

Informationsfusion auf heterogenen Datenbeständen

Oliver Dunemann¹, Ingolf Geist², Roland Jesse², Gunter Saake², Kai-Uwe Sattler²

¹ Nord/LB, Informationsmanagement und Organisation, Systemintegration, Friedrichswall 10, 30151 Hannover

² Fakultät für Informatik, Otto-von-Guericke Universität Magdeburg, Universitätsplatz 2, 39106 Magdeburg

Eingegangen am 27. November 2001 / Angenommen am 17. Juni 2002

Zusammenfassung. Die Informationsfusion als Prozess der Integration und Interpretation heterogener Daten mit dem Ziel der Gewinnung von Informationen einer neuen, höheren Qualität eröffnet eine Vielzahl von Anwendungsgebieten. Dabei erfordert dieser Prozess eine enge Verzahnung der bislang häufig noch isoliert vorliegenden Werkzeuge und Techniken zum Zugriff auf heterogene Datenquellen, deren Integration, Aufbereitung, eine Analyse der syntaktischen, semantischen sowie temporalen Strukturen und Visualisierung derselben. In diesem Beitrag werden Rahmenbedingungen der Informationsfusion ebenso dargestellt wie die sich aus ihnen ergebenden Aufgaben. Es werden Lösungsansätze zur Erstellung einer Workbench vorgestellt, die eine durchgängige Unterstützung von FusionsSchritten ermöglicht. Dabei wird das Ziel einer konsequenten Nutzung von Datenbanktechniken verfolgt.

Schlüsselwörter: Informationsfusion, Datenanalyse, Data Mining, Temporalität, Heterogene Datenquellen

Abstract. Information Fusion is the process of integration and interpretation of heterogeneous data to obtain information of a higher quality. This concept is open to a variety of applications. The process of information fusion requires a tight connection between isolated tools and techniques: accessing heterogeneous data sources, their integration, preparation and transformation, analysis of syntactical, semantic and temporal structures as well as their visualization. In this paper we present the framework for a workbench which supports the individual steps of information fusion in a continuous and uniform manner by using database technology.

Keywords: Information fusion, Data analysis, Data mining, Temporality, Heterogeneous data sources

CR Subject Classification: H.2.4, H.2.5, H.5.2

1 Einleitung und Motivation

„Stellen Sie sicher, dass Sie durch Ihren Wissensdurst nicht in der Flut von Informationen ertrinken.“

(Anthony J. D'Angelo)

Durch die Steigerung der Leistungsfähigkeit der Informationstechnologie ist heute die Verwaltung sehr großer Datenbestände technisch möglich. Da durch die Zunahme des Datenvolumens in gleichem Maße dessen Verständnis erschwert wird, wird die Möglichkeit zur automatisierten Analyse der Daten immer wichtiger.

Die bisherigen Arbeiten in diesem Bereich konzentrierten sich dabei zunächst auf die Erarbeitung von Lösungen für einzelne Teilschritte. So wurden die technischen Voraussetzungen zum Zugriff auf heterogene Datenbestände und Methoden für die Integration geschaffen. Werkzeuge für spezielle Nachbearbeitungs- und Analyseschritte wie beispielsweise Data Mining oder Visualisierung befinden sich in der Entwicklung. Eine Integration dieser Komponenten in Rahmen einer Workbench dient zum einen dazu, den manuellen Bearbeitungsaufwand zum Transformieren der Daten zwischen den Bearbeitungsschritten zu reduzieren. Zum anderen kann eine Instanz, welcher der gesamte Fusionsprozess bekannt ist, Optimierungen in der Art durchführen, dass beispielsweise gemeinsame oder wiederholt auszuführende Schritte erkannt und zusammengefasst werden, oder bereits vorhandene Analyseergebnisse in neuen Anfragen benutzen. Zusätzliche Dienste, die von einer solchen Workbench angeboten werden, sind unter anderem eine Benutzerverwaltung, eine einheitliche Fehlerbehandlung oder verschiedene Möglichkeiten der Visualisierung.

An der Universität Magdeburg wird zur Zeit unter dem Arbeitstitel *INFuse* eine Workbench entwickelt, die die Zusammenführung der Komponenten der Informationsfusion zum Ziel hat [30]. Durch ein offenes und modulares Konzept wird ein Rahmen aus den oben angesprochenen Basisdiensten und einer Benutzerschnittstelle zur Definition und Ausführung von Fusionsprozessen geschaffen, in den weitere Komponenten eingefügt werden können. Dabei werden bereits in der Analysephase Aspekte der Teilaufgaben der Informationsfusion berücksichtigt, indem von Praxisbeispielen ausgehend beispielhaft Fusionsprozesse modelliert werden.

