines ZUI auf und soll daher im Folgenden näher erläutert werden.

Raskin stellt in seinem Buch „The Human Interface“ [Raskin, 2000] die „ZoomWorld“ ein neues Konzept, das auf dem Zoomable User Interface bzw. seinem „Zooming Interface Paradigm“ (ZIP) basiert, vor. Bei der ZoomWorld können Informationsobjekte unterschiedlicher Skalierung auf einer unbegrenzten zweidimensionalen Informationslandschaft platziert werden (siehe Abbildung 14). Die Informationsobjekte sind verschiedenster Natur, wie z.B. Texte oder Bilder, und verschiedenster Herkunft, wie z.B. aus dem lokalen Datenbestand oder aus dem Internet. Die Navigation in der Informationslandschaft erfolgt durch Zooming und Panning. Dabei verändert der Benutzer durch Zooming die Granularität und den Skalierungsgrad der dargestellten Objekte. Bildlich gesprochen nähert sich der Benutzer beim Herinzoomen den Objekten und taucht somit in die Informationslandschaft ein. Beim Herauszoomen entfernt er sich wieder. Durch Panning verschiebt der Benutzer den aktuell sichtbaren Ausschnitt, um so Objekte außerhalb des momentanen Sichtfeldes erblicken zu können.

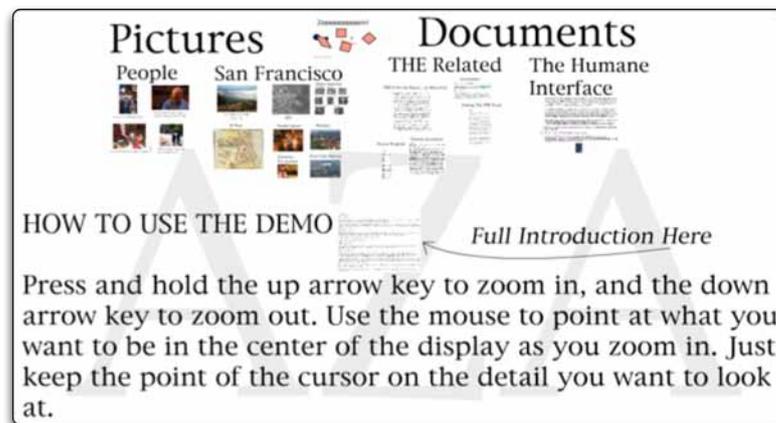


Abbildung 14: Jef Raskins Vision ZoomWorld [Raskin, 2000]

Grundsätzlich muss man zwischen verschiedene Zoom-Techniken unterscheiden. Besonders interessant sind dabei das geometrische und das semantische Zooming. In der ZoomWorld ist das Zoomen rein geometrisch: Vorher klein dargestellt Objekte werden beim Einzoomen optisch vergrößert dargestellt und somit verändert sich deren Skalierung. Die Informationen und ihre visuelle Präsentation bleiben aber unverändert (siehe Abbildung 15). Vergleichbar ist das geometrische Zooming mit dem

Auslösen des Zooms an einer Kamera.

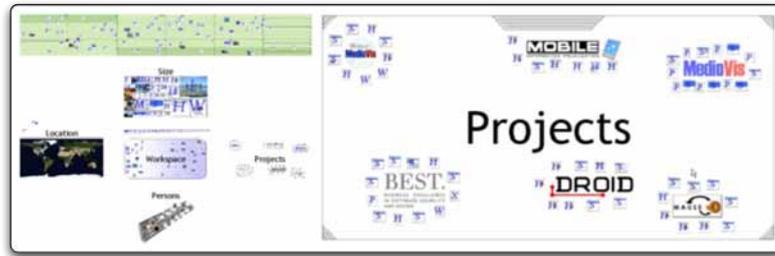


Abbildung 15: Geometrischer Zoom anhand von ZOIL illustriert - links Übersichtsdarstellung, rechts geometrischer Zoom auf Bereich „Projects“ [König, 2006]

Beim semantischen Zooming hingegen ändert sich nicht die Skalierung und damit die Größe sondern der Inhalt eines Informationsobjekts [Perlin u. Fox, 1993]. Dabei wird jedoch meistens semantisches mit geometrischem Zooming kombiniert [Gerken, 2006]. Durch den geometrische Zoom wird, wie beschrieben, der Platz, der einem Objekt zukommt, vergrößert. So kann der durch den semantischen Zoom sich verändernde Inhalt des Objekts den anwachsenden Platz optimal ausnutzen. Damit wird in jeder Zoomstufe eine andere Repräsentation für ein Informationsobjekt gewählt. Beispielsweise wird bei einem PDF-Dokument, wie in Abbildung 16, zuerst nur ein Symbol angezeigt. Mit zunehmender Größe erscheint beim Hineinzoomen nach und nach mehr Inhalt. Zunächst werden die Metadaten, wie Titel und Autor, eingeblendet, bis dann später der komplette Text sichtbar wird und das Dokument im Idealfall bearbeitbar wird.

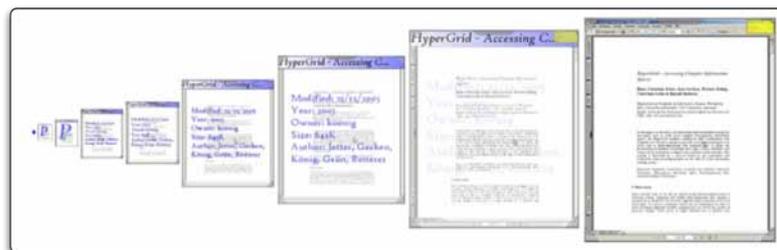


Abbildung 16: Semantischer Zoom anhand einer PDF-Datei in ZOIL illustriert [König, 2006]

Durch das ZUI soll das kognitiv belastende Hin- und Herwechseln zwischen verschiedenen Fenstern vermieden werden. Durch das Zooming und Panning müssen lediglich der Fokus und die Skalierung immer wieder neu definiert werden, der Kontext bleibt aber erhalten. Zudem greift das ZUI Prinzipien der realen Welt auf. So muss nicht ein Menue-Eintrag, Icon oder Button betätigt werden, um die dargestellten Informationen zu verändern. Der Benutzer kann sich, wie in der Realität, direkt auf interessante Elemente hinzu bewegen oder seine Umgebung aus einem größeren

Abstand überblicken. So ist eine „Orientierung in abstrakten, unsichtbaren Hierarchien mit Hilfe von Positionsangaben durch symbolische Zeichenketten wie z.B. Pfade oder URLs, [...] nicht mehr notwendig“ [Reiterer u. a., 2005].

3.5 Die HyperGrid

Mit der HyperGrid wurde eine Synthese aus den traditionellen Konzepten der Tabelle und des Browsers und dem ZIP Raskins geschaffen [Reiterer u. a., 2005] [Jetter u. a., 2005]. Die HyperGrid ist eine Tabellenvisualisierung kombiniert mit semantischen Zoomtechnologien. Durch die Erhaltung einer ordnenden Tabellenstruktur können komplexe heterogene Informationsräume leicht exploriert werden. Zudem ermöglicht diese zoombare Tabelle eine platzökonomische Darstellung, die speziell für analytische Vergleiche und Gegenüberstellungen geeignet ist. Die HyperGrid kann dabei sowohl in zeitgenössischen Informationssystemen, als auch in visionären Umgebungen, wie der ZoomWorld oder ZOIL [König, 2006], beheimatet sein. Der Name der HyperGrid setzt sich aus zwei Schlüsselkomponenten zusammen:

„Hyper“ steht für die Fähigkeit untereinander verwobene Hypertexte- oder Hypermedia-Inhalte in einer Modalität darzustellen. „Grid“ steht für die klare zwei-dimensionale Gitterstruktur, die zur interaktiven Visualisierung verwendet wird. Obwohl viele Aspekte der HyperGrid auf den ersten Blick an eine traditionelle Tabelle oder bekannte zoombare Tabellen wie die TableLens [Rao u. Card, 1994] oder DateLens [Bederson u. a., 2004] erinnern lässt, unterscheiden sich die semantische Gitterstruktur und die Zellen der HyperGrid davon drastisch. Im Folgenden soll dies anhand der Nutzung der Mediodaten demonstriert werden.

Die Mediodaten besitzt eine Vielzahl von Titeln. Die Metadaten dieser Titel werden im MedioVis MediaWarehouse mit Informationen verschiedener Quellen, siehe Kapitel 3.3, angereichert. Daher besitzen die Titel der Mediodaten eine beachtliche Anzahl an Attributen und Metadaten, durch die sie beschrieben werden: Titel, Originaltitel, Untertitel, Poster, Personen, Jahr, Details, Sprache, Film, Herausgeber, Stadt, Institutionen, Medientyp, Ausleihstatus, Fachgebiet, Signatur, Format, Anmerkung und Standort.

Wie in traditionellen Tabellen entspricht in der HyperGrid eine Zeile einem Titel. Jedoch ist nicht jede Spalte einem Attribut zugewiesen, sondern jede Spalte repräsentiert einen Aspect of Interest (AOI) [Rüger u. a., 1996]. Der AOI stellt einen thematischen Gesichtspunkt oder einen Oberbegriff innerhalb des gesamten Informationsraumes dar. In einer sorgfältigen Modellierung, die detailliert von [Reiterer u. a., 2005] beschrieben wird, wurden alle Attribute eines Titels in einem mathematischen Attributraum räumlich angeordnet. So konnten semantisch ähnliche Cluster zusammengeführt werden, um anschließend eine gezielte Darstellung des vom Benutzer gewünschten Teilausschnitts aus der Gesamtinformation zu ermöglichen. Bei

den Titeln der Mediothek wurden drei Aspects of Interest identifiziert. Diese drei AOIs „Titel“, „Beschreibung“ und „Exemplar“ wurden den ersten drei Spalten der HyperGrid zugeordnet. In der Abbildung 17 kann man die HyperGrid mit den drei AOIs, die den verschiedenen Spalten zugewiesen wurden, erkennen.

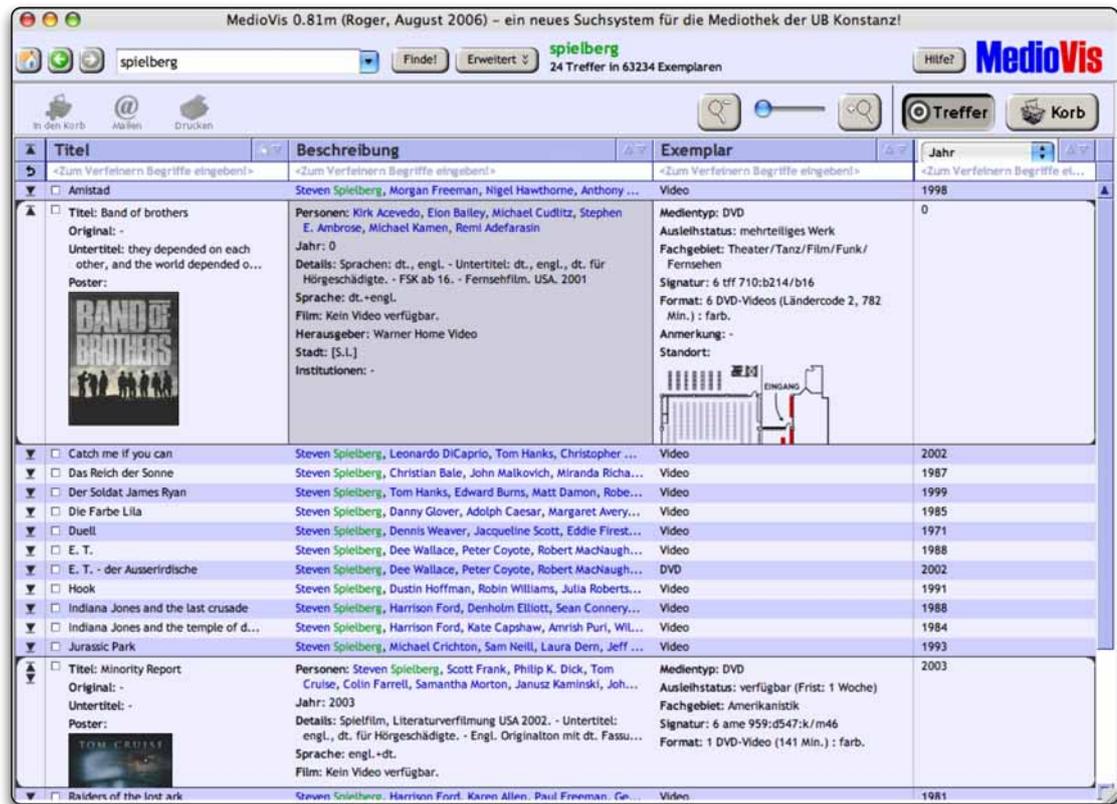


Abbildung 17: Die HyperGrid

Jeder Zelle in einer Spalte werden mehrere Attribute zugeordnet. Daher sind einzelne Tabellenzellen nicht mehr nur statische Informationsträger eines Wertes mit einem Datentyp, sondern die Zellen beinhalten eine Vielzahl an Informationen und Datentypen. Jede Tabellenzelle wird als ein möglicher Ausgangspunkt für die Exploration des Informationsraumes gesehen, wobei die Zeile das Object of Interest (OOI) und die Spalte den AOI definiert. Die dritte Dimension, die vom Benutzer kontrolliert wird, ist der Degree of Interest (DOI) [Rao u. Card, 1994]. Im Gegensatz zur ZoomWorld sind der AOI und der DOI nicht stetig, sondern „im Sinne des semantischen Zoomens abgestuft und in die Tabellenstruktur eingebettet, um die Freiheitsgrade bei der Benutzung im Interesse der Handhabbarkeit einzuschränken und die Erwartungskonformität zu steigern“ [Reiterer u. a., 2005].

Das semantische Zoomen wird durch das vom Benutzer kontrollierte Verändern des DOI erreicht. In der Modellierung des Attributraumes wurden die Attribute eines AOI zu sinnvollen Attributgruppen zusammengefasst. So kann der Informationszu-

wachs bei der Veränderung des DOI in Ebenen abgestuft werden. Durch Zoomen, das heißt durch Klicken des Benutzers mit der linken Maustaste in eine Zelle, wird der DOI des jeweiligen AOI in dieser Zelle erhöht. So ist der DOI für jede Zelle individuell wählbar. Der Klick löst einen kontinuierlichen Zoom in die Zelle aus, wobei diese auf Kosten der Nachbarzellen, unter Beibehaltung der Tabellenstruktur, durch einen animierten Effekt in Breite und Höhe wächst (siehe Abbildung 18). Durch einen Klick auf die rechte Maustaste kann wieder „herausgezoomt“ werden. Damit wird der Detailgrad bzw. DOI der dargestellten Information zu dem jeweiligen Titel reduziert. Zusätzlich zum „stufenweisen Erhöhen“ des DOI kann eine Zeile mit allen Zellen durch einen Klick auf den Flap-In/-Out-Button in der linken Spalte komplett gezoomt werden. Der Übergang von abstrakten Metadaten über detaillierte Metadaten hin zum Volltext stellt eine Variante des **semantischen Zooms** dar.

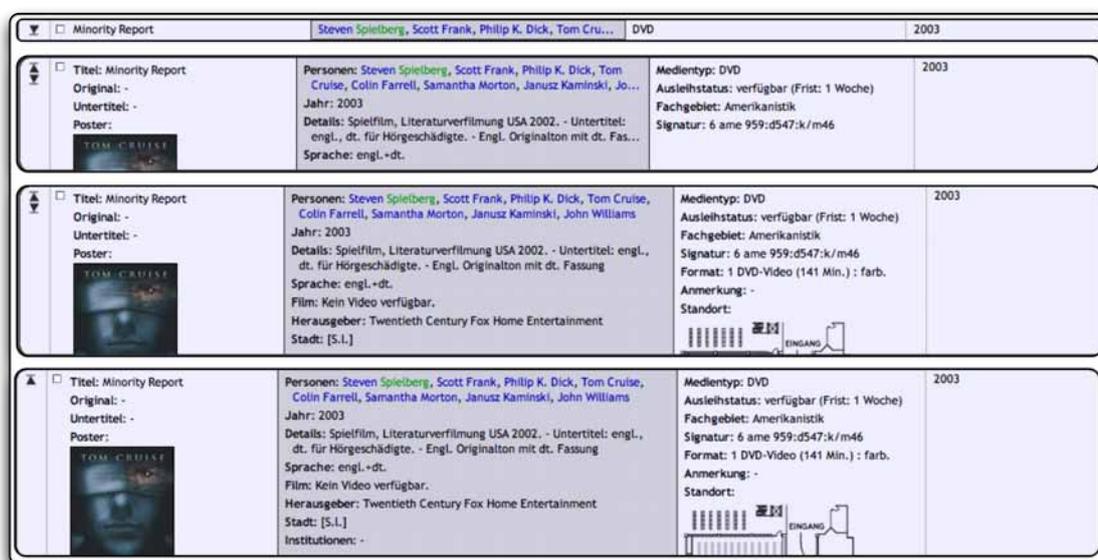


Abbildung 18: Die verschiedenen Zoomstufen des mittleren AOI in der HyperGrid

Alle Entitäten der Mediodaten, das sind z.B. Schauspieler, Autoren, Bücher, Filme, haben untereinander verschiedene relationale Beziehungen, wie 1-zu-1, 1-zu-n oder n-zu-n. Dies bedeutet, ein Film-Eintrag hat z.B. Verweise auf mehrere Schauspieler, während ein Schauspieler-Eintrag Verweise auf seine eigenen Filme enthält. In einer herkömmlichen WIMP-Oberfläche, wie die meisten Seiten des Webs³⁹, wird jede Entität durch eine eigene Webpage repräsentiert. Dies entspricht der traditionellen Art relationale Daten darzustellen. Das Hin- und Herwechseln zwischen den Entitäten Film und Schauspieler führt somit zum kompletten Verlust des visuellen Kontexts. So kann es zu einem starken Umbruch im Layout oder gar in der Modalität und folglich auch zu einer starken kognitiven Belastung des Benutzers kommen

³⁹Beispiel hierfür ist die IMDb: Hier hat jeder Film, aber auch jeder Schauspieler, eine eigene Webpage

[Müller u. Klein, 2002].

Im Informationsraum der angereicherten Mediodaten gibt es Objekte, wie zum Beispiel Schauspieler innerhalb einer Tabelle von Medien/Titeln, die in keinem konsistenten Kontext als eigene Zeile in der Tabelle aufgeführt werden können. Nun könnte man eine komplett neue nur Schauspieler-bezogene HyperGrid anzeigen, auf die der Benutzer z.B. bei Klick auf einen Schauspieler gelangt. Allerdings würde dies zu einem Verlust des kompletten Kontextes führen. Um dieses Problem zu umgehen wurde eine Browserkomponente in die HyperGrid-Zellen integriert. Dadurch können verlinkte Informationen, die sich nicht konsistent in den Tabellenkontext integrieren lassen, oder Inhalte, deren Darstellung die Zellgröße nicht zulässt, in einem überlagernden Browserfenster präsentiert werden. So fungieren alle Objekte, die nicht vollständig in der HyperGrid-Zelle dargestellt werden können, wie die aus dem HTML-Browser bekannten Hyperlinks. Durch Mausklick auf einen Hyperlink werden die zugehörigen Informationen in einem kleinen Browser-Fenster über der aufgerufenen Tabellenzelle dargestellt (siehe Abbildung 19). Dieses Browser-Fenster kann durch Klick mit der linken Maustaste auf den Fensterrahmen bis auf die Größe der kompletten HyperGrid vergrößert und durch Klick mit der rechten Maustaste wieder verkleinert werden. Bei Klick auf das „Kreuz“ wird das vorgelagerte Browser-Fenster geschlossen.

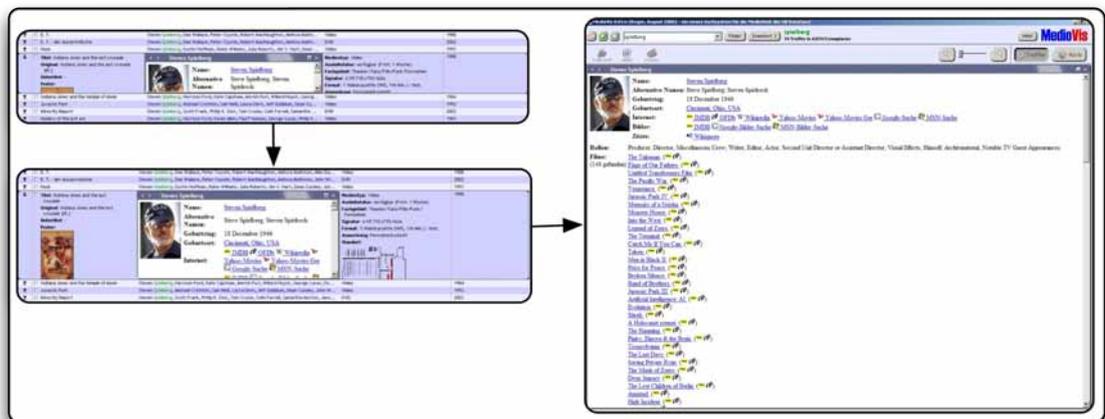


Abbildung 19: Die verschiedenen Zoomstufen der in die HyperGrid integrierten Browserkomponente

So können in das Browser-Fenster beliebige externe Inhalte, wie zum Beispiel Zusatzinformationen zu beteiligten Personen, eingebunden werden. Auch Webinhalte oder Webservices können so direkt integriert werden. Beispielsweise bekommt man durch einen Klick auf den Geburtsort einer Person in den Zusatzinformationen den Ort mit Hilfe von GoogleMaps auf einer Karte, wie in Abbildung 20 illustriert, angezeigt.



Abbildung 20: Die Darstellung von GoogleMaps innerhalb der HyperGrid

Komplexe, platzverbrauchende, multimediale Objekte, wie PDFs und Videos werden ebenfalls in dem vorgelagerten Fenster visualisiert (siehe Abbildung 21). Bei der Betrachtung der PDF-Dateien im Browser-Fenster ist der komplette Funktionsumfang des Adobe Acrobat Readers⁴⁰ zugänglich und verwendbar.



Abbildung 21: Die Darstellung von PDF-Dateien und das Abspielen von Filmen innerhalb der HyperGrid

Für die Nutzung der HyperGrid im Zusammenspiel mit der Mediothek ist eine direkte Anbindung an das bestehende Bibliothekssystem KOALA essentiell. Das Attribut „Ausleihstatus“ dient dabei als direkter Link auf die KOALA-Seite des jeweiligen Titels. Die KOALA-Seite öffnet sich, wie in Abbildung 22, im Browser-Fenster dieser Zelle und man kann den Titel z.B. vormerken oder andere Funktionen des KOALA-Systems verwenden.

⁴⁰Adobe Acrobat Reader wird zum Betrachten von PDFs verwendet, <http://www.adobe.com/de/products/acrobat/>



Abbildung 22: Direkter Zugriff auf das KOALA-System innerhalb der HyperGrid

Die Integration von Hypertext- und Hypermedia-Daten in die HyperGrid-Zellen durch die Browserkomponente führt dazu, dass die Zellen eine Art dynamisches Fenster in den Informationsraum sind und sich damit von herkömmlichen statischen Tabellenzellen enorm unterscheiden. So unterstützt die HyperGrid sowohl das browsing-orientierte, interessen geleitete „Stöbern“ mittels Zoomable User Interface und Browser-Konzept, als auch die analytische Suche durch strukturierte Darstellung der Suchergebnisse in der Tabelle mit der Möglichkeit zur Sortierung und Filterung. Die Zustände der Zellen bleiben auch bei der Anwendung der analytischen Methoden Sortierung und Filterung erhalten.

Das Sortieren wird durch Klick auf den Sortier-Button in dem entsprechenden Spaltenkopf ausgelöst. Als Sortierkriterium dient das erste Attribut innerhalb der Zelle. Das erste Attribut (Titel, Personen, Medientyp) hat die höchste Relevanz für den jeweiligen AOI. Ist ein anderes Attribut von besonderem Interesse, kann es aus der Tiefe des Informationsraumes an die Oberfläche gebracht werden. Hierfür ist es möglich, in der 4. Spalte, der sogenannten benutzer-adaptiven Spalte⁴¹, aus einem Drop-Down Menü das gewünschte Attribut auszuwählen und so die Spalte individuell anzupassen. Anschließend können die Suchtreffer anhand dieses Attributs beispielsweise sortiert und verglichen werden.

Als weiteres analytisches Werkzeug steht die Filterung zur Verfügung. Eingabefelder unterhalb der Spaltenköpfe ermöglichen es, die Suchresultate bezüglich des jeweiligen AOI und der eingegebenen Schlüsselwörter dynamisch zu filtern. Aus jeder Veränderung der Eingabe in das Filterfeld resultiert eine sofortige Aktualisierung der Ergebnismenge in der HyperGrid, wodurch der Benutzer ein direktes visuelles Feedback auf seine Eingabe erhält. Es werden somit nur noch Einträge dargestellt, die dem formulierten Filterkriterium, das aus den Schlüsselwörtern der Eingabefelder besteht, entsprechen. Am linken Rand der ersten Spalte befindet sich der Filter-Reset-Button. Durch einen Klick auf diesen wird das Filterkriterium jeder Spalte zurückgesetzt.

Die HyperGrid ermöglicht eine klare tabellarische Struktur, die das Vergleichen von Suchtreffern speziell durch Funktionen wie Sortierung, Filterung, Suche und Brow-

⁴¹Die benutzer-adaptive Spalte wird auch user-adjustable column (uac) genannte

sing unterstützt. Diese Funktionalitäten sind zur explorativen Erkundung großer Informationsräume unabdingbar [Shneiderman, 1996].

3.6 Hierarchien in MedioVis

Im Datenbestand der Mediothek, wie in Kapitel 2.1.3 beschrieben, gibt es Titel, die Hierarchien angehören. In einer Redesign-Maßnahme [König, 2005] wurde das Datenmodell des MedioVis Frameworks angepasst und hierarchische Strukturen integriert. Momentan werden hierarchische Strukturen allerdings nur in der klassischen Ansicht angezeigt. Die klassische Ansicht folgt dem Prinzip der „Multiple Coordinated Views“ (MCVs) [North u. Schneiderman, 1997]. Hierbei werden verschiedene Ansichten, die semantisch ähnliche Daten darstellen, kombiniert. In der klassischen Ansicht werden TreeTable, Detail-Ansicht und Standort-Ansicht in einer MCV angezeigt.

Sowohl in in der HyperGrid als auch in bekannten Online-Suchsystemen, wie Yahoo und Google, wird dem Anwender nach dem Start ein einfaches Suchfeld angeboten. Durch Eingabe von Schlüsselwörtern in das Suchfeld und anschließendes Betätigen des Suchbuttons wird der gesamte Datenbestand anhand dieser Schlüsselwörter eingeschränkt und in der TreeTable visualisiert.

Die TreeTable visualisiert die Daten ähnlich wie die HyperGrid in einer Tabelle. Jede Zeile repräsentiert ein Objekt der Resultatmenge. Die Spalten entsprechen den Attributen der Suchtreffer. Die Auswahl und die Reihenfolge dieser Attribute werden durch den Assignment Editor [Grün, 2004] frei definiert. Die Spaltenköpfe tragen den Namen des zugeordneten Attributs. Durch einen Klick auf den Sortier-Button im entsprechenden Spaltenkopf wird die Spalte sortiert. Direkt unter den Spaltenköpfen stehen für jede Spalte Eingabefelder bereit, die, wie in der HyperGrid, als Tabellen-Filter dienen. Die Ergebnismenge wird anhand der dort eingegebenen Schlüsselwörter dynamisch gefiltert.

In der TreeTable gibt es im Gegensatz zur HyperGrid keinen semantischen Zoom, jedoch ist eine Detail-Ansicht vorhanden, in der zusätzliche Informationen zu dem jeweiligen Objekt optisch aufbereitet werden. Die Detail-Ansicht wird gemäß dem Brushing und Linking Konzept [Becker u. Cleveland, 1987] beim Überfahren der einzelnen Titel in der TreeTable aktualisiert. Außerdem wird noch eine Standort-Ansicht für den jeweils fokussierten Titel angezeigt. Die Standort-Ansicht beinhaltet eine Standortmarkierung des Titels in einer schematischen Umrisskarte der Mediothek (siehe Abbildung 23).

