

Interaktive Informationsvisualisierung am Beispiel statischer Erhebungen über Standards im E-Business

Diplomarbeit

im Fach Digitales Informationsdesign
Studiengang Informationsmanagement
der
Fachhochschule Stuttgart –
Hochschule der Medien

Silke Kleindienst

Erstprüfer: Prof. Dr. Christian Rathke
Zweitprüfer: Prof. Dr. Wolf-Fritz Riekert

Bearbeitungszeitraum: 15. Juli 2002 bis 12. November 2002

Stuttgart, November 2002

Inhaltsverzeichnis

Inhaltsverzeichnis.....	2
Abbildungsverzeichnis	5
Tabellenverzeichnis.....	6
Kurzfassung	7
Abstract	8
1 Einleitung und Überblick.....	9
1.1 Motivation und Aufgabenstellung	9
1.2 Lösungsansatz.....	9
1.3 Der Aufbau der Diplomarbeit	10
2 Informationsvisualisierung	11
2.1 Bestimmung der Begriffe	11
2.1.1 Visualisierung.....	11
2.1.2 Daten – Information	12
2.1.3 Datenvisualisierung – Wissenschaftliche Visualisierung - Informationsvisualisierung	13
2.2 Geschichtlicher Abriss.....	14
2.3 Bedeutung und Ziel der Informationsvisualisierung.....	17
2.4 Anforderungen an Informationsvisualisierung.....	18
2.5 Informationsvisualisierung im Visualisierungsprozess.....	19
3 Techniken der Informationspräsentation.....	21
3.1 Stabdiagramme	23
3.2 Kreisdiagramme.....	24
3.3 Liniendiagramme.....	25
3.4 Piktogramme.....	27

4	Interaktive Informationsvisualisierung.....	28
4.1	Dynamic Queries	29
4.1.1	SQL-Abfragen	30
4.1.2	Dynamic Queries	31
4.1.3	FilmFinder	33
4.2	MagicLens TM	35
4.3	Overview and Detail.....	37
4.3.1	LifeLines	38
5	Beschreibung und Aufbau des Visualisierungstools	40
5.1	Installation und Inbetriebnahme	43
5.1.1	Windows	43
5.1.2	Unix	44
5.2	VisJex aus Anwendersicht.....	45
5.2.1	Menü	46
5.2.2	PieChart.....	46
5.2.3	BarChart.....	48
5.2.4	Legende	49
5.3	Der innere Aufbau von VisJex	50
5.3.1	MainClass	50
5.3.2	MenuBar	51
5.3.3	ReadFile.....	51
5.3.4	ClickablePie	53
5.3.5	BarChart.....	54
5.3.6	History.....	57
6	Entwicklung und Implementierung des Visualisierungstools.....	58
6.1	Entwicklungsumgebung und Hilfsmittel.....	58
6.1.1	Java Development Kit 1.4.....	58
6.1.2	Eclipse Plattform 2.0.....	59
6.2	Überblick über die Entwicklungsschritte.....	62
6.2.1	Aneignen der notwendigen Technik.....	62
6.2.2	Realisieren von Teilaspekten	63
6.2.3	Zusammenführen der Komponenten	64

7	Projekt „Diplomarbeit“	65
	7.1 Phasenmodell	66
	7.2 Arbeitspakete	67
	7.3 Meilensteine	69
	7.4 MS Project	70
	7.5 JVision.....	70
8	Ausblick.....	71
	Literaturverzeichnis.....	73
	Anhang.....	76
	A: Anforderungsanalyse	76
	Erklärung.....	79

Abbildungsverzeichnis

Abbildung 1 Minards Visualisierung des Russlandfeldzugs	14
Abbildung 2: Florence Nightingales Rose.....	15
Abbildung 3 Harry Becks Plan der Londoner U-Bahn	16
Abbildung 4: Visualisierungsprozess	19
Abbildung 5: Säulendiagramm	23
Abbildung 6: Gestapeltes Balkendiagramm.....	23
Abbildung 7: Kreisdiagramm	24
Abbildung 8: Liniendiagramm 1.....	25
Abbildung 9: Liniendiagramm 2.....	25
Abbildung 10: Flächendiagramm 1	26
Abbildung 11: Flächendiagramm 2	26
Abbildung 12: Piktogramm	27
Abbildung 13: SQL-Statements.....	30
Abbildung 14: FilmFinder.....	33
Abbildung 15: Demo Magic Lens	36
Abbildung 16: LifeLines	38
Abbildung 17: Source-Datei im ASCII- Format	41
Abbildung 18: Source-Datei in MS Excel	41
Abbildung 19: Daten aus der Source-Datei mit VisJex angezeigt.....	42
Abbildung 20: Screenshot DOS-Eingabeaufforderung	43
Abbildung 21: Screenshot Shell	44
Abbildung 22: VisJex - Screenshot	45
Abbildung 23: Screenshot VisJex: Zoomstufe 2	46
Abbildung 24: Screenshot VisJex Zoomstufe 3	47
Abbildung 25: Screenshot VisJex: selektiertes PieSegment	47
Abbildung 26: Screenshot VisJex: BarChart.....	48
Abbildung 27: Screenshot VisJex: Dialogfenster "Weitere Informationen"	48
Abbildung 28: Screenshot VisJex: Legende	49
Abbildung 29: VisJex - Klassendiagramm	50
Abbildung 30: Screenshot source.txt	51
Abbildung 31: Prototypen 1	62
Abbildung 32: Prototypen 2	63
Abbildung 33: Screenshot VisJex.....	64
Abbildung 34: Phasenmodell.....	66
Abbildung 35: Abhängigkeit der Arbeitspakete	68
Abbildung 36: Meilensteine	69

Tabellenverzeichnis

Tabelle 1: Grafikarten	22
Tabelle 2: Interaktionstechniken.....	29
Tabelle 3: Eclipse Funktionalität: Übersicht	60
Tabelle 4: Eclipse: Vor- und Nachteile.....	61
Tabelle 5: Arbeitspakete	67

Kurzfassung

Diese Arbeit beschäftigt sich mit den Themengebieten Informationspräsentation und Visualisierung. Durch die theoretische Auseinandersetzung mit diesen Themengebieten wurden wichtige Merkmale einer Anwendung identifiziert, die nötig sind, um dem Benutzer eines Visualisierungs-Tools das Explorieren einer großen Datenmenge und somit das Treffen einer sachlich fundierten Entscheidung zu erleichtern.

Die identifizierten Merkmale wurden in der Anwendung VisJex exemplarisch verwirklicht. VisJex verbindet Techniken der traditionellen Datenpräsentation mit interaktiven Techniken der Informationsvisualisierung.

Um die Idee des „Overview and Detail“ zu realisieren, auf der die Anwendung basiert, wurde besonders darauf geachtet, dass die bildliche Darstellung der Daten für den Nutzer nicht sequentiell sondern simultan sichtbar ist. So gibt ein PieChart eine Übersicht über die Datenmenge, während ein Balkendiagramm und ein weiteres Dialogfenster Einsicht in die Details der Datenmenge gewähren.

Das Programm VisJex könnte beispielsweise als Teil eines Management-Informationssystem im Rahmen eines Data-Warehouses zum Einsatz kommen.

Schlagworte

Visualisierung, Datenpräsentation, Informationsvisualisierung, Visualisierungswerkzeug

Abstract

This work is dealing with the topics presentation of information and visualization. Important features for a visualization tool were recognized, being necessary to ease exploring a huge data set and consequently coming to a self-justifying decision.

The application VisJex implements these identified features. VisJex connects traditional technologies of data presentation to interactive visualization of information.

The representation of data becomes not succesively but simultaneously visible for the user. So the idea of "overview and detail", on which the application based, is realized. To afford an overview about the amount of data there is a pie chart. To gain an insight into the particular feature of data, there are bar charts and further dialog windows with further information.

The application VisJex could go into action bye example as part of a management information system within a data warehouse.

Key Words

Visualization, representation of data, information visualization, visualization tool

1 Einleitung und Überblick

1.1 Motivation und Aufgabenstellung

Für Entscheidungen verschiedener Art werden häufig Daten als Grundlage genutzt, die nur als Rohdaten vorliegen. Um den Entscheidungsträgern eine Erkundung und Interpretation dieser Daten zu erleichtern, werden Daten visualisiert. Betrachtet man konventionelle Techniken, stellt man verschiedene Schwachstellen fest. Zum Einen wird der Entscheidungsträger nicht selbst interpretierend tätig, sondern verlässt sich auf bereits interpretierte und somit selektierte Daten. Zum Anderen mangelt es an einer interaktiven Technik, die verschiedene Datenmengen und ihre Relevanz zueinander visualisiert.

Am Beispiel der Daten einer Befragung¹ über die Verbreitung und Akzeptanz von Standards im E-Business, die "Führungskräften in Unternehmen einen Überblick über existierende Standardisierungsinitiativen" liefern soll, wird gezeigt, wie eine solche Realisierung aussehen kann.

Für eine Vertiefung in die Problematik der Standardisierungsvorhaben im Bereich E-Business können neben der oben genannten Studie folgende Quellen dienen:

www.bmecat.org

www.etim.de

www.eclass.de

www.opentrans.org

1.2 Lösungsansatz

Mit der Entwicklung des Werkzeuges VisJex (Visual Java Explorer) werden die interpretierenden Phasen der Datenaufbereitung und Präsentation in die Hand des Anwenders gelegt, sodass er näher an den Rohdaten ist als zuvor. Um dem Anwender

¹ [Otto et al.02]

eine bekannte Umgebung zu bieten, in der er interagieren kann, werden konventionelle Präsentationstechniken verwendet und kombiniert. Durch den Aspekt der Interaktivität, den VisJex realisiert, wird das Erforschen einer Datenmenge zu einer spielerischen Erkundungstour durch eine spezifizierte Datenwelt.

1.3 Der Aufbau der Diplomarbeit

Das folgende **Kapitel 2** stellt die Disziplin „Informationsvisualisierung“ als Einführung in das Thema der Diplomarbeit vor. Für Begriffe, mit denen sich die Diplomarbeit beschäftigt, wird an dieser Stelle durch das Zusammenfassen verschiedener Aspekte eine Definition gegeben. Anschließend wird auf die Bedeutung der Informationsvisualisierung eingegangen und ihre Rolle innerhalb des Visualisierungsprozesses von den Rohdaten hin zu einer grafischen Visualisierung erläutert.

Kapitel 3 stellt konventionelle Techniken der Informationspräsentation vor. Es wird besonders auf die Aspekte eingegangen, die notwendig sind, um eine möglichst geringe Verzerrung der Rohdaten zu erlangen und die grafische Darstellung aussagekräftig zu machen.

Was sich hinter dem Begriff „Interaktive Informationsvisualisierung“ verbirgt, erläutert **Kapitel 4**. Darüber hinaus werden ausgewählte Techniken dieser Disziplin exemplarisch vorgestellt.

Kapitel 5 beschreibt das Visualisierungswerkzeug VisJex für den Anwender und gibt ihm Hilfestellung bei der Benutzung. Im Weiteren wird der innere Aufbau von VisJex erläutert.

Auf den Prozess der softwaretechnischen Umsetzung von VisJex wird in Kapitel 6 eingegangen. Die einzelnen Phasen werden hier ausführlich beschrieben.

Kapitel 7 beleuchtet die Diplomarbeit unter dem Aspekt des Projektmanagement und erläutert die dafür gewählte Vorgehensweise.

Einen Ausblick für Funktionen, die innerhalb des Programms VisJex noch realisiert werden könnten, gibt **Kapitel 8**.

2 Informationsvisualisierung

Um einen Überblick darüber zu bekommen, was Informationsvisualisierung meint und welche Techniken momentan angewandt werden, um Daten zu visualisieren, steht am Anfang eine Beschäftigung mit dem Stand der Wissenschaft in diesem Bereich. Daraus entwickelten sich eigene Vorstellungen darüber, mit welchen Hilfsmitteln und Techniken eine Visualisierung von Daten möglich ist. Diese Erkenntnisse bilden die Grundlage für die Entwicklung des Visualisierungstools VisJex.

2.1 Bestimmung der Begriffe

In der Einführung zu dieser Diplomarbeit tauchen einige Begriffe auf, für die es bisher keine allgemeingültige Definition gibt, da die Informationsvisualisierung als Teilgebiet der Informatik eine recht junge Wissenschaft ist. Um eine gemeinsame Basis zu schaffen, beschäftigt sich das zweite Kapitel im weiteren damit, notwendige Begriffe zu definieren.

2.1.1 Visualisierung

Betrachtet man den Begriff „Visualisierung“ in der Vielzahl der Artikel, die zu diesem Thema im Bereich der Computer-Literatur erscheinen, fällt auf, dass er different behandelt wird. Um uns der Bedeutung zu nähern, werden hier verschiedene Definitionen einführend genannt.

***Visualisieren:** etwas optisch so betonen, dass es Aufmerksamkeit erregt; Ideen in ein Bild umsetzen.²*

***Visualization** is an activity which a human being engages in, and [...] it is a cognitive activity. The potential value of visualization [is] gaining insight and understanding.³*

² [Duden90]

³ [Spence01]

Visualization: *The use of computer-supported, interactive, visual representations of data to amplify cognition.*⁴

Visualization *offers a method for seeing the unseen.*⁵

Fasst man die relevanten Aspekte für diese Diplomarbeit zusammen, ergibt sich folgende neue Definition:

Visualisierung ist ein menschlicher, kognitiver, bisweilen computer-unterstützter Vorgang, dessen Ziel es ist, Daten oder Sachverhalte so zu repräsentieren, dass sie nicht nur in ihrem Kontext verständlicher sind, sondern den Betrachter befähigen, aus ihnen eigene, neue Ideen zu entwickeln.

2.1.2 Daten – Information

Daten, bzw. Informationen sind in dem Prozess der Visualisierung die kleinste Einheit, sozusagen die Atome der Visualisierung. Umgangssprachlich wird die Bedeutung der beiden Begriffe nicht unterschieden, sogar der Duden⁶ setzt sie als Synonyme. Zur Verdeutlichung der Beziehung zwischen Daten und Information ziehen wir eine Definition von Albrecht Müller hinzu:

Als Daten bezeichnen wir die symbolische Repräsentation von Sachverhalten.

*Als Information bezeichnen wir ein Bündel von Daten, das in einer propositionalen Struktur zusammengefasst ist.*⁷

Auch John Gundry nimmt eine ähnliche Unterscheidung der beiden Begriffe auf seiner Webseite „Knowledge Ability“ vor. Er bezeichnet Daten als 0-dimensional und Information als 1-dimensional.⁸ Informationen sind demnach Daten, die in Bezug auf ein Subjekt stehen.

⁴ [Card et al. 99]

⁵ [McCormick87]

⁶ [Duden90]

⁷ [Müller97]

⁸ [Gundry02]

2.1.3 Datenvisualisierung – Wissenschaftliche Visualisierung - Informationsvisualisierung

Um die Begriffe in den Kontext der Visualisierung einordnen zu können, folgt hier eine Beschreibung und Klassifizierung.

Datenvisualisierung wird meist als Oberbegriff der Visualisierung beliebiger Daten verwendet. Während Wissenschaftliche Visualisierung die Repräsentation realitätsbezogener Daten z.B. aus der Medizin, Chemie, Meteorologie beschreibt, wird Informationsvisualisierung als Repräsentation abstrakter Daten bezeichnet z.B. das Dateisystem einer Festplatte oder statistische Daten aus einer Befragung. Daneben ist der Begriff Softwarevisualisierung, der sich mit der Darstellung der Arbeitsweise von Computerprogrammen beschäftigt, zu nennen.

Für diese Diplomarbeit wird der Begriff „Informationsvisualisierung“ wie folgt definiert: Informationsvisualisierung ist die Erschließung großer Datenbestände mit Hilfe von Computern unter Einsatz grafischer Mittel, wodurch es dem Betrachter erleichtert wird, Einsicht in den Datengehalt zu erlangen.

2.2 Geschichtlicher Abriss

Geht man in der Geschichte des Menschen so weit zurück, als es weder Zahlen noch Buchstaben gab, erkennt man, dass Visualisierung schon immer das Bedürfnis der Menschen ausdrückte, sich nicht nur auf gesprochene Worte zu verlassen. Höhlenmalereien halten Erlebnisse für kommende Generationen fest. Verzierungen an Gebrauchsgegenständen erzählen Legenden.

Bis die Informationsvisualisierung eine wissenschaftliche Disziplin wurde, war es aber noch ein weiter Weg.

Monsieur Minard, ein Kartograph Napoleons, stellt in einer Illustration des legendären Moskaufeldzuges dar, wie sich die Zahl der Soldaten im Verlaufe der Schlacht und dem Marsch nach Hause durch die vielen Todesfälle drastisch verminderte. Es gelingt ihm, in einer einfachen Grafik den immensen Verlust der Soldaten, abhängig von verschiedenen Variablen, wie Wettereinflüsse, geographische Verhältnisse eindrucksvoll und nachvollziehbar darzustellen.