Im Weiteren ist der Beitrag wie folgt gegliedert. Zunächst wird in Abschnitt 2 der Begriff „Informationsfusion“ definiert und abgegrenzt. Mögliche Anwendungen werden kurz aufgezeigt. Anschließend wird im Abschnitt 3 ein Beispiel entwickelt, welches durchgängig in der Arbeit benutzt wird. Abschnitt 4 stellt den Hauptteil der Arbeit dar und zeigt die Integration verschiedener Methoden in einer Workbench zur Unterstützung der Informationsfusion. Dabei wird auf die Architektur, mögliche Integrations- und Analysemethoden, die Verwaltung der Metadaten sowie die Visualisierung und die Berücksichtigung temporalen Verhaltens während des Fusionsprozesses eingegangen. Den Abschluss dieses Abschnittes bildet die Beschreibung des entstandenen Prototyps. Abschnitt 5 zeigt den aktuellen Stand der Technik auf und stellt verwandte Arbeiten vor. Eine Zusammenfassung und ein Ausblick auf weitere Forschungsschwerpunkte zur Informationsfusion beschließen die Arbeit.

2 Informationsfusion – Begriff und Anwendungen

Unter Informationsfusion wird im hier dargestellten Umfeld¹ der Prozess der Integration und Interpretation von Daten aus heterogenen, verteilten Quellen sowie die darauf aufbauende Konstruktion von Modellen für einen bestimmten Problembereich mit dem Ziel der Gewinnung von Informationen einer neuen, höheren Qualität bezeichnet. Der Begriff der Wissensentdeckung in Datenbanken wird dabei neben dem Zugriff auf verteilte, heterogene Datenbestände, der Visualisierung von Zwischen- und Endergebnissen sowie der Wissensakquisition aus natürlichsprachlichen Quellen um die interaktive Komponente und die Verzahnung der Teilschritte erweitert. Auf der Basis dieser neu generierten Informationen können Anwender aus verschiedenen Gebieten in die Lage versetzt werden, fundierte Entscheidungen zu treffen. Somit ist der Prozess durch die Entdeckung von für den Anwender nützlichen, interessanten Informationen getrieben. Diese Definition erklärt die Informationsfusion als ein interdisziplinäres Gebiet, welches auf Methoden und Techniken verschiedener Bereiche, wie z. B. Datenbanken, Statistik, Maschinelles Lernen und Visualisierung zurückgreift.

Der Fusionsprozess beinhaltet dabei die verschiedenen Aspekte Datenzugriff, Datenintegration, Analyse, Verdichtung, Präsentation und Weiterverarbeitung sowie die Verwaltung der Metadaten. Eine Anforderungsanalyse und eine Beschreibung der einzelnen Bereiche sind in [14] gegeben.

Die Schritte der Integration und Analyse der Daten sowie die Abhängigkeiten untereinander können formal durch die Verwendung eines Graphen modelliert werden. In diesem Graphen repräsentieren die Knoten Datenquellen beziehungsweise Operationen auf diesen Quellen, während die Kanten die Aufeinanderfolge der Operationen und Quellen beschreiben. Somit beschreibt ein solcher Graph einen Fusionsprozess und dient im Weiteren als Modell für ein *Worksheet*, welches die verschiedenen Sichten auf den Prozess modelliert. Hierbei kann der Fusionsgraph einmal direkt grafisch modelliert beziehungsweise implizit durch die Anwendung der verschiede-

nen Operationen auf die Daten in einer „Spreadsheet“-Ansicht erzeugt werden.

In vielen betriebswirtschaftlichen und wissenschaftlichen Anwendungsgebieten werden Aufbereitung und Präsentation von Daten zur Unterstützung von Entscheidungsprozessen benötigt, die durch die Informationsfusion unterstützt werden können. Exemplarisch sei die Analyse von Gensequenzen aus verschiedenen Gen- und Stoffwechseldatenbanken in der Bioinformatik (*Comparative Genomics*) genannt. Ein weiteres Anwendungsszenario, aus dem das im folgenden Abschnitt eingeführte Beispiel entnommen wurde, wird durch die Analyse von Konto- und Kundendaten in Kreditinstituten zum Zweck des *Database Marketing* gebildet [15, 18].

3 Beispiel

In diesem Abschnitt wird beispielhaft ein Anwendungsszenario beschrieben, in dem mit Hilfe der Informationsfusion abwanderungsgefährdete Kunden eines Kreditinstituts identifiziert werden sollen. Mit dieser Information und den Gründen für eine potenzielle Abwanderung kann derselben in bestimmten Fällen durch geeignete Maßnahmen entgegengewirkt werden.