The screenshot shows the Medioviz 0.81m search interface. The search bar contains 'eastwood' and the results show 22 hits. The table below lists the search results:

Titel	Personen	Jahr	Sprache	Medientyp	Signatur	Ausleihstatus
A fistful of Dollars	Leone, Sergio; Eastwood, Clint; Koch...	2001	engl.	Video	6 tff 710:1583/f48	ausgeliehen bis: 24.02.2007
A grammar of spoken English	Mackin, Ronald; Eastwood, John		engl.	diverse	6 fse 340/m14b	mehrteiliges Werk
Bird	Eastwood, Clint; Whitaker, Forest; V...	1988	dt.	Video	6 mus 648.10/e18	verfügbar (Frist: 1 Woche)
Brondo Billy	Eastwood, Clint; Locke, Sondra; Lew...	1980	dt.	Video	6 tff 710:e136/b76	verfügbar (Frist: 1 Woche)
Cue for a drill	Harkess, Shiona; Eastwood, John	1976		Buch	6 fse 340/h16	verfügbar
Die Brücken am Fluss	Eastwood, Clint; Streep, Meryl; LaGr...	1995		Video	6 tff 710:e136/b74	verfügbar (Frist: 1 Woche)
Die Letzten beißen die Hunde	Cimino, Michael; Eastwood, Clint; Br...	1973	dt.	Video	6 tff 710:c573/t49	verfügbar (Frist: 1 Woche)
Ein Fremder ohne Namen	Eastwood, Clint; Bloom, Verna; Hill...	1983	dt.	Video	6 tff 710:e136/h43	verfügbar (Frist: 1 Woche)
For a few Dollars more	Leone, Sergio; Eastwood, Clint; Van...	1999	engl.	Video	6 tff 710:1583/p27	verfügbar (Frist: 1 Woche)
Für eine Handvoll Dollar	Leone, Sergio; Eastwood, Clint; Koch...	1964	dt.	Video	6 tff 710:1583/p27a	verfügbar (Frist: 1 Woche)
Hang 'em high	Post, Ted; Eastwood, Clint; Stevens...	1997	engl.	Video	6 tff 710:p858/h16	verfügbar (Frist: 1 Woche)
In the line of fire - Die zweite Chance	Petersen, Wolfgang; Eastwood, Clint...	1993	dt.	Video	6 tff 710:p484/m60	verfügbar (Frist: 1 Woche)
Le streghe	Visconti, Luchino; Mangano, Silvana...	1966	dt.	Video	6 tff 710:825/s94	verfügbar (Frist: 1 Woche)
Martin Scorsese presents: The Blues	Scorsese, Martin	2003	engl.	DVD	6 tff 710:s423/b59	mehrteiliges Werk
Million dollar baby	Eastwood, Clint; Swank, Hilary; Free...	2005		DVD	6 tff 710:e136/m45	ausgeliehen bis: 20.02.2007

The detailed view for 'Million dollar baby' shows the following information:

- Beschreibung:** [Darsteller:] Clint Eastwood, Hilary Swank, Morgan Freeman ... Kamera: Tom Stern. Nach einer Geschichte aus 'Ropeburns' von F. X. Toole. Drehbuch: Paul Haggis. Produzent und Regie: Clint Eastwood
- Medientyp:** DVD
- Details:** Sprachen: engl., dt.
- Jahr:** 2005
- Personen:** Eastwood, Clint; Swank, Hilary; Freeman, Morgan; Stern, Tom; Toole, F. X.; Haggis, Paul
- Format:** 1 DVD-Video (127 Min.): farb.
- Signatur:** 6 tff 710:e136/m45
- Ausleihstatus:** ausgeliehen bis: 20.02.2007
- Belleibtheit:** 26x ausgeliehen
- Herausgeber:** Kinowelt Home Entertainment
- Stadt:** [Leipzig]
- Fachgebiet:** Theater/Tanz/Film/Funk/Fernsehen

On the right, there is a floor plan diagram with the label 'EINGANG' (Entrance).

Abbildung 23: TreeTable: Ergebnis der Suche nach „Eastwood“

Das Symbol in Form eines auf der Spitze stehenden Dreiecks am linken Rand der TreeTable soll signalisieren, dass ein Objekt hierarchisch ist. Zusätzlich wird dies durch den Mauszeiger signalisiert, der beim Überfahren hierarchischer Titel zu einer Lupe wird, und durch den Ausleihstatus, in dem dann ein hierarchisches Dokument als „mehrteiliges Werk“ gekennzeichnet ist. Durch einen Klick mit der linken Maustaste in die Zeile eines hierarchischen Dokuments werden die untergeordneten Elemente dieses Titels in einer Animation nach unten aufgeklappt und nehmen nun, visuell gruppiert, jeweils eine eigene Zeile in der TreeTable ein (siehe Abbildung 24). So können detaillierte Informationen zu den Unterdokumenten auf Wunsch durch Überfahren in der Detail-Ansicht angezeigt werden. Ein Klick mit der rechten Maustaste auf das Oberdokument klappt die Unterdokumente wieder zu.

A fistful of Dollars	Leone, Sergio; Eastwood, Clint; Koch...	2001	engl.	Video	6 tff 710:1583/f48	ausgeliehen bis: 24.02.2007
A grammar of spoken English	Mackin, Ronald; Eastwood, John		engl.	diverse	6 fse 340/m14b	mehrteiliges Werk
Bird	Eastwood, Clint; Whitaker, Forest; V...	1988	dt.	Video	6 mus 648.10/e18	verfügbar (Frist: 1 Woche)
Brondo Billy	Eastwood, Clint; Locke, Sondra; Lew...	1980	dt.	Video	6 tff 710:e136/b76	verfügbar (Frist: 1 Woche)

A fistful of Dollars	Leone, Sergio; Eastwood, Clint; Koch...	2001	engl.	Video	6 tff 710:1583/f48	ausgeliehen bis: 24.02.2007
A grammar of spoken English	Mackin, Ronald; Eastwood, John		engl.	diverse	6 fse 340/m14b	mehrteiliges Werk
Communication		1984	engl.	Buch	6 fse 340/m14b-d2...	mehrere Exemplare
					6 fse 340/m14b-d2	verfügbar
					6 fse 340/m14b-c2	verfügbar
Mackin, Ronald: A grammar of spoken English		1981	engl.	Buch	6 fse 340/m14b-a	verfügbar
Structures		1983	engl.	Buch	6 fse 340/m14b-c1...	mehrere Exemplare
					6 fse 340/m14b-c1	verfügbar
					6 fse 340/m14b-d1	verfügbar
Bird	Eastwood, Clint; Whitaker, Forest; V...	1988	dt.	Video	6 mus 648.10/e18	verfügbar (Frist: 1 Woche)
Brondo Billy	Eastwood, Clint; Locke, Sondra; Lew...	1980	dt.	Video	6 tff 710:e136/b76	verfügbar (Frist: 1 Woche)

Abbildung 24: Die Animation zum Aufklappen hierarchischer Titel in der TreeTable

„Um den Anwendern bei der ersten Ansicht der Suchresultate einen guten Gesamtüberblick zu ermöglichen“ [König, 2005], werden bei hierarchischen Daten immer nur die Gesamttitel angezeigt. Von einer Hierarchie muss nur ein Titel getroffen werden, damit der Gesamttitel in der Resultatmenge visualisiert wird. So kann es passieren, dass der Gesamttitel von der Suche nicht getroffen wurde, aber trotzdem in der Resultatmenge dargestellt wird. Heißt der Gesamttitel einer Hierarchie gänzlich anders als die Untertitel, so ist es schwer vom Gesamttitel auf dessen Untertitel zu schließen (vergleiche Abbildung 24). Dadurch würde man beim Durchforsten der Resultatmenge dem Gesamttitel eventuell keine Beachtung schenken und folglich auch nie den gesuchten Untertitel entdecken.

Aufgrund der unklaren Visualisierung hierarchischer Daten können daher wichtige Objekte der Resultatmenge übersehen werden. Dies kann man sehr gut anhand der Hierarchie „A grammar of spoken English“ zeigen (siehe Abbildung 25).

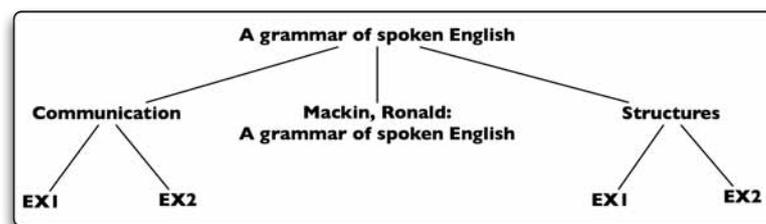


Abbildung 25: Die Hierarchie: „A grammar of spoken English“

Ein Benutzer sucht nun nach „Structures“, dem dritten Buch dieser Reihe „A grammar of spoken English“. Der Benutzer kennt aber nicht den Gesamttitel dieser Hierarchie. Da aber nur der Gesamttitel in der TreeTable dargestellt wird (siehe Abbildung 26), findet der Benutzer den gesuchten Untertitel „Structures“ nur schwer oder gar nicht (vergleiche Abbildung 27).

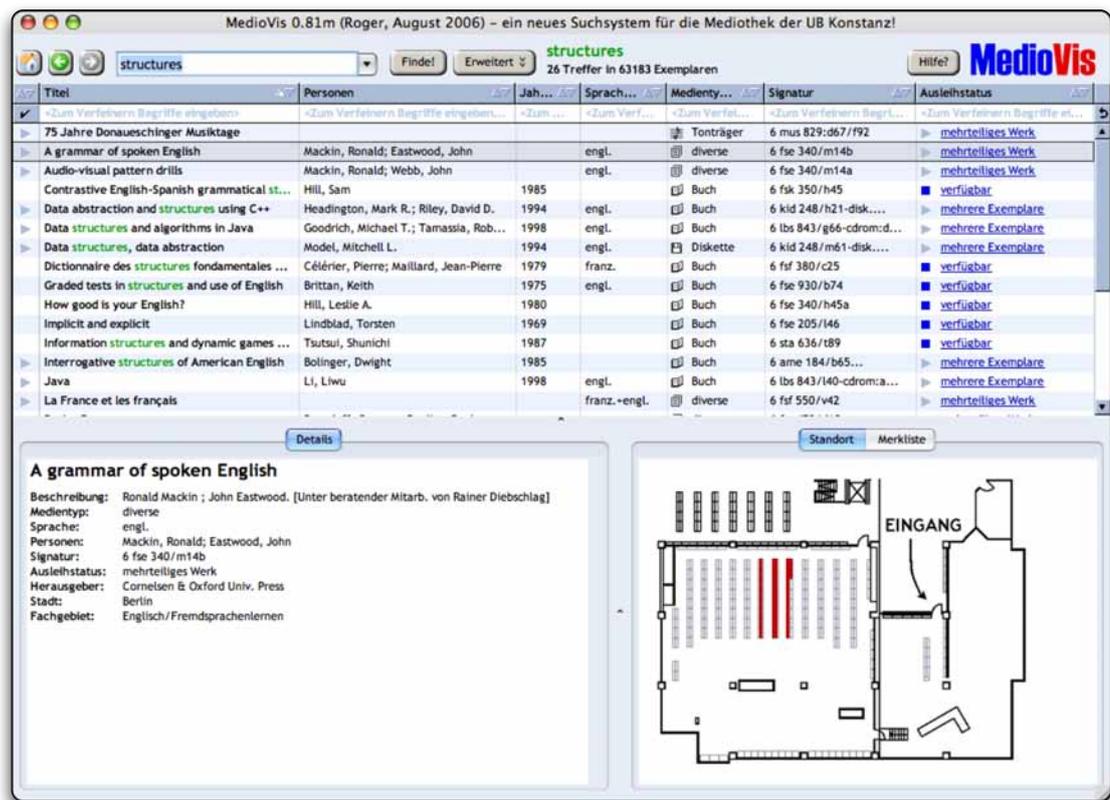


Abbildung 26: TreeTable nach der Suche nach „Structures“

Des Weiteren erwartet man beim Betrachten der Hierarchie, dass diese auch komplett dargestellt wird. In der TreeTable werden jedoch nur die Titel einer Hierarchie angezeigt, die sich tatsächlich auch in der Resultatmenge der Schlüsselwörter befinden. Sucht man nach „Structures“, taucht in der Resultatmenge der TreeTable das mehrteilige Werk mit dem Gesamttitel „A grammar of spoken English“ auf. Exploriert man nun die Hierarchie dieses Werks, gibt es nur einen Titel, nämlich „Structures“ (siehe Abbildung 27). Der Suchende erwartet, dass auch die übrigen Bücher der Reihe angezeigt werden, da sie ja ein Element dieser Hierarchie sind und man die komplette Hierarchie auffinden möchte.

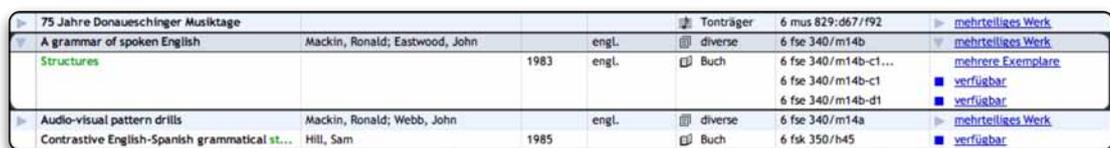


Abbildung 27: TreeTable mit einem aufgeklappten hierarchischen Dokument „A grammar of spoken English“

Problematisch ist auch, dass man nicht mehrere Hierarchie-Ebenen gleichzeitig deutlich darstellen kann. Bei zwei Ebenen funktioniert dieses Konzept noch ziemlich gut, doch sobald man drei Ebenen hat, was im Datenbestand der Bibliothek sehr häufig vorkommt, kann man die jeweiligen Ebenen nicht mehr voneinander unterscheiden

(siehe Abbildung 24). Daraus folgt: Es gibt keine klare Kommunikation der Hierarchie.

3.7 Zusammenfassung und Problemstellung

In Kapitel 3.6 wurde das Interaktions- und Visualisierungskonzept der klassischen Ansicht aufgezeigt. Die TreeTable, die Teil der klassischen Ansicht ist, ist bisher die einzige Darstellung im Visualisierungsrepertoire von MedioVis, die hierarchische Daten darstellen kann. In Kapitel 3.5 wurde eine andere Visualisierung, die HyperGrid, vorgestellt. Mit der HyperGrid wurde eine zoombare Tabellenvisualisierung geschaffen, die sowohl die analytische Suche als auch das interessen geleitete „Stöbern“ in einer gemeinsamen Oberfläche unterstützt. Der Benutzer bestimmt aktiv durch das semantische Zoomen, mit welchem Inhalt und mit welcher Informationsmenge er konfrontiert werden will. Diese selektive Präsentation ermöglicht es dem Benutzer komplexe heterogene Informationsräume ohne kognitive Überlastung zu explorieren.

In der klassischen Ansicht wird hauptsächlich die analytische Herangehensweise, aufgrund der fehlenden Browserkomponente, unterstützt. Zudem können die Informationsräume der klassischen Ansicht im Sinne der Quantität, Dimensionalität und Multimedialität nicht mit denen der HyperGrid mithalten. Des Weiteren kann die HyperGrid sowohl in klassischen WIMP und Online-Umgebungen, als auch in Zoomable User Interface konsistent eingebunden werden. Die klassische Ansicht hingegen würde Letzteres durch das fehlenden Zoomable Interface Paradigma „zerstören“.

Problem der HyperGrid ist die fehlende Möglichkeit Hierarchien darzustellen. Das Konzept der TreeTable lässt sich jedoch nicht in die HyperGrid integrieren. Gründe hierfür sind zum einen die in Kapitel 3.6 schon erwähnten Schwächen der TreeTable, die insbesondere für das Verständnis bzw. für ein effizientes Explorieren von hierarchischen Daten ungünstig sind. Zum anderen sind einige Funktionen, die in der TreeTable verwendet werden, in der HyperGrid bereits in einem anderen Zusammenhang in Gebrauch. Hierzu gehört zum Beispiel das Klicken mit der rechten Maustaste auf eine Zeile. In der HyperGrid wird dadurch das Zoomen in die Zellen ausgelöst, in der TreeTable hingegen werden damit die Unterdokumente eines Wurzel-Dokuments ausgeklappt. Ein weiteres Beispiel ist das Symbol in Form des auf der Spitze stehenden Dreiecks am linken Rand. Ein Klick auf dieses Symbol in der HyperGrid bewirkt das komplette Ein- bzw. Auszoomen eines Objekts, im Gegensatz dazu werden in der TreeTable dadurch die Untertitel ausgeklappt.

Aus dem soeben Beschriebenen resultiert, dass ein neues Visualisierungs- und Interaktionskonzept zur Integration hierarchischer Daten in die HyperGrid entwickelt werden muss. Dieses neue Konzept, das HyperGridXGL heißt, wird im nächsten

Kapitel vorgestellt.

4 HyperGridXGL

Im vorangegangenen Kapitel 3.7 wurde die Ausgangssituation für die Entwicklung eines neuen Konzepts dargelegt. Dabei wurde deutlich, dass das aktuelle Konzept zur Darstellung hierarchischer Daten in MedioVis (siehe Kapitel 3.6) nicht in die HyperGrid integrierbar ist. Daraus ergab sich die Problemstellung, ein neues Visualisierungs- und Interaktionskonzept zur Integration hierarchischer Daten in die HyperGrid zu entwickeln. Im Folgenden werden die an das Konzept gestellten Anforderungen und Ziele sowie die daraus resultierenden Rahmenbedingungen definiert. Daran schließt sich eine Vorstellung des Konzepts der HyperGridXGL an.

4.1 Anforderungen und Ziele

Die HyperGrid ist eine zoombare Tabelle, die Interaktionskonzepte des ZUI verwendet. Das neue Konzept soll in die HyperGrid integriert werden, ohne diese konzeptionell zu „zerstören“. Daher muss das zoombare Tabellen-Konzept in der Darstellung der hierarchischen Daten fortgeführt werden, um so eine vollständige, konsistente Integration in das Interaktions- und Visualisierungskonzept der HyperGrid zu erreichen. So sollen alle Interaktionstechniken, insbesondere das semantische Zooming aber auch das Sortieren und Filtern in der Darstellung der hierarchischen Dokumente beibehalten werden. Das führt dazu, dass auch die Navigations- und Steuerelemente der HyperGrid übernommen werden müssen. Innerhalb der HyperGrid sind sowohl die linke und rechte Maustaste als auch das Scrollrad mit wesentlichen Funktionen belegt. Die Belegung dieser Navigations- und Steuerelemente wurde in Evaluationen [Jetter u. a., 2005] bestätigt und folglich sollen die bisherige Funktionen der HyperGrid auch in der Darstellung der hierarchischen Daten durch die gleiche Belegung gesteuert werden.

Ein weiteres primäres Ziel des neuen Konzepts ist darin zu sehen, dass tatsächlich alle Ebenen der hierarchischen Struktur eines Objekts exploriert werden können. Demnach soll auch klar die Hierarchie, im Sinne der Semantik des Baumes, kommuniziert werden. Diese Aspekte sind für Visualisierungen, die sich mit der Darstellung von hierarchischen Daten beschäftigen, von großer Bedeutung. In der TreeTable, wie in Kapitel 3.6 beschrieben, wurde insbesondere die Kommunikation der Hierarchie vernachlässigt, was zu Verständnisproblemen seitens des Benutzers geführt hat. Daraus folgt: Der Aufbau eines mentalen Modells erweist sich für den Benutzer als schwierig. Wie schon in Kapitel 2.1.1 erwähnt, gleicht das mentale Modell, das zum Verständnis von Hierarchien konstruiert wird, oft dem Aufbau eines Baumes.

Wenn man nun die verschiedenen Darstellungsmethoden für hierarchische Daten, die in Kapitel 2.2 vorgestellt wurden, explizit betrachtet, stellt man schnell fest, dass die meisten Visualisierungen auch aus diesem Grund auf der Idee der Visualisie-

rungstechnik „Baum“ (siehe Kapitel 2.2.1) basieren. Der große Unterschied dieser verschiedenen Visualisierungen besteht im Wesentlichen darin, dass der Baum in unterschiedlichen Ausgestaltungen, zum Beispiel zwei- oder dreidimensional, dargestellt wird. Aus diesem Grund liegt es nahe, auch für die Darstellung hierarchischer Daten in der HyperGrid eine Visualisierung zu wählen, die auf der Grundidee der Visualisierungstechnik „Baum“ basiert. Jedoch ist im Rahmen der HyperGrid die Möglichkeit einen Baum bzw. Objekte als Knoten darzustellen nicht möglich. Eine solche Baumdarstellung würde das Gesamtbild der HyperGrid „zerstören“. Das Strukturierungskonzept der HyperGrid gleicht dem einer Tabelle und unterscheidet sich damit gravierend vom Strukturierungskonzept eines Baumes. In der Tabelle sind die einzelnen Objekte als Zeilen horizontal abgebildet. Betrachtet man im Gegensatz dazu einen Baum, dessen Wurzel oben ist, sind die Ebenen zwar auch zeilenweise dargestellt, jedoch befindet sich in einer Zeile eine Vielzahl von Objekten nebeneinander. So wären in jeder Zeile nicht mehr nur ein Objekt, wodurch die strukturierte Ordnung der Tabelle verloren gehen würde. Folglich würde die Verwendung von baumartigen Visualisierungen einem Bruch der tabellenartigen Ansicht der HyperGrid gleichkommen. Daher ist es naheliegend eine tabellenartige bzw. listenartige Darstellung für hierarchische Daten in der HyperGrid zu wählen. Sichtbare Schwächen offenbart die tabellenartige bzw. listenartige Darstellungsweise der TreeTable (siehe Kapitel 3.6). Dies gilt es durch neue innovative Konzeptideen zu verbessern.

Ein wichtiges Merkmal der HyperGrid ist die Möglichkeit, mehrere Objekte der Resultatmenge miteinander zu vergleichen ohne aber die restlichen aus dem Blickfeld zu verlieren. Aus diesem Grund soll bei der Exploration einer konkreten Hierarchie das fokussierte Untersuchungsobjekt zwar hervorgehoben werden, aber die restliche Resultatmenge sollte im Blickfeld bleiben. So bleibt weiterhin die Möglichkeit für den Benutzer bestehen, verschiedene Objekte, ob diese hierarchischer oder nicht-hierarchischer Natur sind, einem Vergleich zu unterziehen.

4.2 Konzeptideen

Im Verlauf der Entwicklungsphase des Konzepts sind verschiedene Ideen aufgekommen. Diese Ideen sind keine vollausgearbeiteten Konzepte, sondern entsprechen lediglich Ergebnissen aus Brainstorming-Sessions und dienten als Grundlage für das spätere Visualisierungs- und Interaktionskonzept. Im Folgenden werden drei Konzeptideen vorgestellt und deren Vor- und Nachteile diskutiert.

4.2.1 classical

Diese Konzeptidee ähnelt dem Konzept, das derzeit in der TreeTable realisiert ist. Zudem ist sie an die grafische Darstellung der Treeview des Windows Explorers an-

gelehnt. Daher kommt auch der Name „classical“, da es sich um einen klassischen Ansatz handelt.

Wie bei allen Konzepten bekommt der User zu Beginn nur die Suchmaske angeboten. Nach Eingabe der Schlüsselwörter und dem Betätigen des Suchbuttons wird der gesamte Datenbestand anhand der Schlüsselwörter eingeschränkt und in der HyperGrid angezeigt.

Neben der Spalte mit dem Flap-In/-Out-Button (Symbol in Form eines umgekehrten Dreiecks) befindet sich nun eine weitere Spalte, wie man in der Abbildung 28 erkennen kann. In dieser Spalte wird bei den Objekten bzw. Zeilen, die zu einer Hierarchie gehören, ein weiteres Symbol angezeigt. Dieses Symbol, das in der Abbildung 28 rot gekennzeichnet ist, besteht aus einer Dreieckspitze eines Pfeiles und weist nach rechts auf den nachfolgenden Zeilentext. Die Pfeilspitze signalisiert demnach, dass dieses Objekt noch Kind-Knoten besitzt. Falls der Benutzer die Hierarchie eines solchen Objekts explorieren möchte, klickt er auf das Symbol.

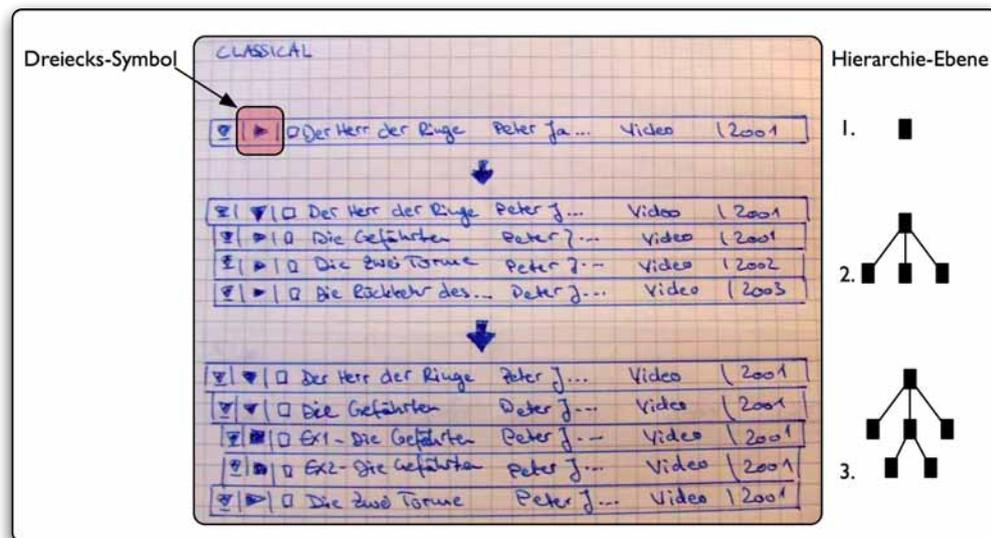


Abbildung 28: Die Konzeptidee „classical“

Durch eine Animation, ähnlich wie in der TreeTable, werden die Elemente der darunter liegenden Hierarchie-Ebene nach unten ausgeklappt (vergleiche Abbildung 28, zweite Hierarchie-Ebene bzw. zweite Tabelle von oben). Das Dreieck des Oberdokuments, das zuvor nach rechts gezeigt hat, zeigt jetzt nach unten auf die ausgeklappten neuen Dokumente der darunter liegenden Ebene. Die Zeilen der ausgeklappten Objekte sind im Vergleich zur Zeile des Vater-Knotens ein wenig nach rechts eingerückt. Durch diese Einrückung soll kommuniziert werden, dass es sich um unterschiedliche Ebenen handelt. Wie alle Objekte einer Hierarchie haben auch die neu ausgeklappten Objekte in der zweiten Spalte von links ein Symbol. Entspricht die Form dieses

Symbols einem Quadrat, dann ist das Objekt ein Blatt, das keine weiteren Kind-Knoten besitzt (siehe Abbildung 28). Ist das Symbol ein Dreieck, das nach rechts zeigt, hat dieses Objekt weitere „Kinder“. Durch Klick auf dieses Symbol werden die Objekte der darunter liegenden Ebene in der oben beschriebenen Animation direkt unter jenem „Vater-Objekt“ ausgeklappt.

Bei dieser Konzeptidee wurde vor allem die Kommunikation der Hierarchie-Ebenen gegenüber dem Konzept der TreeTable wesentlich verbessert. Die Hierarchie-Ebenen können durch die Einrückung voneinander unterschieden werden. Außerdem werden nicht alle Ebenen auf einmal ausgeklappt, sondern können getrennt voneinander geöffnet werden. Zudem wird die Idee zum Navigieren zwischen den Hierarchie-Ebenen mit dem Symbol in Form eines Dreiecks ebenfalls in bekannten Systemen, wie dem Windows Explorer von Windows Vista⁴² sowie dem Finder von Mac OS X⁴³, verwendet. Durch den täglichen Gebrauch jener Systeme ist der Umgang mit diesem Navigationskonzept geübt. Daher sollte die Nutzung der Navigation innerhalb dieser Konzeptidee den meisten Anwendern leicht fallen. Dennoch gibt es einige Nachteile, die dazu führten, dass diese Idee nicht weiterverfolgt wurde:

Zum einen wird durch das Einrücken sehr viel Platz benötigt und zum anderen kann die Darstellung sehr entarten. Durch die gewisse Ähnlichkeit mit der Treeview, kann man aus Erfahrung sagen, dass diese Visualisierung für kleine Hierarchien, insbesondere Hierarchien mit wenigen Elementen, einigermaßen gut geeignet ist, sobald die Hierarchie jedoch größer wird und eine Vielzahl von Knoten umfasst, geht die Übersicht verloren. Ein weiteres Problem, das zuvor schon bei der Beschreibung der TreeTable in Kapitel 3.6 angesprochen wurde, tritt hier in ähnlicher Weise auf: In der Resultatmenge würde sich nur der Gesamttitel einer Hierarchie befinden. Somit würde der Benutzer einige interessanten Objekte gar nicht entdecken. Eine Möglichkeit dieses Problem zu umgehen wäre, Suchtreffer, die gleichzeitig Untertitel sind, direkt von Beginn an mit aufgeklappter Hierarchie darzustellen. So würde man von Anfang an eine wesentlich größere Menge an Objekten anzeigen. Dabei würden viele Objekte visualisiert werden, die eigentlich nicht Teil der Resultatmenge sind. Daher sind beide Möglichkeiten für den Benutzer nur schwer nachzuvollziehen, denn bei der ersten werden Informationen „unterschlagen“ und bei der zweiten wird der Benutzer von zu vielen Informationen „erschlagen“.

4.2.2 finder

Zu dieser Konzeptidee inspirierte das tägliche Arbeiten mit dem Finder, dem Dateimanager unter Mac OS X. Meistens werden Hierarchien von oben nach unten, also vertikal, dargestellt. In der Treeview des Windows Explorers und auch in der

⁴²Windows Vista ist das Betriebssystem von Microsoft, <http://www.microsoft.com/germany/windows/products/windowsvista/default.aspx>

⁴³Mac OS X ist das Betriebssystem von Apple, <http://www.apple.com/de/macosex/leopard/>

TreeTable werden die darunter liegenden Ebenen nach unten ausgeklappt. Der Finder hingegen kommuniziert Hierarchien horizontal (siehe Abbildung 29). Der Pfad einer Datei wird über das Highlighten von links nach rechts gekennzeichnet. Demnach ist in der Abbildung 29 der relative Pfad: Dokumente/uni/Bachelor/BA_papers/hci-kn/.