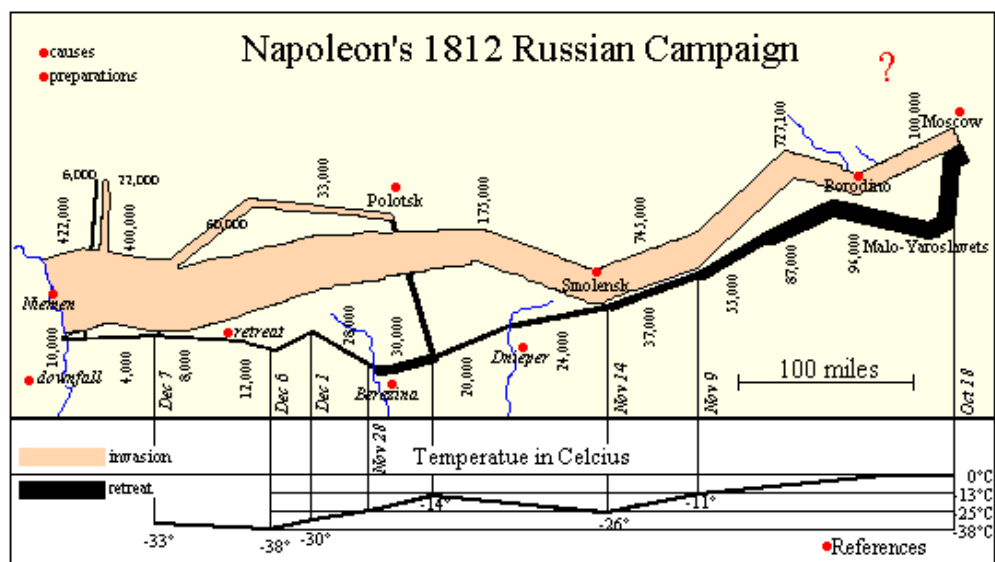


Abbildung 1 Minards Visualisierung des Russlandfeldzugs⁹

⁹ [Hicks2002]

Florence Nightingales Bericht (1858) über die Zustände der medizinischen Versorgung britischer Soldaten während des Krieges auf der Krim enthalten etliche Verbesserungsvorschläge, deren Dringlichkeit sie mit einem Diagramm in der Form einer Rose zu unterstützen versuchte. Ihr Diagramm zeigt nicht nur das Anwachsen der Zahl der Todesfälle, sondern als Vergleichszahl auch die Zahl der Todesfälle in Armeehospitalen in Manchester, England.

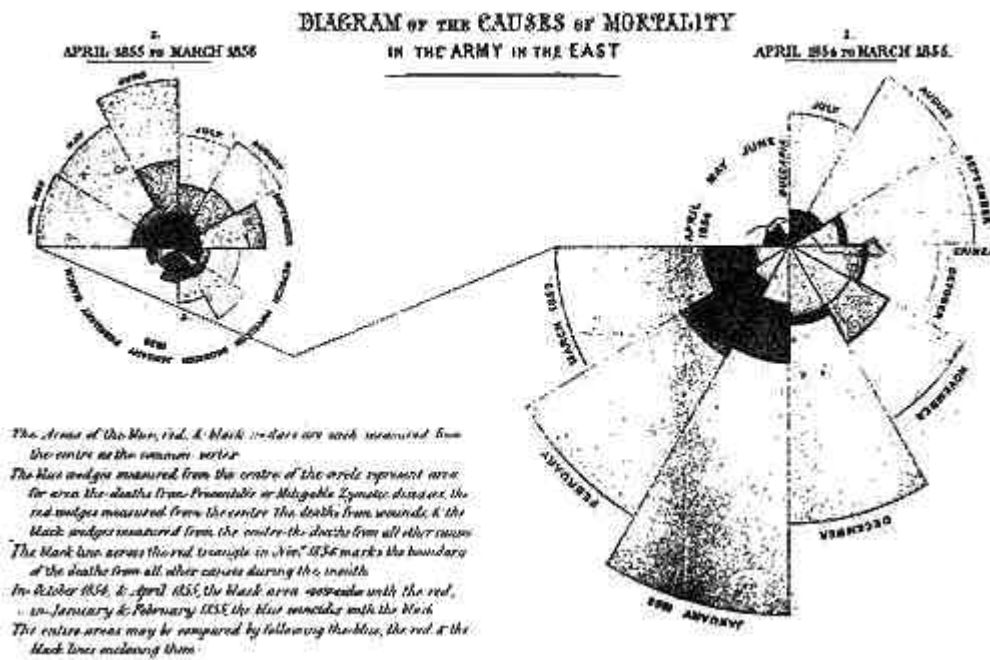


Abbildung 2: Florence Nightingales Rose¹⁰

¹⁰ [Scott98]

Harry Beck hat 1931 die geniale Idee, dass es unter der Erde völlig irrelevant ist, wo man sich geografisch befindet, so lange man in die richtige U-Bahn steigt. Mit dieser Erkenntnis gestaltet er die U-Bahn Karte Londons völlig neu und gibt ihr ein Aussehen, das an elektrische Schaltkreise erinnert. Seit damals wird seine brillante Idee weltweit kopiert und ist uns allen vertraut.



Abbildung 3 Harry Becks Plan der Londoner U-Bahn¹¹

Erst 1967 erscheint das erste Werk, das sich theoretisch mit Visualisierung beschäftigt. Bertin beschreibt in „The Semiology of Graphics“ die Grundelemente von Diagrammen und den Rahmen für ihre Gestaltung.

Ab 1977 erfordern enorme Mengen an Messdaten aus der Wissenschaft (Raumfahrt, Medizin) besondere Visualisierungsanwendungen. Gleichzeitig erscheinen weitere theoretische Betrachtungen zu dieser Problematik.

In „Exploratory Data Analysis“ betont Tukey 1977 den Gebrauch von Bildern, um eine schnelle Einsicht in statistische Datenmengen zu erhalten.

1983 veröffentlicht Tufte seine Theorie der Datengraphen, die die Maximierung der Dichte der nützlichen Informationen betont.

¹¹ [Tube02]

Datenvisualisierung wird ab 1987 als eigenes Forschungsgebiet betrachtet, in dem Wissenschaftler aus unterschiedlichen Bereichen zusammen forschen.

Cleveland und McGill geben 1988 ein Werk mit dem Titel „Dynamic Graphics of Statistic“ heraus, in dem sie erstmals darauf eingehen, wie Datensätze mit vielen Variablen dargestellt werden können.

1989 wird der Begriff „information visualization“ erstmals bei Robertson, Card und Mackinlay erwähnt.

1990 findet die erste IEEE („Eye-triple-E“ = Institute of Electrical and Electronics Engineers) Visualization Conference statt, an der Wissenschaftler aus den Bereichen Geografie, Physik und Computerwissenschaften teilnehmen.

2.3 Bedeutung und Ziel der Informationsvisualisierung

Wir leben in einer Welt, in der die Menge an Informationen fast schon exponentiell zunimmt. Besonders deutlich ist dieser Sachverhalt, wenn wir nach einer bestimmten Information im World Wide Web suchen. Angesichts der Fülle des Angebots wird der Benutzer an die Informationsaufbereitung mindestens zwei Forderungen stellen. Erstens: sie ist so klassifiziert, dass er leicht die für ihn wichtigen Informationen finden kann. Zweitens: die für ihn relevante Information ist so aufbereitet, dass ihm der Informationsgehalt leicht zugänglich ist.

Richten wir den Fokus auf die Welt der Unternehmen, sind wir mit der Aussage konfrontiert, dass Information neben Arbeit, Boden und Kapital als vierter Produktionsfaktor¹² gilt. Wenn ein Manager Information als strategische Waffe einsetzen möchte, ist es von bedeutender Wichtigkeit, dass er auf diese Information schnell zugreifen kann und er sie intuitiv versteht. Den schnellen Zugriff gewährt ihm das Internet oder das firmeneigene Intranet. Aber er steht mit der Information alleine da. Der Autor eines Dokuments oder einer Grafik wird ihm nicht erklärend zur Seite stehen, um die Information zu interpretieren.

¹² [Sapinfo02]

An diesem Punkt wird die Bedeutung der Informationsvisualisierung greifbar. Informationen entfalten ihren Nutzen erst völlig, wenn sich der Informationsgehalt aus sich selbst erschließt. Nicht umsonst gilt die Redensart:

Ein Bild sagt mehr als tausend Worte.

Dennoch darf Informationsvisualisierung nicht als Selbstzweck gesehen werden. Grafik, Farbe und Animation dürfen nicht die Information, die sie repräsentieren, in den Hintergrund drängen.

Das Ziel der Informationsvisualisierung ist es, „geeignete visuelle Repräsentationen einer gegebenen Datenmenge zu erzeugen, um damit eine effektive Auswertung zu ermöglichen“.¹³ Die visuelle Repräsentation dient als Hilfsmittel, die dem Betrachter die Möglichkeit gibt, nicht nur das Äußere der Daten zu sehen, sondern einen Blick auf die inneren Zusammenhänge zu werfen. Dadurch wird er in die Lage versetzt, den Datengehalt zu verstehen und zu bewerten¹⁴.

2.4 Anforderungen an Informationsvisualisierung

Wir haben gesehen, dass Information eine wichtige Voraussetzung für betriebswirtschaftliche Entscheidungen sein kann und dass die Informationsvisualisierung ein Hilfsmittel mit großer Bedeutung in diesem Prozess ist. Welche Anforderungen muss sie erfüllen, um diesem Anspruch gerecht zu werden?

Schumann und Müller definieren drei Anforderungen, denen Informationsvisualisierung genügen muss: Expressivität, Effektivität und Angemessenheit¹⁵.

Von Expressivität wird gesprochen, wenn die Datenmenge unverfälscht ist, d.h. die Visualisierung enthält ausschließlich Daten aus der darzustellenden Datenmenge. Die Struktur und die Art der Daten sind Grundvoraussetzung für die Expressivität.

Effektivität umfasst einerseits die Voraussetzung, die ein Betrachter mitbringt und andererseits die charakteristischen Eigenschaften des Ausgabemediums. Beides muss

¹³ [Schumann00]

¹⁴ [Schumann00]

¹⁵ [Schumann00], [Müller97]

berücksichtigt werden. Weiterhin dürfen Zielsetzung und der Anwendungskontext bei der Visualisierung nicht außer acht gelassen werden.

Unter Angemessenheit wird verstanden, dass der Aufwand der Visualisierung in der Herstellung sowie bei der Betrachtung und Interpretation dem Nutzen entsprechen sollten.

2.5 Informationsvisualisierung im Visualisierungsprozess

In der relevanten Literatur sind verschiedene Modelle des Visualisierungsprozesses zu finden, die unterschiedliche Schwerpunkte aufweisen. An dieser Stelle Ausgangsmodell vorgestellt, das auf den Ideen von [Schumann00] basiert und für die Diplomarbeit entwickelt wurde.

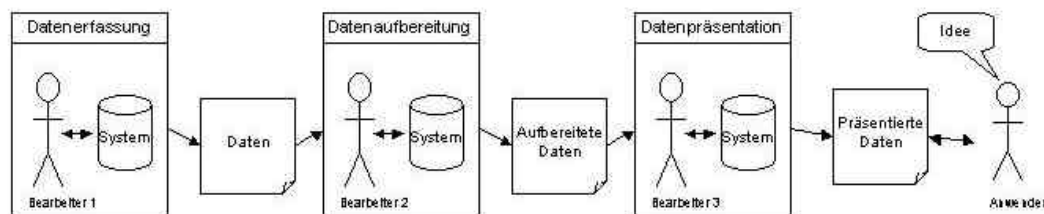


Abbildung 4: Visualisierungsprozess

Wie in der Abbildung oben zu sehen ist, läuft der Visualisierungsprozess, wie er bisher gehandhabt wird, in drei Phasen ab: Datenerfassung, Datenaufbereitung, Datenvisualisierung.

Die Datenerfassung beschäftigt sich mit dem Gewinnen der Rohdaten. Das können Daten aus Computerberechnungen oder Daten aus Messungen sein. Die Datenaufbereitung ist der Prozess der Vervollständigung, der Reduktion oder aber des Filterns der Datenmenge. Die Datenvisualisierung beinhaltet den Übergang der Daten, die nichtgeometrisch vorliegen zu einer geometrischen Abbildung mit spezifischen Attributen.

Jede Phase beinhaltet eine Interaktion zwischen einem Menschen und einem System, wobei die Daten nach einem bestimmten Modell bearbeitet werden. Genau genommen ist jeder Bearbeitungsschritt schon eine Interpretation der Daten. Der Anwender kann

seine Idee nur aus den Interpretationen, die ihm zur Verfügung gestellt werden, entwickeln. Man kann sich die Frage stellen, ob hier nicht schon der Anspruch der Expressivität, die Schumann und Müller fordern, verloren geht.

3 Techniken der Informationspräsentation

Die Beschäftigung mit dem Themenbereich der Informationspräsentation und dem in Bezug setzen zu den statistischen Daten, die innerhalb dieser Arbeit exemplarisch visualisiert werden, hatte zum Ziel Präsentationstechniken zu erkennen, die für eine softwaretechnische Umsetzung der gegebenen Daten geeignet sind.

Zur Darstellung von Informationen ist, wie schon in den vorangegangenen Kapiteln deutlich wurde, die Technik ihrer Präsentation entscheidend. In diesem Kapitel werden ausgewählte traditionelle Techniken beschrieben. Darüber hinaus wird eine Einschätzung dafür gegeben, welcher Grafiktyp für welche Daten am geeignetsten erscheint.

Um Daten nicht zu verfälschen und dem Anwender einen möglichst leichten Zugang zu dem Inhalt der Daten zu gewähren, ist es notwendig, sich an bestimmte statistische Regeln¹⁶ zu halten.

- Die Grafik ist einfach und übersichtlich-
- Die Wahl der Darstellungsart ist auf die Daten abgestimmt.
- Ein Titel erklärt den Sachverhalt, den die Grafik darstellt.
- Die grafische Darstellung ist für sich allein verständlich.
- Es gibt eine klare Beschriftung aller wichtigen Bestandteile (Achsen, Einheiten, Maßstab) einer Grafik.

¹⁶ [Riedwy187]

Bleichmann nimmt folgende Unterteilung von Grafiken vor¹⁷.

Tabelle 1: Grafikarten

Geschäftsgrafiken (bis zu 3 Dimensionen)	Grafiken für mehrdimensionale Werte	Strukturdarstellungen
<ul style="list-style-type: none">- Balkendiagramm- Säulendiagramm- Kurvendiagramm- Punktediagramm- Strukturdiagramm- Verbunddiagramm	<ul style="list-style-type: none">- Netzdiagramm- Chernoff-Faces- Hyberbox- Parallele Koordinaten	<ul style="list-style-type: none">- Baum- Netz- Landkarte

¹⁷ [Bleichmann et al.00]

3.1 Stabdiagramme

Mit Stabdiagramme, zu denen Säulendiagramme (Column Charts) und Balkendiagramme (Bar Charts) zählen, erhält man eine gute Darstellung für Zeitreihen und Verhältniszahlen. Balkendiagramme ordnen die Stäbe horizontal an, Säulendiagramme vertikal. Beide Typen bieten die Möglichkeit, Vergleiche zwischen den Werten einzelner Objekte zu ziehen oder die Entwicklung einer Wertänderung innerhalb eines bestimmten Zeitraumes zu betrachten. Möchte man das Verhältnis von Werten zu einem Ganzen betrachten, bieten sich gestapelte Stabdiagramme an.

Bei der Erstellung von Stabdiagrammen sollte man folgende Gesichtspunkte beachten:¹⁸

- Ordinate liegt bei Null
- Gleiche Breite für alle Stäbe
- Fehlende Werte müssen erwähnt werden
- Problem einer dreidimensionalen Darstellung: Vergleich von Volumina verzerrt die Darstellung
- Kein Schraffieren der Stäbe, da die Größe durch diese Technik verzerrt wird

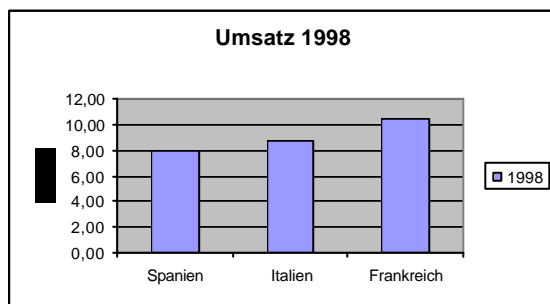


Abbildung 5: Säulendiagramm

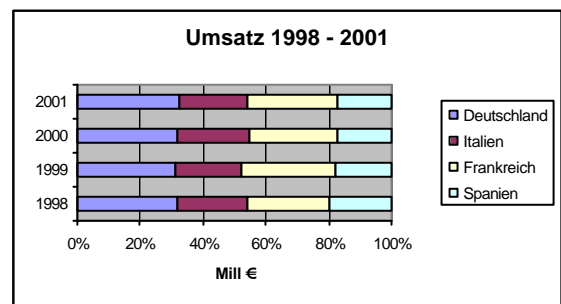


Abbildung 6: Gestapeltes Balkendiagramm

¹⁸ [Riedwyl87], [Krämer01]

3.2 Kreisdiagramme

Kreisdiagramme werden auch Pie Charts genannt und eignen sich am besten für die Darstellung von Gliederungszahlen oder prozentualen Anteilen einer Gesamtmenge.

Um aussagekräftige Kreisdiagramme zu erhalten, sollte man folgende Aspekte berücksichtigen¹⁹:

- Gesamtmenge der Daten ist ersichtlich
- Kreissektoren werden ab dem Startpunkt 12:00 Uhr im Uhrzeigersinn abgetragen
- Nur zwei Segmente lassen viele Täuschungsmanöver zu über die tatsächliche Größe der Segmente zu
- Perspektivische Darstellung verzerrt das Verhältnis der Sektoren

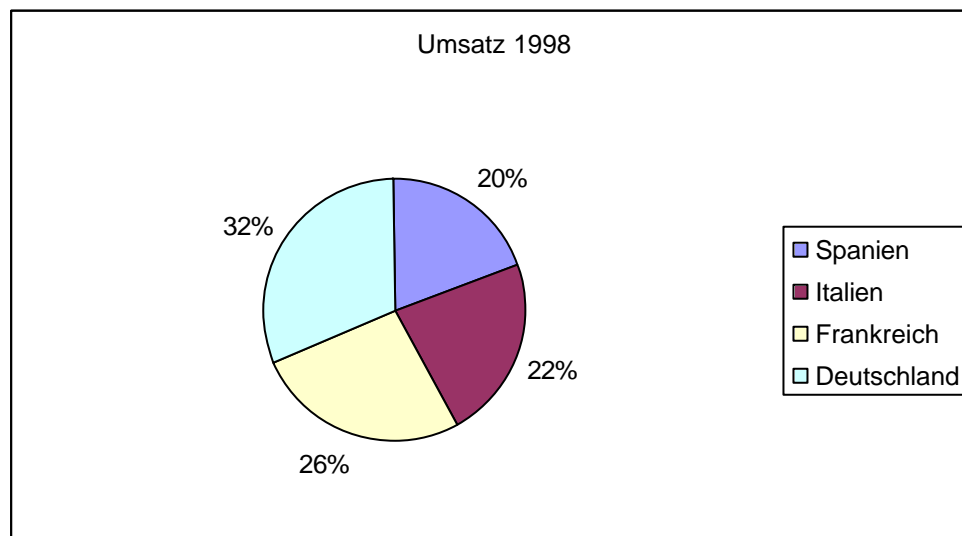


Abbildung 7: Kreisdiagramm

¹⁹ [Riedwyl87], [Krämer01]

3.3 Liniendiagramme

Liniendiagramme werden meist dazu verwendet, nicht exakte Werte sondern Trends aufzuzeigen. Mit dieser Art Diagramm lassen sich eine große Anzahl zeitabhängiger Daten darstellen. Außerdem ist der Vergleich von Datenserien leicht möglich.