Datensammlung und Integration. Das betrachtete Institut verfüge über ein Data Warehouse, in dem historische Kunden- und Kontodaten vorliegen. Außerdem existiere eine Datenbasis mit ehemaligen Kunden, die ihre Kundenbeziehung beendet haben. In dieser liegen auch, soweit bekannt, die Gründe für diese Beendigung bzw. das Ziel der Abwanderung vor. In einem ersten Schritt kann so ein Datensatz erstellt werden, der sowohl abgewanderte wie auch nicht abgewanderte Kunden mit einigen kundenspezifischen Kennzahlen wie den Volumina der verschiedenen Produktarten integriert. Dieser hat etwa das in Tabelle 1 dargestellte Aussehen.

Bereinigung und Anreicherung. Um die Gründe für die Abwanderung von Kunden erkennen zu können, muss dieser Datensatz um Abwanderungen bereinigt werden, die nicht im Einflussbereich des Instituts liegen. So wurde beispielsweise die Beziehung zu dem Kunden 25894 durch dessen Tod beendet, weshalb sie keine Relevanz für die Klassifizierung und Abhängigkeitsanalyse abwanderungsgefährdeter Kunden besitzt. Ein solcher Schritt kann oft nur mit Hilfe manueller Eingriffe des Analytikers oder durch zusätzliche Berücksichtigung weiterer Informationsquellen durchgeführt werden. Das Ergebnis dieser Bereinigung wird mit sozio-demographischen Informationen angereichert, um mehrere Kriterien für eine spätere Klassifizierung zu erhalten. So werden Kundendaten wie das Alter und Informationen aus Selbstbeurteilungen der den Kunden betreuenden Geschäftsstellen mit einbezogen (Tabelle 2). Diese geschäftsstellenspezifischen Informationen sind von 9 (sehr gut) bis 1 (sehr schlecht) skaliert und umfassen die folgenden Kriterien:

- Verkehrsmäßige Lage,
- Räumliche Verhältnisse,
- Konkurrenzaktivitäten,
- Siedlungsstruktur und
- Einwohnerentwicklung.

¹ In der Literatur wird der Begriff der Informationsfusion auch für das Zusammenführen von Sensordaten verwendet [5, 33], hier jedoch liegt der Fokus auf der Fusion von Daten aus Datenbanken.

Tabelle 1. Historische Kundenentwicklung (über drei Jahre)

1999				2000				2001				AbwGrund
Kunde	Giro	Spar	...	Kunde	Giro	Spar	...	Kunde	Giro	Spar	...	
12345	207,20	6000,00		12345	-800,24	7000,00		12345	2198,05	8000,00		
13521	3820,48	4832,99		13521	12,78	0,00						Bank A
17846	1457,58	8569,47		17846	5,46	0,00						Bank B
25894	526,92	7894,15		25894	632,14	8314,02						Tod
29587	-1258,49	0,00		29587	-4897,15	0,00		29587	-3874,95	0,00		
⋮												

Tabelle 2. Bereinigte und angereicherte Kundenentwicklung (über zwei Jahre)

2000						2001						AbwGrund
Kunde	Giro	Spar	Alter	Lage	...	Kunde	Giro	Spar	Alter	Lage	...	
12345	-800,24	7000,00	47	7		12345	2198,05	8000,00	48	7		
13521	12,78	0,00	21	3								Bank A
17846	5,46	0,00	29	2								Bank B
29587	-4897,15	0,00	53	6		29587	-3874,95	0,00	54	6		
⋮												

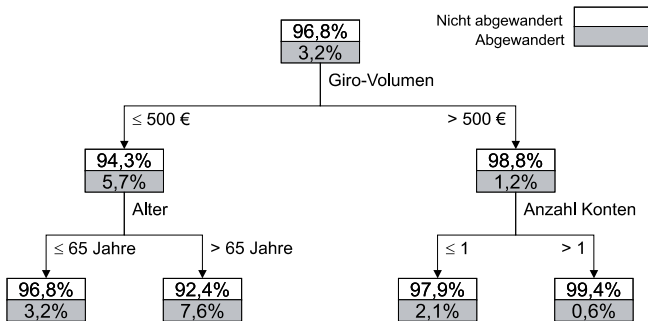


Abb. 1. Entscheidungsbaum von Kundenabwanderungen

Kriterienerstellung. Dieser Datensatz wird nun als Trainingsmenge verwendet, um im Data Mining-Schritt mit Klassifikationen und sequenziellen Mustern die abgewanderten Kunden von den anderen zu unterscheiden. Ergebnisse der Klassifikation könnten bspw. Entscheidungsbäume (Abb. 1) bzw. die folgenden Merkmale überdurchschnittlich abwanderungsgefährdeter Kunden sein (vgl. [18]):

- Kunden älter als 65 Jahre,
- nur eine aktive Kontoverbindung,
- geringes Volumen (unter 500 €),
- reine Anlage- bzw. Finanzierungskunden und
- kurze Dauer der Geschäftsbeziehung (unter 5 Jahre).