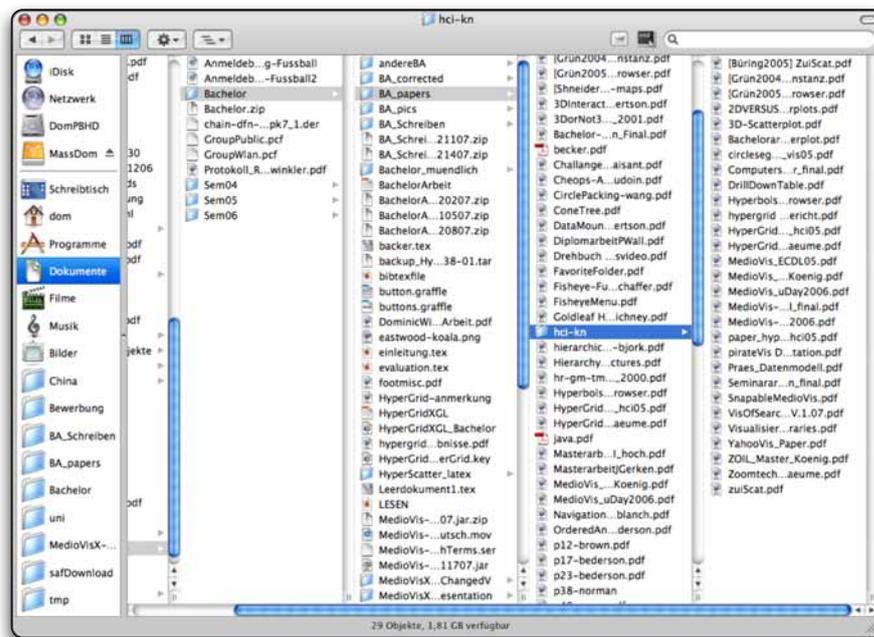


Abbildung 29: Der Finder unter Mac OS X

Aus dem Konzept des Finders resultiert die Idee Hierarchien in der HyperGrid horizontal darzustellen. Denn bisher wird die Fläche in der HyperGrid nur durch den semantischen Zoom beim „Hinein“- bzw. „Nach hinten“-Zoomen geometrisch nach unten hin vergrößert. Deswegen ist es naheliegend, ähnlich wie beim Finder, Hierarchie-Ebenen nicht nach unten auszuklappen, sondern seitlich und somit auch gleichzeitig die Fläche links und rechts zu nutzen.

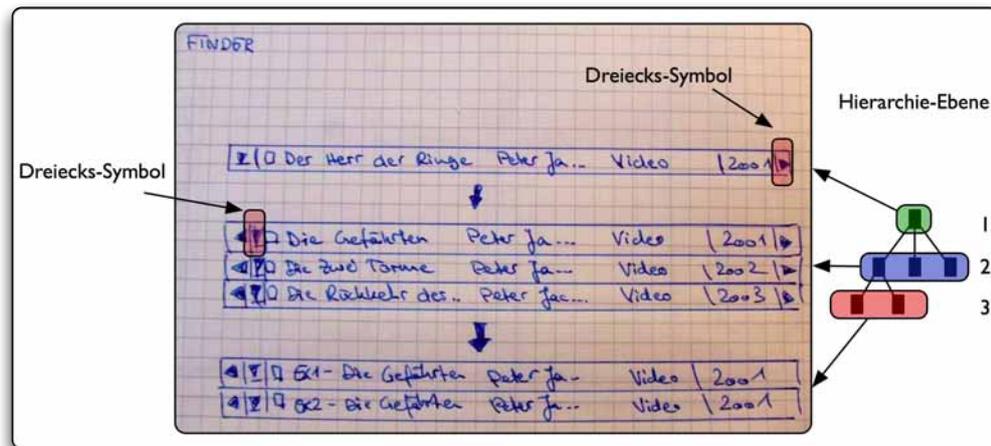


Abbildung 30: Die Konzeptidee „finder“

Wie auch schon in der Konzeptidee 4.2.1 ändert sich nichts an der Suchmaske. Zur Standard-HyperGrid werden zwei neue Spalten, links außen und rechts außen hinzugefügt. Ein Dreieck in der rechten bzw. linken Spalte einer Zeile signalisiert, ob Objekte in den Hierarchie-Ebenen darüber bzw. darunter vorhanden sind (siehe Abbildung 30). Beim Klick auf das am rechten Rand liegende Symbol in Form eines Dreiecks kann man die Elemente der darunter liegenden Ebene betrachten. Entweder werden diese durch eine Animation von rechts eingefahren und überdecken den Vater-Knoten oder der Fokus wird durch den Klick auf das rechte Dreieck durch eine Animation auf eine neue HyperGrid, die rechts außerhalb des sichtbaren Bereichs liegt, verschoben. Ähnliches passiert, wenn man auf das Dreieck, das nach links zeigt, auf der linken Seite klickt. Nur hier werden die Elemente der Hierarchie-Ebene darüber angezeigt.

Das in Kapitel 3.6 und 4.2.1 geschilderte Problem, dass nur der Gesamttitel in der Resultatmenge auftaucht, ist hier nicht mehr gegeben. Bei der Visualisierung in der TreeTable sowie in der classical kann man Hierarchien nur „von oben nach unten“ explorieren. Somit benötigt man immer den Gesamttitel als Startpunkt. Beim finder wäre ein beliebiger Knoten als Startknoten möglich.

Von Nachteil ist, dass man immer nur eine Hierarchie-Ebene sieht und daher nicht weiß, in welchem Kontext diese steht. Darüber hinaus verliert man die restliche Resultatmenge aus dem Blickfeld.

4.2.3 cube

Diese Idee wurde inspiriert von der Figur eines Würfels bzw. eines Quaders. Auch hier bleibt die Suchmaske, bezogen auf die jetzige MedioVis-Version, unverändert. In die HyperGrid wird eine neue Spalte direkt neben der „Flap-In/-Out-Button“-

Spalte eingefügt. Immer dann, wenn das Objekt dieser Zeile zu einer Hierarchie gehört, wird ein Symbol in der neuen Spalte angezeigt (siehe Abbildung 31).

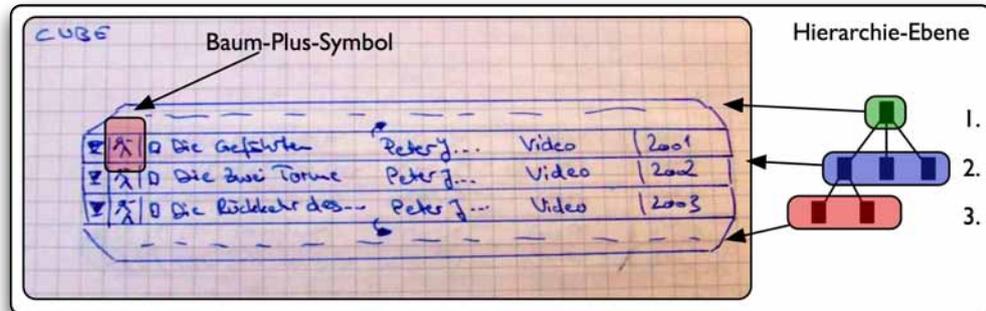


Abbildung 31: Die Konzeptidee „cube“

Das Symbol entspricht einem „Baum“ mit einem „Plus“. Da viele Menschen einen Baum bildlich mit einer Hierarchie in Verbindung bringen und da, wie bereits in Kapitel 2.2.1 erwähnt, eine Hierarchie auch gedanklich oft mit einer Baumstruktur gleichgesetzt wird, liegt es nahe einen Baum als Symbol zu wählen. Das Plus wird häufig als Metapher für das Zoomen bzw. „Erhalten von mehr Informationen“ gewählt. Daher scheint eine Kombination dieser zwei Symbole zu einem Symbol als sinnvoll. Bei einem Klick auf dieses Symbol fährt aus der flachen zweidimensionalen HyperGrid eine Art dreidimensionaler Würfel heraus.

Auf der vorderen Fläche des Würfels sieht man das Objekt, auf das man in der flachen HyperGrid geklickt hat, von dem ausgehend die hierarchische Struktur exploriert werden soll. Auch alle anderen Objekte der selben Hierarchie-Ebene finden sich hier. Auf der oberen Fläche des Würfels wird das Element der darüber liegenden Hierarchie-Ebene des selektierten Objekts angezeigt, auf der unteren Seite des Würfels werden die Elemente der darunter liegenden Ebene sichtbar. Die Elemente auf der Ober- und Unterseite des Würfels werden optisch verzerrt dargestellt. Durch Klick auf den kleinen Pfeil oben auf der vorderen Fläche wird der Würfel durch eine Animation ein wenig nach unten gedreht und man kann die perspektivisch verzerrte Oberseite besser sehen. Will man die Elemente dieser Ebene jetzt genauer „untersuchen“, muss man auf die obere Fläche klicken. Diese wird durch Animation nach vorn gedreht und ist ab sofort die Vorderseite des Würfels. Gleiche Aktionen lassen sich auch mit dem kleinen Pfeil, der unten auf der Vorderseite ist, und bei Klick auf die Unterseite durchführen. Der einzige Unterschied ist, dass die untere Fläche nach vorn gedreht wird. Das Symbol mit dem Baum in der zweiten Spalte ist nun mit einem Minus kombiniert. Durch Klick auf dieses Symbol wird der Würfel wieder eingefahren.

Bei dieser Konzeptidee muss die Zusammengehörigkeit zwischen dem selektierten Element der Vorderseite und den verzerrten Flächen oben und unten deutlich dargestellt werden. Da diese Flächen räumlich ein wenig auseinander sind, kann dies durchaus schwer zu verdeutlichen sein. Zudem ist noch nicht ganz klar, wie man das Selektieren verwirklicht. Schließlich gibt es die Funktion des Selektierens schon in einem anderen Zusammenhang in der HyperGrid: Mit den Objekten, die selektiert sind, können verschiedene Funktionen, wie in den Warenkorb verschieben, ausgeführt werden.

4.3 Wieso HyperGridXGL?

Das Visualisierungs- und Interaktionskonzept zur Integration hierarchischer Daten in die HyperGrid, das im Rahmen dieser Arbeit entstanden ist, basiert auf der Konzeptidee cube. Die in Kapitel 4.2 beschriebenen Konzeptideen wurden ausführlich anhand ihrer jeweiligen Vor- und Nachteile in einer „Experten-Runde“, bestehend aus Mitgliedern des MedioVis-Projekt-Teams, diskutiert. Die Wahl fiel innerhalb der Expertenrunde auf die Konzeptidee „cube“ als Basis. Im Folgenden soll die Begründung für diese Auswahl dargelegt werden.

Die erste Konzeptidee classical versucht durch Hinzufügen neuer Effekte den klassischen Ansatz der TreeTable zu verbessern. Durch die Einbindung der Navigation über das „Dreieck“ und durch das Einrücken der Zeilen ist dies mit Sicherheit gelungen. Vor allem die Kommunikation der Hierarchie konnte dadurch visuell deutlicher dargestellt werden. Jedoch besitzt diese Idee das Problem, dass Hierarchien nur von „top to bottom“ exploriert werden können. So besteht eine hohe Wahrscheinlichkeit, dass ein Verlust an Informationen auftritt. Aber gerade im Rahmen eines visuellen Suchsystems besteht der Anspruch ohne großen Informationsverlust komplexe Informationen flexibel zu präsentieren. Dieser Anspruch kann weder durch die TreeTable noch durch die Konzeptidee „classical“ verwirklicht werden.

Die zweite Konzeptidee „finder“ bietet einen flexiblen Einstieg in eine Hierarchie, um so dem aufgezeigten Anspruch gerecht zu werden. Auf diese Weise kann jedes Element einer Hierarchie als Startpunkt der Exploration in die Hierarchie dienen. Bei der HyperGrid findet der Informationszuwachs zu einem Objekt immer am Ort der Interessensäußerung statt. Der Kontext sowie der Überblick über die Resultatmenge bleibt dabei erhalten. Dieses Konzept sollte auch beim Anzeigen der Hierarchie eines Objekts verfolgt werden. Bei der Konzeptidee „finder“ hingegen wird das Ausgangselement entweder von den Kind-Knoten überlagert oder die Kind-Knoten werden in einer neuen HyperGrid angezeigt. Dabei geht der Kontext der jeweils dargestellten hierarchischen Objekte verloren. Folglich können die angestrebten Ziele und Anfor-

derungen durch die Konzeptidee „finder“ nicht erfüllt werden.

Im Vergleich zu den ersten zwei Konzeptideen überzeugt der „cube“ insbesondere durch seine neuartige Darstellung. Zudem besitzt der „cube“ weder das soeben aufgezeigte Problem der ersten noch das der zweiten Konzeptidee. Da der Informationszuwachs in Form der Darstellung der Hierarchie des Ausgangsobjekts am Ort der Interessensäußerung stattfindet, wird der Kontext innerhalb der Resultatmenge als auch innerhalb der Hierarchie gewahrt. Außerdem kann jedes Element einer Hierarchie als Startpunkt zur Exploration der Hierarchie gewählt werden. Demnach treten die Probleme der ersten zwei Konzeptideen im „cube“ nicht mehr auf. Darüber hinaus wurde der „cube“ von der „Experten-Runde“ als innovativ bezeichnet. Daher wurde die Konzeptidee „cube“ als Grundlage für das Visualisierungs- und Interaktionskonzept HyperGridXGL genommen. Im restlichen Kapitel 4 wird die HyperGridXGL näher beschrieben.

4.4 Anwendungsszenario

Im Folgenden wird das Visualisierungs- und Interaktionskonzept HyperGridXGL anhand eines typischen Beispielszenarios vorgestellt. Dabei wird auf den angereicherten Daten der Haupt-Anwendungsdomäne der Mediothek gearbeitet. Die Verwendung des Systems sowie einzelne Funktionen werden nun exemplarisch durch die Nutzung eines typischen Anwenders der Hauptzielgruppe vorgeführt. Da MedioVis auf über 150 Terminals in der Universitätsbibliothek für den praktischen Einsatz zur Verfügung steht, besteht der Großteil der tatsächlichen MedioVis-Anwender aus Studenten, Doktoranden, Professoren und vereinzelt Externen, die Zugang zu den Terminals der Bibliothek haben. Folglich setzt sich die Hauptzielgruppe aus Erst- und Gelegenheitsnutzern mit durchschnittlichen Computerkenntnissen und gutem Allgemeinverständnis zusammen. Ein Benutzerprofil mit einer Auswertung der DROID-Daten kann man im Redesign des Frameworks [König, 2005] nachlesen.

Julia May⁴⁴ studiert an der Universität Konstanz Englisch und Deutsch im zweiten Semester mit dem Berufsziel Lehrerin im Gymnasialbereich. Sie kennt sich recht gut mit Computern aus, da sie während ihres Studiums den Computer nicht nur zum Anfertigen von Texten benötigt, sondern insbesondere auch zum Recherchieren. Dabei dient ihr vor allem das Internet als Informationsquelle.

Im laufenden Semester besucht Julia u.a. das Seminar „Spoken English“. Um einen Schein in diesem Seminar zu erlangen, muss jeder Teilnehmer ein Referat halten. Der Dozent teilt gleich in der ersten Seminarstunde eine Liste mit Themen für die Referate aus. Julia entscheidet sich, ihr Referat über die Grammatik des heute gesprochenen Englisch zu erstellen. Zusätzlich zum Thema teilt der Dozent eine

⁴⁴fiktive, aber typische MedioVis-Benutzerin

Übersicht mit Literaturhinweisen aus. Ausgestattet mit einem Notizblock geht Julia in die Bibliothek, um sich mit geeignetem Material für ihr Referat einzudecken. In der Bibliothek startet sie MedioVis an einem der öffentlich zugänglichen PC-Terminals.

Das Blatt mit den Literaturhinweisen hat Julia „verlegt“. Sie kann sich aber noch darin erinnern, dass ihr ein Buch eines Autors namens „Eastwood“ über die Kommunikation der Englischen Sprache empfohlen wurde. Daher startet sie eine Suche mit dem Stichwort „Eastwood“. Daraufhin bekommt sie 24 Suchtreffer (siehe Abbildung 32) in der HyperGridXGL (siehe Kapitel 4), einer tabellenartigen Visualisierung, präsentiert.

	Titel	Beschreibung	Exemplar	Jahr
<input type="checkbox"/>	A fistful of Dollars	Sergio Leone, Clint Eastwood, Marianne Koch, Gian M...	Video	2001
<input type="checkbox"/>	A grammar of spoken English	Ronald Mackin, John Eastwood	diverse	0
<input type="checkbox"/>	Bird	Clint Eastwood, Forest Whitaker, Diane Venora, Micha...	Video	1988
<input type="checkbox"/>	Brondo Billy	Clint Eastwood, Sondra Locke, Geoffrey Lewis, Dennis ...	Video	1980
<input type="checkbox"/>	Cue for a drill	Shiona Harkess, John Eastwood	Buch	1976
<input type="checkbox"/>	Die Brücken am Fluss	Clint Eastwood, Meryl Streep, Richard LaGravenese, A...	Video	1995
<input type="checkbox"/>	Die Letzten beißen die Hunde	Michael Cimino, Clint Eastwood, Jeff Bridges, Geoffre...	Video	1973
<input type="checkbox"/>	Ein Fremder ohne Namen	Clint Eastwood, Verna Bloom, Mariana Hill, Ernest Tid...	Video	1983
<input type="checkbox"/>	For a few Dollars more	Sergio Leone, Clint Eastwood, Lee VanCleeef, Gian Ma...	Video	1999
<input type="checkbox"/>	Für ein paar Dollar mehr	Sergio Leone, Clint Eastwood, Lee VanCleeef, Gian Ma...	DVD	2005
<input type="checkbox"/>	Für eine Handvoll Dollar	Sergio Leone, Clint Eastwood, Marianne Koch, Gian M...	DVD	2005
<input type="checkbox"/>	Für eine Handvoll Dollar	Sergio Leone, Clint Eastwood, Marianne Koch, John W...	Video	1964
<input type="checkbox"/>	Hang 'em high	Ted Post, Clint Eastwood, Inger Stevens, Ed Begley, L...	Video	1997
<input type="checkbox"/>	In the line of fire - Die zweite Chance	Wolfgang Petersen, Clint Eastwood, John Malkovich, ...	Video	1993
<input type="checkbox"/>	Le streghe	Luchino Visconti, Silvana Mangano, Totò [Schauspieler...	Video	1966
<input type="checkbox"/>	Million dollar baby	Clint Eastwood, Hilary Swank, Morgan Freeman, Tom ...	DVD	2005
<input type="checkbox"/>	Mystic River	Clint Eastwood, Sean Penn, Tim Robbins, Kevin Bacon, ...	DVD	2004
<input type="checkbox"/>	Network	John Eastwood	diverse	0
<input type="checkbox"/>	Pale rider	Clint Eastwood, Michael Moriarty, Carrie Snodgrass, M...	Video	1985
<input type="checkbox"/>	Piano Blues	Clint Eastwood	DVD	2003
<input type="checkbox"/>	Sinola	John Sturges, Clint Eastwood, Robert Duval, John Cart...	Video	1984
<input type="checkbox"/>	The good, the bad and the ugly	Sergio Leone, Clint Eastwood, Lee VanCleeef, Aldo Glu...	Video	1987
<input type="checkbox"/>	The outlaw Josey Wales	Clint Eastwood, Dan George, Sandra Locke, Forrest C...	Video	1987
<input type="checkbox"/>	Unforgiven	Clint Eastwood, Gene Hackman, Morgan Freeman, Da...	Video	1998

Abbildung 32: Suchergebnis „Eastwood“ in der HyperGridXGL

Sie möchte die Suchtreffer weiter einschränken. Dazu gibt sie beim integrierten Tabellenfilter in der Spalte „Exemplar“ das Stichwort „Buch“ ein. Hierdurch wird die ursprüngliche Resultatmenge auf nur noch drei Titel reduziert. Der erste Titel lautet „A grammar of spoken English“ und scheint zumindestens auf den ersten Blick genau ihrem Referatsthema zu entsprechen. Um sich aber ganz sicher zu sein, exploriert Julia diesen Titel genauer, indem sie durch Klick mit der rechten Maustaste in die Titel-Zelle den semantischen Zoom auslöst. So gelangt sie an mehr Information. Anhand des Symbols in der zweiten Spalte von rechts, auf dem zwei hintereinander

liegende Seiten und ein Plus dargestellt sind, erkennt sie, dass dieser Titel einer Hierarchie angehört und dass hier noch mehr Informationen hinterlegt sind. Durch einen Mausklick auf das Symbol, das der TriggerButton ist, wird der hierarchische Zoom ausgelöst. In einer Animation wird die Zeile mit dem fokussierten Dokument wie eine Art Schublade aufgezo- gen, wobei die obere und untere Fläche nach oben bzw. unten in den 3D-Raum geklappt werden. Auf der Vorderseite der Schublade, der so- genannten DetailedView, befindet sich immer noch der gerade explorierte Titel „A grammar of spoken English“. Auf der unteren perspektivisch verzerrten Fläche, der ContextView, entdeckt Julia einen Titel mit dem Namen „Communication“ (siehe Abbildung 33). Dies scheint ein geeigneter Titel zu sein.

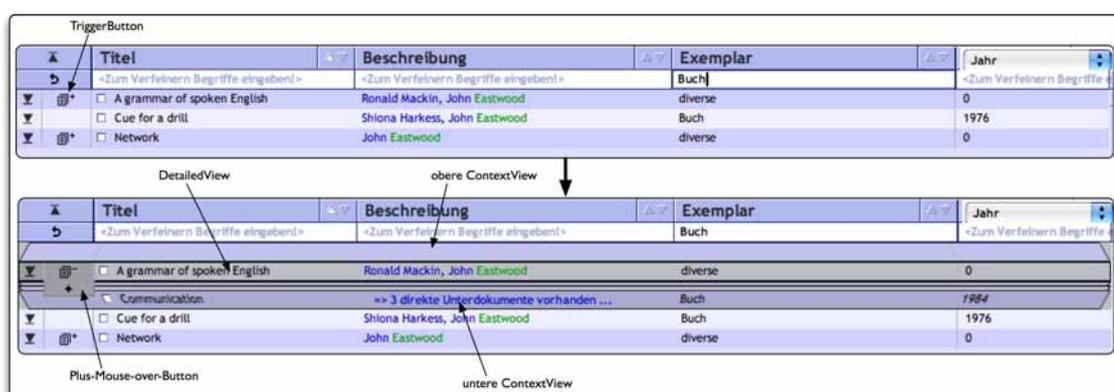


Abbildung 33: Die gefilterte HyperGridXGL mit aufgezoomten Titel „A grammar of spoken English“

So klickt Julia auf den Plus-Mouse-over-Button (MoB), der sich direkt unter dem TriggerButton befindet. Dadurch dreht sich das 3D-Gebilde nach oben. Der Titel „A grammar of spoken English“ wandert in die obere perspektivisch verzerrte Fläche. Davor war diese obere ContextView leer, da der Titel „A grammar of spoken English“ der Wurzel-Knoten der Hierarchie ist und es daher keinen Vater-Knoten zum Anzeigen in der oberen ContextView gab. „Communication“ befindet sich nun in der DetailedView und die untere ContextView beinhaltet den Titel „EX1-Communication“. Daraufhin werden die restlichen zwei Titel der Hierarchie-Ebene von „Communication“ auf der DetailedView nach unten ausgeklappt (siehe Abbildung 34).



Abbildung 34: Die Communication-Ebene mit Plus und Minus MoB

Julia glaubt, dass dies die Bücher sein könnten, die sie als Quellen für ihr Referat benötigt. Jetzt muss sie nur noch herausfinden, ob die Bücher momentan in der

Bibliothek verfügbar sind. Um das in Erfahrung zu bringen fährt sie mit der Maus über die Zeile des jeweiligen Titels. Dabei wird die untere ContextView aktualisiert. Falls die untere ContextView ein Objekt beinhaltet, wie das bei „Communication“ der Fall ist, wird zusätzlich zum Minus-MoB der Plus-MoB angezeigt. Demnach besitzt „Communication“ Kind-Knoten, daher klickt Julia auf den Plus-MoB und rotiert auf die nächste Hierarchie-Ebene. Dort exploriert sie ein Exemplar⁴⁵ von „Communication“ und findet im Attribut „Ausleihstatus“ heraus, dass dieser Titel noch vorhanden ist.

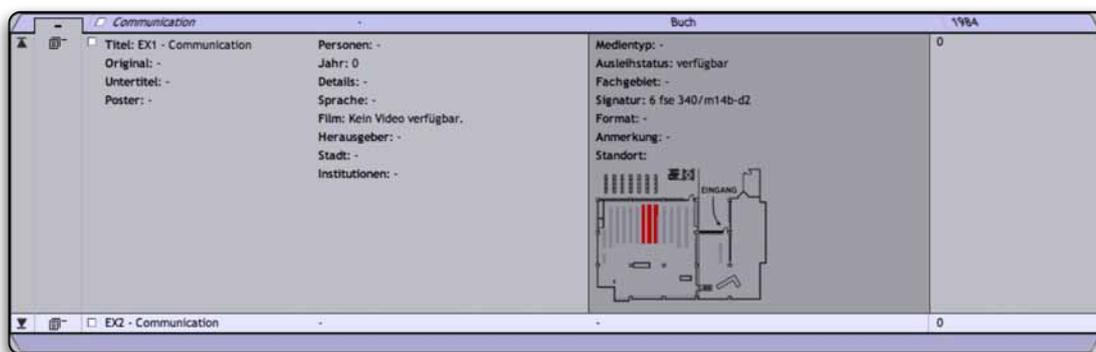


Abbildung 35: Die Exemplar-Ebene des Titels „Communication“

Daraufhin rotiert sie durch Klick auf den Minus-MoB auf die darüberliegenden Hierarchie-Ebene und erkundet ebenfalls die Exemplare der anderen zwei Titel. Julia freut sich, dass alle Titel verfügbar sind. So markiert sie zum Schluss alle drei Titel, indem sie auf das kleine weiße Kästchen vor dem Titel klickt und drückt sich die Informationen zu diesen Titeln über die Druck-Funktion von MedioVis aus. Zufrieden mit ihrer Suche in MedioVis begibt sich Julia zu den Regalen der Bibliothek, um anhand der ausgedruckten Standortskizze die ausgewählten Titel zu finden.

4.5 Das Visualisierung- und Interaktionskonzept HyperGridXGL

Nachdem das Visualisierung- und Interaktionskonzept HyperGridXGL zur Integration hierarchischer Daten in der HyperGrid an einem charakteristischen Beispielszenario vorgestellt wurde, sollen die einzelnen Komponenten im weiteren genau betrachtet werden.

Davor soll hier aber noch auf die Namensgebung eingegangen werden. Wie im Anwendungsszenario gezeigt wurde, entspricht die HyperGridXGL einer Erweiterung der HyperGrid. Betrachtet man das durch den hierarchischen Zoom (siehe Kapitel 4.5.1) ausgelöste 3D-Gebilde, das in Kapitel 4.5.2 detailliert vorgestellt wird, erinnert es an einen liegenden halben handelsüblichen Bleistift. Ein Bleistift basiert auf

⁴⁵Exemplare sind identische Ausführungen eines Titels, die sich lediglich in der Signatur bzw. im Standort von einander unterscheiden.

der Form eines Sechsecks. Nimmt man einen Sechskant-Bleistift und dreht ihn, kann man Parallelen zur Rotations-Animation, die genauer in Kapitel 4.5.3 beschrieben wird, erkennen. Das „XGL“ stammt von he**X**a**G**ona**L**, was im Deutschen „soviel wie sechskant“ heißt. Die HyperGridXGL ist somit eine Kombination aus der HyperGrid und dem 3D-Gebilde zum Darstellen hierarchischer Daten.

4.5.1 Hierarchischer Zoom

Das Benutzen einer Zoom-Funktion führt dazu, dass der Anwender mehr Informationen präsentiert bekommt. Wie schon in Kapitel 3.5 beschrieben, hat die HyperGrid Interaktionstechniken eines ZUIs adaptiert. Daher gibt es in der HyperGrid bereits eine Zoom-Funktion. Dieser schon vorhandene Zoom entspricht der in Kapitel 3.4 definierten Kombination aus geometrischem und semantischem Zooming. Durch Klicken mit der rechten Maustaste in eine Zelle eines Objekts, zoomt man in dessen Informationsraum. Konzeptionell gesehen wird in die Tiefe des Informationsraumes gezoomt, optisch gesehen jedoch vergrößert sich die Zelle durch einen animierten Effekt und der Benutzer bekommt so zusätzliche Informationen präsentiert. Somit erhöht der Benutzer durch das Zoomen in die Zelle den Degree of Interest bezogen auf den jeweiligen Aspect of Interest.