Als Ergänzung zum Liniendiagramm soll hier das Flächendiagramm erwähnt werden, bei dem die Fläche unter der Linie farblich aufgefüllt wird. Wird nur ein Linienzug dargestellt, ist ein Flächendiagramm anschaulicher als ein Liniendiagramm. Ein Vergleich mehrerer Linienzüge lässt sich durch dreidimensionale Flächendiagramme sehr anschaulich darstellen.

Auch für diese Diagrammart gibt es einige erwähnenswerte Regeln für das Design²⁰:

- Beschriftung des Koordinatensystems an der Abszisse und Ordinate
- Keine Verzerrung des Maßstabs
- Ordinateneinheit wird einheitlich angegeben

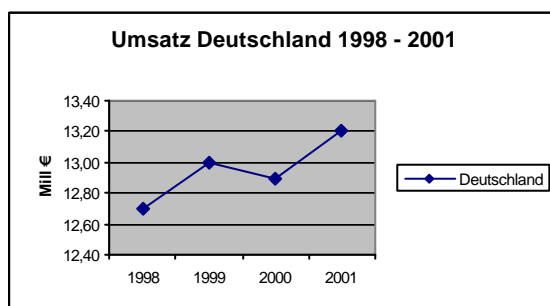


Abbildung 8: Liniendiagramm 1

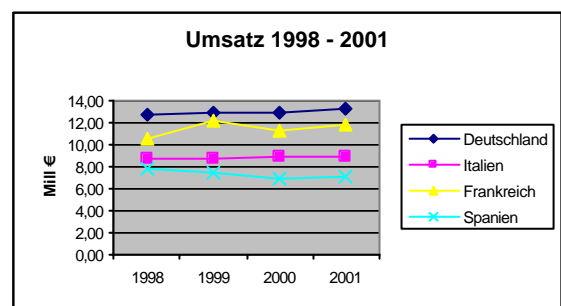


Abbildung 9: Liniendiagramm 2

²⁰ [Riedwyl87], [Krämer01]

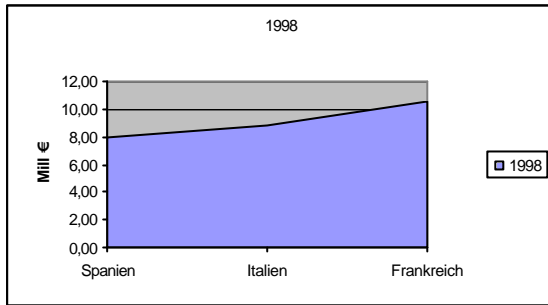


Abbildung 10: Flächendiagramm 1

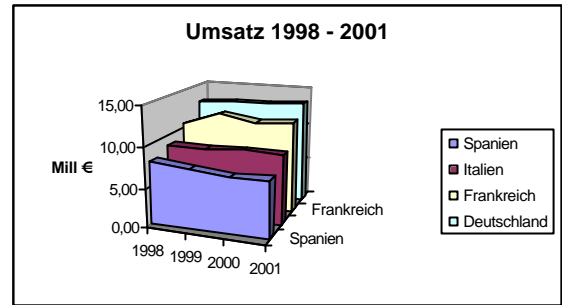


Abbildung 11: Flächendiagramm 2

3.4 Piktogramme

Piktogramme verwenden Figuren anstelle geometrischer Elemente wie Linien, Rechtecke, Kreise oder Flächen um statistische Sachverhalte darzustellen. Beliebt ist diese symbolische, leicht verständliche Darstellung besonders für Unterrichtszwecke und in öffentlichen Medien.

Bei der Darstellung sollte vor allem folgender Aspekt beachtet werden:

- Vielfachprinzip: nicht 1:3, sondern 100:300

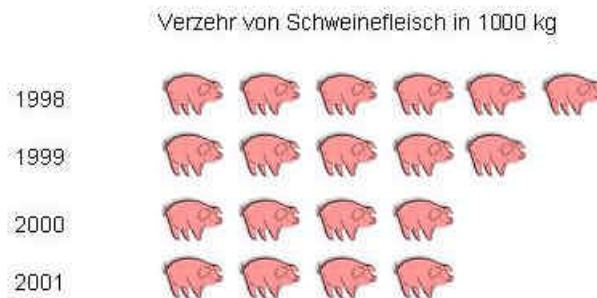


Abbildung 12: Piktogramm

4 Interaktive Informationsvisualisierung

Das vorige Kapitel beschreibt traditionelle Techniken der Visualisierung. Ihnen ist gemein, dass die Darstellungsform „statisch“ ist, d.h. es kann jeweils nur ein Aspekt einer Datenmenge dargestellt werden. Eine Darstellung verschiedener Aspekte ist innerhalb einer Abbildung schwer möglich. In der Auseinandersetzung mit dem Themenbereich wurde die Dringlichkeit eines interaktiven Moments immer deutlicher. Für eine Vertiefung in diese Problematik beschäftigt sich das vierte Kapitel mit Interaktivität. Aus den behandelten Ansätzen und Realisierungen konnten für die Umsetzung in eine Applikation innerhalb dieser Arbeit Techniken identifiziert werden, die in dem Visualisierungstool VisJex eine Umsetzung erfahren.

Informationsvisualisierung hat das Ziel, nicht-geometrische Daten in Bilder zu verwandeln, die das Verständnis des Inhalts einer Datenmenge erleichtern. Interaktive Informationsvisualisierung verfolgt darüber hinaus das Ziel, Datenräume durch geeignete Hilfsmittel zu erforschen. Im Einzelnen werden die Vorteile von Interaktiver Informationsvisualisierung wie folgt beschrieben²¹:

- durch eine schnelle Interaktion wird der Prozess des Verstehens von Daten gefördert
- in einer vorgegebenen Zeit können mehr Möglichkeiten erforscht werden
- bei einer Veränderung von Parametern können Auswirkungen direkt und unmittelbar betrachtet werden.

In der wissenschaftlichen Gemeinde, die sich mit Interaktiver Visualisierung beschäftigt, werden folgende Gesichtspunkte diskutiert: Wie schnell muss Interaktion sein? Welche Aspekte kontrolliert Interaktion? Kann Interaktion von Visualisierung getrennt werden und wenn ja, wie? Welche Techniken existieren, wie kann man sie weiter entwickeln und verbessern?

Die Antworten auf diese Fragen ändern sich mit der Geschwindigkeit, in der die bisher verwendete Technologie voran schreitet.

²¹ [Shneiderman92]

Shneidermann kategorisiert Interaktionstechniken nach dem Kriterium, das modifiziert werden soll.

Tabelle 2: Interaktionstechniken²²

Modifies Data Transformations	Modifies Visual Mappings	Modifies View Transformations
Dynamic Queries	Dataflow	Direct selection
Direct Walk	Pivot tables	Camera movement
Details-on-demand		Magic lens
Attribute walk		Overview and detail
Brushing		Zoom
Direct manipulation		

In den folgenden Kapiteln werden einige Techniken aus der oben dargestellten Tabelle 1 näher beschrieben.

4.1 Dynamic Queries

Der Begriff „Dynamic Queries“ wurde 1992 von Williamson und Shneiderman das erste Mal in einer Abhandlung beschrieben. Dynamic Queries²³ ist die visuelle und intuitive Antwort auf herkömmlichen SQL-Abfragen. Jede Abfrage-Sprache erfordert hohe Einarbeitungs- und Lernzeiten vom Anwender, um durch korrekte Syntax und Semantik zu einem Ergebnis zu gelangen.

²² [Shneiderman92]

²³ = visuelle Abfragen

4.1.1 SQL-Abfragen

Betrachtet man herkömmliche Methoden, um Daten durch Abfragen aus einem großen Datenbestand zu selektieren, lässt sich folgendes Prinzip von zwei Arbeitsschritten feststellen:

- Eine Abfrage mit Parametern wird formuliert und auf den Datenbestand angewandt.
- Die so gefilterten Daten werden meist in simpler Tabellenform dargestellt.

Die unten stehende Abbildung soll diesen Prozess noch einmal verdeutlichen.

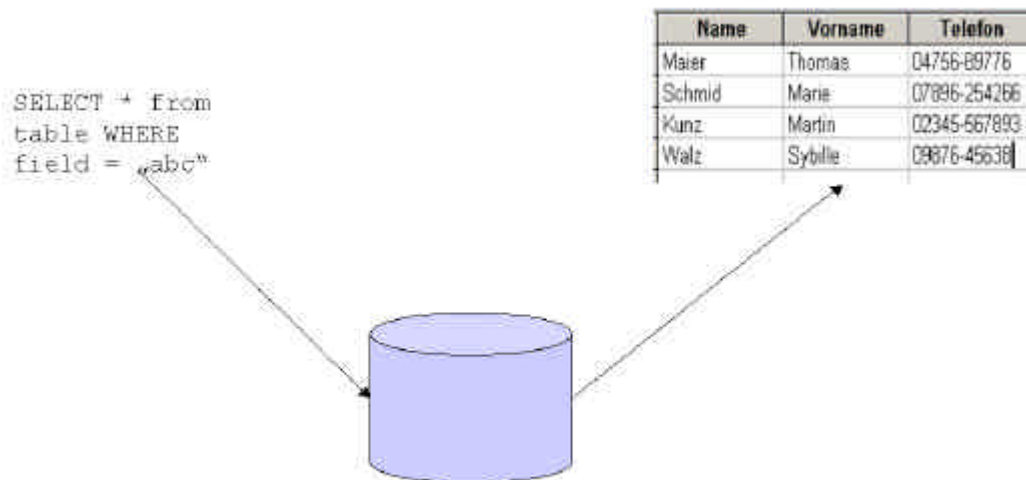


Abbildung 13: SQL-Statements

Der Vorteil der beschriebenen Methode liegt darin, dass sehr spezifische und leistungsfähige Abfragen gestaltet werden können, wenn der Anwender über das entsprechende Know-How verfügt.

Dennoch lassen sich auf den ersten Blick mehrere Nachteile erkennen. Wie oben genannt muss der Anwender Zeit investieren, um die Abfragesprache zu erlernen. Der Aufwand, um exakte Abfragen zu schreiben ist nicht unerheblich. Um sinnvolle Abfragen schreiben zu können, muss der Anwender die Struktur der Datenbank mit ihren Besonderheiten genau kennen. Aufgrund zu starker oder zu schwacher Einschränkung der Parameter können die Suchergebnisse unbrauchbar sein, weil keine, zu wenige oder sehr viele Daten als Ergebnis geliefert werden.

Das Ergebnis von traditionellen Abfragemethoden wird in verschiedener Form dargestellt. Meist sind es einfache Tabellen oder Masken, wobei eher technische Aspekte und weniger die Anforderungen des Anwenders das Layout bestimmen. Eine weitere beliebte Darstellungsform sind Berichte. Hier liegt nicht mehr reines Datenmaterial vor sondern eine Interpretation desselben.

Die dargestellten Probleme und Schwachstellen der traditionellen Techniken lassen sich teilweise durch Dynamic Queries lösen. Dynamic Queries bieten die Möglichkeit, sich vom zeitaufwändigen und fehlerträchtigen Suchen zu lösen, um auf spielerische Art und Weise in einem Datenraum auf Entdeckungsreise zu gehen.

4.1.2 Dynamic Queries

Die Idee der Dynamic Queries macht es möglich, auf Benutzeroberflächen Eingabe- und Ausgabebereich nicht mehr gesondert darstellen zu müssen. Oft ist die Funktionalität der beiden Bereiche nicht klar getrennt. Man spricht dann von „Tight-Coupling“. Bei Tight-Coupling kann ein Ausgabebereich dergestalt manipuliert werden, dass er sofort eine Eingabe verarbeitet und darstellt, ohne dass in einem speziellen Eingabebereich Suchparameter eingegeben werden müssen. Anders ausgedrückt ist der Ausgabebereich nicht mehr nur ein Ausgabebereich, sondern wird um die Funktion der Eingabe erweitert.

Meist werden aber hinreichend bekannte Steuerelemente einer grafischen Benutzeroberfläche wie Schalter, Auswahllisten und Schieberegler als Eingabeelemente benutzt. Neuere Entwicklungen wie Alpha Slider, Data Visualization Slider und MagicLenses als Elemente zur Eingabe können eine Anwendungsoberfläche intuitiver gestalten.

Bei einer Anwendung, die auf der Idee der Dynamic Queries basiert, unterscheiden sich die Arbeitsschritte völlig von einer herkömmlichen Anwendung. Man kann sich vorstellen, dass beim Start standardmäßig eine Datenmenge in einer geeigneten Darstellungsform angezeigt wird. Der Anwender erkundet und selektiert diese Datenmenge mit Hilfe der Eingabeelemente. Dynamic Queries geben dem Anwender ein intuitives Werkzeug in die Hand, mit der er die gewünschte Selektion aus der gegebenen Datenmenge einfach und schnell bewerkstelligen kann. SQL-Abfragen tolerieren keine Fehler. Dieses Problem taucht bei Dynamic Queries prinzipiell nicht auf, da der Anwender nur aus korrekten Parametern wählen kann und sich keine

Gedanken über die richtige Eingabe dieser Parameter machen muss, da sie zum Beispiel durch einen Schieberegler mit einer Unter- und Obergrenze fest vorgegeben sind.

Als Demonstration einer solchen Abfragetechnik wird in einem kleinen Exkurs der „FilmFinder“ kurz vorgestellt, der Anfang der neunziger Jahre in einer Forschungsgruppe um Ben Shneiderman als Pilot-Projekt entstanden ist.

4.1.3 FilmFinder

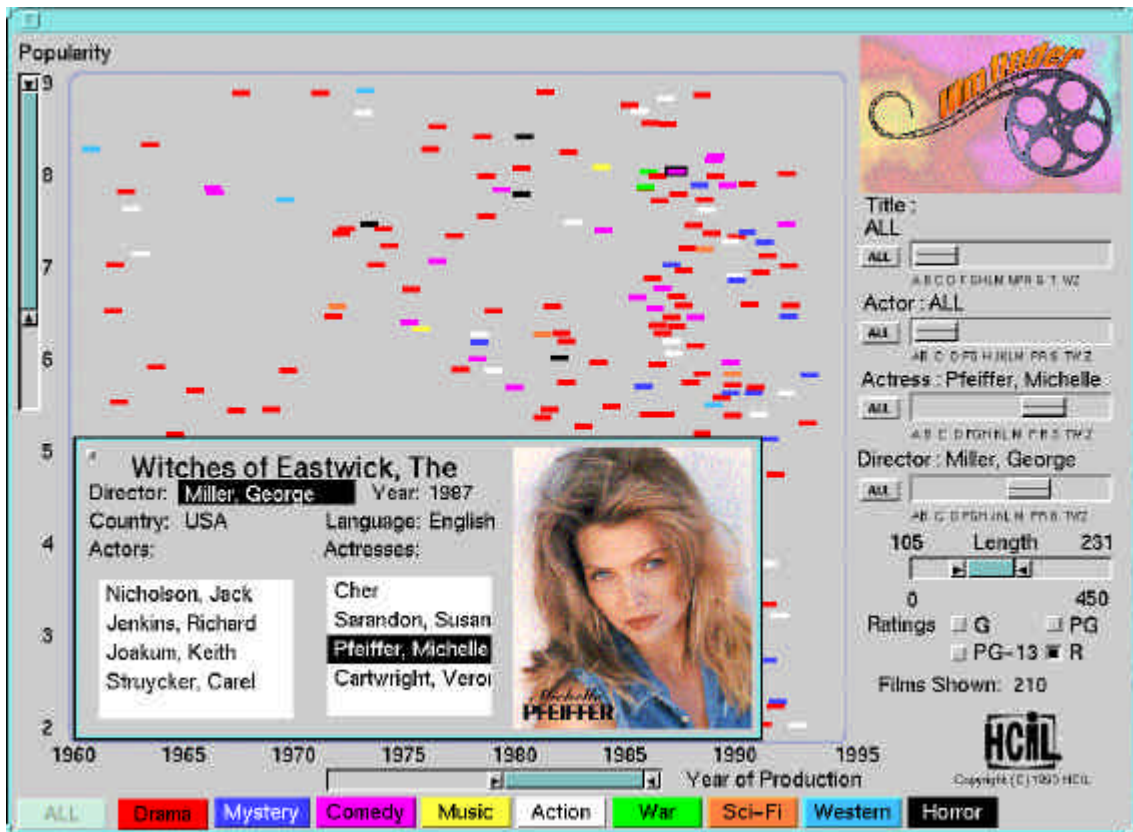


Abbildung 14: FilmFinder²⁴

Der Bildschirm des FilmFinder ist grob in die zwei großen Bereiche Eingabe und Ausgabe unterteilt, wobei der Ausgabebereich in der Mitte liegt und die verschiedenen Eingabebereiche um den Ausgabebereich herum platziert wurden.