Sequenzielle Muster ähneln den Assoziationsregeln, nur dass sie Entwicklungen im Zeitablauf abbilden.

Kriterienanwendung. Mit diesen Ergebnissen kann ein Kundenbestand in Hinblick auf abwanderungsgefährdete Kunden untersucht werden. Allerdings stellt nicht jeder Abbruch einer Kundenbeziehung einen monetären Verlust für die Bank dar. Daher sind in einem weiteren Schritt die profitablen von den nicht profitablen Kundenverbindungen zu trennen. An dieser

Stelle werden weitere Kriterien erstellt. Hierzu kann beispielsweise eine Potenzialanalyse nach dem *Customer Lifetime Value*-Prinzip durchgeführt werden [16].

Nachbereitung. Durch einen weiteren Fusionsprozess können somit abwanderungsgefährdete Kunden, die eine lukrative Entwicklung der Kundenbeziehung erwarten lassen, extrahiert werden. Da auch die Gründe bzw. die Richtung der Abwanderung vermutet werden können (beispielsweise schlechte Lage der Filiale und Abwanderung hin zu einer Direktbank), kann dieser in einem Teil der Fälle gezielt entgegen gewirkt werden. Die Abbildung 2 zeigt den gesamten Prozess im Überblick, wobei die genutzten und entstandenen Tabellen, die Operationen bzw. Unterprozesse sowie die erzeugten Modelle (Kundenklassifikation, Kundenpotential) dargestellt werden.

4 Werkzeugunterstützung für die Informationsfusion

Eine enge Verzahnung der Integration heterogener Daten mit ihrer Aufbereitung und Analyse bildet die Basis der Informationsfusion. Das Ineinandergreifen der einzelnen Bestandteile ermöglicht die interaktive und iterative Arbeitsweise innerhalb des gesamten Prozesses – von Integration bis Evaluation der Daten. Ein solcher Prozess benutzt Methoden aus einem Vorrat von Werkzeugen, die innerhalb der Workbench angeboten werden.

4.1 Architektur

Eine Workbench zur Unterstützung der Informationsfusion muss eine effiziente und interaktive Analyse großer, zum Teil heterogener Datenbestände ermöglichen. Diese Aufgaben umfassen die Definition und Ausführung von Anfragen, die Transformation von Daten sowie die Anwendung von Analyseoperationen und die Visualisierung der Zwischen- und Ender-

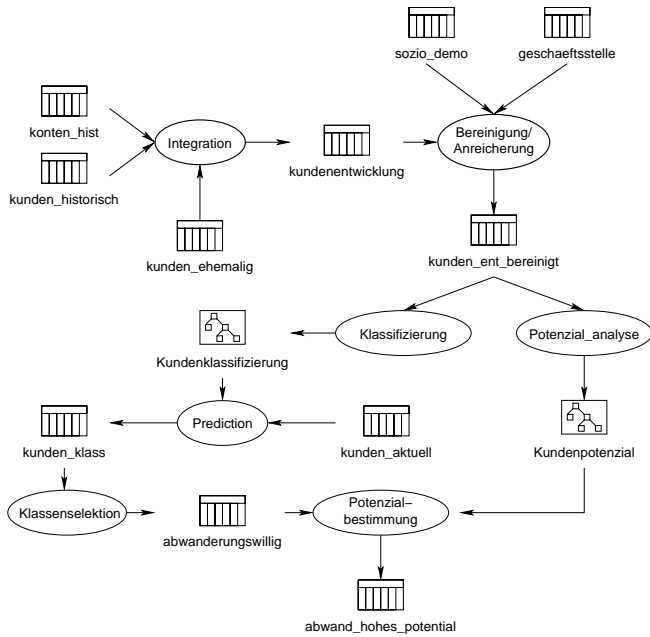


Abb. 2. Beispielprozess

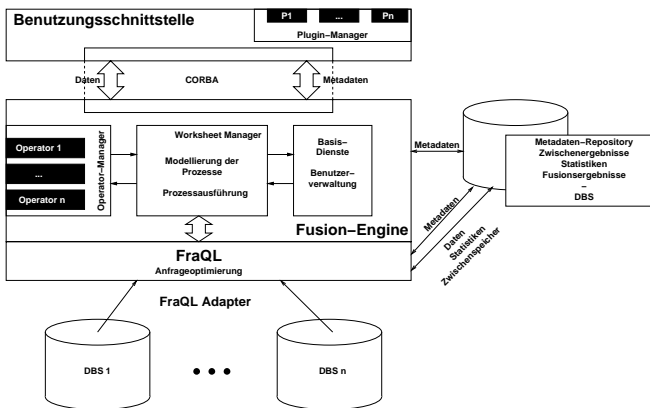


Abb. 3. Architektur der Workbench

gebnisse. Vergleichbare Anforderungen sind auch in OLAP-Anwendungen zu finden, so dass für die Fusionsworkbench ein den OLAP-Systemen ähnlicher Architekturansatz gewählt wurde (vgl. Abb. 3).