Zu diesem schon vorhandenen semantischen Zoom, der den Degree of Interest, bezogen auf die Information eines Objekts, erhöht, soll eine neue Zoom-Funktion hinzugefügt werden. Durch diese neue Zoom-Funktion, den hierarchischen Zoom, wird der DOI, bezogen auf die Hierarchie, verändert. So wird durch Klick auf einen TriggerButton, der in Kapitel 4.5.3 näher beschrieben wird, der hierarchische Zoom zu einem Objekt, welches einer Hierarchie angehört, ausgelöst. Optisch gesehen wird dabei nicht „hineingezoomt“, um den Degree of Interest zu erhöhen, sondern die Information kommt auf den Anwender mit Hilfe einer 3D-Animation zugefahren. Die Objekte der Hierarchie des TriggerDocs, der Titel auf den der hierarchische Zoom angewendet wurde, werden dann mit einem 3D-Effekt innerhalb der HyperGrid präsentiert, was im nächsten Kapitel 4.5.2 näher beschrieben wird. Somit wird in der HyperGridXGL der beibehaltene semantische Zoom der HyperGrid sowie der hinzugefügte hierarchische Zoom vom Benutzer, wie von [Shneiderman, 1996] in der Visual Information Seeking Mantra beschrieben, direkt kontrolliert bzw. manipuliert.

4.5.2 Visualisierung

Nach dem Start von MedioVis mit der HyperGridXGL als Visualisierung, wird dem User ein einfaches Suchfeld angeboten. Durch die Eingabe eines Schlüsselworts in das Suchfeld und anschließendes Betätigen des Suchbuttons, wird der gesamte Datenbestand anhand dieses Schlüsselworts eingeschränkt und in der HyperGridXGL

visualisiert. Die HyperGridXGL visualisiert alle Daten in einer zoombaren Tabelle. Jede Zeile repräsentiert ein Objekt der Resultatmenge. Zu der Standard-HyperGrid wird eine neue Spalte eingefügt, die sich direkt rechts neben der Spalte mit den Flap-In/-Out-Buttons befindet (siehe Abbildung 36). In dieser Spalte befinden sich bei den Objekten bzw. Zeilen, die hierarchischer Natur sind, die so genannten Trigger-Buttons, die genauer in Kapitel 4.5.3 beschrieben werden.

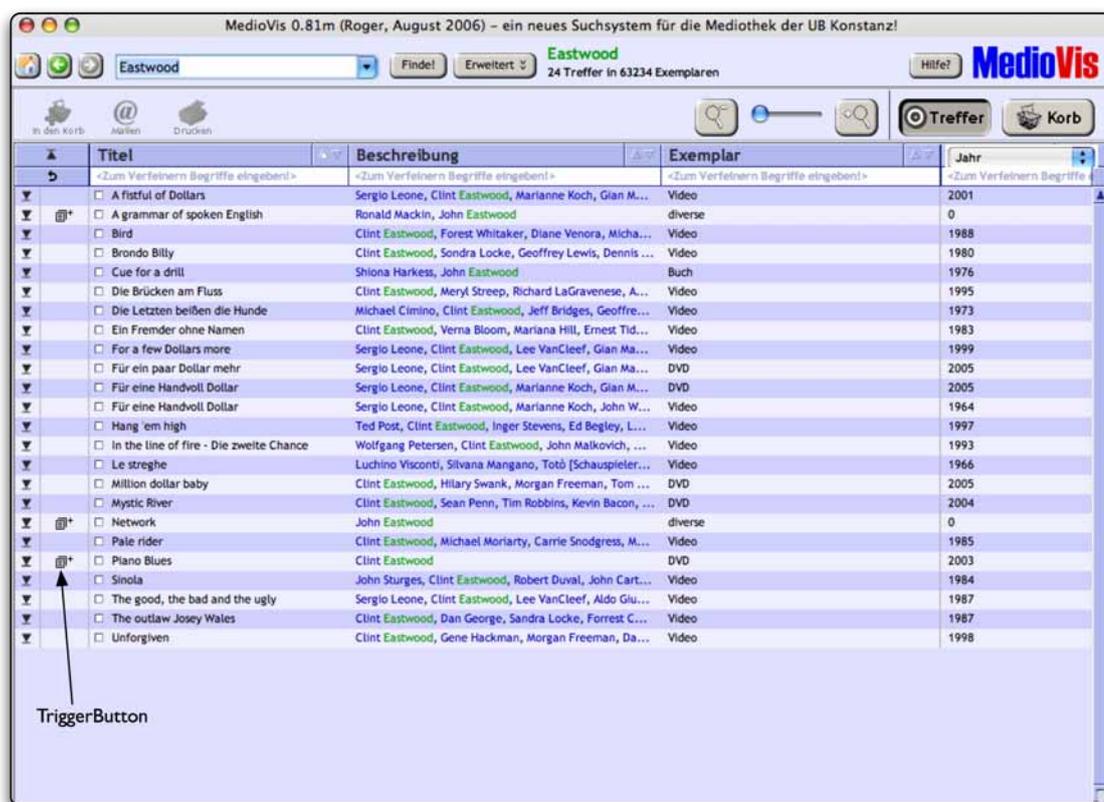


Abbildung 36: Die HyperGridXGL

In der TreeTable wird nur der Wurzel-Knoten bzw. Gesamttitel einer Hierarchie direkt nach der Suche in der TreeTable visualisiert. So muss man vom Gesamttitel auf dessen Untertitel schließen, was aber nicht immer möglich ist. Im Unterschied dazu werden in der HyperGridXGL alle von der Suche getroffenen Titel angezeigt. So werden weder Titel angezeigt, die überhaupt nicht getroffen wurden, noch werden Titel in der ersten Ansicht der Resultatmenge unterschlagen (vergleiche Kapitel 4.1). Ein Titel kann mehrere identische Exemplare besitzen, die aber nicht in der Resultatmenge angezeigt werden. Durch Anzeigen dieser Exemplare in der Resultatmenge würde man eine Redundanz an Information darstellen und würde sozusagen anderen Suchtreffern „den Platz wegnehmen“. Daher macht es Sinn, Exemplare eines Titels erst beim Explorieren der Hierarchie eines Objekts zu visualisieren. Zudem kann man in der HyperGridXGL die komplette Hierarchie eines Titels explorieren. Somit werden beim hierarchischen Zoom alle Knoten einer Hierarchie und nicht nur, wie in

der TreeTable, die von der Suche getroffenen Knoten angezeigt (siehe Abbildung 37).

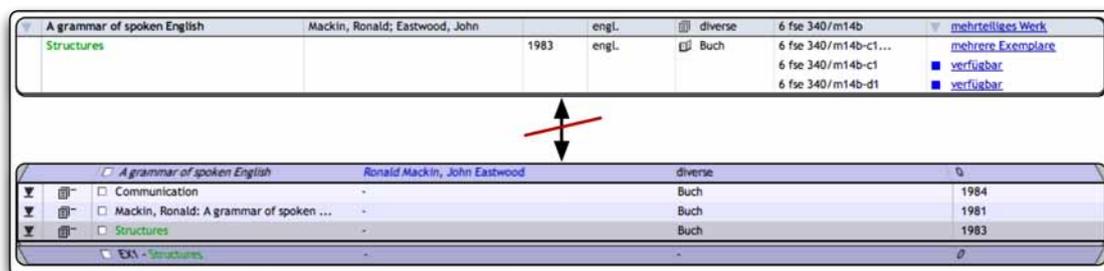


Abbildung 37: Unterschiedliches Ergebnis bei Suche nach Structures zwischen HyperGridXGL (unten) und TreeTable (oben)

Eines der größten Probleme bei der Visualisierung hierarchischer Daten ist die extrem große Informationsmenge, die benutzerfreundlich aufbereitet werden muss. Um die Masse an Information auf einer begrenzten Fläche darstellen zu können, benötigt man explizite Visualisierungsstrategien und -techniken. Bei solchen Visualisierungsstrategien und -techniken wird über das Layout und das graphische Design versucht, die benötigte Information in eine Ansicht zu packen. Verschiedene Design-Ideen, wie z.B. der Fisheye-Effekt, wurden bei der Star-Analyse in Kapitel 2.2 vorgestellt.

Ein heute üblicher Computer-Monitor ist im Vergleich zu einem Arbeitsplatz in der „realen Welt“, wie zum Beispiel einem Schreibtisch, verhältnismäßig klein [D. Austin Henderson u. Card, 1986]. Außerdem ist der Umfang der Information durch das menschliche Wahrnehmungsvermögen sehr begrenzt. Das menschliche Auge versucht jedoch mit der Masse an Information in der realen Welt durch die Kombination aus fließendem Übergang von fokalem Sehen (Detailansicht) zu peripherem Sehen (Gesamtüberblick) fertig zu werden [Resnikoff, 1989].

In der Visualisierung der Perspective Wall⁴⁶ [Mackinlay u. a., 1991] wird diese Technik der Integration einer ContextView und einer DetailedView in einer Ansicht auf linear-strukturierte Informationsräume angewandt. Die Ausgabe linear-strukturierter Information hat ein extremes Seitenverhältnis, das heißt, dass ein Ausgabefenster sehr breit sein müsste, um alles darzustellen. Schließlich liegen alle Elemente nebeneinander auf einer Ebene. Die Perspective Wall löst dieses Problem, indem dieser sehr breite Ausgabebereich auf eine dreidimensionale Wand gelegt wird (siehe Abbildung 38).

⁴⁶Die Perspective Wall wurde von XEROX Palo Alto Research Center entwickelt und wird heute von INXIGHT als TimeWall vertrieben, <http://www.inxight.com/products/sdks/tw/>

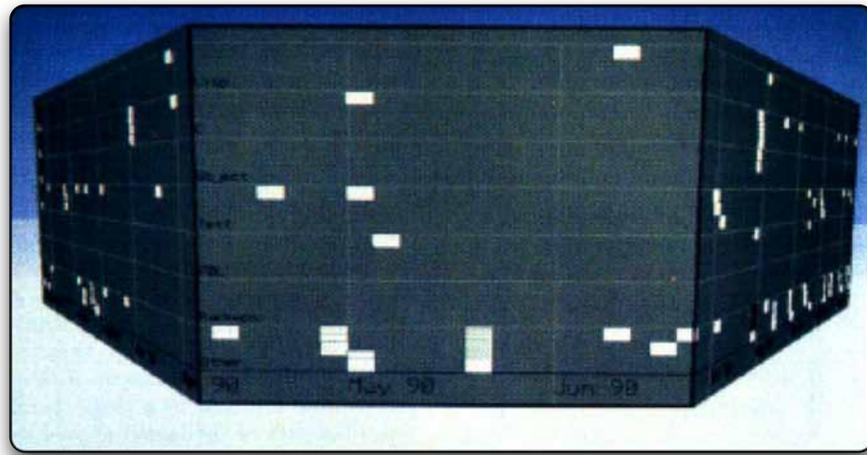


Abbildung 38: Die Perspective Wall [Mackinlay u. a., 1991]

Durch diese Anordnung wird die Ansicht in drei Teile aufgeteilt. Der mittlere Teil, die **DetailedView**, befindet sich „nahe“ zum Betrachter und zeigt die Elemente mit allen Details. Die „äußeren“ Teile, **ContextViews** genannt, werden vom Betrachter weggefaltet. Die Elemente dieser Bereiche werden dadurch perspektivisch verzerrt dargestellt [Peters, 2005]. Durch die Verzerrung im dreidimensionalen Raum, ist diese Darstellung sehr platzsparend. Obendrein erlaubt sie einen weichen Übergang zwischen Detailansicht und Überblick, was mit der Sehfähigkeit des menschlichen Auges positiv korreliert.

In der HyperGridXGL wird zum Darstellen der Hierarchie eines Titels eine ähnliche Visualisierungstechnik wie in der Perspective Wall verwendet. Aus der Tabelle der HyperGrid wird eine Art Schublade nach vorne ausgefahren. Auf der Vorderseite der Schublade befindet sich die **DetailedView**. Die Ober- sowie Unterseiten der Schublade, die sogenannten **ContextViews**, werden perspektivisch verzerrt im 3D-Raum „abgeklappt“. Da dieses Gebilde in Abbildung 39 an eine ausgezogene Schublade erinnert, wird dieser 3D-Konstrukt Schubladen-Effekt genannt. So kann man im Schubladen-Effekt eine um 90 Grad gedrehte Anlehnung an das Konzept der Perspective Wall entdecken. Allerdings werden hier hierarchische Daten und nicht wie bei der Perspective Wall, linear-strukturierte Daten verwendet bzw. abgebildet.

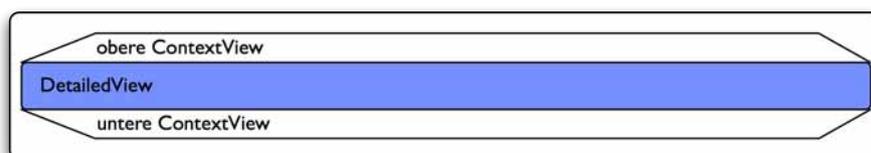


Abbildung 39: Der Schubladen-Effekt

Auf allen drei Flächen befinden sich Objekte der „gezoomten“ Hierarchie. Diese Objekte werden wie in der Tabellenstruktur der HyperGrid zeilenweise visualisiert. Auf der ContextView, also auf den weggefalteten Flächen oben als auch unten, sind diese Zeilen perspektivisch verzerrt dargestellt. Jede dieser drei Flächen repräsentiert eine Hierarchie-Ebene eines Objekts. Die obere ContextView beinhaltet somit das Vater-Objekt. In der DetailedView, das heißt in der dem Betrachter „nächstgelegenen“ Fläche, werden alle Elemente der Hierarchie-Ebene des Trigger-Docs angezeigt. Die zeilenweise angeordneten Elemente der DetailedView besitzen die gleiche Funktionalität wie die HyperGrid, lediglich die Hintergrundfarbe der Zeilen unterscheidet sich zur HyperGrid. Demnach ist die DetailedView „eine kleine HyperGrid in der HyperGrid“. In der DetailedView kann folglich auch der aus der restlichen HyperGrid schon bekannte semantische Zoom verwendet werden. So kann der Benutzer in der DetailedView, wie der Name schon sagt, alle Details eines Objekts durch Erhöhen des DOIs explorieren. Auf der unteren ContextView wird ein Kind-Objekt des fokussierten Objekts der DetailedView perspektivisch verzerrt dargestellt. Im Gegensatz zu der hellen Farbgebung der DetailedView sind die ContextViews ziemlich dunkel, wobei die obere ContextView ein wenig heller wie die untere ist. Dadurch und durch die hellen Farben der DetailedView soll der 3D-Effekt betont werden.

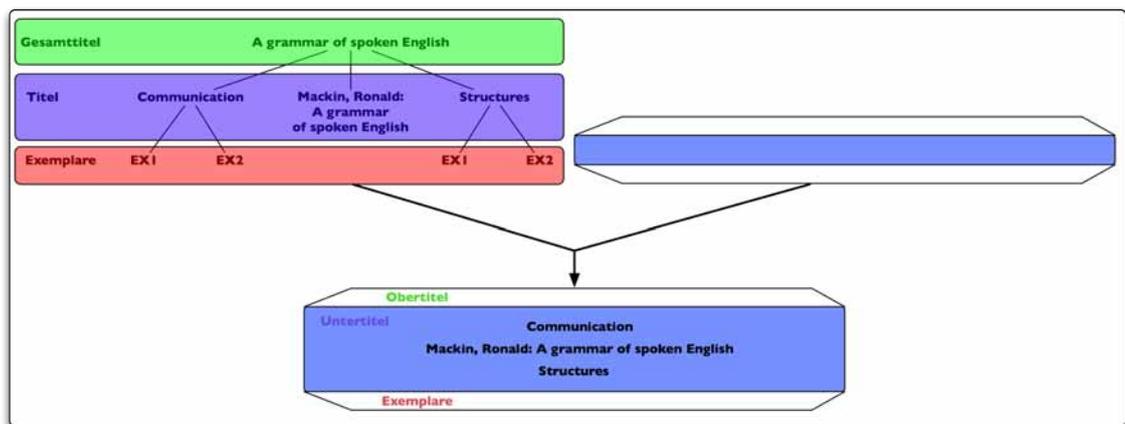


Abbildung 40: Synthese aus 3D-Schubladen und Hierarchie

Die Anzahl der Kind-Knoten des fokussierten Objekts der DetailedView wird mit Hilfe des **Stacks** angedeutet. Der Stack greift die Metapher eines Papier- bzw. Dokumentenstapels auf. So wird die Anzahl der Kind-Elemente durch hintereinander liegende bzw. aufeinander gestapelte ContextViews angedeutet (siehe Abbildung 41). Bei bis zu fünf Kind-Objekten wird mit Hilfe dieses „Visual Clues“ die genaue Anzahl angezeigt. Ab fünf Kinder wird immer nur ein Stack mit konstant fünf gestapelten ContextViews dargestellt.



Abbildung 41: 3D-Schublade mit Stack

Durch die ContextViews wird der direkte Kontext eines Elements in dessen Hierarchie veranschaulicht. Somit kann die HyperGridXGL durch eine gute Visualisierung der selektierten Hierarchie-Ebene und die ständige Darstellung der direkten Vor- und Nachfahren glänzen.

4.5.3 Interaktion

Der hierarchische Zoom wird durch einen Klick auf den **TriggerButton** ausgelöst. Der Standard-HyperGrid wurde, wie schon erwähnt, eine weitere Spalte hinzugefügt. Diese Spalte befindet sich direkt rechts neben der Flap-In/-Out-Button Spalte. Bei jedem Titel, der einer Hierarchie angehört, ist ein TriggerButton in dieser Spalte platziert.

Der Trigger-Button soll signalisieren, dass der hierarchische Zoom auf die Hierarchie des jeweiligen Titels per Klick ausgelöst werden kann. Daher ist es wichtig, dass dieser Button mit einem Symbol versehen ist, das sowohl für Hierarchie als auch für die Zoom-Funktion steht. Fast überall auf der Welt wird der Zoom mit einer Lupe oder einem Plus gleichgesetzt. Insbesondere im Bereich der Informatik bedeutet der Klick auf ein Plus oder eine Lupe, dass präsentierte Daten unterschiedlichster Ausgestaltung vergrößert werden; man zoomt also in einen Informationsraum hinein. Aus diesem Grunde ist es nachvollziehbar, dass man ein „Plus“ als Metapher für den Zoom-In verwendet. Folglich wird dann ein „Minus“ für den Zoom-Out benutzt.

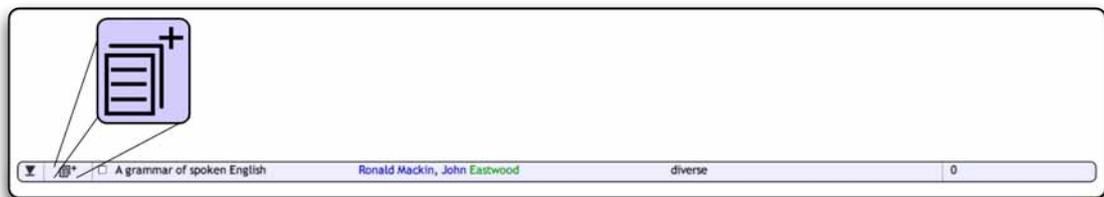


Abbildung 42: Der TriggerButton

Viele verbinden bildlich einen Baum mit einer Hierarchie. Daher wurde bei der Konzeptidee „cube“ auf dem TriggerButton das „Plus“ symbolisch mit einem Baum

verknüpft. Jedoch stellte sich in einer informellen Benutzerbefragung [Winkler, 2006] heraus, dass das Baum-Symbol des „cube“ zu klein und undeutlich ist. Da der für das Symbol zur Verfügung stehende Platz sehr gering ist, konnte man den Baum auf dem Button zudem nicht vergrößern. So galt es ein neues Symbol stellvertretend für den Baum bzw. die Hierarchie zu finden. Grundsätzlich wird durch den hierarchischen Zoom die Hierarchie eines Titels angezeigt, was gleichbedeutend mit dem Darstellen von weiteren Objekten und damit mehr Informationen ist. So wird hier die Metapher eines Papier- bzw. Dokumentenstapels, die auch der Stack verwendet, aufgegriffen. Zwei hintereinander bzw. aufeinander liegende Seiten stehen demnach symbolisch für eine Vielzahl von Dokumenten und damit für eine Hierarchie. Sinnvollerweise ist das Symbol des TriggerButtons also eine Kombination aus diesen zwei hintereinander liegenden Seiten und einem Plus bzw. Minus.

Klickt der Benutzer auf den TriggerButton, wird eine 3D-Animation ausgelöst. Zuerst werden die Zeilen unter dem TriggerDoc, dem Element, auf dessen TriggerButton der Benutzer geklickt hat, nach unten geschoben. Dabei entsteht ein grauer Zwischenraum zwischen dem TriggerDoc und den restlichen Suchtreffern. Anschließend löst sich die Zeile des TriggerDocs von der darüber liegenden Zeile und fährt nach unten. Gleichzeitig werden direkt ober- und unterhalb dieser Zeile die ContextViews in den 3D-Raum ausgeklappt. Durch das langsam anwachsende Aufklappen der ContextViews wird ein 3D-Effekt erzeugt. Auf der oberen ContextView befindet sich das Vater-Objekt des TriggerDocs, während sich auf der unteren ContextView ein Kind-Knoten befindet. Gibt es keine Kind-Knoten, bleibt die untere ContextView leer. Dementsprechend bleibt auch die obere ContextView leer, wenn schon das TriggerDoc dem Wurzel-Knoten einer Hierarchie entspricht und es daher keinen Vater-Knoten besitzt. Die Objekte der DetailedView sollen wie die restliche Resultatmenge der HyperGrid sortiert sein. Darum schieben sich die Elemente auf der Vorderseite der Schublade je nach Position in der DetailedView nach unten auf. Im Anschluss daran wird der Stack, je nach Anzahl der Kind-Knoten, zwischen dem letzten Dokument der DetailedView und der unteren ContextView „aufgezogen“. Zum Schluss der Animation wird das TriggerDoc markiert, indem es durch Verdunkeln der Hintergrundfarbe von den anderen Elementen abgehoben wird (siehe Abbildung 43).

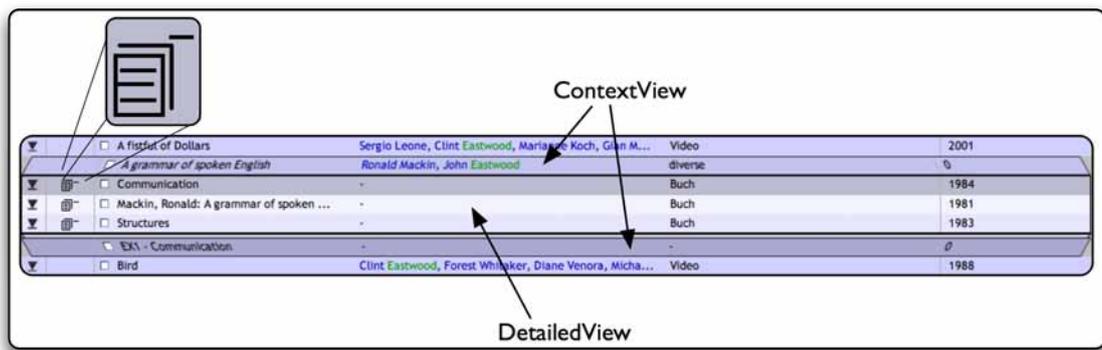


Abbildung 43: Aufgezoomte Hierarchie in der HyperGridXGL

Die HyperGrid der DetailedView unterscheidet sich nicht nur in der Farbgebung von der HyperGrid der Resultatmenge. Auch das Symbol auf dem Trigger-Button ist verändert. Der Trigger-Button wird in den Zeilen der DetailedView nicht mehr zum Auslösen des hierarchischen Zooms verwendet, sondern soll durch Klicken die „Schublade“ wieder einfahren. Symbolisch hierfür wird das „Plus“, das für das Einzoomen steht, durch ein „Minus“ ersetzt.

Die einzelnen Zeilen bzw. Objekte der DetailedView werden fokussiert und farblich markiert, sobald man mit der Maus darüber fährt. Konsistent dazu wird auch ein Kind-Knoten des aktuell fokussierten Objekts in der unteren ContextView angezeigt. Zusätzlich wird auch noch der Stack angepasst und in der Mitte der unteren ContextView wird ein textueller Hinweis, der die Anzahl der Kind-Knoten beinhaltet, platziert. Folglich ändern sich die untere ContextView und der Stack durch Bewegen der Maus über die verschiedenen Zeilen der DetailedView wie in Abbildung 44. Durch diese Interaktion soll dem Anwender sofortiges Feedback gegeben werden. Durch das Feedback wird zudem veranschaulicht, ob ein Objekt weitere Nachfahren besitzt oder nicht. Bei diesem Mouse-Over-Effekt spricht man daher von einem „Visual Clue“.

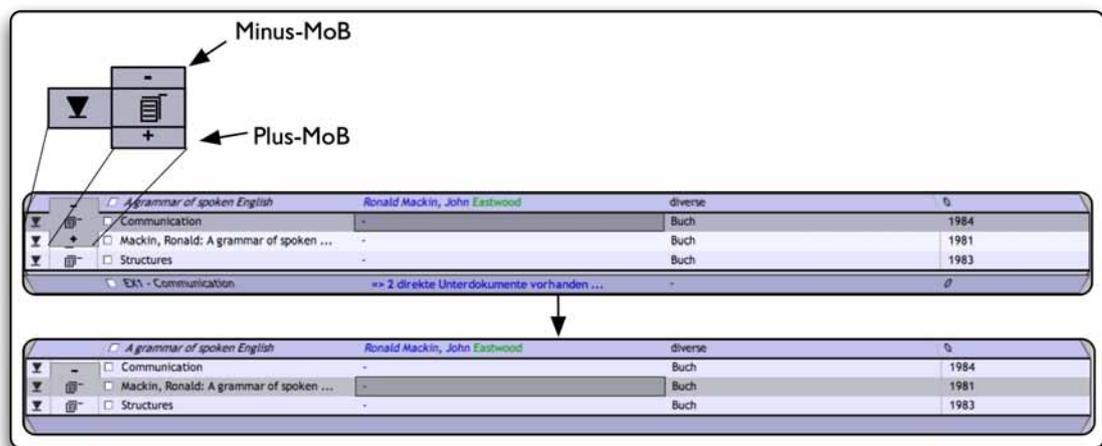


Abbildung 44: Direktes Feedback beim Überfahren der ContextView

Ein weiterer Mouse-Over-Effekt lässt in der mit der Maus überfahrenen Zeile der DetailedView die Mouse-over-Buttons erscheinen (siehe Abbildung 44). Durch Klick auf einen Mouse-over-Button, oder kurz MoB, kann man zwischen den verschiedenen Hierarchie-Ebenen hin- und hernavigieren. Die MoBs befinden sich ober- bzw. unterhalb des TriggerButtons. So gibt es bei Objekten, die sowohl einen Vater-Knoten als auch einen Kind-Knoten besitzen, einen Minus-MoB und einen Plus-MoB. Durch Klick auf den Plus-MoB eines Titels der DetailedView wird auf die Hierarchie-Ebene der Kind-Knoten „rotiert“. Rotieren bedeutet in diesem Zusammenhang, dass sich die Kind-Ebene in einer Rotations-Animation auf die DetailedView dreht. Zu Beginn dieser Animation werden alle Elemente der DetailedView durch das geklickte Element zusammengeschoben. Anschließend dreht sich das 3D-Gebilde dann nach oben. Dadurch gelangt der geklickte Titel auf die obere ContextView. Der auf der unteren ContextView angezeigte Kind-Knoten befindet sich jetzt vorne auf der Schublade und ist markiert. Dessen Kind-Knoten wiederum wird auf der unteren ContextView angezeigt. Zum Schluss der Rotations-Animation werden alle Elemente der jetzigen DetailedView aufgeschoben und der Stack des markierten Elements eingeblendet.

Wenn auf den Minus-MoB geklickt wird, wird die gleiche Animation nur in die andere Richtung durchgeführt. Dementsprechend dreht sich das 3D-Gebilde nach unten. Das geklickte Element der DetailedView wandert auf die untere ContextView. Die Elemente der Vater-Hierarchie-Ebene rotieren dann auf die Vorderseite der Schublade. Der Vater-Knoten des vorher angeklickten Elements wird markiert. Dessen Vater-Knoten wiederum wird nun auf der oberen Context-View angezeigt. Sollte ein Titel keine „Kinder“ haben, so wird bei diesem Titel nur der Minus-MoB angezeigt und man kann nur „nach oben“ rotieren. Ist ein Element der Wurzel-Knoten einer Hierarchie kann man folglich nur „nach unten“ navigieren und es wird daher nur der Plus-MoB angezeigt.

Zusätzlich zu den MoBs hat man auch die Möglichkeit zum Rotieren auf die Context-Views zu klicken, was dem Prinzip der „Direct Manipulation“ [Shneiderman, 1996] entspricht. Bei Klick auf die obere ContextView rotiert man auf die nächst höhere Hierarchie-Ebene. Bei Klick auf die untere ContextView hingegen rotiert man auf die Kind-Hierarchie-Ebene des markierten Titels der DetailedView. Wie bereits beschrieben, wird die untere ContextView beim Überfahren der Titel der DetailedView aktualisiert. Diese Aktualisierung wird mit einem Delay-Effekt verwirklicht. Durch diesen Effekt wird die untere ContextView erst dann aktualisiert, wenn ein Objekt der DetailedView eine gewisse Zeit fokussiert ist. Bei schnellem Überfahren der Elemente der DetailedView wird die untere ContextView nicht verändert. Nur so ist möglich, vom obersten Titel der DetailedView mit der Maus auf die ContextView zu gelangen ohne dass sich das Kind-Element der ContextView ständig verändert.