Verschiedene Techniken werden zur Verfügung gestellt, um eine Selektion zu erreichen. Links des Ausgabebereiches sieht man einen Schieberegler (Popularity), der Filme nach Popularität anzeigt. Unterhalb des Ausgabebereiches können durch verschiedenfarbige Schalter Filme nach Genre gewählt werden. Ein weiterer Schieberegler (Year of Production) erlaubt die Selektion nach Produktionsjahr des Filmes. Auf der rechten Seite des Ausgabeteils kann aus der Gesamtdatenmenge sehr differenziert nach Titel, Schauspieler und weiteren Kriterien durch Schieberegler

²⁴ [Sony CSL 2002]

selektiert werden. Durch diese Technik ist es fast unmöglich, dass der Anwender einen einzelnen falschen Wert auswählt, da nur existierende Werte innerhalb eines Bereichs mit einer unteren und oberen Grenze vorgegeben werden.

Der Ausgabebereich ist mit kleinen farbigen Rechtecken übersät, die einzelne Filme repräsentieren. Diese Art der Darstellung wird „Starfield Display“ genannt. Die farbige Kennzeichnung korrespondiert mit den Farben der Schalter, die unterhalb des Ausgabebereichs zu sehen sind. Damit erfolgt eine Zuordnung der Filme zu dem jeweiligen Genre.

Im unteren Teil des Ausgabebildschirm ist ein weiterer Rahmen zu sehen, der Details zu einem selektierten Filmtitel anzeigt, sobald dieser durch einen Mausklick ausgewählt wurde. Die hier verwendete Technik nennt man „Details-on-Demand“.

An dem Beispiel des FilmFinder kann man die Prinzipien direkter Manipulation²⁵, wie Shneiderman sie beschreibt, gut erkennen und es ist sofort klar, dass die Technik der „Dynamic Queries“ die Forderungen an eine benutzerfreundliche Visualisierung erfüllen:

- Komponenten der Abfrage werden visuell dargestellt
- Ergebnisse der Abfrage werden visuell dargestellt
- Rasche, zunehmende und reversible Kontrolle der Abfragen ist möglich
- Auswahl erfolgt durch Zeigen nicht durch Eintippen.
- Sofortiges und anhaltendes Feedback ist gewährleistet

²⁵ [Shneiderman92]

4.2 MagicLens TM

Anfang der 90er Jahre beschäftigten sich Fishkin und Stone mit Erweiterungen der Dynamic Queries. Sie entwickelten eine Methode, die als „Movable Filters“ bekannt ist. Durch Kombination von Movable Filters und der Technik des Starfield Display schufen sie die MagicLenses. Starfield Displays unterstützen interaktives Filtern und Zoomen auf Scatterplot Displays²⁶, während Movable Filters mannigfaltige gleichzeitige visuelle Umformungen und Abfragen auf Daten unterstützen.

Die Technik der MagicLenses stellt man sich am einfachsten so vor, als würde man einen Text lesen und eine spezielle Lupe für die Worte zu Hilfe nehmen, deren Bedeutung man nicht kennt. Die Technik verlangt also zwei Ebenen:

- Ebene 0, auf der die Daten liegen (Starfield Display)
- Ebene 1, auf der die Filtertechnik interagiert (Movable Filters).

Movable Filters setzen Operanden von herkömmlichen Abfragestatements visuell um. Das heisst, die Linse wird mit Schaltern und Schieberegler so versehen, dass die Werte der Filterfunktion kontrolliert und die Verknüpfungsoperationen definiert werden können. Zusammengesetzte Abfragen können ganz leicht erreicht werden, indem man mehrere Movable Filters übereinander legt.

²⁶ Anzeigen von Daten als Punkte in einem zweidimensionalen Raum

Das unten stehende Beispiel in Abbildung 7 macht die Wirkungsweise noch einmal deutlich. Das verschiebbare Dreieck färbt die darunter liegenden Polygone gelb ein, während das verschiebbare Quadrat die unter ihm liegenden Farben der geometrischen Formen dunkler anzeigt. Kombiniert man beide Linsen werden die Flächen, die von beiden Linsen bedeckt sind, in einem dunkleren Gelb eingefärbt.

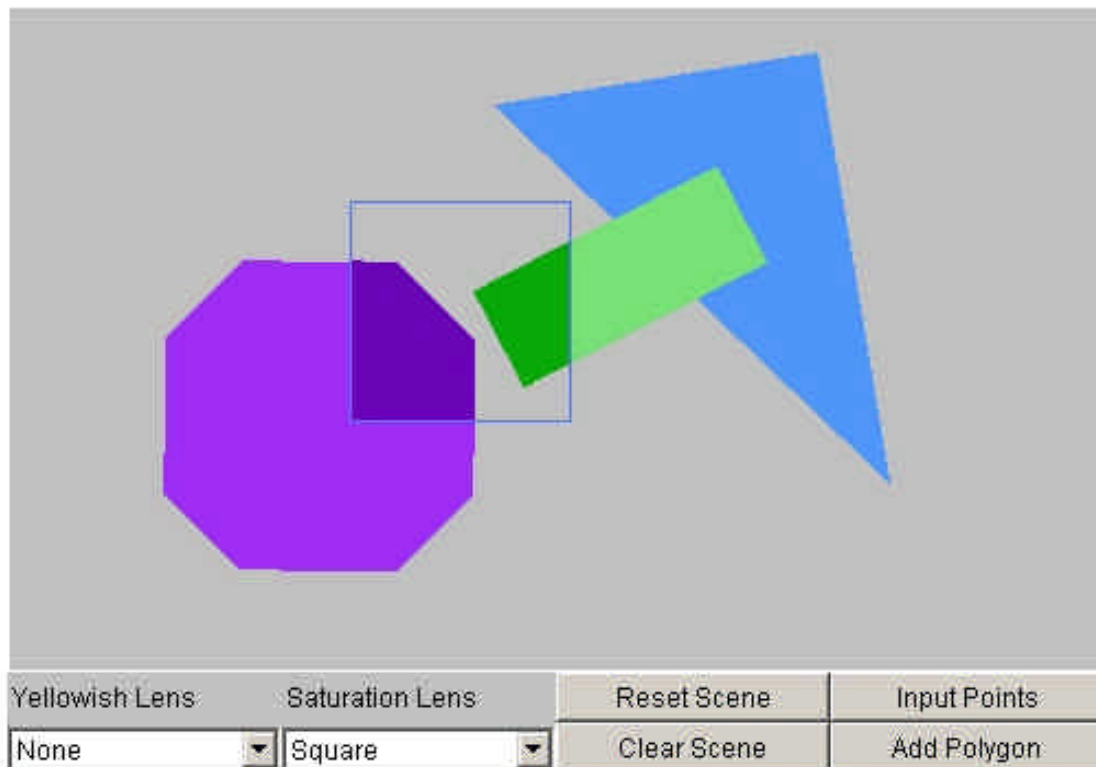


Abbildung 15: Demo Magic Lens²⁷

Die Vorteile der Technik der MagicLens liegen nach Shneiderman²⁸ und Spence²⁹ in den folgenden Aspekten.

Bestechend ist natürlich die Attraktivität der Metapher, die die MagicLens umsetzt. Sie appelliert direkt an die kindliche Seite des menschlichen Wesens: ein magisches Glas, das uns Einblick in das geheimnisvolle Innere einer Datenmenge gewährt.

²⁷ [Parc02]

²⁸ [Shneiderman92]

²⁹ [Spence00]

Daneben gibt es natürlich weitaus pragmatischere Argumente. Mit MagicLens steht ein direkter Manipulationsmechanismus für allgemeine boolesche Abfragen zur Verfügung, der mannigfaltige simultane Ansichten gewähren kann. MagicLens bietet eine einheitliche Technik, um alternative Sichten auf die Daten zu unterstützen. Daten müssen bei dieser Technik nicht zu dem Hilfsmittel gebracht werden, sondern umgekehrt wird das Hilfsmittel zu den Daten gebracht. Der Bildschirmausschnitt für Detailinformationen ist begrenzt, sodass nicht die Gefahr einer Detailschwemme besteht. Durch ein nichtverändertes Umfeld um die Detailansicht und die Linse herum, wird der wertvolle Kontext der Daten erhalten.

4.3 Overview and Detail

Unter dem Begriff „Overview and Detail“ versteht man eine Interaktion, die einerseits Sicht auf die Gesamtheit einer Datenmenge gewährt, während gleichzeitig die Analyse einer Teilmenge der Daten erfolgen kann.

Bei sehr großen Datenmengen ist es von besonderer Wichtigkeit, den Blick für die Gesamtmenge der Daten zu behalten. Dieser Blick reduziert den Suchaufwand, erlaubt das Erkennen von Mustern und unterstützt den Anwender darin, die nächsten Schritte zu wählen. Kurz gesagt erlaubt Overview and Detail einem Anwender den Überblick zur Orientierung und die Detailansicht für weiteres Arbeiten.

Man unterscheidet zwischen „spatial zooming“, bei dem die Detailansicht die Vergrößerung einer Teilmenge der Übersicht ist und „semantic zooming“, bei dem der Inhalt der Ansichten gleich ist, das Erscheinungsbild der Daten sich aber ändert. Ein Beispiel für semantic zooming wären Landkarten.³⁰

An dem Produktbeispiel „LifeLines“, das in der unten stehenden Abbildung mit einem Screenshot gezeigt wird, werden die Vorteile von „Overview and Detail“ erläutert.

³⁰ [Shneiderman92]

4.3.1 LifeLines

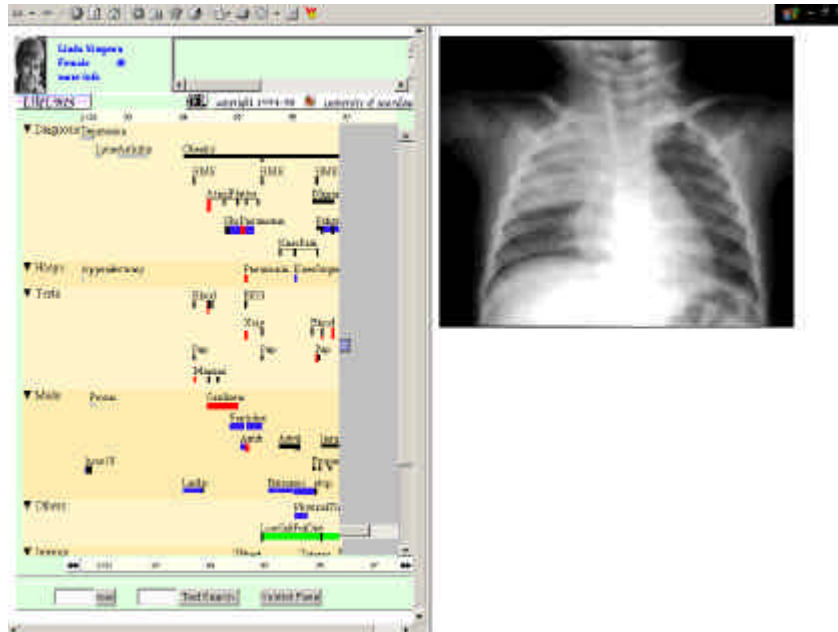


Abbildung 16: LifeLines³¹

Durch die Technik von „Overview and Detail“, die in dem Produkt LifeLines zur Anwendung kommen, wird eine allgemeine Visualisierungsumgebung für Personengeschichte geschaffen, wie sie zum Beispiel in der Medizin für die Krankheitsgeschichte von Patienten verwendet werden kann. Die Vorteile, die eine solche Umsetzung bieten, liegen in diversen Aspekten:

Der Anwender erhält einen Überblick an seinem Bildschirm, gleichzeitig werden mannigfaltige Facetten auf verschiedene Protokollierungen gezeigt. Stellt man sich unser Beispiel von Krankengeschichten in einem herkömmlichen Sinne vor, lägen die verschiedenen Daten wie Röntgenbilder, Laborergebnisse oder Medikationen bei unterschiedlichen Ärzten oder Fachärzten und müssten erst mühsam zusammengetragen werden.

Während Zeitlinien für Zustände oder in dem gezeigten Beispiel von LifeLines für eine medizinische Verfassung stehen, symbolisieren Icons diskrete Ereignisse, in obigem Beispiel Arztbesuche. Linienfarbe und Dicke der Zeitlinien illustrieren Wichtigkeit und Beziehungen zwischen verschiedenen Vorgängen.

³¹ [CS02]

Die hier vorgestellte Technik reduziert die Möglichkeit, wichtige Informationen zu verlieren oder nicht zu berücksichtigen, wie es bei dem oben beschriebenen Szenario der herkömmlichen Verwaltung einer Patientenakte vorstellbar ist. Durch die Darstellung verschiedener Gesichtspunkte auf ein Objekt, hier der Patient, ist es leichter, Anomalien, Trends oder Zusammenhänge zu erkennen.

Zusammenfassend lässt sich sagen, dass ein fortschreitender Zugriff auf Details, während der Blick auf die Gesamtheit der Informationen gewährt bleibt, eine gut informierte Entscheidung erleichtert. Als Vergleich kann man sich ein Puzzle vorstellen, dessen Bild immer deutlicher wird, je mehr Einzelteile auf dem Tisch liegen.

5 Beschreibung und Aufbau des Visualisierungstools

Die Erkenntnisse, die durch die Beschäftigung mit den Themengebieten Visualisierung, Techniken der Informationspräsentation und Interaktivität gewonnen wurden, erfahren eine teilweise Umsetzung in dem Visualisierungstool VisJex.

Im Einzelnen sind das folgende Aspekte:

Darstellungsform

Um eine Realisierung der Anwendung innerhalb der vorgegebenen Zeit möglich zu machen, wurden zwei traditionelle Techniken der Informationspräsentation gewählt: Pie Charts und Bar Charts. Diese verschiedenen Darstellungsarten visualisieren zwei verschiedene Blickwinkel auf die gegebene Datenmenge. Die unterschiedlichen Visualisierungsformen erscheinen gleichzeitig auf dem Bildschirm, so dass die Verbindung und sich daraus ergebende Konsequenzen der Daten für den Benutzer deutlich werden können.

Interaktivität

Interaktivität wurde als eine sehr wichtige Voraussetzung erkannt, um dem Anwender eine spielerische und damit intuitive Erkundung einer Datenmenge ermöglichen zu können. In der Anwendung VisJex wird dieser Aspekt durch Zoomen, Selektierung der Daten durch Fokussierung auf eine weitere Eigenschaft der Datenmenge per Mausklick und Details-on-Demand realisiert.

Eine genauere Beschreibung der Funktionen erfolgt in den nächsten Kapiteln.

Um zu verdeutlichen, welchen Nutzen die Anwendung VisJex für den Benutzer bringt, werden an dieser Stelle verschiedene Aufbereitungen gezeigt, die statistische Daten erfahren können.

Die Daten, die mit VisJex visualisiert wurden, lagen als Quelle in einer Text-Datei vor, die im ASCII-Format gespeichert wurde. Wie in der folgenden Abbildung deutlich wird, ist die Datei unübersichtlich, schwer lesbar und ohne weitere Bearbeitung nicht interpretierbar.


```

Schluessel 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
Mitarbeiter 34 1 210 5 543 2813 108 1623 678 478 789 12 56 145 356 23
Umsatz 5 35 4 125 225 18 80 100 95 125 23 34 45 67 123 1 2 3 4 5
Zugehoerigkeit Industrie Industrie Industrie Industrie Industrie
ExterneWarenschluessel 1 1 2 3 1 1 2 3 1 1 2 3 1 1 2 3 1 1 2 3
InterneWarenschluessel 2 1 2 3 2 1 2 3 2 1 2 3 2 1 2 3 2 1 2 3
Katalogdatenaustausch ja nein nein ja nein ja nein nein ja nein j
BMEcat gut schlecht schlecht schlecht gut schlecht schlecht schlecht
ColorsZoom1 13209 16711680 16776960
ColorsZoom2 13209 10066431 16711680 16743552 16776960 16777113
ColorsZoom3 13209 6711039 10066431 13421823 16711680 16743552 16751001
ColorsYellow 13434777 13434726 13434676 13421568 16776960 16777062 1
ColorsBlue 10092492 3394815 39372 26316 13209 102 10066431 6684877 67110
ColorsRed 16724940 16732240 13369344 10813473 6684723 13382400 1673775
ColorsBarChart 10092492 3394815 39372 26316 13209 102 10066431 6684877 6

```

Abbildung 17: Source-Datei im ASCII-Format

Sieht man sich die vorliegende Datei mit dem Programm MS Excel wie in der unten stehenden Abbildung an, erkennt man eine deutliche Verbesserung. Die Zugehörigkeit der Daten ist erkennbar, die Daten sind leichter lesbar, aber eine Interpretation ist immer noch schwierig. Allerdings bietet MS Excel als Feature die Umwandlung der Daten in Grafiken an, was den Umgang mit den Daten erleichtert.

	A	B	C	D	E	F	G
1	Schluessel	1	2	3	4	5	6
2	Mitarbeiter	34	1	210	5	543	2813
3	Umsatz	5	35	4	125	225	18
4	Zugehoerigkeit	Industrie	Industrie	Industrie	Industrie	Großhandel	Industrie
5	ExterneWarenschluessel	1	1	2	3	1	1
6	InterneWarenschluessel	2	1	2	3	2	1
7	Katalogdatenaustausch	ja	nein	nein	ja	nein	ja
8	BMEcat	gut	schlecht	schlecht	schlecht	gut	schlecht
9	ColorsZoom1	13209	16711680	16776960			
10	ColorsZoom2	13209	10066431	16711680	16743552		16776960
11	ColorsZoom3	13209	6711039	10066431	13421823	16711680	16743552
12	ColorsYellow	13434777	13434726	13434676	13421568	16776960	16777062
13	ColorsBlue	10092492	3394815	39372	26316	13209	102
14	ColorsRed	16724940	16732240	13369344	10813473	6684723	13382400
15	ColorsBarChart	10092492	3394815	39372	26316	13209	102

Abbildung 18: Source-Datei in MS Excel

Als unmittelbaren Vergleich folgt in der nächsten Abbildung ein Screenshot der Anwendung VisJex, das die oben gezeigten Daten in Grafiken abbildet. Um die Interaktivität erfahrbar zu machen, ein Explorieren der Datenmenge zu erlangen, sollte das Programm direkt ausprobiert werden.