Die Basis wird von der *Fusion-Engine* gebildet, die im Kern aus einem Multidatenbank-Anfragesystem (in Abb. 3 die FRAQL-Schicht²) besteht. Dieses Anfragesystem ermöglicht einen transparenten Zugriff auf verschiedene Datenquellen und stellt Mechanismen zu deren Integration bereit [40,43]. Die Fusion-Engine umfasst weiterhin eine lokale Datenbank für temporäre Daten (z.B. Materialisierungen und Metadaten) und Ergebnisse sowie die eigentlichen Fusionsoperatoren, die ähnlich gespeicherten Prozeduren direkt auf den integrierten Datenbeständen ausgeführt werden können. Die Fusionsoperatoren werden vom Operator-Manager dynamisch in die Workbench eingebunden. Der Worksheet-Manager verwaltet komplette Fusionsprozesse, indem die Abhängigkeiten und Zustände einzelner Fusionschritte verarbeitet werden. Hier-

bei können voneinander unabhängige, ausführbare Operationen (es existieren keine Vorgänger oder alle Vorgängeroperationen sind bereits erfolgreich ausgeführt worden) parallel abgearbeitet werden.

Die Benutzungsschnittstelle (GUI) wird durch das Workbench-Frontend bereitgestellt. Dieses ist zunächst ein grafisches Analyse- und Definitionswerkzeug. So können interaktiv Integrations- und Fusionsoperationen ausgeführt, Anfragen formuliert und die Ergebnisse visualisiert werden. Die angebotene Funktionalität hängt dabei von den Rechten des angemeldeten Anwenders ab. Unter anderem wird zwischen Prozessnutzer (zum Beispiel ein Controller in einem Kreditinstitut) und Prozessersteller (Knowledge-Ingenieur) unterschieden. Zusätzlich können durch die Trennung von GUI und Fusion-Engine spezialisierte Werkzeuge zur Programmsteuerung eingesetzt werden. So ist denkbar, für spezielle Nutzer ein Frontend zu entwickeln, welches lediglich die von ihnen geforderte Funktionalität, nicht aber Administrationsmöglichkeiten anbietet.

Die Architektur der Workbench ist mit der Trennung in Fusion-Engine und Frontend mit Ansätzen aus dem OLAP-Bereich vergleichbar. Ein wesentlicher Unterschied besteht jedoch darin, dass die zu analysierenden Daten nicht vorab extrahiert, transformiert, bereinigt und redundant in einem Warehouse abgelegt werden. Statt dessen ermöglicht die Verwendung eines Multidatenbank-Anfragesystems innerhalb der Fusions-Engine den transparenten Zugriff auf die Quellen und die Anwendung von Transformations- und Integrationsoperationen. Auf diese Weise können einerseits erste Analysen durchgeführt werden, ohne dass zuvor Daten aufwendig migriert und transformiert werden müssen. Andererseits können die aktuellen Daten betrachtet werden. So lassen sich relevante Datenausschnitte selektieren und Operationen parametrisieren. Für die tiefere Analyse können die Ergebnisse einzelner Schritte anschließend materialisiert werden, um so eine xeffiziente Ausführung zu erreichen.

4.2 Metadatenverwaltung

Für eine flexible Integration und Analyse mit Hilfe einer Reihe von verschiedenen Werkzeugen ist eine zentrale Metadatenverwaltung von entscheidender Bedeutung. Dazu müssen Daten über die Objekte *Datenquellen*, *Operatoren* und *Worksheets* modelliert werden.

Datenquelle. Unter Datenquellen werden zunächst Eingangsdaten in Form von Relationen oder Dateien verstanden. Ergebnisse eines Fusionschritts können ihrerseits Datenquellen eines nachfolgenden Schritts bilden.

Die während der Datenanalyse entdeckten Muster oder Hypothesen werden als Modelle bezeichnet. Diese werden im globalen Schema abgelegt. Die Modelle beschreiben neben den gefundenen Mustern auch deren Qualität über eine Interessantheitsfunktion. Weiterhin wird für jedes Modell dessen mögliche Weiterverwendung, wie z.B. die Art der Daten, die klassifiziert werden können, vermerkt.