Wie bereits erwähnt stehen in der DetailedView alle Funktionen der HyperGrid zur Verfügung. So kann z.B. der semantische Zoom auf das fokussierte Objekt angewandt werden. Genauso kann die gesamte Resultatmenge sortiert oder durch den Tabellenfilter eingeschränkt werden.



Abbildung 45: Filtern, Sortieren, semantischer Zoom und Browserkomponente sind in die HyperGridXGL integriert

Beim Filtern in der HyperGrid wurden bisher alle Attribute eines AOIs eines Titels mit den Schlüsselbegriffen des Filters verglichen. Falls ein Attribut, egal ob momentan in der HyperGrid sichtbar oder noch in der Tiefe des Attributraumes versteckt, dem Filterkriterium entspricht, wird der Titel nicht herausgefiltert. Dadurch kann es passieren, dass Titel in der gefilterten Resultatmenge angezeigt werden, bei denen zumindestens auf den ersten Blick nicht zu erkennen ist aus welchem Grund sie nicht herausgefiltert wurden. Ist ein Titel hierarchisch, wird in der HyperGridXGL sogar dessen komplette Hierarchie dynamisch gefiltert. Dadurch werden beim HierarchieFilter nicht nur Attribute eines Objekts, sondern auch die Attribute aller der Hierarchie angehörenden Objekte mit dem Filterbegriff verglichen. Folglich bleibt eine noch größere Menge an nichtgefilterten Titeln erhalten. Es wird diese Filter-Methode verwendet, da beim Filtern von nur sichtbaren Attributen wesentliche Informationen unterschlagen werden können.

In der HyperGrid können die Suchtreffer sortiert werden. Es wird immer nach dem Attribut mit der höchsten Relevanz bzw. der obersten Position des jeweiligen AOIs, das heißt der jeweiligen Spalte, sortiert. Ist keine Hierarchie ausgezoomt, wird in der HyperGridXGL so sortiert wie bisher (siehe Kapitel 3.5). Sind jedoch eine oder mehrere Hierarchien verschiedener Titel der Resultatmenge ausgezoomt, so wird das TriggerTitel-Sorting verwendet. Beim TriggerTitel-Sorting wird die Resultatmenge so wie bisher sortiert, nur dass zusätzlich noch die aufgezoomten Hierarchien nach ihrem jeweiligen TriggerTitel⁴⁷ sortiert werden.

Bei dieser Methode kann es allerdings passieren, dass der TriggerTitel einer Hierarchie nicht sichtbar ist. Der TriggerTitel kann sich durch vorheriges Rotieren der Hierarchie-Ebenen auf einer momentan nicht angezeigten Ebene befinden. Folglich würde der Benutzer nicht sofort erkennen können, nach welchem Kriterium diese Hierarchie sortiert wurde. Um dies zu umgehen, könnte man z.B. nach dem momentan angezeigten Titel der oberen ContextView sortieren. Problem hierbei ist, dass beim Herauszoomen aus einer Hierarchie nur noch der TriggerTitel in der HyperGridXGL angezeigt wird. Falls nun nicht nach dem TriggerTitel sortiert wurde, würde beim Herauszoomen ein „Sprung“ stattfinden. Der TriggerTitel müsste von der Sortierposition des Objekts der ContextView auf die eigene Position in der Resultatmenge „springen“. Damit würde der Benutzer aus seinem Kontext gerissen werden und die Orientierung verlieren. Demnach wird das TriggerTitel-Sorting als Sortier-Methode in der HyperGridXGL verwendet, da der Kontext beibehalten wird.

⁴⁷TriggerTitel = TriggerDoc = Startelement

4.6 Umsetzung

Einen großen Teil dieser Arbeit nahm die praktische Umsetzung des Visualisierungs- und Interaktionskonzepts HyperGridXGL ein. Die HyperGridXGL wurde als eine neue eigenständige Visualisierung in die aktuelle Version des MedioVis Frameworks (0.81m) integriert. Dabei wurde die HyperGridXGL wie das komplette MedioVis Framework in JAVA⁴⁸ realisiert. Das Ergebnis der Ausimplementierung des Konzepts ist ein Prototyp, mit dessen Hilfe durch die in Kapitel 5 vorgestellte Evaluation das Visualisierungs- und Interaktionskonzept der HyperGridXGL auf seine Gebrauchstauglichkeit getestet wurde.

Das MedioVis Framework setzt sich aus vielen unabhängigen Modulen zusammen. Abstrahiert kann man das Framework in vier Schichten unterteilen:

- **Datenquellen**
- **Datenanfrage und Input**
- **Datenmodell**
- **Visualisierung und Interaktion**

Entsprechende Details zu den verschiedenen Schichten sowie zur Programmierung des Datenmodells und der HyperGrid können der Entwicklung von MedioVis [Grün, 2004] sowie dem Redesign des Frameworks [König, 2005] entnommen werden. Bei der Realisierung der HyperGridXGL spielten im wesentlichen die letzten zwei Schichten eine wichtige Rolle. So wurde das Datenmodell an die neuen Anforderungen angepasst bzw. dementsprechend erweitert. Diese Veränderungen des Datenmodells nahmen den größten Teil der praktischen Umsetzung ein. Danach wurde auf der Grundlage des aktualisierten Datenmodells eine neue Visualisierung und deren Interaktionen erstellt.

Festzuhalten ist, dass das bisher programmierte einem Prototypen entspricht, der daher noch nicht die Stabilität einer finalen Fassung erreicht hat. In der Vorbereitung der Evaluation wurden auf dem PC des Usability Labs Probleme des Prototypen in Form von Zeichen-Fehlern festgestellt. Diese Fehler hängen mit der Unterstützung von Hyper-Threading⁴⁹ des Lab-PCs zusammen. Indem man das Hyper-Threading abschaltete bzw. die Zugehörigkeit des Java-Prozesses des Prototypen auf dem Lab-PC einem CPU zuordnete, konnte man die auftretenden Probleme beseitigen. So konnte die Evaluation schlussendlich ohne Probleme durchgeführt werden.

⁴⁸JAVA 1.5, Sun Microsystems, <http://java.sun.com>

⁴⁹Durch Hyper-Threading können Prozesse auf einem CPU parallel verarbeitet werden, <http://www.intel.com/technology/hyperthread/>

Im Folgenden soll nun geschildert werden, auf welche Weise das Konzept programmiertechnisch gelöst wurde. Dafür wurden explizit zwei Teilprobleme mit interessanten Lösungswegen ausgesucht, wobei diese durch Schaubilder vereinfacht dargestellt werden sollen.

4.6.1 Datenmodell

Wie schon erwähnt beschäftigt sich der bei weitem größere Teil der Programmierarbeit mit der Anpassung und Erweiterung des Datenmodells. Ein wichtiges Ziel des Konzepts ist die Visualisierung aller Suchtreffer in der HyperGridXGL. Somit sollen sich nicht nur Gesamttitel in der Resultatmenge befinden, sondern alle gefunden Titel. Dadurch kann es passieren, dass mehrere Titel einer Hierarchie in der Resultatmenge zu finden sind und damit als Startpunkt für die Exploration jener Hierarchie genutzt werden. Folglich kann ein und derselbe Titel als Suchtreffer und als Titel einer aufgezoozten Hierarchie eines anderen Suchtreffers vorkommen. Dies setzt voraus, dass ein Titel mehrfach in der HyperGridXGL visualisiert werden kann. Jeder Titel, das heißt einzelne DVDs, CDs oder Bücher, entspricht im MedioVis-Datenmodell einer Instanz der Klasse Document. Referenzen dieser Document-Objekte wiederum werden im DocManager gespeichert, wobei der DocManager für das Verwalten dieser Objekte zuständig ist. So wird beim Visualisieren eines Documents und auch bei der Interaktion mit einem visualisierten Document auf den DocManager zugegriffen. Dabei werden sowohl die Attribute aus den Daten eines Documents ausgelesen als auch Zustände, wie z.B. „gefiltert“ oder „nicht gefiltert“ oder die Position in der Visualisierung, verändert. Die Document-Objekte werden in einem Hash abgespeichert, wobei der Hashkey über die verschiedenen Attribute eines Documents errechnet wird. Da ein Document immer die gleichen Attribute hat, ist es im bisherigen Datenmodell lediglich möglich, ein Document einmal im Hash abzulegen. Problem dabei ist aber, dass in der HyperGridXGL ein Document mehrfach vorkommen und damit auch verschiedene Zustände, wie Objekt einer DetailedView oder Objekt der Suchtreffermenge, besitzen kann. Diese Zustände sollen nun gesondert in einem weiteren Objekt abgespeichert werden. In der Abbildung 46 soll gezeigt werden, wie das Zusammenspiel zwischen Visualisierung und Datenmodell nach der Einführung des DocID-Objekts auf einer abstrakten vereinfachten Ebene funktioniert.

In der Abbildung 46 sieht man auf der linken Seite symbolisch für das Datenmodell den DocManager und auf der rechten Seite symbolisch für die Visualisierung die HyperGridXGL. Auf eine Interaktion des Benutzers mit der HyperGridXGL, wie suchen, zoomen, sortieren oder filtern, folgt eine Neu-Sortierung der dargestellten Titel in der Tabelle der HyperGridXGL. Bei der Sortierung werden die IDs⁵⁰ der Document-Objekte der angestrebten Reihenfolge entsprechend in einem Array

⁵⁰Die ID ist die Zahl über die das Document eindeutig identifiziert werden kann.

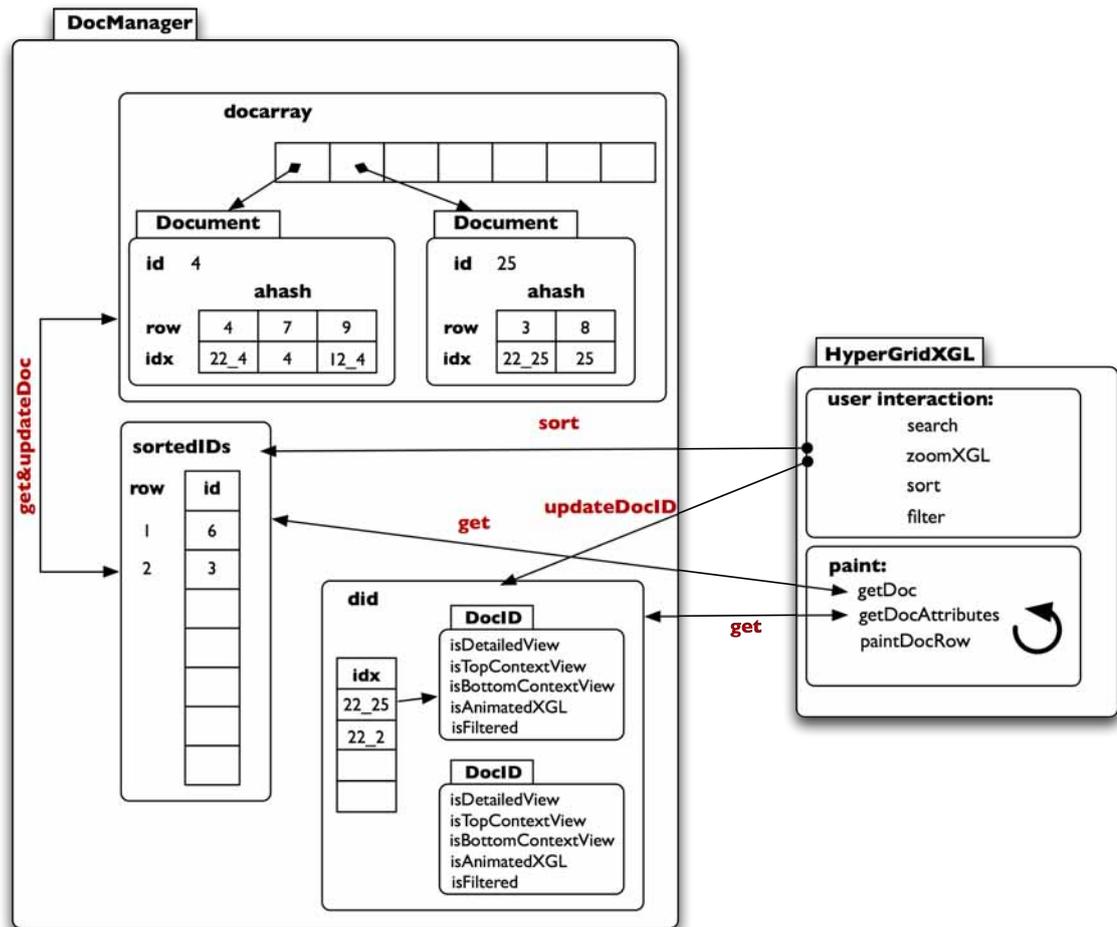


Abbildung 46: Zusammenspiel aus Datenmodell und Visualisierung

(sortedIDs) im DocManager umsortiert. Während der Umsortierung der IDs im sortedIDs-Array wird auch der Hash (ahash) eines jeden Documents aktualisiert. In diesem Hash zeigt jede Zeile als Hashkey auf den Hashvalue idx. Eine idx wie 22_25 besteht aus der ID des Documents (25). Falls das Document zusätzlich noch Teil einer Hierarchie ist, wird die eigene ID mit der ID des TriggerDocs (22) kombiniert. Gleichzeitig werden auch noch die Eigenschaften des jeweiligen Documents der jeweiligen Zeile in Form von DocID-Objekten abgespeichert. Im Anschluss an die Neu-Sortierung wird die HyperGridXGL neu gezeichnet. Dabei werden die Documents in der Reihenfolge des sortedIDs-Arrays „angepackt“. Um alle Eigenschaften eines Documents auszulesen, muss über einen weiteren Hash (did) das jeweilige DocID-Objekt angeschaut werden. Die DocID-Objekte können dann über den vom ahash-Hash bekannten idx-Wert des Docs in der jeweiligen Zeile im did-Hash erreicht werden. Wenn alle Eigenschaften bekannt sind, kann das jeweilige Document-Objekt gezeichnet werden.

4.6.2 Visualisierung

Zwar nahm das Datenmodell den weitaus größeren Teil der Programmierarbeit ein, dennoch beschreibt die Benutzeroberfläche den *wichtigeren* Teil dieser Arbeit. Denn die Benutzeroberfläche, bestehend aus der Visualisierung und dem Interaktionsangebot der HyperGridXGL, stellt den Berührungspunkt zwischen MedioVis und dem Benutzer dar. Aus technischer Sicht ist insbesondere die Darstellung der ContextViews interessant. In der HyperGridXGL wird jede Zelle einer Zeile in einem eigenen Schritt gezeichnet. Dabei befinden sich alle Zellen der HyperGridXGL auf einem Panel⁵¹. Für die perspektivische Verzerrung der ContextViews wird ein Image-Filter verwendet. Dieser Image-Filter kann Images anhand von vorgegebenen Parametern perspektivisch verzerren. Um die Tabelle der HyperGridXGL darzustellen, wird jede Zeile von oben nach unten durchgegangen und dementsprechend auf das Panel gezeichnet. Das Zeichnen der ContextViews funktioniert jedoch auf eine andere Weise.

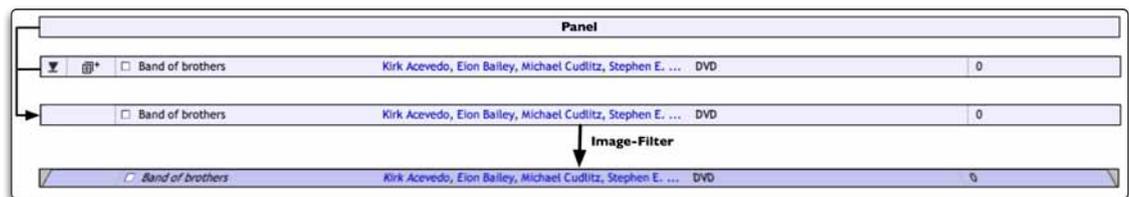


Abbildung 47: Pre-Processing bei der Erzeugung der ContextView

Sobald der hierarchische Zoom eines Titels ausgelöst wird, findet wie beschrieben eine Neu-Sortierung der dargestellten Dokumente statt. Während dieser Neu-Sortierung wird Pre-Processing für das Visualisieren der ContextViews betrieben. Zum Darstellen einer ContextView wird ein Panel mit den Ausmaßen einer Zeile erstellt. Auf dieses Panel werden alle Zellen der Zeile des ContextView Titels gezeichnet. Wie man in Abbildung 47 erkennen kann, werden dabei für die ContextView Komponenten, wie z.B. der Flap-In/-Out-Button und der TriggerButton, weggelassen. So erhält man auf dem Panel die vereinfachte Darstellung einer Zeile wie sie nun auf den ContextViews zu sehen ist. Anschließend wird das Panel in ein Image-Objekt umgeschrieben. Dieses Image-Objekt wird in einem Hash abgelegt, um so eine spätere Verwendung zu ermöglichen. Sobald die Umsortierung der Titel abgeschlossen ist, beginnt eine schrittweise Animation des hierarchischen Zooms. Bei jedem Animationsschritt wird die HyperGridXGL neu gezeichnet. Zu Beginn der Animation werden die ContextViews „anwachsend“ in den 3D-Raum ausgeklappt. Dieser Effekt wird durch schrittweises perspektivisches Verzerren des zuvor erstellten Image-Objekts erreicht. Daher wird vor jedem Animationsschritt mit Hilfe eines Image-Filters das Image-Objekt, genau durch Parameter bestimmt, perspektivisch

⁵¹Das Panel entspricht der einfachsten JAVA Container Klasse, die „space“ für andere Components zur Verfügung stellt

verzerrt und auf ein Neues im Hash abgespeichert. So kann bei jedem Repaint-Vorgang der HyperGridXGL an der Stelle der ContextView das perspektivisch verzerrte Image-Objekt, das im Hash gespeichert wurde, gezeichnet werden.

Der aus dieser praktischen Umsetzung entstandene Prototyp wird im nächsten Kapitel mit Hilfe einer Evaluationsstudie auf seine Gebrauchstauglichkeit getestet.

5 Evaluation

Im Rahmen einer formativen Evaluation des Interaktions- und Visualisierungskonzepts der HyperGridXGL wurden unabhängige Versuchspersonen beauftragt, die neuartige Darstellung hierarchischer Daten innerhalb der Tabellenstruktur der HyperGrid zu testen. Als Anwendungsszenario und Datengrundlage für diese Studie kam der Datenbestand der Mediothek Konstanz mit gespiegelten Daten verschiedener Quellen, wie in Kapitel 2.1.3 näher beschrieben, zum Einsatz.

Im Wesentlichen konzentrierte sich dieser Test auf das Aufdecken möglicher Usability-Probleme sowie auf den grundsätzlichen Umgang der Versuchspersonen mit der Benutzung von Hierarchien. Hierzu wurde ein Test-Design entworfen, das einen Methodemix aus qualitativem Benutzertest, halbstrukturiertem Interview und Erhebung quantitativer Daten mittels des Fragebogens AttrakDiff [Hassenzahl u. a., 2003] umfasst.

5.1 Ziele und Absichten des Tests

Mit Ausnahme des schon erwähnten AttrakDiff-Fragebogens wurden in der Studie keine weitere quantitativen Daten erhoben. Das Ergebnis dieses Fragebogens ist aufgrund der geringen Anzahl an Versuchspersonen mit Vorbehalt zu interpretieren. Trotz des Vorbehalts lässt das Ergebnis eine Tendenz erkennen. Die grundsätzliche Zielsetzung bestand darin, das Visualisierungs- und Interaktionskonzept der HyperGridXGL auf seine Benutzerfreundlichkeit zu testen. Daher wurde bei der Ausarbeitung des Methodik-Designs insbesondere Wert auf die Erlangung qualitativer Ergebnisse gelegt. So sollte gewährleistet sein, dass während der Evaluation möglichst umfassend alle eventuell vorhandenen Usability-Probleme aufgedeckt werden und zugleich die subjektive Meinung jeder Versuchsperson zum Gesamtkonzept in Erfahrung gebracht werden kann.

Darüber hinaus lassen sich unterschiedliche Zielsetzungen bezüglich der jeweiligen Phasen der Untersuchung feststellen. Nachdem alle Versuchspersonen das Demonstrationsvideo⁵² zur Benutzung der HyperGrid verfolgt hatten, sollten sie selbstständig die HyperGrid erkunden. So konnte gewährleistet werden, dass alle Versuchsperson zu Beginn des eigentlichen Tests den gleichen Wissensstand in Bezug auf den Umgang mit der HyperGrid hatten. Schließlich sollte in dieser Untersuchung nicht die HyperGrid auf mögliche Usability-Probleme bzw. ihre Benutzerfreundlichkeit geprüft werden, sondern die Versuchsperson sollte sich ausschließlich auf das Testen des neuen Interaktions- und Visualisierungskonzepts zur Integration hierarchischer

⁵²Unter <http://hci.uni-konstanz.de/MedioVis> kann ein 9-minütiges Demonstrationsvideo der MedioVis-Oberfläche betrachtet werden

Daten, die HyperGridXGL, konzentrieren.

Nachdem die Möglichkeiten und die Funktionsweise der HyperGrid den Versuchspersonen klar waren, begann die zweite Phase der Evaluation der HyperGridXGL. In dieser Phase sollten die Versuchspersonen ohne jegliche Aufgabenstellung die Erweiterung der HyperGrid explorieren. Fragen seitens der Versuchspersonen wurden vom Versuchsleiter mit Tipps und Hinweisen beantwortet, jedoch nicht mit einer kompletten Erklärung. Die Versuchspersonen sollten die Antworten zu ihren Fragen selbst finden. Demnach bestand die Zielsetzung dieser Phase darin, zu untersuchen, inwieweit die Benutzer in der Lage sind, die neuen Möglichkeiten des Systems selbstständig zu erkennen.

In Phase drei sollte dann mit Hilfe konkreter Aufgabenstellungen untersucht werden, ob den Teilnehmern ein selbstständiger Umgang mit dem vorgestellten System möglich ist. Die Aufgaben erfüllten ihren Zweck, indem sie die Versuchsperson zum Benutzen des kompletten Funktionsumfangs der HyperGridXGL veranlassten, was den Versuchspersonen bei einer freien Exploration ohne eigene Motivation nur bedingt möglich gewesen wäre.

Die vierte und letzte Phase diente dazu, den vorangegangenen Test gemeinsam mit dem jeweiligen Teilnehmer zu rekapitulieren. So wurde der Teilnehmer auf konkrete Funktionen sowie auf aufgetretene Probleme nochmals angesprochen und zu einer umfassenden Aussage aufgefordert. Zum Schluss wurden die Stärken und die Schwächen des Systems diskutiert, um so weitere Anregungen zu Verbesserungen zu erlangen.

5.2 Testmethode

5.2.1 Versuchspersonen

Am Test nahmen insgesamt acht Versuchspersonen, vier männliche und vier weibliche, teil. Die erste Versuchsperson diente zudem als Pre-Tester, das heißt mit ihrer Hilfe wurde das Test-Design, insbesondere die Testaufgaben, für die restlichen Tests optimiert. Alle Versuchspersonen sind Studenten an der Universität Konstanz, sie gehören jedoch sechs unterschiedlichen Fachbereichen an, wodurch eine gewisse Heterogenität insbesondere in Bezug auf vorhandene Computererfahrung gewährleistet werden konnte. Die getestete Erweiterung der HyperGrid zur Integration hierarchischer Daten war allen Teilnehmern neu und unbekannt, jedoch verfügten sie über eine unterschiedliche Erfahrungstiefe mit MedioVis in der derzeitigen Bib-Version⁵³. Nähere Informationen zu den einzelnen Teilnehmern können der Tabelle in der Ab-

⁵³Die aktuelle Version von MedioVis in der Bibliothek enthält lediglich die klassische Visualisierung (TreeTable) zur Darstellung von Hierarchien

bildung 48 entnommen werden.

VP	Geschlecht	Alter	Studiengang	PC-Nutzung /Tag in Stunden	Internet-Nutzung /Tag in Stunden	Informationsquelle im internet	Erfahrung mit Suchmaschinen in Jahren	Zufriedenheit mit Suchergebnissen *	Kennt MedioVis	Benutzung von MedioVis †
1	männlich	25	Information Engineering	2-3	2-3	google, Links von Freunden/ Zeitschriften	3-5	4	nein, aber davon schon gehört	nie
2	weiblich	23	Sprachwissenschaften	0-1	0-1	google, Links von Freunden	1-3	3	ja	1
3	männlich	20	Deutsche Literatur	1-2	0-1	Suchmaschinen	1-3	3	nein, aber davon schon gehört	nie
4	weiblich	20	Spanisch/Deutsch BA	3+	3+	google, Links von Freunden	3-5	4	nein, aber davon schon gehört	nie
5	männlich	21	Literatur-Kunst-Medien	2-3	2-3	google, Links von Freunden	1-3	4	nein, aber davon schon gehört	nie
6	weiblich	22	Literaturwissenschaften	2-3	2-3	google, Links von Zeitschriften	5+	3	ja	2
7	männlich	27	Verwaltungswissenschaften	3+	2-3	google, Links von Freunden/ Zeitschriften	3-5	3	ja	3
8	weiblich	24	Englisch/Deutsch/ Geschichte	1-2	1-2	google, Links von Freunden,spiegel.de,ub.uni-konstanz.de	1-3	3	ja	1

* 1 = völlig unzufrieden, 5 = vollkommen zufrieden

† 1 = sehr selten, 5 = mehrmals pro Woche

Abbildung 48: Übersicht der Versuchspersonen

5.2.2 Testumgebung

Die Evaluationsstudie wurde an der Universität Konstanz im Usability Lab (D208) der Arbeitsgruppe Mensch-Computer Interaktion durchgeführt. Das Usability Lab ist ein kleiner, heller Raum mit der für die Universität Konstanz üblichen Büroausstattung.

Während des Tests war neben der Versuchsperson noch eine weitere Person, der Versuchsleiter, im Lab anwesend. Dieser war für den reibungslosen Ablauf des Tests zuständig und zugleich der Ansprechpartner der jeweiligen Versuchsperson. Der Versuchsleiter begrüßte die Versuchsperson, wies sie in den Test ein, händigte die Fragebogen und die Aufgabenstellungen aus und führte die abschließenden Interviews durch. Insbesondere während der explorativen Phasen eins und zwei konnten die Versuchspersonen ihre allgemeinen Fragen an ihn richten.

Eine weitere Person, unsichtbar für die jeweilige Versuchsperson, verfolgte den Test in einem angrenzenden Büro mit Hilfe der RemoteViewer Funktionalität der Aufzeichnungssoftware Morae.

5.2.3 Hard- und Software Equipment der Testumgebung

Betriebssystem:	Microsoft Windows XP SP2
Java Version:	JAVA 1.5.0_10
PC:	Intel Pentium IV 3 GHz 1 GB Arbeitsspeicher
Monitor:	WACOM Cintiq 17SX LCD Tablet
Auflösung:	1280x1024px
Farbtiefe:	32bit
Webcam:	Logitech QuickCam Notebook Pro

5.2.4 Testwerkzeuge

Für die Aufzeichnung sowie für die anschließende Auswertung des Videomaterials wurde die Software Morae Version 1.1.0 der Firma Techsmith⁵⁴ verwendet. Mit Hilfe von Morae wurde der Bildschirminhalt mit sämtlichen Eingaben und Mausbewegungen aufgezeichnet. Eine Webcam mit integriertem Mikrofon, die recht unauffällig auf dem TFT-Monitor platziert wurde, übernahm die Dokumentation der Frontalansicht und der Kommentare der jeweiligen Versuchsperson. Mit Hilfe dieser umfangreichen Aufzeichnungen war es im Anschluss daran möglich, den Test lückenlos nachzuvollziehen, zu analysieren und auszuwerten.

Durch die RemoteViewer Funktion von Morae konnte der Test auch an einem zweiten PC in einem weiteren Raum von einer weiteren Person verfolgt werden.

Anzumerken wäre zu dieser Untersuchung noch, dass die Aufzeichnungssoftware die Systemgeschwindigkeit leicht beeinträchtigte und beispielsweise die Rotations-Animation nicht immer völlig „ruckelfrei“ bzw. flüssig ablaufen konnte. Inwieweit dies die Versuchspersonen in ihren Wertungen zur HyperGridXGL beeinflusste, ist nicht bewertbar.

5.2.5 Testaufgaben

Der Einsatz der Aufgaben fand nur in der zweiten Phase des Tests statt. Basierend auf dem Hauptanwendungsszenario von MedioVis, der „Mediothek“, und auch auf deren angereichertem Datenbestand, wurde bei der Gestaltung der Testaufgaben vor allem auf folgende Gesichtspunkte Wert gelegt:

⁵⁴Weitere Information zu der verwendeten Software sowie eine Trial-Version kann unter <http://www.techsmith.com/morae.asp> heruntergeladen werden

Im Vordergrund stand die Benutzung von Hierarchien, bezogen auf die verschiedensten Use-Cases, die auch bei der Nutzung von KOALA bzw. der klassischen MedioVis-Version entstehen könnten. Demnach sollten die Testaufgaben speziell das Benutzen der Hierarchie beinhalten. Beim Gestalten der Aufgaben war es unter anderem wichtig die Integration des hierarchischen Zooms in die HyperGrid zu testen. Zudem sollte mit den Aufgaben möglichst der komplette Funktionsumfang der HyperGridXGL abgedeckt werden.