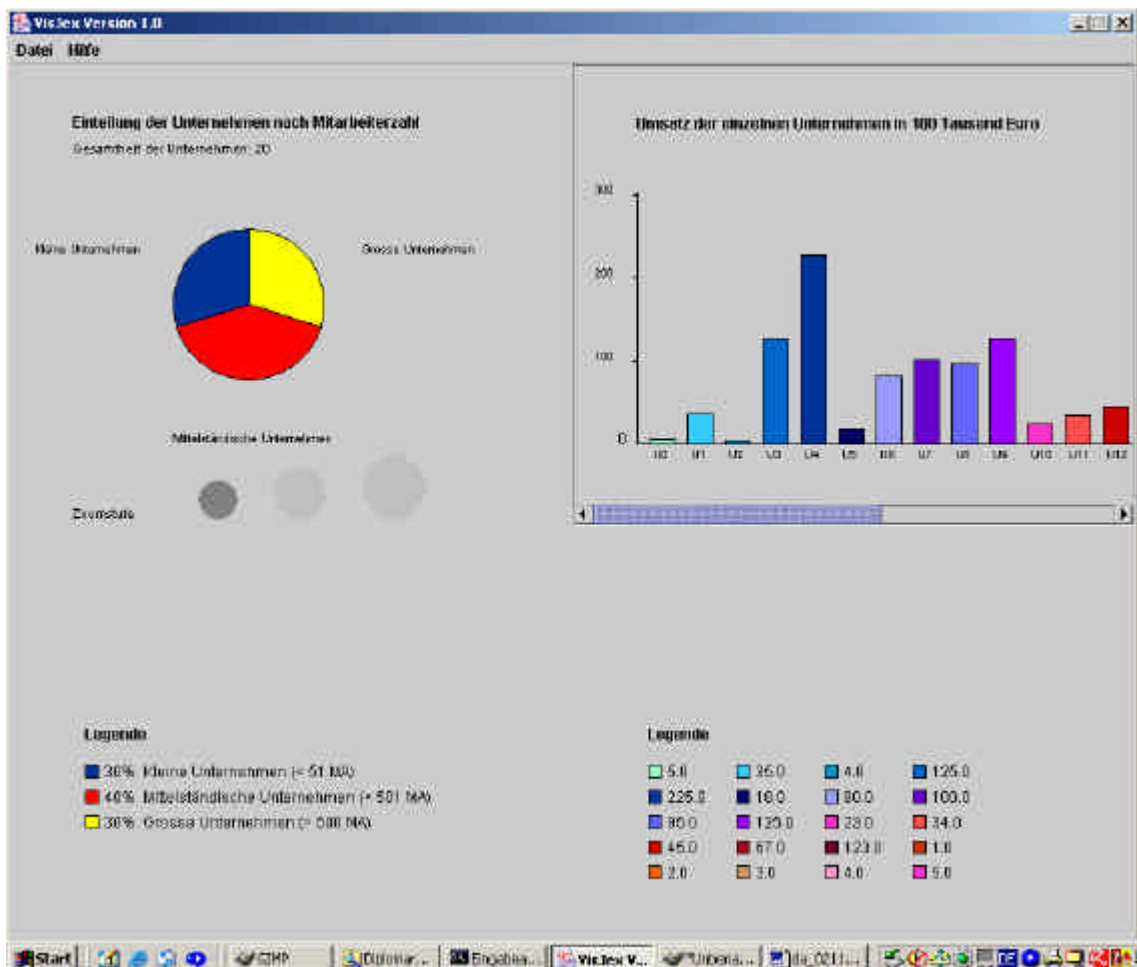


Abbildung 19: Daten aus der Source-Datei mit VisJex angezeigt

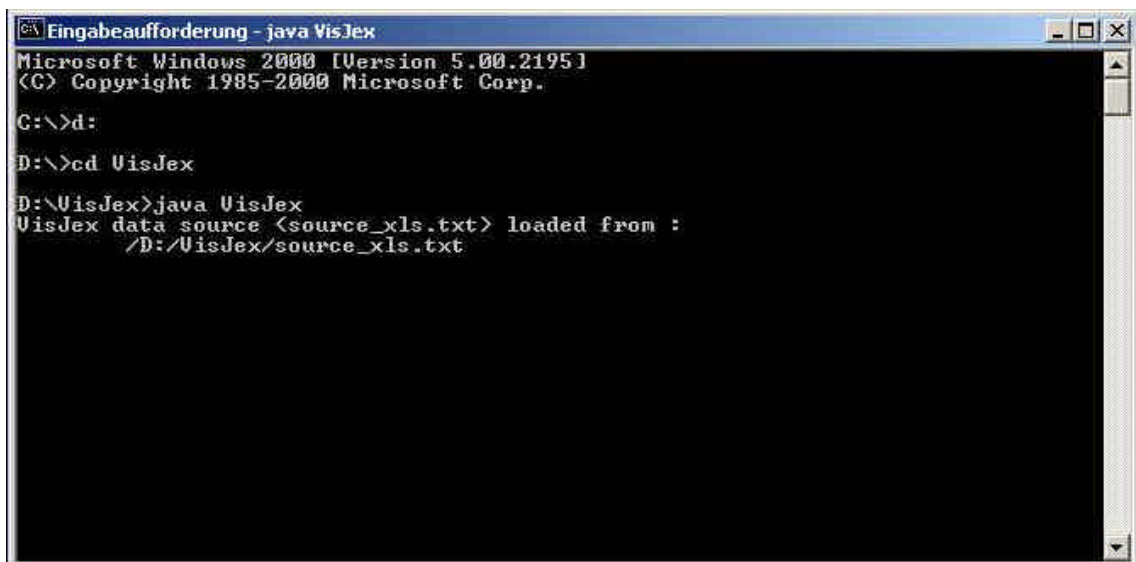
5.1 Installation und Inbetriebnahme

Um sicher zu gehen, dass die Anwendung problemlos läuft, sollte auf dem Rechner das JDK 1.4 installiert sein. Das JDK finden Sie im Verzeichnis JDK 1.4.1 auf der CD-ROM.

5.1.1 Windows

Kopieren Sie das komplette Verzeichnis VisJex auf Ihren Rechner.

Rufen Sie die DOS-Eingabeaufforderung auf und wechseln Sie in das Verzeichnis VisJex.



```
Eingabeaufforderung - java VisJex
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.
C:\>d:
D:\>cd VisJex
D:\VisJex>java VisJex
VisJex data source <source_xls.txt> loaded from :
/D:/VisJex/source_xls.txt
```

Abbildung 20: Screenshot DOS-Eingabeaufforderung

Geben Sie folgenden Befehl ein:

```
java VisJex
```

Die Anwendung VisJex startet und Sie können beginnen.

Schließen Sie die Anwendung über Datei > Beenden.

5.1.2 Unix

Kopieren Sie das komplette Verzeichnis VisJex auf Ihren Rechner.

Starten Sie die Shell und wechseln Sie in das Verzeichnis VisJex.

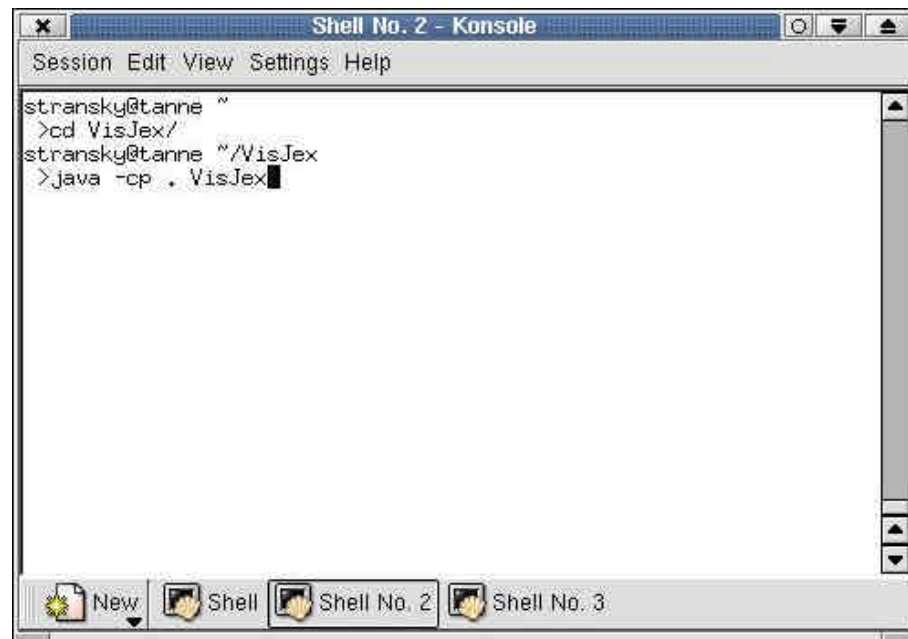


Abbildung 21: Screenshot Shell

Geben Sie folgenden Befehl ein:

```
java -cp . VisJex
```

Die Anwendung VisJex startet und Sie können beginnen.

Schließen Sie die Anwendung über Datei > Beenden

5.2 VisJex aus Anwendersicht

Die nachstehende Abbildung zeigt die Elemente der Anwendung VisJex. Im Weiteren folgt für den Benutzer eine kleine Einweisung in die Handhabung des Programms.

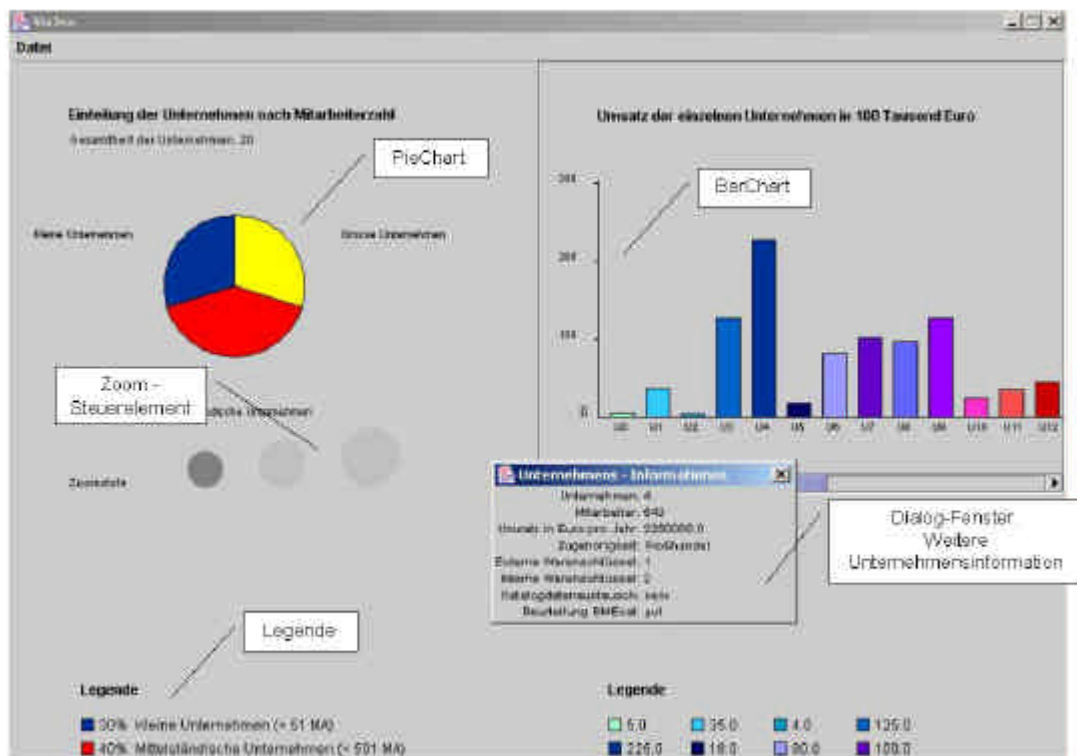


Abbildung 22: VisJex - Screenshot

5.2.1 Menü

Das Menü hat zwei Menüpunkte mit Untermenüs.

Datei

Klicken Sie auf Datei und dann auf Beenden, wenn Sie VisJex beenden möchten.

Hilfe

Brauchen Sie Hilfe bei der Benutzung von VisJex, wählen Sie den Menüpunkt Hilfe und dort das Untermenü Inhalt aus. Sie finden dort einen ähnlichen Hilfetext.

5.2.2 PieChart

Das PieChart wird beim Aufruf von VisJex mit der Datenmenge „Mitarbeiter“ gezeichnet. In der ersten Ansicht erfolgt eine Einteilung der Datenmenge in Kleine -, Mittelständische- und Grosse Unternehmen.

Das PieChart bietet zwei Funktionen:

Zoomen

Klicken Sie mit der linken Maustaste auf einen Kreis des Zoom-Steuer-elementes, um tiefer in die Datenmenge des PieCharts „einzutauchen“. Der Klick veranlasst die Anzeige des PieCharts mit neuen Daten. Zur Auswahl stehen drei Zoomstufen, die erste wird standardmäßig angezeigt.



Abbildung 23: Screenshot VisJex: Zoomstufe 2



Abbildung 24: Screenshot VisJex Zoomstufe 3

Selektieren

Um ein Segment des PieCharts zu selektieren, klicken Sie mit der linken Maustaste auf ein Segment. In der Komponente BarChart können Sie nun eine Teilmenge der Datenmenge „Umsatz“ sehen, die zu der selektierten Datenmenge der Gesamtmenge „Mitarbeiter“ gehört. Das selektierte PieSegment wird durch Herausrücken aus dem gesamten PieChart markiert.

Das Selektieren wurde aus Zeitgründen nur für die Zoomstufe eins realisiert.

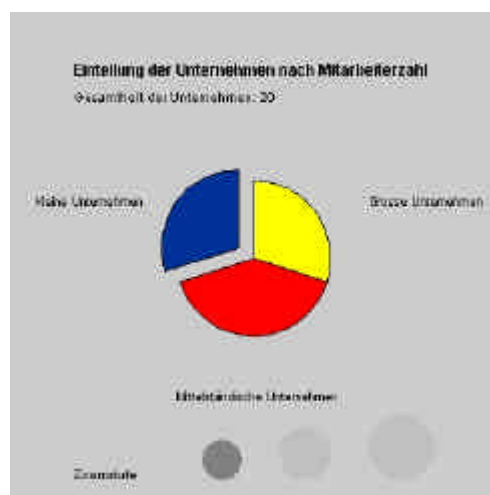


Abbildung 25: Screenshot VisJex: selektiertes PieSegment

5.2.3 BarChart

Beim Aufruf des Programms wird das BarChart mit den Daten der Gesamtmenge „Umsatz“ gezeichnet. Nach dem Selektieren einer Teildatenmenge durch Klicken auf ein PieSegment zeigt es nur noch die Daten der gewählten Teilmenge an.

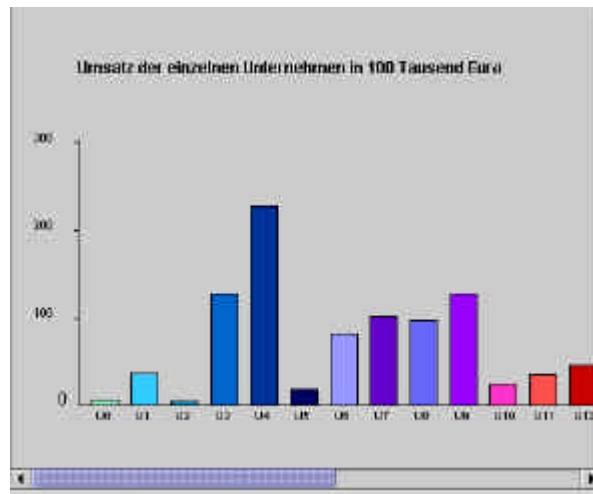


Abbildung 26: Screenshot VisJex: BarChart

Das BarChart bietet neben der Visualisierung der Daten eine weitere Funktion:

Details-on-Demand

Klicken Sie mit der linken Maustaste auf ein Bar. Es öffnet sich ein Dialogfenster mit weiteren Informationen zu dem durch Anklicken gewählten Unternehmen.

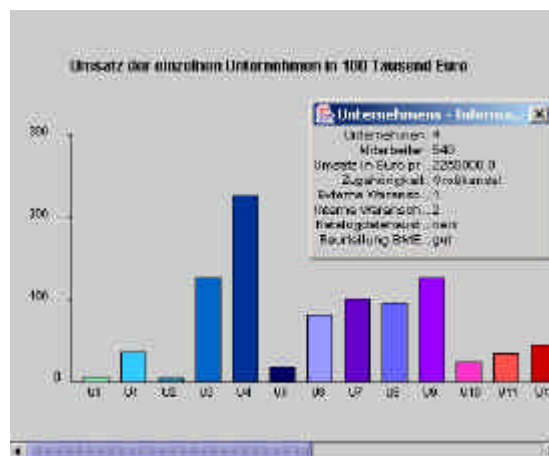


Abbildung 27: Screenshot VisJex: Dialogfenster "Weitere Informationen"

5.2.4 Legende

In der Legende können Sie die genauen Zahlen wieder zu den einzelnen grafischen Abbildungen ablesen.