Operator. Operatoren stehen jeweils für eine Aktion innerhalb des Fusionsprozesses. Sie werden neben ihrem Namen durch ihre Ein- und Ausgabeparameter beschrieben. Zur Auswahl eines Operators wird dieser in eine Klassenhierarchie eingeordnet, die die Verwaltung der Operatoren strukturiert.

² FRAQL steht für **F**ederated **Q**uery **L**anguage.

Hierdurch kann beispielsweise zwischen Analyse- und Integrationsoperatoren unterschieden werden. Eine weitere Klasse von Operatoren sind die „Routing“-Operatoren, die mit Hilfe von Abhängigkeiten und Bedingungen den Workflow innerhalb eines Worksheets steuern. Von jedem Operator-Typ können in einem Worksheet beliebig viele, unterschiedlich parametrisierte Instanzen zum Einsatz kommen. Diese nutzen und erzeugen Relationen und Modelle.

Worksheet. Ein Worksheet beinhaltet einen Fusionsprozess. Wie Operatoren haben auch Worksheets Ein- und Ausgabeparameter, wodurch die Schachtelung von verschiedenen Prozessen ermöglicht wird. In einem Prozess werden Operatoren und Datenquellen verknüpft und somit die Erstellungsschritte einer bestimmten Datenquelle beschrieben. Eine Worksheet-Instanz stellt wiederum die vollständig parametrisierte Form eines Prozesses dar, welche in andere Prozesse als Teilprozess eingebettet werden kann.

4.3 Aufbereitung und Integration der Daten

Bevor die Analyseoperationen ausgeführt werden können, müssen alle dafür notwendigen Daten und Modelle in das einheitliche, (objekt-)relationale Modell abgebildet werden, welches die Operatoren verarbeiten können. In den Aufbereitungs- und Integrationsschritten werden die Datenquellen in dieses Modell transformiert, wobei die Datenqualität kontrolliert und gegebenenfalls verbessert werden muss. Ohne qualitativ hochwertige Daten lassen sich keine relevanten und zur Prognose geeigneten Analyseergebnisse ableiten. Für diesen Schritt sind oft manuelle Eingriffe des Analysten notwendig.

Im Data Warehouse-Bereich werden Data-Cleaning-Werkzeuge bzw. Programme zur Extraktion, zur Transformation und zum Laden von Daten (ETL) zur Aufbereitung und Integration der Daten benutzt. Um diese Aktivitäten zu ermöglichen, muss ein Werkzeug folgende Eigenschaften aufweisen:

Integration der Methoden. Mit Hilfe verschiedener Aktionen und Algorithmen zur Konflikterkennung und -lösung wird die Aufbereitung der Daten durchgeführt. Da sich verschiedenartige Konflikte gegenseitig bedingen können und somit nicht sofort zu erkennen sind, ist eine Integration der Werkzeuge zur Lösung unterschiedlicher Integrationskonflikte in einem System notwendig.

Schnelle Reaktionszeiten. Die Erkennung und Auflösung von Konflikten erfordert einen Dialog mit dem Anwender. Um diese interaktive Arbeitsweise zu ermöglichen, müssen die Werkzeuge erste Ergebnisse der Aufbereitungsschritte schnell an den Anwender ausgeben, so dass dieser sie in einem frühen Stadium beurteilen kann. Hierzu werden zunächst Stichproben aus der Datenmenge untersucht, um anschließend die erstellten Transformationsschritte auf den gesamten Datenbestand (im Idealfall iterativ) auszuweiten.

Grafische Benutzerführung. Die Interaktion mit den Werkzeugen soll möglichst durch eine durchgängige grafische Benutzerführung erleichtert werden. So können die Zeiten zur Einarbeitung in komplexe Programmumgebungen verringert sowie die Entdeckung und Lösung von Konflikten während der Aufarbeitung und Integration der Daten erleichtert werden. Zur Unterstützung der Iteration im Integrationsprozess

ist eine Undo-Funktion von zentraler Bedeutung. Insbesondere bei komplexen Operationen auf großen Datenbeständen ist diese im Normalfall nicht trivial zu bestimmen. Folglich sollten alle Aktionen zunächst virtuell ablaufen und nicht sofort materialisiert werden, damit deren Auswirkungen vorab eingeschätzt werden können.