Es muss darauf hingewiesen werden, dass bei den verwendeten Testaufgaben die Übertragbarkeit auf reale Einsatzzwecke nicht im Vordergrund stand. Daher wurden im Verlauf der Untersuchung weder Taskzeit noch Erfüllungsgrad gemessen.

Die Aufgaben und deren Reihenfolge, in der sie zu bearbeiten waren, wurden innerhalb des Testteams eingehend diskutiert und nach dem ersten Test, dem sogenannten Pre-Test, nochmals angepasst.

Der genaue Wortlaut der einzelnen Aufgaben ist dem Anhang A zu entnehmen.

5.2.6 Test-Design

Das Test-Design der hier beschriebenen Untersuchung entspricht einem formativen User-Test, welcher auf einer qualitativen Methodik basiert, die entsprechend der Zielsetzung das Aufdecken von Usability-Problemen ermöglicht, jedoch nicht die Prüfung von Hypothesen unterstützt. Da bei der qualitativen Forschung teilweise das Problem besteht, valide Ergebnisse zu erhalten, ist zu empfehlen, das Test-Design aus einem Methodenmix, wie auch hier angewandt, zu gestalten [Gerken, 2004]. In der vorliegenden Untersuchung wurde eine **teilnehmende Beobachtung mit einer semi-strukturierten Befragung** in Form eines Interviews kombiniert. Da der Teilnehmer in der ersten und zweiten Phase des Tests die Möglichkeit hat, mit dem Versuchsleiter zu kommunizieren, der Versuchsleiter aber in der dritten Phase eine rein beobachtende Stellung einnimmt, kann man hier von einer **teilnehmenden Beobachtung** bzw. lediglich von einer **Beobachtung** sprechen.

5.2.7 Ablauf des Tests

Nach der Begrüßung der Versuchsperson durch den Versuchsleiter, wurde diese zunächst gebeten, vor dem Testrechner Platz zu nehmen und sich in einem vorbereiteten Willkommenschreiben über den Inhalt des Tests zu informieren. Bereits hier wurde für die Versuchsperson klargestellt, dass die HyperGridXGL bei dieser Untersuchung auf dem Prüfstand steht und nicht ihr Benutzer. Folglich sah sich die Versuchsperson in der Rolle des Prüfers, welcher dabei helfen soll, die HyperGridXGL auf Benutzungsprobleme und Fehler zu prüfen und zu verbessern. Sie sollte von Anfang an das Gefühl erhalten, sie sei als kompetenter Gesprächs- und Diskussionspart-

ner dazu aufgerufen, Probleme aufzudecken und - noch wichtiger - Verbesserungen anzuregen.

Anschließend wurde der Versuchsperson erklärt, dass der Test per Video aufgezeichnet wird. Um die Aufzeichnung und deren Verwendung zu legitimieren, wurde vorab eine schriftliche Einverständniserklärung eingeholt. Vor Beginn des Tests war ein Pre-Test-Fragebogen auszufüllen. Dieser Fragebogen sollte von den Versuchspersonen demographische Daten und weitere Informationen hinsichtlich der Computernutzung und Erfahrung, insbesondere im Umgang mit MedioVis, erfassen. Der genaue Wortlaut des Pre-Test-Fragebogens kann dem Anhang C entnommen werden.

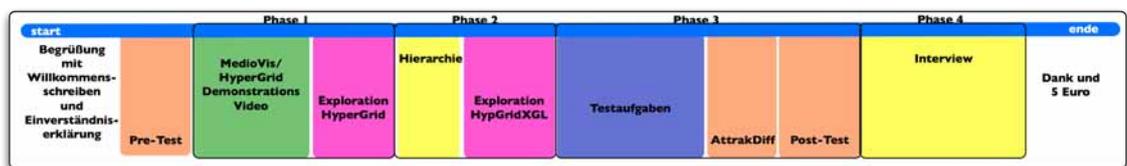


Abbildung 49: Zeitlicher und inhaltlicher Ablauf des Tests

Die erste Phase des Tests begann mit dem Starten des Demonstrationsvideos zur Nutzung der HyperGrid. Nachdem alle für den Test relevanten Funktionen der HyperGrid erklärt und demonstriert wurden, stoppte der Versuchsleiter das Video. Die Versuchsperson sollte nun das soeben im Video beobachtete Procedere „live“ an der HyperGrid ausprobieren und gleichzeitig möglichst alle Gedanken laut äußern⁵⁵. Ziel hierbei war lediglich zu testen, ob die Versuchsperson die Funktionalität, die im Video vorgestellt wurde, verstanden hat und anwenden kann. Der Versuchsperson wurde erklärt, dass es in dem Test nicht um die HyperGrid geht, sondern um eine Erweiterung der HyperGrid und dass es wichtig sei, die Funktionsweise der HyperGrid verstanden zu haben. Sobald sich die Versuchsperson sicher im Umgang mit der HyperGrid fühlte, ging es weiter zur nächsten Phase.

Zu Beginn der zweiten Phase wurde der Versuchsperson anhand eines trivialen Beispiels (siehe Abbildung 50) das Über- und Unterordnungsprinzip einer Hierarchie erklärt.

⁵⁵Im Bereich des Usability Engineering wird diese Methode „thinking aloud“ genannt

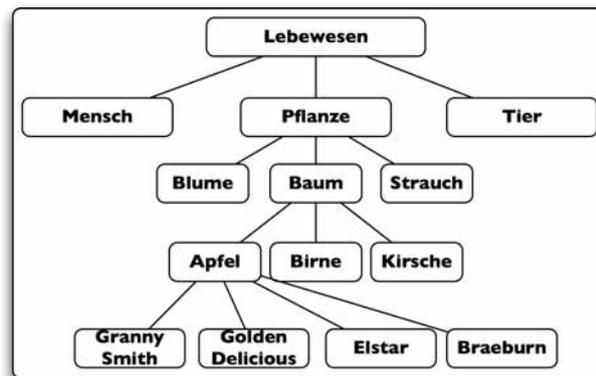


Abbildung 50: Beispiel-Hierarchie

Anschließend sollte die Versuchsperson, zunächst ohne Anleitung, die HyperGridXGL erkunden. Ergänzend wurde in diesem Zusammenhang noch vorgegeben, dass Unterschiede und Erweiterungen zur HyperGrid festzuhalten seien. Wie in der ersten Phase sollte die Versuchsperson ihre Gedanken zum Test laut äußern. Ziel war dabei herauszufinden, inwieweit ein Benutzer die Erweiterung entdecken würde, ohne darüber informiert zu sein, und wie er mit den Neuerungen zurechtkommt. Wenn in dieser Phase Fragen aufkamen, gab der Versuchsleiter Tipps und Hinweise, um so dem Teilnehmer das Beantworten seiner eigenen Frage zu ermöglichen. Die Versuchsperson wurde gebeten, selbst das Signal zu geben, dass sie die HyperGridXGL ausreichend lang untersucht hat und sich dabei über die Funktionalitäten klar geworden ist.

In Phase 3 erhielt die Versuchsperson vom Versuchsleiter konkrete Aufgabenstellungen zur selbstständigen Bearbeitung. Auf Karteikarten festgehaltene Aufgaben wurden der Versuchsperson der Reihe nach vorgelegt. Die Versuchsperson wurde gebeten, die Aufgaben laut vorzulesen und sie anschließend zu bearbeiten. Wie schon in den vorherigen Phasen sollte die Versuchsperson möglichst laut alle Gedanken zur Bearbeitung der Aufgabe äußern. Sobald - der eigenen Meinung zufolge - die Aufgabe erfüllt war oder für nicht lösbar eingeschätzt wurde, sollte dies signalisiert werden. Sofern eine Aufgabe von der Versuchsperson als nicht lösbar eingestuft wurde, gab der Versuchsleiter Tipps und Hinweise zur Problemlösung. Im Anschluss an das Bearbeiten der Testaufgaben wurde die Versuchsperson gebeten, den AttrakDiff-Fragebogen sowie den Post-Test-Fragebogen auszufüllen. Durch den AttrakDiff-Fragebogen [Hassenzahl u. a., 2003] wurde der Eindruck bezüglich Attraktivität, hedonischer und pragmatischer Qualität, welche die HyperGridXGL hervorgerufen hatte, quantitativ festgehalten. Mit Hilfe des Post-Test-Fragebogens wurde die Meinung der Versuchsperson in Bezug auf Stärken und Schwächen schriftlich erfasst. Außerdem wurden sonstige Anmerkungen und das Gesamturteil erfragt.

Im Verlauf der Bearbeitung des Post-Test-Fragebogens entstand oftmals schon eine Konversation zwischen Versuchsperson und Versuchsleiter, wobei die Überleitung zur vierten Phase, dem semi-strukturierten Interview, zumeist stufenlos erfolgte. Im Interview wurden die während des Tests beobachteten Probleme thematisiert. So wurde in Erfahrung gebracht, an welchen Stellen der jeweilige Teilnehmer möglicherweise Probleme hatte. Zum Schluss wurden die Versuchspersonen noch zu ihren Verhaltensweisen in Bezug auf den Umgang mit Hierarchien befragt.

Der Leitfaden zum Interview kann dem Anhang B entnommen werden.

Insgesamt betrug die Dauer eines Tests ca. 45 Minuten. Jeder Teilnehmer erhielt eine Aufwandsentschädigung von 5 Euro.

5.3 Ergebnisse

Im Folgenden sollen zunächst die während der Evaluation aufgetretenen Usability-Schwächen und -Probleme aufgegriffen und anhand von Screenshots illustriert werden (Kapitel 5.3.1). Anschließend werden konkrete Redesign-Vorschläge vorgestellt, die aus der Diskussion mit einzelnen Teilnehmern resultieren (Kapitel 5.3.2). Danach werden die im Verlauf des Tests bewährten bzw. positiv aufgefallenen Aspekte zusammengestellt (Kapitel 5.3.3). Zum Schluss wird das Ergebnis des AttrakDiff-Fragebogens vorgestellt (Kapitel 5.3.4) und interpretiert, wobei an dieser Stelle nochmals gesagt sei, dass angesichts der geringen Teilnehmerzahl kein Anspruch auf Repräsentativität erhoben werden kann.

5.3.1 Usability-Schwächen

In erster Linie werden Usability-Probleme mit der HyperGridXGL angesprochen, jedoch sind im Verlauf der Evaluation und in der darauf folgenden Analyse einige Usability-Probleme innerhalb des Gesamtkonzepts der HyperGrid aufgedeckt worden. Feedback zur HyperGrid sowie einige konkrete Redesign-Vorschläge wurden dem MedioVis-Projekt-Team in einer Präsentation vorgestellt und anschließend diskutiert. Daher werden weder die aufgedeckten Probleme noch die Redesign-Vorschläge zur HyperGrid im weiteren Verlauf dieser Arbeit erörtert.

Die Testpersonen haben drei Usability-Schwächen identifiziert:

- **Inkonsistenz bei der Exemplar-Ebene**
- **Mehrfachverortung von Dokumenten**
- **Buttonproblematik innerhalb der HyperGridXGL**

Darüber hinaus weisen die meisten innovativen Konzepte das „Usability-Problem“ **Kennenlernen** auf. Im vorliegenden Fall ist eine Versuchsperson sehr vertraut im

Umgang mit herkömmlichen Systemen wie dem Windows Explorer⁵⁶ oder dem Midnight Commander⁵⁷, die den TreeBrowser (siehe Kapitel 2.2.3) als Visualisierung verwenden. Beim Explorieren von Hierarchien hatte die Versuchsperson daher die Erwartung, dass die Darstellung einer „alt-bekanntem“ Visualisierung ähneln würde und fand es deswegen sehr schwer, sich auf eine neue, noch nie zuvor gesehene Visualisierung einzustellen. Nach einer *Eingewöhnungsphase*, in der die HyperGridXGL ausführlich genutzt werden konnte, war der Umgang mit dem neuen System relativ unproblematisch. Die Ansicht, dass man beim *First* bzw. *Initial Contact*, also beim ersten Betrachten (siehe Abbildung 36), nicht sofort weiß, wie die HyperGridXGL zu bedienen ist, wurde auch von drei weiteren Versuchspersonen geteilt. Eine dieser Versuchspersonen meinte zudem, dass der Prozess des Kennenlernens bei einer neuen, ihr unbekanntem Software völlig normal sei, da man ja „nicht alles von Beginn an verstehen könne“. Es muss also gefolgert werden, dass sowohl die HyperGridXGL als auch die HyperGrid beim ersten Kontakt vor allem aufgrund der Interaktionselemente und des Informationszuwachses in einer Zelle einen sehr komplexen Eindruck machen kann.

Inkonsistenz bei der Exemplar-Ebene Wie schon in einer früheren informellen Benutzerbefragung [Winkler, 2006] festgestellt, gibt es eine Inkonsistenz bei der Darstellung der Exemplar-Ebene. Falls mehrere Exemplare eines Titels in der Mediothek vorhanden sind, werden die Exemplare wie weitere untergeordnete Titel auf der unteren ContextView visualisiert (siehe Abbildung 51). Im Fall, dass von einem Titel einer Hierarchie nur ein Exemplar vorhanden ist, bleibt die untere ContextView wie in Abbildung 52 leer.



Abbildung 51: Ein Titel besitzt mehrere Exemplare. Untere ContextView beinhaltet die Exemplare, zusätzlich wird noch der Minus-MoB angezeigt

⁵⁶Der Windows Explorer ist ein Datei-Manager, welcher von Microsoft für deren Betriebssystem Windows entwickelt wurde.

⁵⁷Der Midnight Commander ist ein Datei-Manager, welcher hauptsächlich auf Linux-Betriebssystemen zum Einsatz kommt und ausschließlich im Text-Mode fungiert <http://www.ibiblio.org/mc/>



Abbildung 52: Der Titel entspricht gleichzeitig einem Kind-Knoten und einem Exemplar. Der Ausleihstatus weist direkt auf diesen Titel, die untere ContextView bleibt leer

Theoretisch sollte in diesem Fall ebenfalls ein Exemplar auf der unteren ContextView angezeigt werden, da ja schließlich ein Exemplar dieses Titels vorhanden ist. Allerdings wären dann auch alle Titel, die bisher keiner Hierarchie angehören und demzufolge nur einfach in der Mediothek vorhanden sind, hierarchischer Natur. So wäre jeder einzeln vorhandene Titel eine Art Container, der eine Hierarchiestufe über dem Exemplar steht. Dies würde zu einer extremen Komplexitätserhöhung innerhalb der HyperGridXGL führen, da plötzlich alle Titel Hierarchien angehören, ganz gleich, ob diese Hierarchie nur aus einem Titel und dessen Exemplar oder tatsächlich aus einer Hierarchie mit z.B. zwanzig Elementen besteht. Bei der Hierarchie von „A grammar of spoken English“ hätte der Titel „Mackin, Ronald: A grammar of spoken English“, der nur einfach in der Mediothek vorhanden ist, eine weitere Hierarchie-Ebene, wie in Abbildung 53 auf der rechten Seite zu sehen ist. Da dieser Ansatz zu einer extremen Komplexitätserhöhung führen würde, werden Hierarchien wie auf der linken Seite in Abbildung 53 dargestellt, was jedoch die Inkonsistenz beinhaltet, dass nicht alle Exemplare tatsächlich auf der Exemplar-Ebene angezeigt werden.

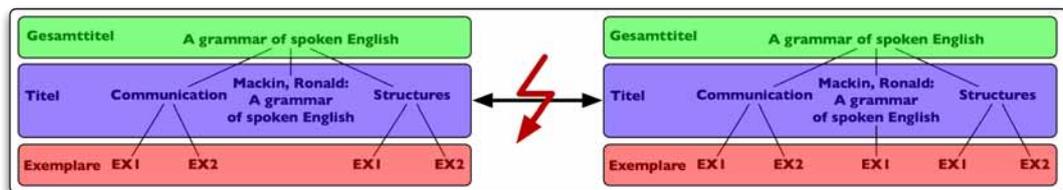


Abbildung 53: Die Hierarchie von „A grammar of spoken English“ links in der Variante, in der sie bisher visualisiert wird und rechts in der „konsistenten“ Variante

Andererseits scheint das oben dargestellte Problem, dass die Exemplar-Ebene nicht gefunden wird, in erster Linie nicht nur in der Inkonsistenz zu liegen, sondern viel mehr darin, dass bei der Frage nach der Anzahl der Exemplare eines speziellen Titels, sieben der acht Teilnehmer zuerst das Attribut Ausleihstatus gefunden haben, in dem der tatsächliche Ausleihstatus, z.B. „verfügbar“ oder „ausgeliehen“ oder „mehrere Exemplare“ bzw. „mehrteiliges Werk“, vermerkt ist. Die Information dieses Attributs war für die meisten klar als Antwort auf die Frage nach der Anzahl an Exemplaren zu erkennen. Daher sind die Versuchspersonen nicht mehr davon ausgegangen, dass die eigentlichen Exemplare auf der nächsten Hierarchie-Ebene

zu finden sind. Dort sind die Exemplare als eigene Dokumente, mit dem Kürzel „EX“ und der Nummer des jeweiligen Exemplars vor dem Titel, dargestellt, wie in Abbildung 54 zu sehen ist. Nachdem der Versuchsleiter einen Tipp gegeben hatte, dass es eine weitere Hierarchie-Ebene gibt, rotierten die Versuchspersonen auf die nächst niedrigere Hierarchie-Ebene. Als dann alle Dokumente der untersten Ebene auf der DetailedView erschienen, war für alle Versuchspersonen erkennbar, wie viele Exemplare tatsächlich vorhanden sind. Jedoch fanden sie diese Art der Darstellung unlogisch, insbesondere nachdem sie herausgefunden hatten, dass es die oben erwähnte Inkonsistenz gibt. Ein Teilnehmer merkte außerdem an, dass er nicht sofort von „EX1-Communication“ auf das 1. Exemplar des Titels Communication zu schließen in der Lage war, was für ihn eine weitere Schwierigkeit bedeutete.

▾	A fistful of Dollars	Sergio Leone, Clint Eastwood, Marianne Koch, G...	Video	2001
▾	Communication	-	Buch	1994
▾	EX1 - Communication	-	-	0
▾	EX2 - Communication	-	-	0
▾	Bird	Clint Eastwood, Forest Whitaker, Diane Venora, ...	Video	1988

Abbildung 54: Exemplare des Titels „Communication“ sind hier auf der DetailedView visualisiert.

Mehrfachverortung von Dokumenten Mehrfachverortung von einzelnen Titeln bedeutet, dass ein und derselbe Titel in der HyperGridXGL mehrfach visualisiert wird. Dieser Fall tritt sehr selten und nur in ganz bestimmten Konstellationen auf. Dazu müssen in der Suchtreffermenge in der HyperGridXGL mehrere Titel einer Hierarchie erscheinen. Ein Beispiel hierfür, welches auch als Aufgabe im Usertest vorkam, ist die Suche nach „Band of Brothers“, bei der es zwei Suchtreffer gibt (siehe Abbildung 55).

Titel	Beschreibung	Exemplar	Jahr
Band of brothers	Kirk Acevedo, Eion Bailey, Michael Cudlitz, Stephen E. ...	DVD	0
Specials: Originaldokumentation "We ...	Mark Cowen, Ron Livingston	DVD	2002

Abbildung 55: Suche nach „Band of Brothers“

Die Teilnehmer sollten herausfinden, wie viele Teile der Reihe „Band of Brothers“ im Bestand der Mediothek vorzufinden sind. Dazu zoomten alle in die Hierarchie des oberen Titels (Band of Brothers). Alle Versuchspersonen rotierten anschließend auf die darunterliegende Hierarchie-Ebene jenes Titels und erfassten direkt, dass die Mediothek aus dieser Reihe fünf Teile und ein „Special“ vorzuweisen hat (vergleiche

Abbildung 56).

Band of Brothers		Kirk Acevedo, Eion Bailey, Michael Cudlitz, Stephen E. ...	DVD	2002
[-]	Specials : Originaldokumentation "We ...	Mark Cowen, Ron Livingston	DVD	2002
[-]	Teil 1. Currahee	Phil Alden Robinson, Richard Loncraine	DVD	2002
[-]	Teil 3. Carentan - Brennpunkt Norma...	Mikael Salomon, David Nutter	DVD	2002
[-]	Teil 5. Keuzungen	Tom Hanks, David Leland	DVD	2002
[-]	Teil 7. Durchbruch	David Frankel, Tony To	DVD	2002
[-]	Teil 9. Warum wir kämpfen	David Frankel, Mikael Salomon	DVD	2002
[-]	Specials : Originaldokumentation "We ...	Mark Cowen, Ron Livingston	DVD	2002

Abbildung 56: Nach dem Aufzoomen der Hierarchie des ersten Suchtreffers

In der nächsten Aufgabe wurde nach der Anzahl von Exemplaren jedes Teils gefragt. In vorangegangenen Aufgaben wurde das Problem des Herausfindens der Anzahl an Exemplaren eines Titels schon bearbeitet. Daher war bei dieser Testaufgabe ein ganz deutlicher Lerneffekt zu entdecken und fünf der acht Versuchspersonen konnten direkt die Anzahl der Exemplare einzelner Titel nennen. Dazu zoomten sie in die Zeile der jeweiligen Titel. Durch das Nichtvorhandensein einer weiteren Hierarchie-Ebene und anhand des Ausleihstatus konnte alle fünf Teilnehmer die Anzahl der Exemplare nennen. Drei Teilnehmer versuchten über einen anderen Weg die Anzahl der Exemplare herauszufinden. Sie zoomten zusätzlich die Hierarchie des zweiten Suchtreffers auf. Da der zweite Suchtreffer mit vollem Titel „Specials : Originaldokumentation We stand alone together: the men of easy company [DVD] Tom Hanks and Steven Spielberg present **Band of Brothers** = Wir waren wie Brüder“ heißt, wurde dieser bei der Suche nach „band of brothers“ ebenfalls getroffen. Hier handelt es sich um einen untergeordneten Titel des Gesamttitels Band of Brothers. Somit gehören sowohl der erste Suchtreffer „Band of Brothers“ als Wurzel-Knoten als auch der zweite Suchtreffer „Specials: Originaldokumentation...“ als Kind-Knoten zur identischen Hierarchie. Die drei Versuchspersonen bekamen nun die Hierarchien, die, wie man in Abbildung 57 sehen kann, aus den exakt gleichen Titeln bestehen, dargestellt.

Band of Brothers		Kirk Acevedo, Eion Bailey, Michael Cudlitz, Stephen E. ...	DVD	2002
[-]	Specials : Originaldokumentation "We ...	Mark Cowen, Ron Livingston	DVD	2002
[-]	Teil 1. Currahee	Phil Alden Robinson, Richard Loncraine	DVD	2002
[-]	Teil 3. Carentan - Brennpunkt Norma...	Mikael Salomon, David Nutter	DVD	2002
[-]	Teil 5. Keuzungen	Tom Hanks, David Leland	DVD	2002
[-]	Teil 7. Durchbruch	David Frankel, Tony To	DVD	2002
[-]	Teil 9. Warum wir kämpfen	David Frankel, Mikael Salomon	DVD	2002

Band of Brothers		Kirk Acevedo, Eion Bailey, Michael Cudlitz, Stephen E. ...	DVD	2002
[-]	Specials : Originaldokumentation "We ...	Mark Cowen, Ron Livingston	DVD	2002
[-]	Teil 1. Currahee	Phil Alden Robinson, Richard Loncraine	DVD	2002
[-]	Teil 3. Carentan - Brennpunkt Norma...	Mikael Salomon, David Nutter	DVD	2002
[-]	Teil 5. Keuzungen	Tom Hanks, David Leland	DVD	2002
[-]	Teil 7. Durchbruch	David Frankel, Tony To	DVD	2002
[-]	Teil 9. Warum wir kämpfen	David Frankel, Mikael Salomon	DVD	2002

Abbildung 57: Nach dem Aufzoomen des ersten und zweiten Suchtreffers

Zwei der drei oben erwähnten Versuchspersonen war nicht klar, warum zwei gleiche Hierarchien in einem Suchergebnis in Erscheinung treten. Sie konnten nicht nachvollziehen, dass beide Titel der Suchtreffermenge der gleichen Hierarchie angehören und

dass man folglich beim Hineinzoomen in die Hierarchie dieser Titel auch die gleiche Hierarchie doppelt visualisiert bekommt. Daher dachten sie, jeder Teil von Band of Brothers sei zweimal vorhanden. Wichtig dabei ist, dass eine dieser zwei Versuchspersonen, der sogenannte Pre-Tester war. Beim Pre-Tester war die Reihenfolge der Testaufgaben anders. Die Aufgabe, bei der das beschriebene Problem auftrat, war bei ihm gleich zu Beginn zu beantworten. So konnte man bei ihm keinen Lern-Effekt erwarten, wodurch sein Ergebnis mit Vorsicht zu betrachten ist.

Lediglich eine Versuchsperson erkannte, dass die gleiche Hierarchie zweimal dargestellt wurde. Dem Teilnehmer war klar, dass beide Suchtreffer der gleichen Hierarchie angehören und beim Zoomen in die Hierarchien beider Titel die selbe Hierarchie visualisiert wurde.

Demzufolge kann die Mehrfachverortung einzelner Dokumente, falls sie dem Benutzer überhaupt auffällt, zur Verwirrung führen. Anzumerken ist zusätzlich noch, dass diese Verwirrung erst bei der Frage nach der Anzahl der Exemplare einzelner Titel auftrat und daher ein Folgeproblem der oben erwähnten Inkonsistenz ist.

Buttonproblematik innerhalb der HyperGridXGL Die im Umgang mit Buttons entstehende Problematik innerhalb der HyperGridXGL ist schon aus der HyperGrid bekannt. In der HyperGrid gibt es nur einen Button, den Flap-In/-Out Button am linken Rand, mit dessen Hilfe eine Zeile komplett gezoomt werden kann. In der Untersuchung wurde dieser Button in der Regel erst dann benutzt, wenn auch andere Buttons, wie der **TriggerButton**, verwendet wurden. Oft wurde der TriggerButton in der explorativen Phase 2 erst nach einem Hinweis des Versuchsleiters gefunden und benutzt. Zwei Versuchsperson fanden heraus, dass für sie ein Symbol wie das des TriggerButtons in der HyperGridXGL mehr Information darstellt, jedoch folgerten sie daraus Unterschiedliches. Für die eine Versuchsperson war es lediglich ein Symbol, welches durch seine Symbolik weitere Information geben soll, wobei sie Schwierigkeiten bei der Interpretation der Symbolik hatte. Für die andere Versuchsperson stand das Symbol für einen Button, durch dessen Funktionalität man zu mehr Information gelangt. Dies lässt darauf schließen, dass zum einen teilweise die Affordance für einen Button fehlt und zum anderen die Symbolik des Buttons schwer zu deuten ist.

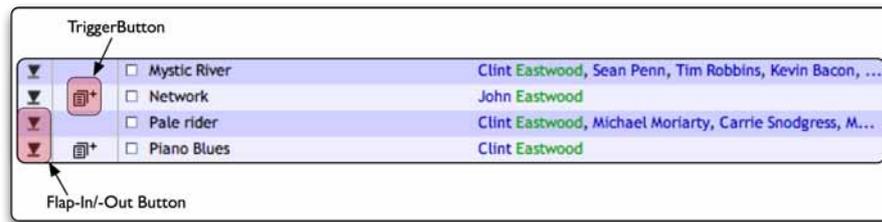


Abbildung 58: Flap-In/-Out Button und TriggerButton

Affordance [Norman, 1999] ist die wünschenswerte Eigenschaft eines User Interfaces, die den Benutzer intuitiv die richtige Funktionalität und den richtigen Gebrauch des jeweils Dargestellten implizieren lässt. Übertragen auf einen Button würde dies bedeuten, dass der Button den Benutzer durch sein Erscheinungsbild zum Drücken bzw. Klicken animieren soll. Grundsätzlich wird durch das Erscheinungsbild eines Buttons beim Benutzer eine Erwartung erzeugt, die durch frühere Erfahrung in der Nutzung von Buttons entstanden ist. Da ein Button in der Regel auf einer zweidimensionalen Fläche mit einem 3D-Effekt ein wenig angehoben dargestellt wird, er in der HyperGridXGL hingegen ausschließlich als 2D-Symbol eingearbeitet ist, könnte dies zumindestens einer der Gründe für die Buttonproblematik sein.

Die schon erwähnte Symbolik des TriggerButtons, insbesondere das „Plus“, lässt nach Meinung aller Versuchspersonen auf mehr Information schließen, dennoch konnte der Zusammenhang zwischen einer Hierarchie und dem verwendeten Symbol nicht sofort erkannt werden. Nachdem schließlich auf den TriggerButton geklickt wurde und die Hierarchie ausgefahren wurde, war allen Versuchspersonen in kürzester Zeit die Funktionalität des Buttons klar. Eine Versuchsperson erklärte, dass der Unterschied zwischen Plus- und Minus-TriggerButton erst beim zweiten Hinschauen zu erkennen war. Ferner konnte man feststellen, dass der Minus-TriggerButton erst entdeckt wurde, als die Versuchsperson nach einer Funktion zum Schließen des hierarchischen Zooms gesucht hat. Dies könnte daran liegen, dass der Minus-TriggerButton durch die oben und unten angrenzenden MoBs, wie in Abbildung 59 zu sehen ist, „visuell überladen“ wird und daher erst nach einiger Zeit entdeckt wird.