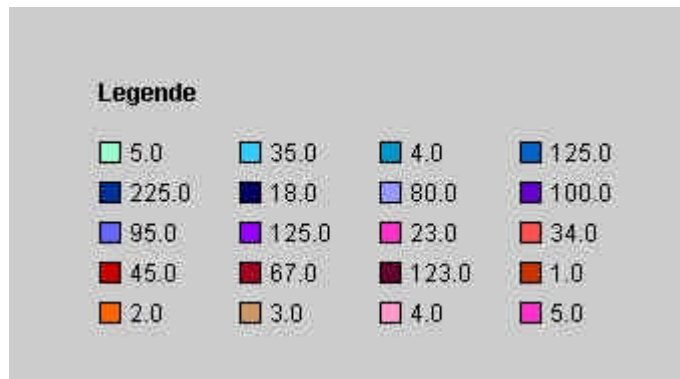


Abbildung 28: Screenshot VisJex: Legende

5.3 Der innere Aufbau von VisJex

VisJex besteht aus verschiedenen Klassen mit jeweils eigenen Methoden, die unten kurz beschrieben werden. Eine erste Übersicht bietet das folgende Klassendiagramm.

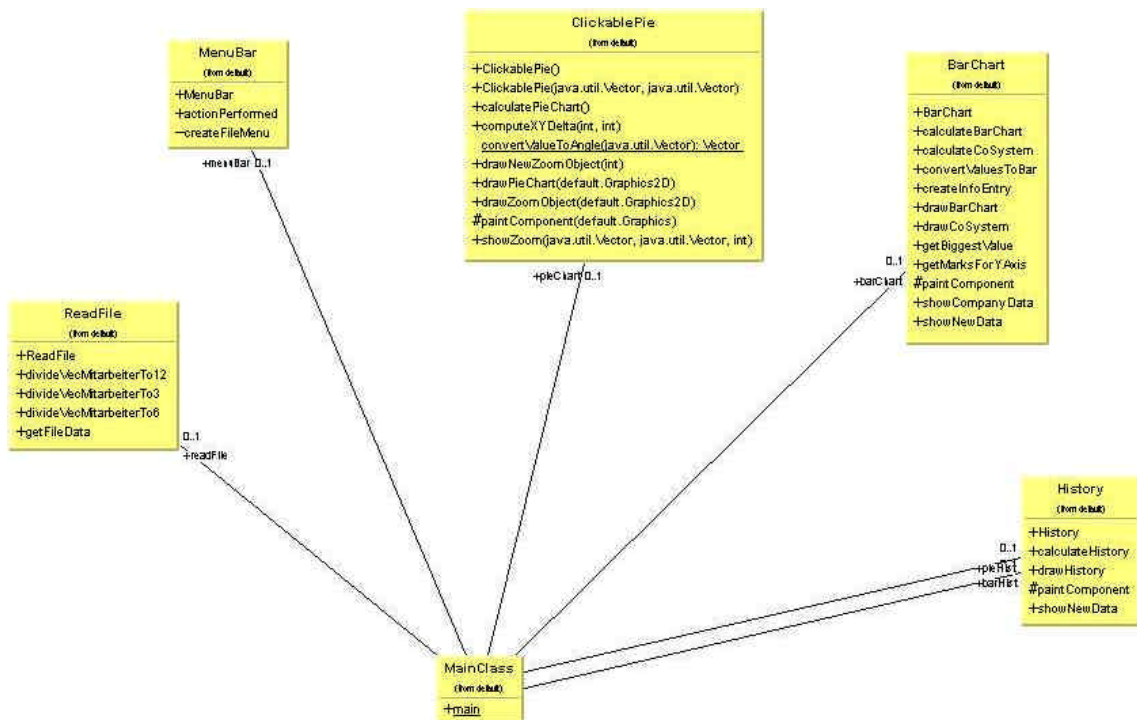


Abbildung 29: VisJex - Klassendiagramm

5.3.1 MainClass

MainClass hat innerhalb des Programms VisJex zwei verschiedene Aufgaben:

- Instanzieren der Komponenten
- Layouten der einzelnen Komponenten

5.3.2 MenuBar

Die Klasse MenuBar erleichtert das Erstellen und Verwalten der einzelnen Menüpunkte. Sie hat innerhalb VisJex keine weitere Funktionalität.

5.3.3 ReadFile

ReadFile hat folgende Methoden:

- getFileData()
- divideMitarbeiterTo3()
- divideMitarbeiterTo6()
- divideMitarbeiterTo12()

Beim Instanzieren eines Objektes der Klasse ReadFile während des Programmstarts wird die Datei, in der die zu visualisierende Datenmenge steht, mitgegeben. Dabei wird die Quelldatei zeilenweise ausgelesen und der Inhalt jeweils in eigene Vektoren geschrieben.

Beispiel

```
Mitarbeiter 34 1 210 5 543 108 11 117 678
Umsatz 5 35 4 125 225 18 80 100 95 125
Zugehoerigkeit Industrie Industrie Industr
Externe Warenschlüssel 1 1 2 3 1 1 2 3 1 1
Interne Warenschlüssel 2 1 2 3 2 1 2 3 2 1
Katalogdatenaustausch ja nein nein ja n
BMEcat gut schlecht schlecht schlecht g
ColorsZoom1 180 13762560 16766720
ColorsZoom2 13762560 16766720 180 1376256
ColorsYellow 13434777 13434726 13434676
ColorsBlue 10092492 3394815 39372 26316 1
ColorsRed 16724940 16732240 13369344 108
```

Abbildung 30: Screenshot source.txt

Zeile 1 „Mitarbeiter“ wird gelesen und der Inhalt ohne das erste Element „Mitarbeiter“ in den Vektor „mitarbeiter“ geschrieben. Genauso wird für die weiteren Zeilen verfahren.

Die Methode **getFileData()** gibt diese Vektoren zurück. Sie verlangt als Argument einen Descriptor, der dem ersten Token³² einer Zeile entspricht.

Beispiel

In der Quelldatei steht folgende Zeile:

```
Mitarbeiter 34 1 210 5 543 108 ...
```

Mit dem Methodenaufruf `getFileData(„Mitarbeiter“)` wird die Zeile `Mitarbeiter ...` identifiziert und der Vektor „mitarbeiter“ zurück gegeben.

Die Methode **divideMitarbeiterTo3()** bearbeitet die so gewonnenen Vektoren weiter. Nach bestimmten Kriterien wird der Vektor „mitarbeiter“ in drei Vektoren aufgeteilt, deren Inhalte die Menge der Unternehmen enthält, die zu den Kleinen-, Mittelständischen- und Grossen Unternehmen gezählt werden können. Mit den Daten aus dem Vektor „mitarbeiter“ werden auch alle weiteren relevanten Daten aus den anderen Vektoren in jeweils eigene Vektoren geschrieben, die die jeweiligen Datenmengen zu den KMUs (Kleine-, Mittelständische- und Grosse Unternehmen) enthalten.

Beispiel

```
Mitarbeiter 34 1 210 5 543 108
           11 117 678 73 789 12 ...
Umsatz5    35 4 125 225 18 80
           100 95 125 23 ...
```

In den Vektor `veckU` werden der Wert „34“ mit dem Index 0 geschrieben, gleichzeitig wird der dazugehörige Umsatz „5“ mit dem gleichen Index in den Vektor `veckUumsatz` geschrieben. So wird gewährleistet, dass der Zugriff auf zusammengehörende Werte möglich ist.

³² Token = hier: kleinste logische Einheit einer Zeile, z.B. Wort, Zahl; getrennt durch Separatoren, hier Tabstopps.

Im Weiteren ermittelt `divideVecMitarbeiterTo3()` die Größe der Vektoren `vecKU`, `vecMU`, `vecGU` (= Menge der Unternehmen mit Zuordnung zu KMUs). Diese drei Werte werden in den Vektor `vecValues` geschrieben.

Die Methoden `divideVecMitarbeiterTo6()` und `divideVecMitarbeiterTo12()` arbeiten ähnlich wie die Methode `divideVecMitarbeiterTo3()`. Allerdings wurde aus Zeitgründen darauf verzichtet, die dazugehörigen Vektoren mit auszulesen.

5.3.4 ClickablePie

Die Klasse `ClickablePie` beschreibt, welche Attribute ein `PieChart` hat. Außerdem enthält sie Methoden, um aus den gewonnenen Vektoren der Klasse `ReadFile`, das `PieChart` berechnen und zeichnen zu können:

- `convertValueToAngle(Vector)`
- `paintComponent(Graphics)`
- `calculatePieChart()`
- `computeXYDelta(int, int)`
- `drawPieChart(Graphics2D)`
- `drawZoomObject(Graphics2D)`
- `drawNewZoomObject(Graphics2D)`
- `showZoom(Vector, Vector, int)`

Der Vektor `vecValues`, den die Methode `ReadFile.divideMitarbeiter()` zurückliefert, wird mit der Methoden **`convertValuesToAngle()`** weiterverarbeitet. `convertValuesToAngle` errechnet unter Mithilfe der Summe der Vektorelemente den Winkel der jeweiligen Werte und schreibt sie in den Vektor `vecAngles`.

Beispiel

```

vecValues = (34, 1, 16);
sum = (34 + 1 + 16) = 51;
angle(34) =      vecElement * 36033 / sum
           = 34 * 360 / 51 = 240;
vecAngles(240, 7, 113);

```

Mit diesem Vektor `vecAngles` berechnet die Methode **calculatePieChart()** das PieChart. Innerhalb der Methode `calculatePieChart()` wird die Methode `computeXYDelta()` aufgerufen, die berechnet, um welche Koordinaten das PieSegment nach dem Auswählen verschoben werden soll, um eine Markierung zu erreichen. Danach zeichnet Methode **drawPieChart()** das errechnete PieChart. Beide Methoden werden in **paintComponent()** aufgerufen. `paintComponent()` selbst wird bei der Instanziierung eines Objektes der Klasse `ClickablePie` automatisch aufgerufen.

Die Methoden **drawZoomObject()** und **drawNewZoomObject()** sind für das Zeichnen und Neu-Zeichnen des Steuerelementes, das das Zoomen für das PieChart auslöst, zuständig.

Die Methode **showZoom()** beschreibt den Fall, wenn durch Zoomen ausgelöst, das PieChart neue Daten anzeigen soll.

5.3.5 BarChart

Die Klasse `BarChart` beschreibt, welche Attribute ein `BarChart` inklusive Koordinatensystem und einzelne `BarCharts` (Balkendiagramme) besitzt. Ausserdem enthält sie Methoden, um aus den gewonnenen Vektoren der Klasse `ReadFile`, das `BarChart` berechnen und zeichnen zu können:

Die Klasse `BarChart` besitzt drei Gruppen von Methoden. Die erste Gruppe beschreibt Methoden des Koordinatensystem, die zweite Methoden des `BarCharts` (Balkendiagramm) selbst, die dritte beschreibt Methoden, die die gesamte Klasse `BarChart` betreffen

Methoden des Koordinatensystem:

³³ 360 entspricht 360° eines Kreises

- calculateCoSystem(Vector)
- drawCoSystem(Graphics)
- getBiggestValue(Vector)
- getMarksForYAxis(double)

Methoden des BarChart (Balkendiagramm):

- calculateBarChart(Vector, Vector)
- drawBarChart (Graphics)
- convertValuesToBar(Vector)

Methoden der gesamten Klasse BarChart:

- paintComponent(Graphics)
- showNewData(Vector, Vector, int)
- showCompanyData(int, int, int, int)
- createInfoEntry(Container aContainer, String aDescription, String aValue)

Beim Instanzieren eines neuen Objektes der Klasse BarChart, bei dem als Parameter ein Vektor verlangt wird, werden folgende drei Methoden aufgerufen: getBiggestValue(), getMarksForYAxis(), convertValuesToBar().

Die Methode **getBiggestValue()** errechnet aus dem Vektor, der dem Konstruktor mitgegeben wird den größten Wert. Sie liefert als Rückgabewert biggestValue.

Die Methode **getMarksForYAxis** verlangt als Parameter biggestValue. Mit diesem Wert und der vorgegebenen Länge der y-Achse berechnet sie die Anzahl der y Achsenabschnitte und somit deren Länge. Als Rückgabewert liefert sie den Wert maxValue, der den nächstgrößte Wert aufgerundet auf 10, 100, 1000 usw. angibt.

Die Methode **convertValuesToBar** berechnet aus den Werten des Vektors, die dem Konstruktor der Klasse BarChart mitgegeben wird, die Länge der BarCharts in ganzen Pixeln (Balkendiagramm) in Abhängigkeit von den Werten.

Beispiel

```
vecUmsatz = (5, 35, 225);  
  
int achsLength = 200;  
  
int maxValue = 300;  
  
barLength (35) = vecElement * achsLength  
                / maxValue = 35 * 200 / 300 =  
                23;  
  
vecBarLength(3, 23, 150);
```

Die Methode **paintComponent()**, die bei der Instanziierung eines neuen BarCharts Objektes automatisch aufgerufen wird, ruft ihrerseits die Methoden `calculateCoSystem()`, `drawCoSystem`, `calculateBarChart()` und `drawBarChart()` auf.

Die Methode **calculateCoSystem()** erhält als Parameter den Vektor `vecUmsatz`. Aus den Werten dieses Vektors und dem Wert `maxValue` errechnet sie das Koordinatensystem mit y-Achsenabschnitten und der dynamischen Länge der x-Achse.

Die Methode **drawCoSystem()** ist für das Zeichnen des errechneten Koordinatensystems zuständig.

Die Methode `calculateBarChart()` erwartet als Parameter den Vektor `vecBarLength`. Mit den Werten dieses Vektors errechnet sie die Position der einzelnen Bars (Balken).

Die Methode **drawBarChart()** ist für das Zeichnen der einzelnen Bars zuständig.

Soll das BarChart neue Werte anzeigen, ist die Methode **showNewData()** zuständig.

Bei einem Maus-Klick auf ein BarChart, wird die Methode **showCompanyData()** aufgerufen. Aus den vorhanden Vektoren `vecMitarbeiter`, `vecUmsatz`, `vecZugehoerigkeit` usw., die durch Instanziierung eines Objektes der Klasse `ReadFile` erzeugt werden, liest diese Methode alle Daten zu einem Unternehmen und zeigt sie in einem Dialogfenster an.

Mit der Methode `createInfoEntry()` wird die Schrift der Labels innerhalb des Dialogfensters formatiert.

5.3.6 History

Die Klasse History ist für das Zeichnen der Legenden der beiden Visualisierungskomponenten zuständig. Sie besitzt die Methoden:

- paintComponent(Graphics)
- calculateHistory(Vector, Vector)
- drawHisotry(Graphics)
- showNewData(Vector, Vector, Vector)

Beim Instanzieren eines neuen Objektes der Klasse History werden dem Konstruktor Vektoren mit den anzuzeigenden Werten, ein Vektor mit spezifischen Farbwerten und der Legendentext mitgegeben.

Die Methode **paintComponent()**, die wie bei den anderen Klassen auch beim Instanzieren automatisch aufgerufen wird, ruft wiederum die Methoden calculateHistory() und drawHistory() auf.

Aus den Werten des Vektors, der dem Konstruktor mitgegeben wird, errechnet die Methode **calculateHistory()** Anzahl der einzelnen Legendenquadrate und deren Position.

Für das Zeichnen der Legende ist die Methode **drawHistory()** zuständig.

Wie bei den schon beschriebenen Klassen ist die Methode **showNewData()** dafür zuständig, was programmtechnisch passieren soll, wenn eine Legende mit neuen Daten gebraucht wird.

6 Entwicklung und Implementierung des Visualisierungstools

6.1 Entwicklungsumgebung und Hilfsmittel

Um dem zeitgemäßen Anspruch der Portabilität von Software gerecht zu werden, wurde VisJex in der Programmiersprache Java geschrieben. Das Programm wurde ausschließlich unter dem Betriebssystem Windows 2000 entwickelt. Getestet wurde es unter Windows 2000 und unter Linux. Als Java-Entwicklungsumgebung kamen das Java Development Kit 1.4 in Verbindung mit Eclipse 2.0 zum Einsatz. Das Klassendiagramm wurde mit JVision 2.1 erstellt.

6.1.1 Java Development Kit 1.4

Java ist eine plattform-unabhängige Programmiersprache, die Anfang der 90er Jahre von Sun Microsystems entwickelt wurde. Ihre Anwendungsgebiete reichen von fensterbasierten Anwendungen über Applets bis hin zu Entwicklungen von Client/Server- und Datenbankmodulen.

Java bietet Editionen für verschiedene Anwendungsbereiche:

- Java 2 Standard Edition (J2SE) als Desktop-Variante für Fenster-Anwendungen, die auf verschiedenen Betriebssystemen laufen.
- Java 2 Enterprise Edition (J2EE) für mehrschichtige Client/Server Anwendungen.
- Java 2 Micro Edition (J2ME) als ressourcensparende Edition für kleine, mobile Geräte

Java ist objektorientiert. Objektorientierte Programmierung konzentriert sich auf die Erstellung von Objekten (= Klassen) mit Daten und Methoden. Im Idealfall sind Klassen wieder verwertbar, d.h. sie können bei fortschreitender Implementierung immer wieder verwendet werden.

In Java wird der Code zu maschinennahem Bytecode kompiliert. Dadurch ist die Ablauffähigkeit des Programms nach dem Compilieren auf jedem Rechner garantiert.

Das JDK steht unter www.java.sun.com jedem zum Download zur Verfügung. Außerdem steht das JDK 1.4.1 auf der beiliegenden CD-Rom im Verzeichnis JDK 1.4.1 zur Installation bereit.

6.1.2 Eclipse Plattform 2.0

Im November 2001 brachte IBM, „das weltweit größte Unternehmen für Informationstechnologie“³⁴, eine Development-Workbench im Wert von 40 Millionen Dollar in die neue Open-Source-Community Eclipse.org ein. Von James Governer, einem Analysten des IT-Beratungsunternehmens Illuminata wurde dieses Vorgehen als „IBM's most audacious open source gambit so far“³⁵ bezeichnet.