Die Integration der Integrationswerkzeuge erfolgt über die Multidatenbanksprache FRAQL, welche Primitive zur Integration und Aufbereitung der Daten bereitstellt. Hierbei können alle Integrationskonflikte [13] behandelt werden. Dazu werden Konzepte zur Erweiterung von DBMS benutzt, z.B. nutzerdefinierte Aggregat- und Gruppierungsfunktionen. Weiterhin existieren erweiterte Join- und Union-Operatoren sowie Konzepte zur Lösung von Metadaten-Konflikten. Somit existiert ein Framework für Datenintegration und -aufbereitung auf Basis der Multidatenbanksprache FRAQL [43], das von der Workbench benutzt wird.

Zur Verbesserung der Antwortzeit erfolgt die Anwendung der Integrationsoperatoren zunächst auf einer Stichprobe des gesamten Datenbestandes. Diese wird durch bekannte Sampling-Verfahren generiert [47]. Diese Verfahren wurden in die dem System zu Grunde liegende Datenzugriffsschicht integriert, um die Effizienz weiter zu steigern [11,35]. Hierdurch kann die Zugriffsschicht bereits eine Optimierung durchführen und die Menge der übertragenen und bearbeiteten Daten limitieren [42]. Allerdings können auf einer Stichprobe nicht alle Konflikte erkannt werden: Zur Entdeckung von Ausreißern und Datenfehlern muss weiterhin die gesamte Datenmenge betrachtet werden.

Zur Erstellung der Integrationssichten ist eine grafische Benutzerführung notwendig. Diese wird in die Workbench eingebettet und somit die Verbindung zur Datenanalyse geschaffen. Für die Benutzerinteraktion wurde eine Spreadsheet-Ansicht gewählt, die eine Stichprobe der Daten anzeigt und die Ausführung der Integrationsoperatoren erlaubt sowie deren Einfluss auf die Stichproben-Daten darstellt. Die Datendarstellung wird durch weitere Visualisierungsmethoden (s. Abschnitt 4.5) ergänzt.

Alle Operationen fließen somit zunächst in eine Sichtdefinition ein, sind also effizient zu modifizieren, und eine Undo-Funktion ist einfach zu implementieren. Eng damit verbunden ist die Berücksichtigung von temporalen Eigenschaften des Fusionsprozesses. Zur grafischen Unterstützung ist diese in einer Visualisierung des Prozesses selbst sowie seiner zu unterschiedlichen Zeitpunkten anfallenden Ergebnisse zu berücksichtigen. Ein entsprechendes Modell, welches auch die zeitlichen Eigenschaften von Nutzerinteraktionen berücksichtigt, wird in Abschnitt 4.6 beschrieben.

Mit dieser Form der interaktiven Integration auf Basis von Stichproben können frühzeitig Konflikte in den Daten erkannt werden. Hat der Anwender unter Verwendung der vorgestellten Integrationsmechanismen die Daten entsprechend seinen Wünschen aufbereitet, ist es möglich die Ergebnisse zu materialisieren, um weitere Analyseschritte zu beschleunigen.

4.4 Datenanalyse

In den meisten Anwendungsfällen wird allein durch die Integration verschiedener Datenquellen noch kein Gewinn erzielt. Gerade bei einer größeren Anzahl von Quellen bleiben auf

Grund des resultierenden Datenvolumens interessante Aspekte oft verborgen. Daher sind die integrierten Daten weiter zu analysieren, um etwa Muster, Tendenzen, Regelmäßigkeiten oder Ausreißer aufzudecken. Das Suchen von Mustern und Zusammenhängen in Daten (*Data Mining*) als Teil des *Knowledge Discovery in Databases* (KDD) ist ein Forschungsbereich, der zunehmend an Bedeutung gewinnt [27]. Dabei befindet er sich an der Schnittstelle zwischen verschiedenen Bereichen wie beispielsweise Statistik, Datenbanken, Entdeckung von Mustern, Optimierung und Visualisierung [2]. In der Vergangenheit wurde eine Vielzahl von Verfahren entwickelt [19]. In Verbindung mit Zugriffs- und Integrationsmechanismen für heterogene Datenquellen versprechen diese Techniken neue, vielfältige Einsatzmöglichkeiten.