Abbildung 59: Plus- und Minus-TriggerButton, sowie MoBs

Der **Mouse-over-Button** (MoB) wurde überwiegend zum Rotieren zwischen den Hierarchie-Ebenen verwendet, infolgedessen hat der MoB eher die Affordance eines Buttons. Trotzdem wurde von einer Versuchsperson der Wunsch geäußert, man sollte den MoB noch etwas plastischer machen, indem man zum Beispiel die Form des MoBs an die Form der verzerrten Flächen der ContextView anpasst. So würde man einen 3D-Effekt erzeugen und könnte zusätzlich deutlicher die Zusammengehörigkeit zwischen jeweiligem Button und ContextView zeigen.

Weitere Usability-Schwächen und -Probleme Bei den bisherigen Konzepten zur Darstellung von Hierarchien wird die komplette Hierarchie auf einmal visualisiert. Damit ist auch der **Wurzel-Knoten** einer Hierarchie dauerhaft sichtbar. Im Konzept der HyperGridXGL wird der Wurzel-Knoten jedoch nicht immer dargestellt, was zunächst von keiner Versuchsperson kritisiert wurde. Bei genaueren Nachfragen im Interview stellten vier Teilnehmer fest, dass es in manchen Situationen wichtig sein könnte, den Wurzel-Knoten zu sehen. Jedoch stellten sie direkt danach in Frage, ob es tatsächlich wichtig oder eigentlich eher interessant wäre. „Außerdem könne man auch hinrotieren“, wurde von zwei Versuchspersonen angemerkt. Interessant ist auch der alltägliche Umgang der Versuchspersonen mit Hierarchien in Form des Navigierens im Dateisystem eines Betriebssystems. Sieben der acht Versuchspersonen verwenden zu Hause den „Arbeitsplatz“ von Windows, um im Dateisystem zu navigieren. Dies ist insofern interessant, da beim „Arbeitsplatz“ lediglich der Pfad eines Knotens bzw. eines Ordners oder einer Datei textuell angezeigt wird, jedoch nicht die komplette Hierarchie visualisiert wird.

Die **Rotation** führte bei einigen Versuchspersonen beim Initial Contact zu Irritationen, da dies eine nie zuvor gesehene Animation war. Allerdings sagten auch fünf der sieben Teilnehmer, dass die Animation direkt nach der ersten Rotation verstanden wird. Die restlichen zwei Teilnehmer hingegen schienen teilweise überfordert zu sein. Bei jeder Animation gaben sie Laute der Verwunderung, wie „oh“, wider. Speziell für eine dieser zwei Versuchspersonen stellte die Animation zu viel Bewegung dar. Diese Versuchsperson, die zum Beispiel auch den MidnightCommander verwendet, hat, eigenen Angaben zufolge, tendenziell Schwierigkeiten, sich mit neuen Visualisierungen anzufreunden.

Der **textuelle Hinweis** wurde von nur einer Versuchsperson während des Tests gesehen. Damit lässt sich die Erkenntnis ableiten, dass der textuelle Hinweis zu klein bzw. zu undeutlich in Erscheinung tritt. Auf Befragung empfanden alle den textuellen Hinweis als hilfreich. Fraglich ist jedoch, ob dieser Hinweis nur in Bezug auf die Testaufgaben hilfreich ist oder auch im alltäglichen Gebrauch eine Hilfe wäre.

Der **Stack** hingegen wurde von keiner Versuchsperson entdeckt. Erst nachdem der Versuchsleiter im Interview darauf hingewiesen hat, wurde der Stack wahrgenom-

men. Auch die Funktionalität des Stack war nur drei Teilnehmern sofort klar, den restlichen erst nach einer entsprechenden Erklärung durch den Versuchsleiter. Teilweise wurde der Stack sogar für einen Grafikfehler gehalten, was mit Sicherheit daran liegt, dass überwiegend in die Mitte des Bildschirms geschaut wird, man den Stack aber erst so richtig an den Rändern, wie in Abbildung 60, erkennen kann.



Abbildung 60: Stack und textueller Hinweis

5.3.2 Redesign-Vorschläge

In der Analyse des Testmaterials haben sich im wesentlichen drei Haupt-Probleme, die in Kapitel 5.3.1 beschrieben wurden, herausgestellt. Da diese Probleme (Exemplar-Ebene, Button, Mehrfachverortung) als kritisch einzustufen sind, wurden dafür Redesign-Maßnahmen erarbeitet. Dabei wurden Redesign-Vorschläge, die in der Diskussion mit den Versuchspersonen während der Interviews aufgetaucht sind, verwendet und weiterentwickelt.

Exemplar-Ebene Aus zwei Gründen wurde ein Exemplar als ein eigenes Dokument in einer Zeile dargestellt. Exemplare eines Titels können an verschiedenen Standorten, zum Beispiel in der Mediothek und im Semesterapparat eines Dozenten, stehen. Somit muss die Möglichkeit bestehen, dass bei jedem Exemplar ein anderer Standortplan dargestellt werden kann. Außerdem können Exemplare einen unterschiedlichen Ausleihstatus haben: ein Exemplar kann momentan verliehen sein, während das andere Exemplar noch verfügbar ist. Folglich war es einleuchtend und logisch Exemplare auf einer weiteren Hierarchie-Ebene als eigene Dokumente auszuweisen. Die Ergebnisse des Usertests zeigen auf, dass die Versuchspersonen Schwierigkeiten mit der Logik dieser Darstellungsweise haben.

Aus datentechnischer Sicht beschreibt die Exemplar-Ebene zwar eine weitere Hierarchie-Ebene, jedoch sind für den Benutzer lediglich der Ausleihstatus und der Standort eines Exemplars relevant. Um den Benutzer nicht mit einer weiteren Ebene kognitiv zu belasten, wurde die Exemplar-Ebene im Zuge der Redesign-Maßnahmen beseitigt. Dadurch werden Exemplare nicht mehr auf einer eigenen Hierarchie-Ebene dargestellt, womit zusätzlich die Inkonsistenz zwischen Titeln, die ausschließlich ein Exemplar haben, und Titeln, die mehrere Exemplare besitzen, unterbunden wird.

Informationen wie Standort und Ausleihstatus müssen aber in spezifischer Art und Weise in der HyperGridXGL präsentiert sein. Denn eine weitere Erkenntnis aus dem

User-Test war, dass der Benutzer es als am interessantesten empfindet, zu wissen wie viele Exemplare eines Titels vorhanden sind, wie viele dieser Exemplare momentan verfügbar sind und wo diese Exemplare in der Mediothek stehen. Die bisherige Information „mehrere Exemplare“, die dem Benutzer anhand des Ausleihstatus übermittelt wurde, ist weniger aussagekräftig als die Angabe der vorhanden und zurzeit verfügbare Anzahl an Exemplaren. Daher wurde das Attribut Ausleihstatus der Erwartung des Benutzers angepasst. Im Ausleihstatus wird jetzt die Anzahl an Exemplaren aus der Gesamtanzahl dargestellt.

Um die Anzahl an Exemplaren zu erfahren, zoomen die Versuchspersonen im Test zuerst in die Tiefe des Informationsraums eines Titels. Erst anschließend, wenn überhaupt, in einem zweiten Schritt rotieren sie auf die nächste Hierarchie-Ebene. Nun kann der Benutzer die vorhandene Anzahl an Exemplaren direkt aus dem Ausleihstatus ablesen. So ist kein weiteres Rotieren in die nächste Hierarchie-Ebene nötig. Will man den genauen Ausleihstatus zu den einzelnen Exemplaren erfahren, muss man auf „Ausleihstatus“ klicken. Dadurch wird ein Browser-Fenster geöffnet. In diesem Browser-Fenster werden dann der Standort sowie Titel und Ausleihstatus eines Exemplars in einer 3-spaltigen Tabelle, wie in Abbildung 61 illustriert, visuell aufbereitet. In dieser Tabelle wird jedes Exemplar in einer Zeile dargestellt. Falls in der HyperGridXGL auf den Ausleihstatus eines Dokuments geklickt wird, welches den Wurzel-Knoten einer Hierarchie verkörpert, erscheinen in der sich öffnenden Tabelle alle Exemplare aller Untertitel. In der linken Spalte der Tabelle wird der Standortplan eines Exemplars gezeigt. In der mittleren Spalte hingegen werden der Titel und die Signatur des Exemplars visualisiert. In der rechten Spalte wird der Ausleihstatus des jeweiligen Exemplars und ein „Ampellicht“, das entweder „rot“ für ausgeliehen oder „grün“ für verfügbar leuchtet, angezeigt. Der Ausleihstatus im Browser-Fenster dient auch als direkter Link zum jeweiligen Exemplar des Koala-Systems der Bibliothek. Die Tabelle ist absteigend nach Verfügbarkeit und Titel der Exemplare sortiert. Exemplare, die verfügbar sind, stehen daher oben in der Tabelle. So kann man auf einen Blick sehen, welche Exemplare noch vorhanden sind.

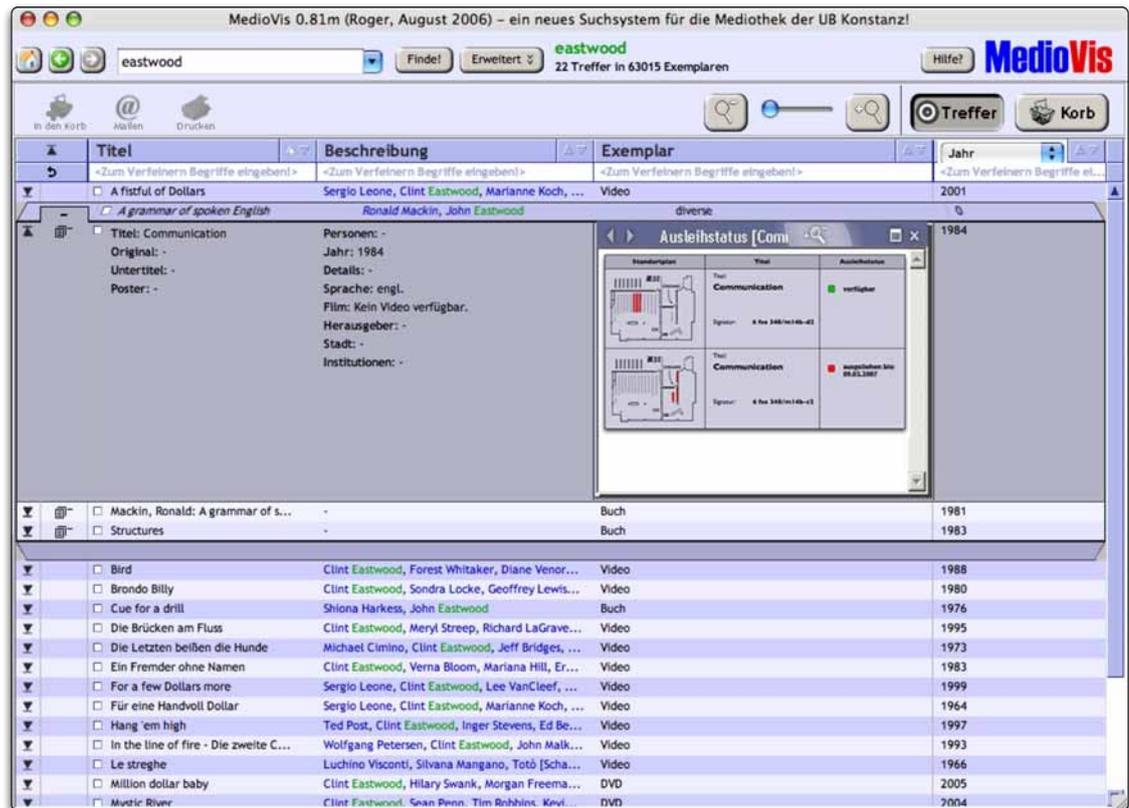


Abbildung 61: Die Ausleihstatus-Tabelle in einem Browser-Fenster

Wie schon erwähnt ist der Ausleihstatus eines Titels in der HyperGridXGL nicht mehr der direkte Link zum Koala-System, sondern ein Link zur soeben beschriebenen Ausleihstatus-Tabelle. Den direkten Link zum Koala-System aus der HyperGridXGL wurde beseitigt. Ein großer Mehrwert von MedioVis ist, dass man sich nur beim Vormerken und Ausleihen eines Titels mit dem Koala-System „auseinandersetzen muss“. Klickte man seither auf den Ausleihstatus eines Gesamttitels, wurde auch die Seite des Gesamttitels im Koala-System aufgerufen. Um schließlich auf die Seite eines tatsächlich vorhandenen Exemplars zu gelangen, musste man sich mit der schwer verständlichen Hierarchien-Navigation des Koala-Systems auseinandersetzen. So ging ein Mehrwert von MedioVis verloren. Daher wird nun die Ausleihstatus-Tabelle zwischengeschaltet. Durch einen Klick auf den Ausleihstatus in der Tabelle gelangt man sofort auf die Seite des Exemplars ohne Interaktion mit den Navigations-elementen des Koala-Systems.

Bisher wurde jedes Exemplar in einer eigenen Zeile visualisiert. So konnte der Standortplan eines Exemplars einfach in die Tiefe des Informationsraums einer Zeile eingebunden werden. Da nun nicht mehr einzelne Exemplare in einer eigenen Zeile dargestellt werden, muss der Standortplan auf eine andere Weise mit den restlichen Informationen verknüpft werden. Seither wurde bei einem Titel, der mehrere Exemplare besaß, nur ein Standortplan bereit gehalten, ganz gleich, ob tatsächlich alle

Unterdokumente auch an diesem Ort standen.

In der Redesign Phase sind zwei verschiedene Ideen aufgekommen, um die Standort-Information zu einem Exemplar elegant zu lösen. Die erste Idee basiert auf der soeben vorgestellten Tabelle, die in einem Browser-Fenster dargestellt wird. Hierbei wird der bisherige Standortplan durch eine Aneinanderreihung von kleinen Standortplänen ersetzt. So werden auf dem gleichen Platz des bisherigen Standortplans alle Standortpläne der Unterdokumente in einer visuell der Treemap ähnlichen Graphik angezeigt (siehe Abbildung 62).

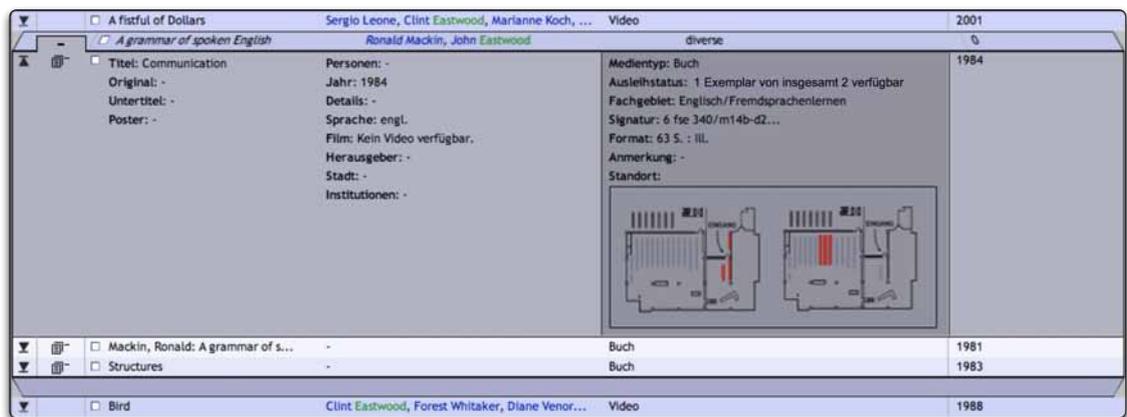


Abbildung 62: Standortplan-Treemap und Ausleihstatus

Beim Überfahren dieser Graphik wird diese abgedunkelt und ein transparenter visual clue „click to enlarge“ hinzugefügt (vergleiche Abbildung 63). Dazuhin verändert sich der Mauszeiger von einer Lupe zu einer Hand, um dem Benutzer zusätzlich zu signalisieren, dass man hier klicken kann. Bei einem Klick auf die Graphik wird die schon vorgestellte Tabelle in einem Browser-Fenster geöffnet.

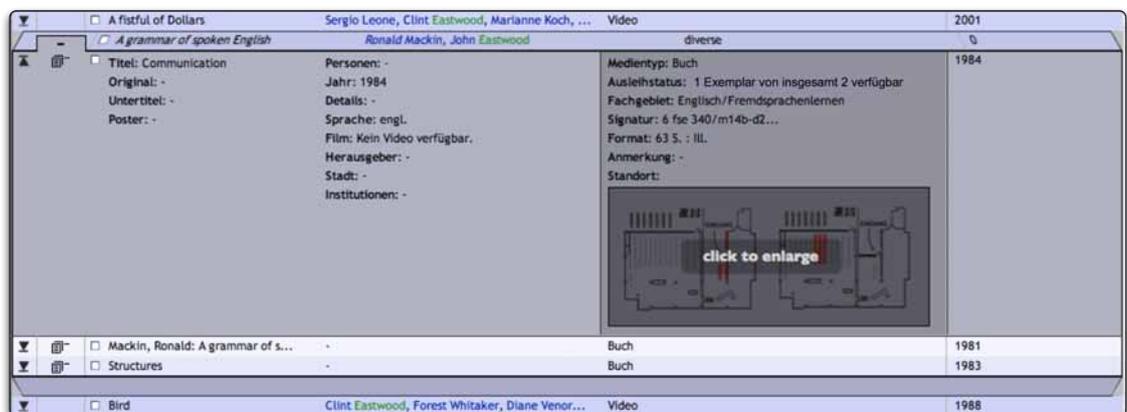


Abbildung 63: Standortplan-Treemap mit Mouse-Over-Effekt und Ausleihstatus

Die zweite Idee verfolgt das Konzept des ZUI. Dabei wird der bisherige Standortplan eines Titels durch einen Standort-Scatterplot ersetzt. Im Scatterplot, einem Punktdiagramm, das als zusätzliche Visualisierung zur HyperGrid in MedioVis eingebunden werden kann, werden die gefundenen Titel entsprechend ihrer Attribute in einem kartesischen Koordinatensystem positioniert. Im Standort-Scatterplot werden die Exemplare des Titels entsprechend ihres Standorts in Form von farbigen Punkten auf einer Umrisskarte der Mediothek in einem Koordinatensystem dargestellt (siehe Abbildung 64). Rote Punkte entsprechen Exemplaren, die momentan ausgeliehen sind, grüne Punkte entsprechen Exemplaren, die gegenwärtig verfügbar sind. Nun kann man wie im Scatterplot in das Punktdiagramm zoomen. Beim Zoomen dient das auf der Umrisskarte liegende Gitternetz des Koordinatensystems durch dementsprechendes Vergrößern bzw. Verkleinern der Abstände der Gitter-Linien zueinander, als Orientierung. Je näher man zu einem Punkt zoomt, desto mehr Information erhält man. So werden aus den Punkten Icons, die symbolisch die Art des Exemplars wiedergeben. Fährt man mit der Maus über ein Icon, bekommt man den Titel, die Signatur, sowie den Ausleihstatus als Tooltip angezeigt. Klickt man auf ein Icon, gelangt man direkt auf die Seite des Exemplars im Koala-System.

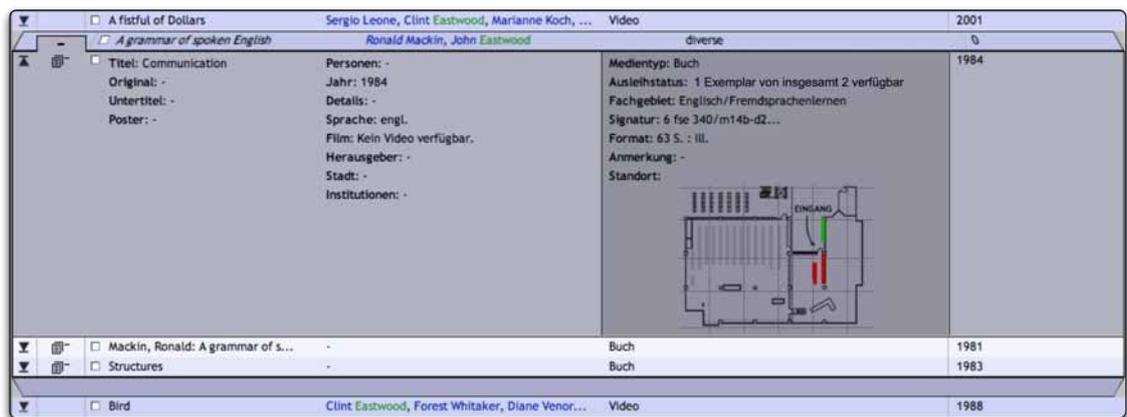


Abbildung 64: Standort-Scatterplot

Abschließend muss man feststellen, dass sich die zwei Ideen drastisch unterscheiden. Die erste Idee entspricht wahrscheinlich eher der Erwartungskonformität des Benutzers. Hier wurde das Konzept der Browserkomponenten aufgegriffen, bei dem durch Klick auf ein Objekt in einer Zelle ein übergelagertes Browser-Fenster mit weiterer Information geöffnet wird. Da dies schon durch die Nutzung der restlichen Funktionen der HyperGridXGL bekannt ist, könnte diese Idee zumindestens zu Beginn als einfacher zu bedienen erscheinen. Allerdings wird man vermutlich die zweite Idee als die elegantere Lösung sehen, sobald man damit vertraut ist. Diese Idee greift das Prinzip des ZUI auf und bietet daher im Gesamtkonzept der HyperGridXGL eine sehr passende Lösung.

Button Der Test hat gezeigt, dass einige Versuchspersonen Probleme im Umgang mit den Buttons hatten. Diese resultieren in erster Linie aus der fehlenden Affordance der Buttons. Aus den Interviews wurden erste Ideen generiert, wie man die Schwächen beheben könnte. Die Vorschläge bezogen sich dabei meistens auf die Anmutung der Buttons. Im Redesign wurde daher Wert auf eine plastischere Gestaltung der Buttons gelegt.

So wurden die Symbole auf den Mouse-over-Buttons mit abgerundeten Rechtecken umrahmt. Der Rahmen bewirkt eine Abgrenzung gegenüber dem TriggerButton, wobei jetzt jeder Button deutlich zu erkennen ist. Klickt man zum Beispiel auf den Zoom-Reset-Button der HyperGridXGL, scheint dieser „nach hinten gedrückt“ zu werden. Durch diesen Effekt erhält der Benutzer ein Feedback, das die Affordance eines Buttons deutlich steigert. Daher wird nun ein ähnlicher Effekt beim Klicken auf die MoBs erzeugt.

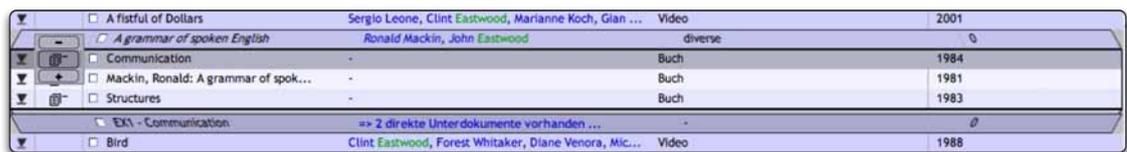


Abbildung 65: Umrahmte TriggerButton und MoBs

Mehrfachverortung Die Mehrfachverortung von Dokumenten tritt nur in sehr seltenen Fällen auf, ist jedoch von zwei der acht Versuchspersonen als verwirrend empfunden worden. Wie schon erwähnt tritt dieses Problem nur dann auf, wenn zwei oder mehr Titel einer Hierarchie durch die Suche getroffen wurden. Ist dies der Fall, werden alle getroffenen Titel in der HyperGridXGL als eigene Dokumente in der Resultatmenge dargestellt. Nun kann es passieren, dass der hierarchische Zoom auf zwei Titel einer Hierarchie angewandt wird und damit zwei gleiche Hierarchien in der HyperGridXGL dargestellt werden. Dies kann, wie im Usertest zweimal geschehen, zu einer Verwirrung seitens des Benutzers führen. Dem Benutzer ist nicht sofort klar, dass zwei Titel zu einer Hierarchie gehören und demnach auch die gleiche Hierarchie mehrfach dargestellt werden kann.

Um dieses Problem zu entschärfen, werden jetzt nicht mehr alle Suchtreffer angezeigt. Werden der Container einer Hierarchie, das heißt der Wurzel-Knoten, sowie dessen Untertitel aufgrund der Namen der einzelnen Titel getroffen, wird nur noch der Container dargestellt. Dies ist die logische Konsequenz aus der Analyse der Testergebnisse.

In den Testaufgaben gab es eine Aufgabe, bei der nach „Band of Brothers“ gesucht werden musste. Als Suchtreffermenge erhielt man den Gesamttitel „**Band of**

brothers“ sowie einen Untertitel von „Band of Brothers“, „Specials: Originaldokumentation We stand alone together: the men of easy company [DVD] Tom Hanks and Steven Spielberg present **Band of brothers** = Wir waren wie Brüder“. Beide Titelnamen enthalten den Suchbegriff „band of brothers“, daher kann man auch vom Untertitel auf Band of Brothers schließen. In dem hier beschriebenen Fall wird daher das Unterdokument nicht in der Suchtreffermenge angezeigt. So kann man Verwirrungen, wie in Kapitel 5.3.1 beschrieben und auf den Abbildungen 56 und 57 zu erkennen ist, gezielt unterbinden.

5.3.3 Positive Aspekte

Trotz einiger offensichtlicher Probleme, insbesondere in Bezug auf die Exemplar-Ebene, gab es ein sehr positives Gesamtfeedback. Die meisten Aufgaben konnten ohne größere Probleme gelöst werden. Sowohl im Interview als auch auf dem Post-Test-Fragebogen sollten die Versuchspersonen ihr Gesamturteil zur HyperGridXGL abgeben. Im Post-Test-Fragebogen wurden sie aufgefordert, auf einer Skala von 1 bis 7, wobei 1 sehr gut und 7 schlecht ist, ihre Meinung einzutragen. Der Mittelwert liegt bei 2,5.

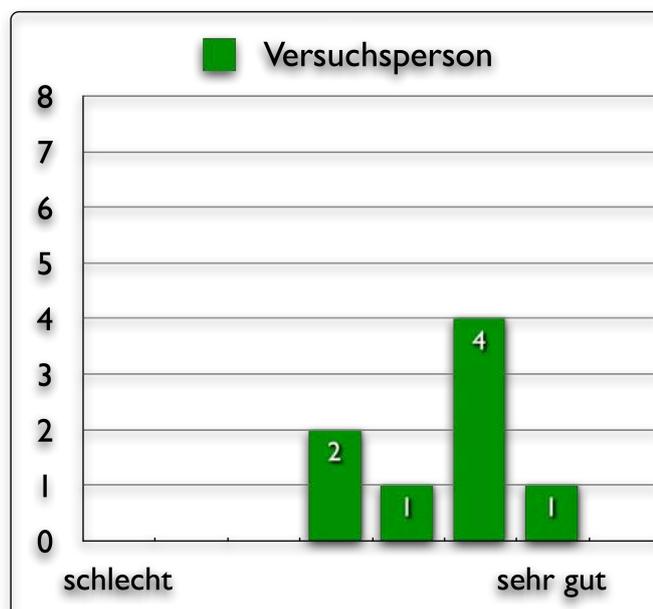


Abbildung 66: Gesamturteil: Skala des Post-Test-Fragebogens

Die schriftlichen Äußerungen in Bezug auf die Stärken der HyperGridXGL und auch die verbalen Äußerungen im Interview waren sehr positiv. Alle Versuchspersonen empfanden die visuelle Darstellung gut und schlüssig. Sieben der Acht Versuchspersonen bezeichneten die Oberfläche als sehr ansprechend. In Bezug auf die Anmutung wurde vor allem die Animation gelobt. Für vier der acht Versuchspersonen war die

Rotation intuitiv und zudem sehr gut animiert. Die Rotation sei insbesondere nicht übertrieben und würde zum grundlegenden Verständnis beitragen. So wurde das Prinzip des Hin- und Hervernavigierens in der Hierarchie schnell verstanden. Außerdem wurde geäußert, dass die Rotation Spaß am Navigieren bereite. Wie schon in der Seminararbeit erwähnt, ist „joy of use“ ein wichtiger Aspekt dieses Konzepts, dessen Vorhandensein von den Teilnehmern bestätigt wurde.

Neben dem Aspekt „joy of use“ konnte im Test festgestellt werden, dass alle Versuchspersonen insbesondere in der dritten Phase beim Bearbeiten der Testaufgaben schnell den Umgang mit dem unbekanntem System erlernten. Dies wurde auch von allen Versuchspersonen im Interview bestätigt. Sie bemerkten, dass für sie die erste Berührung mit dem System teilweise verwirrend war, dass die Anwendung aber einen Selbsterklärungsprozess initiierte und dadurch alles sehr schnell klar wurde. Auch bei der Analyse der Aufgaben war zu erkennen, dass spätestens ab der vierten Testaufgabe ein deutlicher „Lerneffekt“ eintrat. Demnach beinhaltet dieses Konzept neben dem Aspekt des „joy of use“ auch den weiteren Aspekt „easy to learn“.