Jocelyne Attal, die Vizepräsidentin bei IBM für „Application and Integration Middleware Marketing“ beschreibt die Intention von IBM für diese Handlung als Frage des „Erfolgs oder Versagen [...] im Kampf zwischen Microsoft und IBM“ um die Aufmerksamkeit von Entwicklern, die „einfache Programme“ bis hin zu „hochentwickelten und komplexen Anwendungen“ konstruieren“.³⁶

Während zur Zeit der Gründung die Eclipse-Community 150 führende Software-Anbieter umfasst, sind es im September 2002 schon mehr als 175, die an der Erweiterung der Software in Form von Tools arbeiten. Die Zahl der Nutzer, die die kostenfreie Software aus dem Web bis September 2002 herunter geladen haben, beträgt laut IBM zwei Millionen³⁷.

Eclipse bietet eine universale Entwicklungsoberfläche für alle Prozesse der Implementierung. Sie integriert das Testen, die Abstimmung der Performance und das Debugging. Die Software von Eclipse soll die bisher kosten- und zeitaufwendige Integration von Tools fremder Hersteller erleichtern, um die Produktivität der Entwickler zu steigern.

Eclipse 2.0 erschien im September 2002 mit einer Reihe neuer Features, die eine „einfachere Handhabung, schnellere Laufzeiten sowie die bessere Integration von

³⁴ [IBM01]

³⁵ [Governer01], „kühnster Open-Source-Streich von IBM“

³⁶ [Attal01]

³⁷ [Eclipse02]

Tools aus verschiedenen Betriebssystemen³⁸ ermöglichen. Eclipse unterstützt nun das JDK 1.4. Ein „quick fix“ Modul garantiert automatische Fehlerbeseitigung. Seh- oder Körperbehinderte können Eclipse mit Hilfe des Tastaturanschlags anstelle von einer Maus besser bedienen.

Die folgende Tabelle zeigt im Überblick die wichtigsten Funktionalitäten von Eclipse.

Tabelle 3: Eclipse Funktionalität: Übersicht³⁹

	Eclipse
Runtime und OS	Wahlweise verschiedene JDKs (1.2 – 1.4), definierbar auf Projektebene, Windows, Solaris, Linux u.a.
Editor	Intern: Syntax-Highlighting, semantische Auto-Vervollständigung, Einblendung von Java-Doc, Refactoring, Formatierer
Debugger	Intern: HotSwap Hot Code Replace (ab Sun Java SDK 1.4), Inspect, Enabling/Disabling von Breakpoints (Bedingungen definierbar), remote Debugging
Code-Verwaltung	Intern über eigene History, extern über integrierte Anbindung an CVS oder kommerzielle Versionierungssysteme
Deployment	Extern über integrierte Anbindung an Jakarta Ant
Kosten	keine

³⁸ [IBM02]

³⁹ [Wystup02]

Die Vor- und Nachteile von Eclipse wie Wystup sie beschreibt, werden in der unten stehenden Tabelle zusammen gefasst:

Tabelle 4: Eclipse: Vor- und Nachteile⁴⁰

Vorteile	Nachteile
<ul style="list-style-type: none"> - Verbindet die wichtigsten Features (inkrementeller Compiler, Debugger etc.) aus VA mit einem hohen Maß an Flexibilität und Transparenz - File-basiertes Repository - Geringe Kosten - Hohe Akzeptanz bei Herstellern und in der Open-Source Community - Unterstützung für Linux - Einfache Integration von tomcat mit JSP-Debugging - Einfache Erweiterbarkeit über Plugin-Mechanismus 	<ul style="list-style-type: none"> - Kein UI-Builder - Keine stabile GTK-Version für Linux verfügbar trotz Open-Source-Projekt - Geringe Geschwindigkeit der Motif-GUI unter Linux - Ungewissheit über die weitere Entwicklung / Investitionssicherheit

Bei der Implementierung von VisJex waren einerseits der Aspekt einer einheitlichen Plattform für die unterschiedlichen Funktionen, die Eclipse bietet und die bei der Entwicklung einer Software nötig sind, hilfreich. Andererseits natürlich die Professionalität des Produktes bei einem nicht entstehenden Kostenaufwand.

⁴⁰ [Wystup02]

6.2 Überblick über die Entwicklungsschritte

Bei der Implementierung von VisJex wurden drei Entwicklungsschritte angewandt:

- Aneignen der notwendigen Technik und Umsetzung in minimalistische Prototypen
- Realisieren von Teilaspekten des Programms durch Implementierung von Prototypen
- Zusammenführen der daraus entstandenen Prototypen in VisJex

6.2.1 Aneignen der notwendigen Technik

Neben einer Wiederholung der Grundkonzepte der Programmiersprache Java nahm die Vertiefung in die Pakete `javax.swing`, `java.awt`, `java.util` und darin besonders in die Klassen `Graphics2D` und `StringTokenizer` einen großen Raum ein.

Daraus sind folgende Mini-Prototypen entstanden:

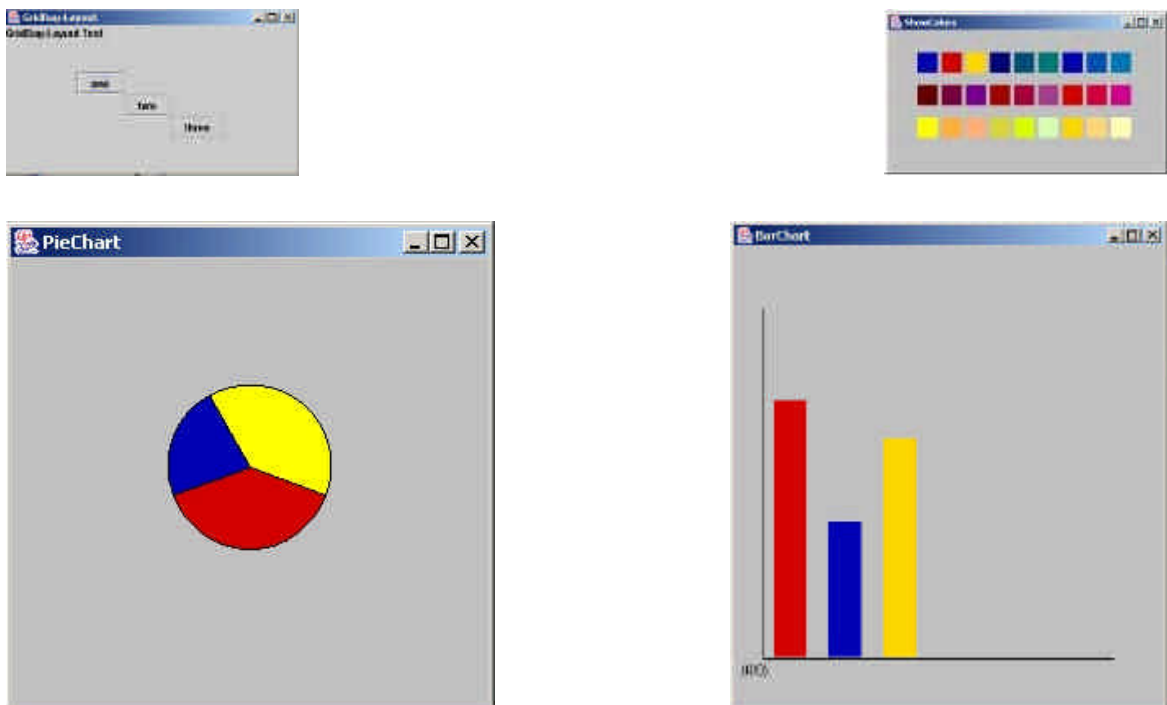


Abbildung 31: Prototypen 1

6.2.2 Realisieren von Teilaspekten

In einem zweiten Schritt wurden einzelne Komponenten aufbauend auf den Miniprototypen für das Programm VisJex mit der dazugehörigen Logik implementiert. Die Objektorientiertheit der Programmiersprache Java unterstützt dieses Vorgehen und erleichtert die modularisierte Umsetzung in Klassen.

Als Ergebnis entstanden prototypisch die unten gezeigten Komponenten in jeweils einer eigenen Klasse. Im Lauf der Implementierung erhielten diese Klassen weitere Funktionalitäten.

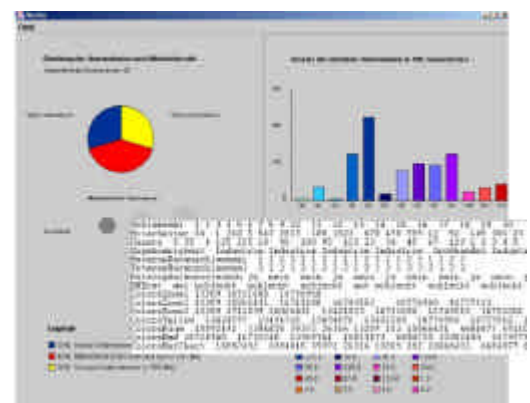
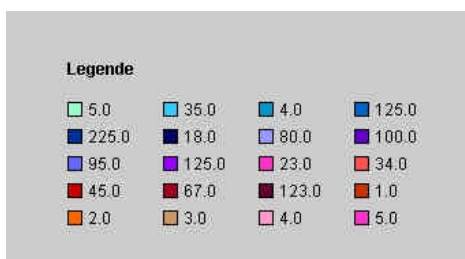
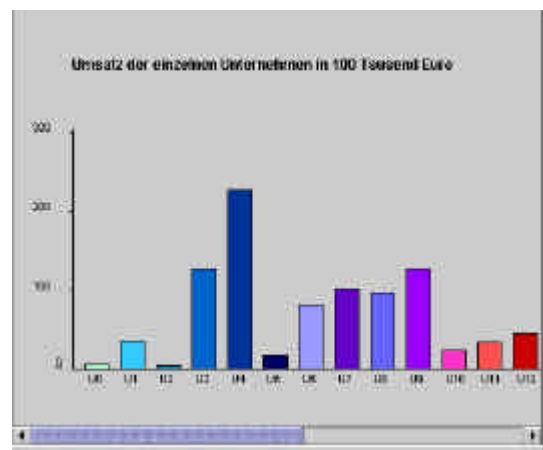


Abbildung 32: Prototypen 2

6.2.3 Zusammenführen der Komponenten

Als letzter Schritt erfolgte das Zusammenführen der Klassen in einer Anwendung, dem Programm VisJex.

Die nächste Abbildung zeigt einen Screenshot des Programm VisJex in Aktion.

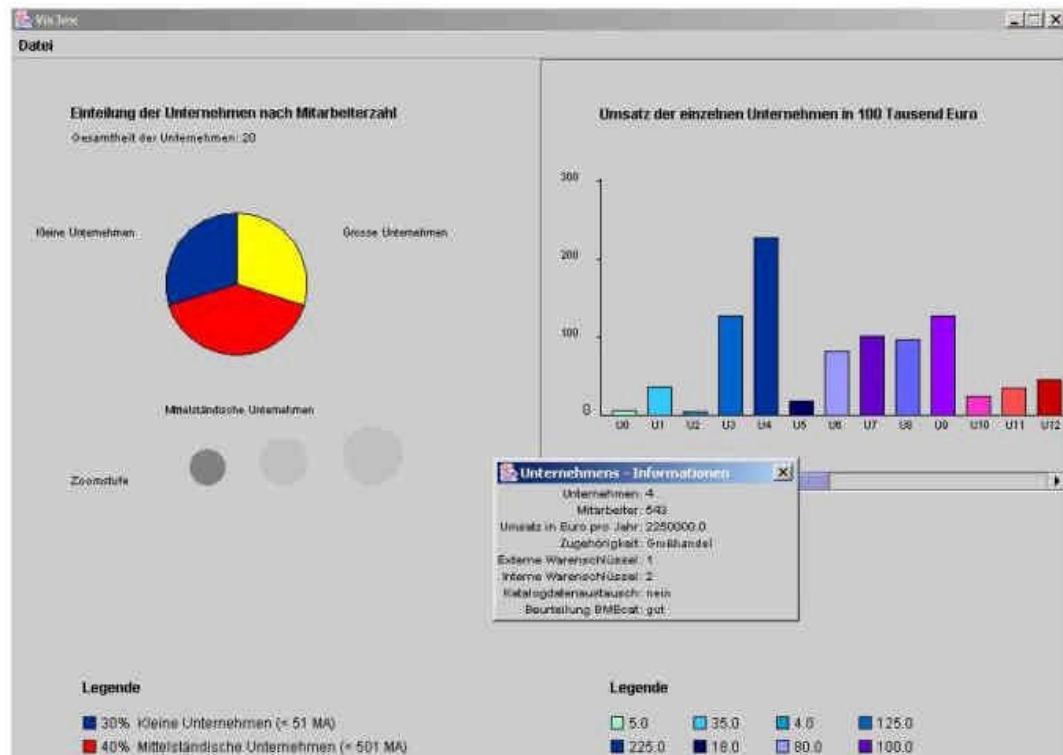


Abbildung 33: Screenshot VisJex

7 Projekt „Diplomarbeit“

Die Diplomarbeit ist mit Rahmenbedingungen ausgestattet, die von Beginn an festgelegt waren. Das betrifft das Ergebnis, den zeitlichen Rahmen und die Ressourcen, die zur Verfügung stehen. Vergleicht man diese Aspekte mit Definitionen, die für Projektmanagement gegeben werden, wird klar, dass auch die Diplomarbeit ein Projekt ist.

Definition der DIN 69901⁴¹

Ein Projekt ist ein Vorhaben, das im wesentlichen durch Einmaligkeit der Bedingungen in ihrer Gesamtheit gekennzeichnet ist, wie z.B. Zielvorgabe, zeitliche/finanzielle/personelle Begrenzung, Abgrenzung gegenüber anderen Vorhaben, spezifische Organisation

Definition von business-wissen.de⁴²

Folgende Ziele sollten erfüllt sein, um eine Aufgabe zum Projekt zu machen:

- *Es sind klare Ziele vorgegeben*
- *Aufgabe ist zeitlich begrenzt*
- *Es existieren finanzielle oder andere Begrenzungen*
- *Aufgabe ist vergleichsweise komplex*
- *Aufgabe ist für das Unternehmen relativ neu*

Nachdem der Status der Diplomarbeit so definiert wurde, war es folgerichtig, ansatzweise Prinzipien des Projektmanagements zu verwenden. Umgesetzt wurde ein Phasenmodell, Arbeitspakete, Meilensteine, die zeitliche Überwachung und Verwaltung des Projektes mit der Software MSProject.

⁴¹ [DIN87]

⁴² [BusinessWissen02]

7.1 Phasenmodell

Für das kleine Projekt „Diplomarbeit“ wurde das unten abgebildete einfache Modell verwendet. Die Vorteile des Modells wurden innerhalb der Projektdurchführung sichtbar. Es

- lässt das Projekt transparenter werden
- unterstützt systematisches Vorgehen
- schafft Übersicht
- erleichtert die Kontrolle

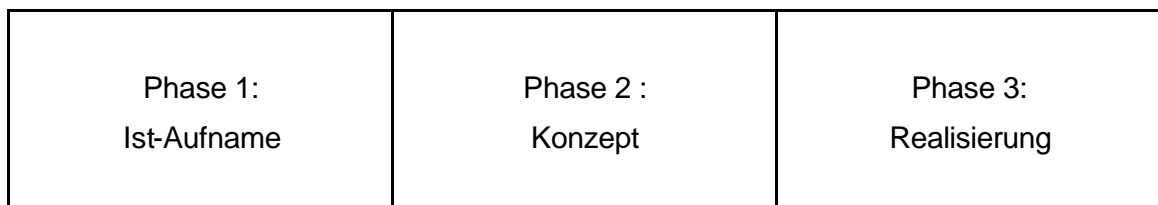


Abbildung 34: Phasenmodell⁴³

Den Phasen wurden Arbeitspakete zugeordnet, zu denen jeweils Meilensteine als Ergebnisse, bzw. Zwischenergebnisse definiert wurden.

⁴³ [Litke/Kunow00]

7.2 Arbeitspakete

Auf der Grundlage des Phasenmodells wurden Arbeitspakete definiert und den Phasen zugeordnet. Während Phasen das übergeordnete Projektproblem einer ersten Aufteilung unterwerfen, haben Arbeitspakete das Ziel, Probleme so weit hierarchisch zu unterteilen bis keine Unterteilung mehr möglich ist, weil die Probleme in einem engen sachlichen Kontext stehen.

Die Vorteile der Aufteilung in Arbeitspakete liegen darin, dass man gezwungen wird, Probleme in Teilprobleme zu unterteilen und sie zu spezifizieren. Gedanklich erfordert dieser Prozess eine sehr intensive Beschäftigung mit den Problemen, was zu einem Abbau von Unsicherheit in Bezug zur Aufgabenstellung führt.

Eine Übersicht der Arbeitspakete und die Zuordnung zu den entsprechenden Phasen zeigt die folgende Tabelle.

Tabelle 5: Arbeitspakete

Phase	Arbeitspaket	
	AP01	Projektmanagement
Ist-Aufnahme	AP02	Diplomarbeit – Wissenschaftliche Ausarbeitung
Konzept	AP03	Prototypen
	AP04	Anforderungsanalyse
	AP05	Spezifikation
	AP06	Entwurf
Realisierung	AP07	Implementierung
	AP08	Dokumentation
	AP09	Testbetrieb
	AP10	Abnahme

Die Abhängigkeiten der einzelnen Arbeitspakete und die besondere Stellung des Arbeitspaketes Projektmanagement, das unterstützend alle anderen Arbeitspakete begleitet, verdeutlicht nachstehende Abbildung.