Im Vorfeld des Tests tauchte des öfteren in Diskussionen die Frage auf, ob die Darstellung von Hierarchien im Rahmen von MedioVis und speziell im Problemraum der Mediothek überhaupt notwendig ist. Daher war es sehr interessant herauszufinden, wie die Versuchspersonen mit Hierarchien umgehen und daraus folgend, ob für sie eine Integration der Hierarchien in die HyperGrid sinnvoll erscheint. Alle Versuchspersonen waren sich sicher, dass Hierarchien im Zusammenhang mit MedioVis einen großen Mehrwert bieten. Sie waren sich einig, dass der Hauptvorteil darin besteht, dass man so bei Recherchen an noch mehr Informationen zu einem Titel gelangen kann. Mit Informationen sind hier zum Beispiel Zusammengehörigkeiten unter Titeln, das Auffinden von Vater-, Geschwister-, Kinder-Knoten zum jeweiligen Titel und das Entdecken von unbekanntem Dokumenten der gleichen Reihe/Hierarchie gemeint.

Ein Use-Case, den alle Versuchspersonen als Begründung für die sinnvolle Nutzung von Hierarchien aufführten, ist der folgende: Beim Explorieren der Informationen zu einem bekannten Titel stellt man fest, dass dieser Titel einer Hierarchie angehört. Daraufhin durchforstet man die Hierarchie. Dabei wird man auf einige Titel der Hierarchie aufmerksam, die man eventuell später für die Erstellung einer Hausarbeit benötigen könnte. Wäre die Möglichkeit der Exploration der Hierarchie eines Titels nicht gegeben, hätte man diese interessanten Titel nicht gefunden.

Demnach war für die meisten Versuchspersonen beim Explorieren einer Hierarchie das Auffinden anderer, verwandter Titel wichtig. So können Titel, die aufgrund ihrer Beziehung in irgendeiner Weise zusammen gehören, gefunden werden. Eine Versuchsperson erwähnte, dass sie Schwierigkeiten beim Explorieren von Hierar-

chien im KOALA-System der Bibliothek hat. Sie findet insbesondere die Darstellung von Hierarchien sehr benutzerunfreundlich. Dagegen ist ihr die Exploration von Hierarchien mit Hilfe der HyperGridXGL leicht gefallen. Generell bezeichneten die Versuchspersonen die grafische Darstellung der Hierarchie mit den dunklen verzerrten ContextViews und der vorderen DetailedView als sehr gelungen. Fazit der Versuchspersonen: Hierarchien werden mit der Hilfe der HyperGridXGL anschaulich visualisiert.

5.3.4 AttrakDiff-Ergebnisse

Die Auswertung des AttrakDiff-Fragebogens vermittelt einen groben Eindruck bzw. eine Tendenz und sollte angesichts der geringen Teilnehmerzahl mit Vorsicht interpretiert werden. Bei acht Teilnehmern hätte eine umfangreiche statistische Auswertung keine wissenschaftliche Relevanz. Außerdem wurde ein Test-Design gewählt, das als höchst reaktiv angesehen werden kann. Es ist davon auszugehen, dass das Ergebnis des AttrakDiff-Fragebogens durch das häufige Kommunizieren der Versuchsperson mit dem Versuchsleiter beeinflusst wurde. So ist das entstandene Ergebnis nicht auf seine Grundgesamtheit übertragbar. Der Fragebogen wurde, wie im Ablauf (Kapitel 5.2.7) beschrieben, direkt nach dem Bearbeiten aller Testaufgaben von den Versuchspersonen ausgefüllt. Die durchschnittliche Ausprägung der hedonischen und pragmatischen Qualität lässt eine positive Gesamttendenz erkennen. Wie man anhand des Schaubildes 67 feststellen kann, wird die HyperGridXGL von den acht Versuchspersonen als eher begehrt eingestuft.

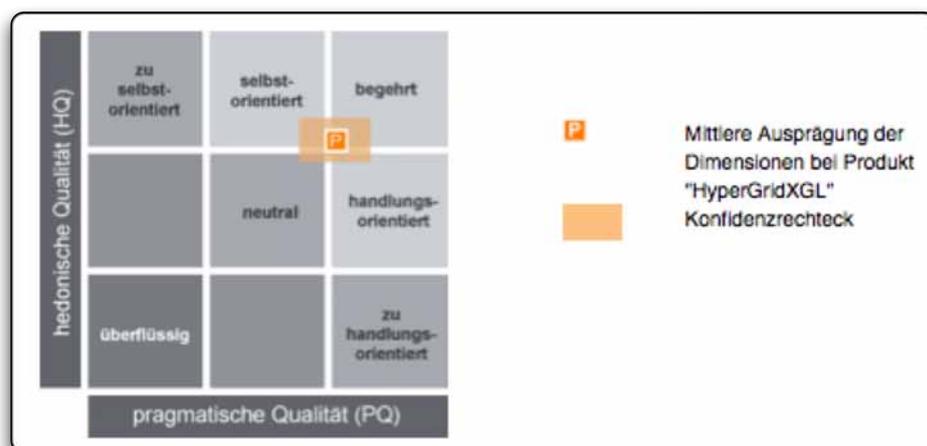


Abbildung 67: Durchschnittliche Ausprägung der hedonischen und pragmatischen Qualität mit Konfidenz-Rechteck

6 Zusammenfassung und Ausblick

Wie [Beaudoin u. a., 1996] im einführenden Zitat feststellten, ist die Art, wie Hierarchien dargestellt werden, entscheidend. Nur durch eine effektive Visualisierungsmethode können die Informationen, die Hierarchien präsentieren, dem Benutzer kommuniziert werden. Demnach sind Hierarchien nicht nur ein Strukturierungswerkzeug komplexe heterogene Informationsräume, sondern geben auch anhand der Struktur Informationen preis. Insbesondere die Darstellung großer hierarchischer Informationsmengen und die anschließende Erfassung der Informationen durch den Benutzer gestaltet sich als schwierig. So haben sich visuelle Suchsysteme, die durch geeignete Darstellung der Daten und gleichzeitiges Bereitstellen explorativer Methoden wie Browsen und analytischer Methoden wie Suchen, Filtern und Sortieren, als unentbehrliche Hilfsmittel erwiesen.

Mit MedioVis wird in der AG Mensch-Computer Interaktion der Universität Konstanz ein innovatives visuelles Suchsystem entwickelt. Dabei ist die HyperGrid, eine zoombare Tabelle, die bisher keine Hierarchien darstellen kann, die Hauptvisualisierung dieses Systems. Im Rahmen dieser Arbeit wurde ein Visualisierungs- und Interaktionskonzept zur Integration hierarchischer Daten in die HyperGrid entwickelt. Dieses Konzept, namens HyperGridXGL, glänzt in erster Linie durch eine vollständige, konsistente Einbindung in das Visualisierungs- und Interaktionskonzept der HyperGrid. Speziell die Verwendung der hierarchischen Zoom-Funktion wurde im Zusammenspiel mit dem semantischen Zoom der HyperGrid in der Evaluation gelobt und scheint damit eine gelungene Lösung zu sein. So kann mit Hilfe des hierarchischen Zooms die komplette hierarchische Struktur eines Objekts exploriert werden. Der semantische Zoom kann nach wie vor auf jedes Objekt mit dem Ziel einer Erhöhung des Informationsgehalts angewandt werden.

Die Hierarchie wird auf einem schubladenähnlichem 3D-Gebilde dargestellt. So wird die fokussierte Hierarchieebene mit den direkten Vor- und Nachfahren im 3D-Raum angezeigt, womit „Fokus und Kontext“ in einer Ansicht zu sehen sind. Durch sequentielles Navigieren von Ebene zu Ebene kann die komplette Hierarchie exploriert bzw. betrachtet werden. Eine klare, anschauliche Kommunikation der Hierarchie erweist sich dabei als eine Stärke der HyperGridXGL.

Zur Illustration der konzeptionellen und technischen Machbarkeit wurde die HyperGridXGL praktisch umgesetzt und in die neueste Version von MedioVis integriert. Anschließend wurde der entstandene Prototyp im Rahmen einer Evaluation getestet. Mit Hilfe dieser Evaluation konnten wichtige Ergebnisse erzielt werden. Dabei ging es in der Studie, insbesondere in den Interviews hauptsächlich um das Aufdecken von Usability-Schwächen und Problemen und nicht um das „Herausfinden positiver Aspekte“. Die Versuchspersonen identifizierten im wesentlichen drei Schwächen:

- Inkonsistenz bei der Exemplar-Ebene
- Mehrfachverortung von Dokumenten
- Buttonproblematik innerhalb der HyperGridXGL.

Diese Usability-Schwächen sollen durch die im Anschluss an die Evaluation entworfenen und in der Arbeit vorgestellten Redesign-Maßnahmen beseitigt werden. Demnach gilt es, die Redesign-Maßnahmen in der nächsten Phase der Weiterentwicklung der HyperGridXGL praktisch in Form weiterer Prototypen umzusetzen und durch Benutzer-Tests auf ihre Gebrauchstauglichkeit zu testen.

Aufgrund des guten Gesamteindrucks in der Evaluation scheint das Visualisierungs- und Interaktionskonzept der HyperGridXGL für den Einsatz in der Bibliothek der Universität Konstanz gerüstet zu sein. So könnte man neben den Tests im Usability-Labor weitere Erkenntnisse durch eine Langzeitevaluation über den Live-Einsatz in der Bibliothek erlangen.

A Aufgaben

Eastwood

Wie heißt der Autor Eastwood mit Vornamen, der eine Reihe von Grammatik-Büchern über die Englische Sprache verfasst hat?

Gehört dieses Buch einer Hierarchie an?

Wie heißen die einzelnen Bücher aus dieser Hierarchie?

Wie oft sind die jeweiligen Bücher im Bestand der Mediothek vorhanden?

Herr der Ringe

Herr der Ringe ist eine Trilogie. Welche Teile dieser Film-Trilogie, die ab dem Jahre 2001 und unter Regie von Peter Jackson entstanden ist, sind im Bestand der Mediothek?

Band of Brothers

Wie viele Teile von Band of Brothers sind im Bestand der Mediothek vorhanden?

Wie oft ist jeder Teil vorhanden bzw. wie viele Exemplare gibt es?

Lizenz zum Töten

Lizenz zum Töten gehört zu welcher Sonder-Edition?

Aus wie vielen Teilen besteht diese Sonder-Edition?

B Interview-Leitfaden

Hast Du offene Fragen, ist dir noch irgendwas an dem Programm unklar?

War dir klar, was Hierarchien sind und zu was diese genutzt werden?

Generelle Meinung zu den Neurungen an MedioVis: gut/schlecht geeignet um Hierarchien zu explorieren?

Wäre es für dich wichtig, die Wurzel einer Hierarchie zu sehen?

Falls Windows-User: Benutzt du den Arbeitsplatzöder den Windows Explorer? Schaust du auf die Breadcrumbs?

TriggerButton

War dir klar was passieren würde, wenn du auf den Button (TriggerButton) in der 2.Spalte, sieht aus wie zwei hintereinanderliegende Blätter mit einem Plus, klickst?

Wie würdest Du beschreiben, was da passiert ist?

Wieso hast du darauf geklickt?

Hast du die Plus/Minus Symbolik verstanden?

Wurdest du durch die Lupe mit dem Plus verwirrt?

ContextView und DetailedView

Dein erster Gedanke als du die Animation nach dem Klick auf den Button in der 2.Spalte gesehen hast?

War dir klar was dort dargestellt wurde?

Unterschiede zwischen den verzerrten Flächen oben und unten und der vorderen Fläche?

ContextView

Hast du den textuellen Hinweis auf der unteren verzerrten Fläche (BottomContext-View) überhaupt bemerkt?

Hast du die unteren hintereinanderliegenden verzerrten Flächen (Stack) überhaupt bemerkt?

Hast du den Sinn vom Stack erkannt? Sinnvoll?

Konnte dieser visuelle Hinweise helfen?

Mouse-over-Button (MoB)

Was dachtest du, als du die Plus- und Minus-Flächen ober-und unterhalb des Button in der 2. Spalte gesehen hast?

War dir klar was passieren würde bei einem Klick auf diese Flächen?

Wie würdest du beschreiben was dann nach einem Klick passiert ist?

Rotation

War die Animation verwirrend?

War die Drehrichtung verwirrend?

Dokumente

Ist dir das Mehrfachvorkommen von Dokumenten aufgefallen?

C Pre-Test-Fragebogen

Pre-Test Fragebogen

Herzlichen Dank, dass Sie sich bereit erklärt haben an dieser Untersuchung teilzunehmen. Bevor wir anfangen, benötigen wir von Ihnen noch einige Angaben zu Ihrer Person und Ihrer bisherigen Erfahrung mit Computern.
Wir möchten Ihnen hiermit noch einmal mitteilen, dass alle Daten vertraulich behandelt werden.

Zur Person

Alter: _____

Geschlecht:

- männlich
 weiblich

Schulbildung:

- Hauptschule
 Realschule
 Gymnasium
 Sonstiges

Studium/Ausbildung: _____

Beruf: _____

Computer/Internet - Erfahrung

Besitzen Sie momentan einen eigenen Computer?Ja Nein **Wieviele Stunden verbringen Sie pro Tag an einem Computer?**0 – 1 Stunde 1 – 2 Stunden 2 – 3 Stunden Mehr als 3 Stunden **Wie viel Zeit verbringen Sie davon im Internet?**0 – 1 Stunde 1 – 2 Stunden 2 – 3 Stunden Mehr als 3 Stunden **Benützen Sie das Internet um nach Informationen zu suchen?**Ja Nein **Wie gelangen Sie an die Informationen, die Sie im Internet suchen?
(Mehrfachauswahl möglich)**Suchmaschine

(google, yahoo, msnsearch)

Wenn ja, welche? _____

Link von Freunden/Bekannten Link in Zeitung/Zeitschrift/Werbung Sonstiges

Wenn ja, wie genau _____

Wie viel Erfahrung haben Sie im Umgang mit Internetsuchmaschinen wie „google“?

0 – 1 Jahr

1 – 3 Jahre

3 – 5 Jahre

Mehr als 5 Jahre

Wie zufrieden sind sie im gewöhnlichen mit Ihren Suchergebnissen? (1 = völlig unzufrieden, 5 = vollkommen zufrieden)

1

2

3

4

5

Haben Sie schon einmal MedioVis benutzt?

Ja

Nein, aber habe davon gehört

Nein, habe auch noch nie davon gehört

**Wenn ja, wie häufig benutzen Sie MedioVis?
(1 = sehr selten, 5 = mehrmals pro Woche)**

1

2

3

4

5

D Post-Test-Fragebogen

Post-Test Fragebogen

Zum Abschluss möchten wir Sie bitten, noch drei kurze Fragen zu beantworten, die sehr wichtig für die Bewertung von MedioVis sind.

Nochmals herzlichen Dank, dass Sie an dieser Untersuchung teilgenommen haben.

Wie sind Sie mit der neuen Version von MedioVis zum Explorieren von Hierarchien zu Recht gekommen?

Sehr gut Schlecht

Wo sehen Sie die Stärken?

Wo sehen Sie die Schwächen?

Gibt es noch sonstige Anmerkungen Ihrerseits zu MedioVis?

E Anhang

CD-ROM

Dieser Arbeit liegt eine CD-ROM bei, welche

- eine digitale Version der vorliegenden Arbeit (Bachelorarbeit-HyperGridXGL-DWinkler.pdf),
- den Programmcode (MedioVisX-HyperGridXGL)
- sowie eine ausführbare JAR-Version des Prototypen (MedioVis-HyperGridXGL-DWinkler.jar)

beinhaltet.

Kontaktinformation

Dominic Winkler

Email: dominicwinkler@gmail.com

Abbildungsverzeichnis

1	Baum	9
2	Baum mit Beispieldatensatz „Der Herr der Ringe“	11
3	Baum-Visualisierung	12
4	Listendarstellung des UNIX-Befehls „ls“	14
5	Windows Explorer	15
6	Hyperbolic Browser	17
7	Bifocale Hyperbolic Browser	17
8	slice-and-dice Layout	19
9	clustered und squarified Treemaps	20
10	PhotoMesa	21
11	Zoomable Treemap	22
12	Cone Tree [Robertson u. a., 1991]	22
13	SpaceTree [Plaisant u. Grosjean, 2002]	24
14	ZoomWorld	29
15	Geometrischer Zoom	30
16	Semantischer Zoom	30
17	Die HyperGrid	32
18	Die verschiedenen Zoomstufen des mittleren AOI in der HyperGrid	33
19	Die verschiedenen Zoomstufen der Browserkomponente	34
20	Die Darstellung von GoogleMaps innerhalb der HyperGrid	35
21	PDF-Dateien und Filme innerhalb der HyperGrid	35
22	Direkter Zugriff auf das KOALA-System innerhalb der HyperGrid	36
23	TreeTable: Ergebnis der Suche nach „Eastwood“	38
24	Die Animation zum Aufklappen hierarchischer Titel in der TreeTable	39
25	Die Hierarchie: „A grammar of spoken English“	39
26	TreeTable nach der Suche nach „Structures“	40
27	TreeTable mit einem aufgeklappten hierarchischen Dokument	40
28	Die Konzeptidee „classical“	45
29	Der Finder unter Mac OS X	47
30	Die Konzeptidee „finder“	48
31	Die Konzeptidee „cube“	49
32	Suchergebnis „Eastwood“ in der HyperGridXGL	52
33	Die gefilterte HyperGridXGL	53
34	Die Communication-Ebene mit Plus und Minus MoB	53
35	Die Exemplar-Ebene des Titels „Communication“	54
36	Die HyperGridXGL	56
37	Unterschiede zwischen HyperGridXGL und TreeTable	57
38	Die Perspective Wall [Mackinlay u. a., 1991]	58
39	Der Schubladen-Effekt	58
40	Synthese aus 3D-Schubladen und Hierarchie	59

41	3D-Schublade mit Stack	60
42	Der TriggerButton	60
43	Aufgezoomte Hierarchie in der HyperGridXGL	62
44	Direktes Feedback beim Überfahren der ContextView	63
45	Filtern, Sortieren, semantischer Zoom und Browserkomponente	64
46	Zusammenspiel aus Datenmodell und Visualisierung	68
47	Pre-Processing bei der Erzeugung der ContextView	69
48	Übersicht der Versuchspersonen	73
49	Zeitlicher und inhaltlicher Ablauf des Tests	76
50	Beispiel-Hierarchie	77
51	Titel mit mehreren Exemplaren	79
52	Titel mit nur einem Exemplar	80
53	Vergleich von bisheriger Hierarchie mit konsistenter Hierarchie	80
54	Exemplar-Ebene des Titels Communication	81
55	Suche nach „Band of Brothers“	81
56	Nach dem Aufzoomen der Hierarchie des ersten Suchtreffers	82
57	Mehrfachverortung von Dokumenten	82
58	Flap-In/-Out Button und TriggerButton	84
59	Plus- und Minus-TriggerButton, sowie MoBs	84
60	Stack und textueller Hinweis	86
61	Die Ausleihstatus-Tabelle in einem Browser-Fenster	88
62	Standortplan-Treemap und Ausleihstatus	89
63	Standortplan-Treemap mit Mouse-Over-Effekt und Ausleihstatus	89
64	Standort-Scatterplot	90
65	Umrahmte TriggerButton und MoBs	91
66	Gesamturteil: Skala des Post-Test-Fragebogens	92
67	Ausprägung der hedonischen und pragmatischen Qualität	94

Literatur

- [Beaudoin u. a. 1996] BEAUDOIN, Luc ; PARENT, Marc-Antoine ; VROOMEN, Louis C.: Cheops: a compact explorer for complex hierarchies. In: *VIS '96: Proceedings of the 7th conference on Visualization '96*. Los Alamitos, CA, USA : IEEE Computer Society Press, 1996, S. 87–ff.
- [Becker u. Cleveland 1987] BECKER, Richard A. ; CLEVELAND, William S.: Brushing scatterplots. In: *Technometrics* 29 (1987), Nr. 2, S. 127–142
- [Bederson 2001] BEDERSON, Benjamin B.: PhotoMesa: a zoomable image browser using quantum treemaps and bubblemaps. In: *UIST '01: Proceedings of the 14th annual ACM symposium on User interface software and technology*. New York, NY, USA : ACM Press, 2001, S. 71–80
- [Bederson u. a. 2004] BEDERSON, Benjamin B. ; CLAMAGE, Aaron ; CZERWINSKI, Mary P. ; ROBERTSON, George G.: DateLens: A fisheye calendar interface for PDAs. In: *ACM Trans. Comput.-Hum. Interact.* 11 (2004), Nr. 1, S. 90–119
- [Bederson u. a. 2002] BEDERSON, Benjamin B. ; SHNEIDERMAN, Ben ; WATTENBERG, Martin: Ordered and quantum treemaps: Making effective use of 2D space to display hierarchies. In: *ACM Trans. Graph.* 21 (2002), Nr. 4, S. 833–854
- [Blanch u. Lecolinet 2006] BLANCH, Renaus ; LECOLINET, Éric: Navigation Techniques for Zoomable Treemaps. In: *Conference Companion of UIST 2006*, 2006, 49-50
- [Bruls u. a. 2000] BRULS, M. ; HUIZING, K. ; WIJK, J. van: Squarified Treemaps. In: *In Proc. of Joint Eurographics and IEEE TCVG Symp. on Visualization (TCVG 2000) IEEE Press*, 2000, S. 33–42
- [D. Austin Henderson u. Card 1986] D. AUSTIN HENDERSON, Jr. ; CARD, Stuart: Rooms: the use of multiple virtual workspaces to reduce space contention in a window-based graphical user interface. In: *ACM Trans. Graph.* 5 (1986), Nr. 3, S. 211–243
- [Gerken 2004] GERKEN, Jens: *Evaluation eines Metadatenbrowsers - Liste vs. Leveltable*, Universität Konstanz, Bachelor-Arbeit, 3 2004
- [Gerken 2006] GERKEN, Jens: *Orientierung und Navigation in zoombaren Benutzerschnittstellen unter besonderer Berücksichtigung kognitions-psychologischer Erkenntnisse*, University of Konstanz, Master-Arbeit, Dec 2006
- [Grün u. a. 2005] GRÜN, C. ; GERKEN, J. ; JETTER, H.C. ; KÖNIG, W. ; REITERER, H.: MedioVis - a User-Centred Library Metadata Browser,. In: *Research and Advanced Technology for Digital Libraries 2005*, Springer Verlag, Sep 2005

- [Grün 2004] GRÜN, Christian: *Entwicklung eines visuellen Metadaten-Browsers für die Mediothek Konstanz*, Universität Konstanz, Master-Arbeit, Dec 2004
- [Haist u. Göbel 2002] HAIST, J. ; GÖBEL, S.: INVISIP: Metadata-based Information Visualization Techniques to Access Geodata Archives and to Support the Site Planning Process. In: *INVISIP: Metadata-based Information Visualization Techniques to Access Geodata Archives and to Support the Site Planning Process*, 2002
- [Hassenzahl u. a. 2003] HASSENZAHL, Marc ; BURMESTER, Michael ; KOLLER, Franz: AttrakDiff: Ein Fragebogen zur Messung wahrgenommener hedonischer und pragmatischer Qualität. In: *Mensch & Computer*, 2003
- [Jetter 2002] JETTER, Hans-Christian: Hyperbolic Treebrowser - der hyperbolische Browser für hierarchische Daten / Universität Konstanz. 2002. – Forschungsbericht
- [Jetter 2007] JETTER, Hans-Christian: *Informationsarchitektur und Informationsvisualisierung für die Post-WIMP Ära*, Master-Arbeit, Jan 2007
- [Jetter u. a. 2005] JETTER, H.C. ; GERKEN, J. ; KÖNIG, W. ; GRÜN, C. ; REITERER, H.: HyperGrid - Accessing Complex Information Spaces. In: *People and Computers XIX - The Bigger Picture, Proceedings of HCI 2005, Vol. 1*, 2005
- [König 2005] KÖNIG, Werner A.: MedioVis Framework: Dokumentation und Redesign des Datenmodells zur Unterstützung von hierarchischen Strukturen komplexer Informationsräume / University of Konstanz. 2005. – Forschungsbericht
- [König 2006] KÖNIG, Werner A.: *Referenzmodell und Machbarkeitsstudie für ein neues Zoomable User Interface Paradigma*, University of Konstanz, Master-Arbeit, Jun 2006
- [Lamping u. a. 1995] LAMPING, John ; RAO, Ramana ; PIROLI, Peter: A focus+context technique based on hyperbolic geometry for visualizing large hierarchies. In: *CHI '95: Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA : ACM Press/Addison-Wesley Publishing Co., 1995, S. 401–408
- [Mackinlay u. a. 1991] MACKINLAY, Jock D. ; ROBERTSON, George G. ; CARD, Stuart K.: The perspective wall: detail and context smoothly integrated. In: *CHI '91: Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA : ACM Press, 1991, S. 173–176
- [Müller u. Klein 2002] MÜLLER, F. ; KLEIN, P.: Visual Information Retrieval with the SuperTable + Scatterplot. In: *Information Visualisation - IV02*, IEEE, Jul 2002, S. 70–75

- [Mußler u. Reiterer 2000] MUSSLER, G. ; REITERER, H.: INSYDER - An Information Assistant for Business Intelligence. In: *INSYDER - An Information Assistant for Business Intelligence*, ACM press, 2000, S. 112–119
- [Norman 1999] NORMAN, Donald A.: Affordance, conventions, and design. In: *interactions* 6 (1999), Nr. 3, S. 38–43
- [North u. Schneiderman 1997] NORTH, Chris ; SCHNEIDERMAN, Ben: A Taxonomy of Multiple Window Coordinations. 1997 (CS-TR-3854). – Forschungsbericht
- [Perlin u. Fox 1993] PERLIN, Ken ; FOX, David: Pad: an alternative approach to the computer interface. In: *SIGGRAPH '93: Proceedings of the 20th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM Press, 1993, S. 57–64
- [Peters 2005] PETERS, Detlef: *Zugriff auf linear geordnete Daten mittels dreidimensionaler animierter Visualisierung*, Universität Paderborn, Diplomarbeit, März 2005
- [Plaisant u. Grosjean 2002] PLAISANT, C. ; GROSJEAN, J. a.: SpaceTree: Supporting Exploration in Large Node Link Tree, Design Evolution and Empirical Evaluation, 2002, 57–64
- [Rao u. Card 1994] RAO, Ramana ; CARD, Stuart K.: The table lens: merging graphical and symbolic representations in an interactive focus + context visualization for tabular information. In: *CHI '94: Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA : ACM Press, 1994, S. 318–322
- [Raskin 2000] RASKIN, Jef: *The Humane Interface: New Directions for Designing Interactive Systems*. Addison-Wesley Professional, 2000. – ISBN 0201379376
- [Reiterer u. a. 2005] REITERER, H. ; JETTER, H.C. ; KÖNIG, W. ; GERKEN, J. ; GRÜN, C.: Zoomtechniken zur Exploration komplexer Informationsräume am Beispiel "HyperGrid", 2005
- [Reiterer u. Klein 2003] REITERER, H. ; KLEIN, P.: Metadata Visualization with VisMeB. In: *7th International Conference on Information Visualisation 2003*, IEEEComputer Society, Jul 2003, S. 600–605
- [Resnikoff 1989] RESNIKOFF, Howard L.: *The illusion of reality*. New York, NY, USA : Springer-Verlag New York, Inc., 1989
- [Robertson u. a. 1991] ROBERTSON, George G. ; MACKINLAY, Jock D. ; CARD, Stuart K.: Cone Trees: animated 3D visualizations of hierarchical information. In: *CHI '91: Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA : ACM Press, 1991, S. 189–194

- [Roth 2003] ROTH, Gerhard: *Fühlen, Denken, Handeln. Wie das Gehirn unser Verhalten steuert*. Frankfurt am Main : Suhrkamp, 2003
- [Rüger u. a. 1996] RÜGER, Michael ; PREIM, Bernhard ; RITTER, Alf: Zoom navigation exploring large information and application spaces. In: *AVI '96: Proceedings of the workshop on Advanced visual interfaces*. New York, NY, USA : ACM Press, 1996, S. 40–48
- [Shneiderman 1992] SHNEIDERMAN, Ben: Tree visualization with tree-maps: 2-d space-filling approach. In: *ACM Trans. Graph.* 11 (1992), Nr. 1, S. 92–99
- [Shneiderman 1996] SHNEIDERMAN, Ben: The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In: *IEEE Visual Languages*. College Park, Maryland 20742, U.S.A., 1996 (UMCP-CSD CS-TR-3665), S. 336–343
- [Shneiderman 2006] SHNEIDERMAN, Ben: *Treemaps for space-constrained visualization of hierarchies*. <http://www.cs.umd.edu/hcil/treemap-history/index.shtml>. Version: April 2006, Abruf: Januar 2007
- [Wake u. Fox 1995] WAKE, William C. ; FOX, Edward A.: SorTables: a browser for a digital library. In: *CIKM '95: Proceedings of the fourth international conference on Information and knowledge management*. New York, NY, USA : ACM Press, 1995, S. 175–181
- [Wattenberg 1999] WATTENBERG, Martin: Visualizing the stock market. In: *CHI '99: CHI '99 extended abstracts on Human factors in computing systems*. New York, NY, USA : ACM Press, 1999, S. 188–189
- [Winkler 2006] WINKLER, Dominic: *HyperGridXGL - Ein Interaktionskonzept zur Integration hierarchischer Daten in die HyperGrid*. Februar 2006
- [Xiang u. a. 2005] XIANG, Yang ; CHAU, Michael ; ATABAKHSH, Homa ; CHEN, Hsinchun: Visualizing criminal relationships: comparison of a hyperbolic tree and a hierarchical list. In: *Decis. Support Syst.* 41 (2005), Nr. 1, S. 69–83