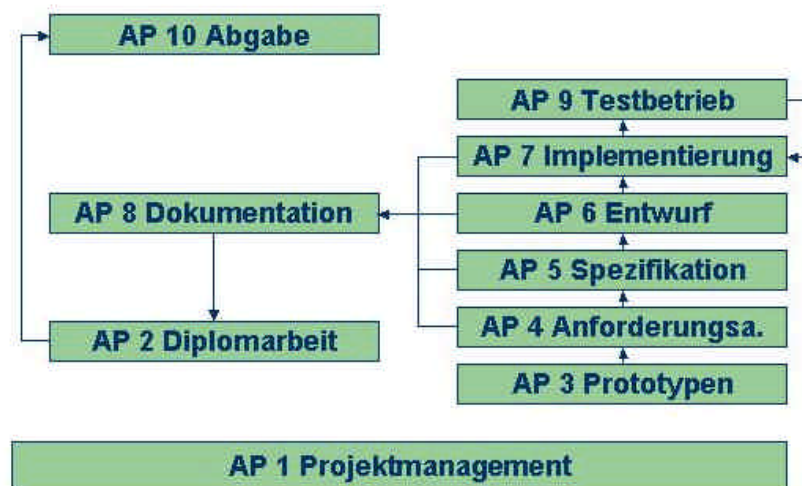


Abbildung 35: Abhängigkeit der Arbeitspakete

Arbeitspakete haben nicht nur einen zeitlich fest umrissenen Rahmen, sondern auch ein vordefiniertes Ziel oder Ergebnis, die als Meilensteine festgelegt werden.

7.3 Meilensteine

Meilensteine definieren einerseits das Ende eines Arbeitspaket, andererseits auch das Ende einer Phase und beschreiben damit als Zwischenergebnisse die Voraussetzung für den Beginn der nachfolgenden Phase.

Die Vorteile der Meilensteine sind:

- Vordefinierte Zwischenergebnisse
- Zwischenergebnisse als Abschluss einer Phase verdeutlichen den bisherigen Erfolg
- Beleg des Projektfortschritts

Ein Ausschnitt aus dem mit MSProject erstellten Projektplan gibt einen Überblick über die definierten Meilensteine des Projekts „Diplomarbeit“.

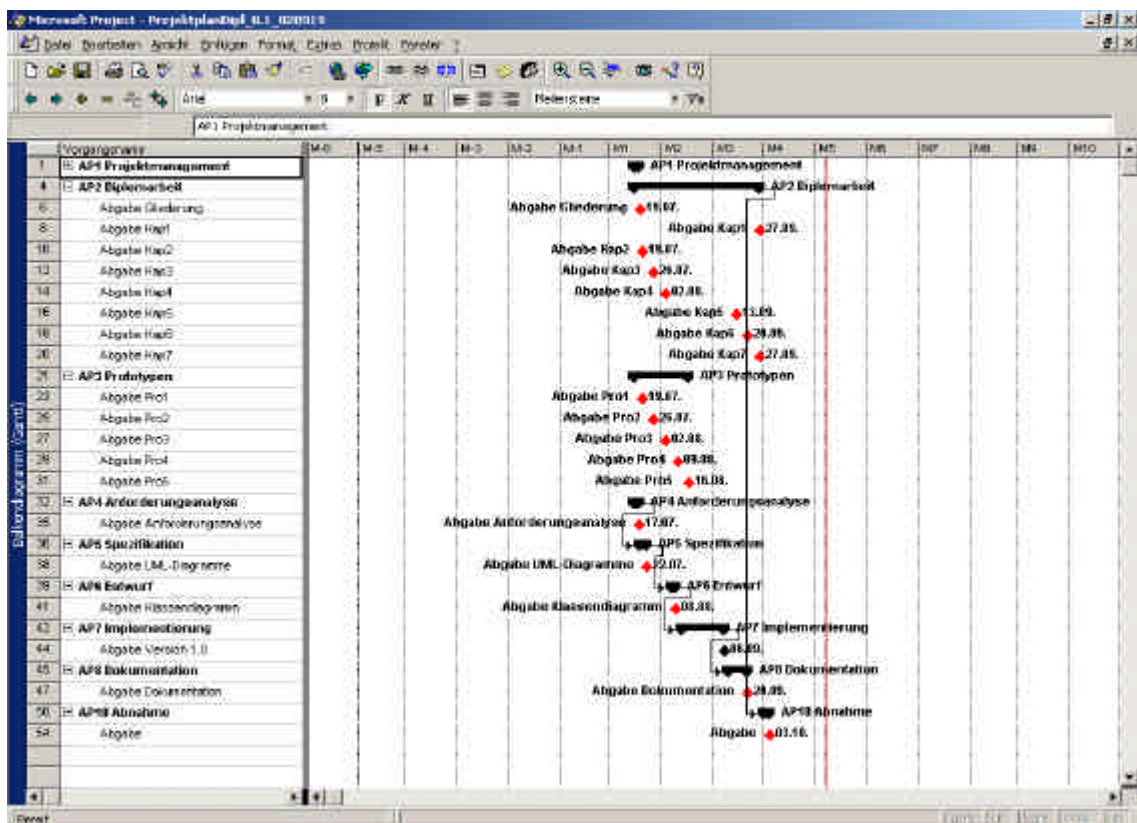


Abbildung 36: Meilensteine

7.4 MS Project

Als unterstützendes Werkzeug für das Projektmanagement wurde MS Project 98 eingesetzt. Aus der fast verwirrenden Vielfalt der Funktionen, die MS Project bietet, wurde nur eine rudimentäre Auswahl getroffen, die hauptsächlich das Verwalten und Darstellen der Arbeitspakete mit ihren Meilensteinen betraf.

7.5 JVision

JVision ist ein Produkt von Object Insight. Die Anwendung ist ein Werkzeug für UML (Unified Modelling Language). Es läuft sowohl unter Windows als auch unter den Betriebssystemen Sun Solaris und Linux. Neben Klassendiagrammen unterstützt JVision die Erstellung von Use-Case- und Sequenzdiagramme aus der Programmiersprache Java. Als weitere Features bietet die Software eine HTML-Dokumentationsgenerierung und Roundtrip-Engineering von Java-Code. Innerhalb dieser Diplomarbeit wurde JVision 1.2.1 für die Erstellung von Klassendiagrammen eingesetzt. Diese Version ist für akademischen, nicht-kommerziellen Gebrauch frei einsetzbar. Daneben steht eine 30-Tage Trial-Version der Version 2.1 auf der Homepage⁴⁴ zum Download zur Verfügung.

⁴⁴ <http://www.object-insight.com/download/>

8 Ausblick

Als Ergebnis dieser Diplomarbeit ist der Prototyp eines Visualisierungstools entstanden, der aufgrund der relativ kurzen Bearbeitungszeit nur rudimentäre Funktionen haben kann.

Um das Tool in möglichst vielen Bereichen einsetzen zu können, wäre eine Erweiterung um folgende Funktionen nötig:

Datei einlesen

In der aktuell vorliegenden Version ist es nicht möglich, ein beliebiges Dateiformat als Datenpool zu verwenden. Momentan kann das System Text-Dateien, bzw. jede Datei, die in ein txt – Format gebracht wird, lesen. Es wäre denkbar, ein Format wie zum Beispiel XML als Standard der Datenherkunft zu definieren. Damit wäre sicher gestellt, dass die Daten beim Einlesen korrekt in das System gelangen.

Weitere Darstellungsformen

Momentan bietet VisJex nur zwei Präsentationsformen an: Pie Chart und Bar Chart. Es ist nicht möglich zwischen beiden Formen zu wählen. Möglich wäre, in einer Kombinationsbox die Art der Darstellungsform als Wahlmöglichkeit dem Benutzer zu überlassen. Das heißt, die Ausgangsgrafik wäre nicht automatisch ein Pie Chart, sondern eine beliebige, wählbare Grafikart. Das gleiche Prinzip könnte man für das Bar Chart anwenden. Denkbar wäre auch, dass in der abhängigen Darstellung (Bar Chart) eine Tabelle mit den kompletten Datensätzen zur Wahl stünde.

Wechsel zwischen verschiedenen Datenmengen

Die bisher realisierte Applikation bietet keine Möglichkeit, die erste Grafik (Pie Chart) mit einer frei wählbaren Datenmenge darzustellen. Das gleiche gilt für die abhängige Grafik (Bar Chart). Durch den Einsatz eines Listenfeldes oder Kombinationsbox könnte man für beide Darstellungen die Auswahl verschiedener Datenmengen ermöglichen.

Speichern und Drucken

Die Ergebnisse, die der Anwender während seines Explorierens durch die vordefinierte Datenwelt sieht, können in der vorliegenden Version weder gespeichert noch gedruckt werden. Um sie weiter verarbeiten zu können, sollte eine Speicher- und Druckfunktion implementiert werden.

Dateneingabe und –modifizierung

Momentan arbeitet VisJex mit einem statischen Datenpool, das heißt, die Daten die vorhanden sind, können nur in der Source-Datei verändert oder erweitert werden. Um ein Hilfsmittel für den täglichen Gebrauch zu sein, besonders in Bereichen, wo täglich Datenänderungen vorliegen, ist eine Erweiterung um diese Funktionen über eine benutzerfreundliche Schnittstelle unabdingbar.

Literaturverzeichnis

- [Attal01] Attal, Jocelyne in "Ein Kampf zwischen Microsoft und IBM" von Volker Weber, c't 25/2001
- [Bleichmann et al. 00] Bleichmann, U.; Mann, T.M.; Mußler, G.; Reiterer, H.: Visualisierung von entscheidungsrelevanten Daten für das Management, HMD, Heft 212, April 2000
- [BusinessWissen02] <http://www.business-wissen.de>: Portal rund um das Thema Business
- [Card et al. 99] Card, S.K., Mackinlay, J.D., Shneiderman, B.: Readings in Information Visualization – Using Vision to Think, Morgan Kaufman Publishers, San Francisco, 1999
- [CS02] <http://www.cs.umd.edu>: Departement of Computer Science, University of Maryland, USA
Zugriff am 17.09.2002
- [DIN87] DIN 69901: Projektwirtschaft, Projektmanagement; Begriffe; 1987
- [Duden90] Duden – Fremdwörterbuch, Dudenverlag, Mannheim u.a., 1990
- [Eclipse02] www.eclipse.org
Zugriff am 14.09.2002
- [Governer01] Governer, James in "IBM donates \$40m to open source" von John Geraldts in Silicon Valley, 06-11-2001, www.vnunet.com
Zugriff am 14.09.2002
- [Gundry02] www.knowab.co.uk
Zugriff am 10.09.2002

- [Hicks2002] Hicks, Kelvin: Student Information Science, University of Texas, Austin
<http://www.ddg.com/LIS/InfoDesignF96/Kelvin/Napoleon/map.html>
Zugriff am 20.08.2002
- [IBM01] <http://www.ibm.com>
Zugriff am 10.10.2002
- [IBM02] Internationaler Einsatz der neuen Eclipse-Plattform Version 2.0, September 2002, <http://www.ibm.com>
Zugriff am 10.10.2002
- [Krämer01] Krämer, W.: Statistik verstehen – Eine Gebrauchsanweisung, Piper, München, 2001
- [Litke/Kunow00] Litke, Hans-D.; Kunow, Ilona: Projektmanagement, Haufe,Planegg, 2000
- [McCormick87] McCormick, B., DeFanti, T., Brown, R.: Visualization in scientific computing and computer graphics, ACM SIGGRAPH, 1987
- [Müller97] Müller, A.v., Denkwerkzeuge für Global Player. In: U. Krystek, E. Zur, Hrsg. Internationalisierung – Eine Herausforderung für die Unternehmensführung, Berlin
- [Otto et al. 02] Otto, Boris; Beckmann, Helmut; Kelkar, Oliver; Müller, Sylvia: E-Business-Standards – Verbreitung und Akzeptanz, Fraunhofer IRB Verlag, Stuttgart, 2002
- [Parc02] <http://www.parc.com/>
Zugriff am 10.09.2002
- [Riedwyl87] Riedwyl, Grafische Gestaltung von Zahlenmaterial, 1987

- [Sapinfo00] SAP INFO Ausgabe 69 2000; www.sapinfo.net
Zugriff am 23.09.2002
- [Schumann00] Schumann, H.; Müller, W.: Visualisierung. Springer, Berlin u.a., 2000
- [Shneiderman92] Shneiderman, B.: Designing the User Interface – Strategies for Effective Human-Computer Interaction, Addison-Wesley, Reading u.a., 1992
- [Scott98] Biographies of Women Mathematicians Web Site, Agnes College, 1998 <http://www.agnesscott.edu/lriddle/women/nightpiechart.htm>
Zugriff am 4.09.2002
- [Sony CSL02] Sony Computer Science Laboratories, Tokyo, Japan
- [Spence01] Spence, R.: Information Visualization, Addison-Wesley, Harlowe u.a., 2001
- [Tube02] www.thetube.com, Webauftritt der Londoner U-Bahn, 2002
Zugriff am 4.09.2002
- [Wystup02] Wystup, Frederik: Eclipse – lohnt sich der Umstieg?, Vortrag innerhalb des Java Forums Stuttgart 2002, <http://www.jfs2002.de>
Zugriff am 8.10.2002

Anhang

A: Anforderungsanalyse

Projektthema

Grafisch aufbereitete Informationen sind i.A. nicht interaktiv, d.h. sie erlauben keine Beeinflussung durch den Betrachter. Interagieren mit Darstellung ist aber häufig nützlich, wenn z.B. Betrachtungswinkel (real oder im übertragenen Sinne) verändert werden. Solche Veränderungen sollen unmittelbar und direkt durchgeführt und durch die grafische Darstellung unmittelbar reflektiert werden.

Die Studienarbeit soll die Möglichkeit des interaktiven Explorierens mit grafischen Informationsstrukturen aufarbeiten und einen ausgewählten Aspekt mit Hilfe von Java realisieren.

Beschreibung des Projektes

Bei der Ansicht und Bewertung großer Datenmengen aus der Statistik treten oftmals Probleme auf, wenn Abhängigkeiten verschiedener Kriterien dargestellt werden sollen. Konventionelle Methoden zur Informationspräsentation erlauben meist nur die Visualisierung eines einzigen Sachverhaltes. Meist kann nur eine kleine Anzahl an Datenelementen abgebildet werden. Eine direkte Möglichkeit der Visualisierungsmanipulation ist dem Anwender nicht gegeben.

Vor dem Hintergrund dieser Problematik soll ein Visualisierungswerkzeug (VisJex) erstellt werden, das die Vorteile konventioneller Techniken der Informationspräsentation um den Aspekt der Interaktion erweitert. Durch Kombination verschiedener Aspekte des Datenbestand wird die Visualisierung von Abhängigkeiten erreicht.

VisJex soll dem Anwender gleichzeitig die Ansicht auf zwei Aspekte des Datenbestandes ermöglichen, wobei die eine abhängig von der anderen ist. Das Hauptobjekt wird als Tortendiagramm realisiert. Auf das Diagramm sind zwei Funktionen anwendbar:

Drehschalter-Funktion

Mit dieser Funktion kann der Anwender die Teilmenge der Daten auswählen, über die er unter einem anderen Aspekt mehr erfahren möchte. Beide Aspekte werden innerhalb des gleichen Bildschirmausschnitts zu sehen sein. Für die Visualisierung wird eine Methode der Informationspräsentation gewählt, die ein anderer Diagrammtyp (z.B. Balkendiagramm) ist.

Zoom-Funktion

Der Anwender kann durch Interaktion mit der Maus auf das Tortendiagramm, das die Gesamtheit der Daten repräsentiert, zoomen. Das Zoomen bewirkt eine Verfeinerung der Abstufungen, die die Kreissegmente von einander abgrenzen. Das heißt, die Anzahl der Daten innerhalb der Teilmenge wird kleiner. Die Anzahl der Klassifizierung der Datengesamtheit nimmt zu.

Das zweite Objekt, das von dem Hauptobjekt abhängig ist, bietet eine tiefergehende Sicht auf die Teilmenge der Daten, die ausgewählt wurde. Die Darstellung beider Objekte soll einen Standardwert besitzen, der durch den Anwender verändert werden kann. Falls eine zeitliche Realisierung möglich ist, sollen für das zweite, von dem Tortendiagramm abhängige Objekt folgende Funktionen bereit gestellt werden:

der Aspekt, unter dem die Teilmenge der Daten dargestellt wird und die Methode der Informationspräsentation können vom Benutzer gewählt werden.

Grafische Benutzerschnittstelle

Die Bedienung des Werkzeuges wird für den Benutzer möglichst intuitiv und einfach gestaltet. Mit den oben genannten Vorgaben sieht ein erster Entwurf wie folgt aus:

GUI-Visualisierungstool

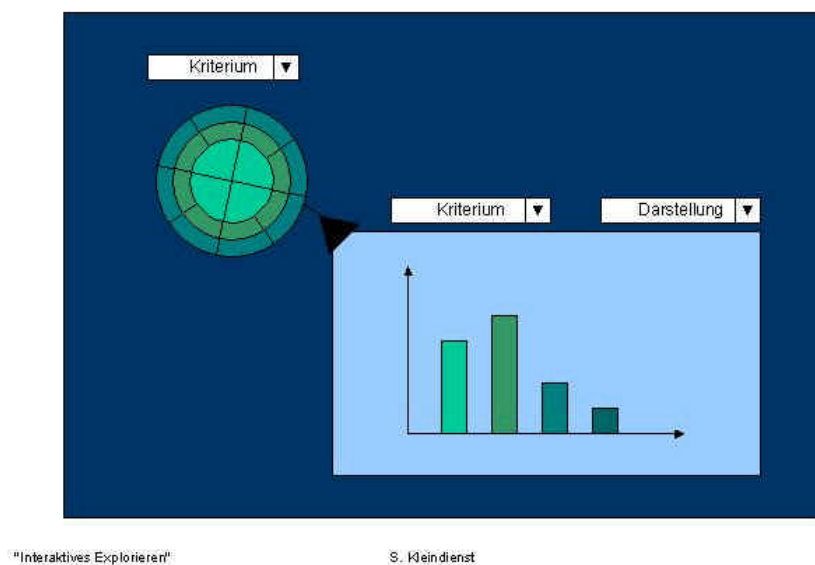


Abbildung: Grafische Benutzeroberfläche

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Diplomarbeit selbständig angefertigt habe. Es wurden nur die in der Arbeit ausdrücklich benannten Quellen und Hilfsmittel benutzt. Wörtlich oder sinngemäß übernommenes Gedankengut habe ich als solches kenntlich gemacht.

Ort, Datum

Unterschrift