Ein Defizit aktueller Ansätze zur automatischen Datenanalyse in großen Datenbeständen – im Vergleich zu OLAP-Anwendungen, die eher eine anwendergesteuerte, navigierende Form unterstützen – ist die unzureichende Kopplung zum Datenbanksystem. So arbeiten viele Data Mining-Tools hauptsächlich speicherbasiert und sind damit zwar sehr schnell, aber hinsichtlich der zu untersuchenden Datenmenge beschränkt. Obwohl vielfach die zu analysierenden Daten bereits in DBMS vorliegen, werden diese selten für Data Mining-Operationen eingesetzt. Ein Grund hierfür ist, dass moderne DBMS kaum Unterstützung für Data Mining-Verfahren in Form spezieller Operatoren oder Optimierungsstrategien anbieten. Ein neues Problem, welches durch den erhöhten Abstraktionsgrad gegenüber OLAP entsteht, ist, wie die Anfragen zu formulieren sind: Wie kann ein Anwender dem System mitteilen, wonach es suchen soll, wenn das Ziel der Suche noch nicht genau bekannt ist? Hierzu wurden auch für diesen Bereich Anfragesprachen wie DMQL [26] und MSQL [29] entwickelt. Der Standardentwurf SQL/MM Part 6 enthält ebenfalls eine Schnittstelle für Data-Mining-Abfragen mittels SQL [45]. Eine enge Kopplung von Data Mining-Verfahren und DBMS bietet eine Reihe von Vorteilen, wie die Nutzung der durch das DBMS bereitgestellten Zugriffsstrukturen und Optimierungsstrategien, der Speicherverwaltung für die Bearbeitung großer Datenmengen sowie der ausgereiften Parallelisierungsmechanismen moderner Systeme [12]. Erst mit diesen Möglichkeiten wird ein interaktives *Ad-hoc Mining* möglich [8].

Vor diesem Hintergrund wird im Rahmen der hier vorgestellten Workbench eine enge Verbindung zwischen den Analysetechniken und der Datenbankfunktionalität angestrebt. So werden die einzelnen Analyseoperationen als SQL-Programme ähnlich gespeicherten Prozeduren implementiert und in der Fusions-Engine ausgeführt. Auf diese Weise lassen sich einzelne SQL-Anweisungen als Teil einer Analyseoperation direkt auf den Quelldaten bzw. auf den (materialisierten) Ergebnisrelationen anwenden.

Die Umsetzung von Data Mining-Verfahren auf der Basis von SQL ist eine aktuelle Herausforderung. Erste Arbeiten beschäftigen sich im Wesentlichen mit Klassifikationsverfahren [10] und der Ableitung von Assoziationsregeln [39]. Dabei wurde deutlich, dass noch Performance-Probleme bestehen, die durch neue Datenbankprimitive und Optimierungstechniken zu lösen sind [8]. Der erste Schritt ist also die Identifikation von Primitiven, die eine möglichst große Anzahl von Data Mining-Verfahren unterstützen. Dazu sind die Verfahren zu analysieren und gemeinsame, laufzeitintensive Anfragetypen zu bestimmen.

Tabelle 3. CC-Tabelle

attrib_name	attrib_value	class	count
Giro	100,00	abgewandert	687
Spar	2000,00	nicht abgew.	154.830
Alter	64	nicht abgew.	87.533
Lage	5	abgewandert	34
⋮	⋮	⋮	⋮

In Abschnitt 3 wurde der Entscheidungsbaum als ein für das Beispiel relevantes Data Mining-Verfahren genannt. Anhand dieses Beispiels soll im Folgenden die Entwicklung von oben angesprochenen Datenbankprimitiven aufgezeigt werden. Für die Erstellung von Entscheidungsbäumen sind viele Algorithmen wie ID3, C4.5, PUBLIC, SLIQ oder SPRINT bekannt, die auf dem Greedy-Prinzip basieren. In der ersten Phase wird der Baum aufgebaut. Dazu wird an der Wurzel mit dem gesamten Datensatz begonnen und ein Kriterium gesucht, anhand dessen der Datensatz am besten partitioniert wird. Für das Beispiel wird das Giro-Volumen identifiziert. Die Definition des Kriteriums ist in den einzelnen Entscheidungsbaum-Verfahren unterschiedlich. Anschließend wird der weitere Baum rekursiv mit den entsprechenden Partitionen aufgebaut. In der zweiten Phase werden Teilbäume des Baumes abgeschnitten (Pruning), um eine zu genaue Anpassung an die Trainingsdaten (Overfitting) zu verhindern.

Die Bestimmung des Punktes, an dem eine Aufspaltung stattfindet, ist während der Aufbauphase der aufwändigste Teil. Hierbei müssen für jeden Knoten alle Tupel bestimmt werden, die zu der entsprechenden Partition gehören. Anschließend wird für diese Partition das Aufspaltungskriterium für alle noch nicht berücksichtigten Attribute berechnet. Hierzu ist eigentlich kein Zugriff auf die Basisdaten notwendig, da lediglich die Häufigkeiten von Attributkombinationen relevant sind. Diese Informationen können auch aus einer einfachen Tabelle entnommen werden, die aus Attributnamen, Attributwert, Klassenzugehörigkeit und Anzahl besteht. Bei numerischen Attributen kann eine Diskretisierung sinnvoll sein, um die Anzahl der Daten gering zu halten. Eine solche Struktur wird als *Class Count-Tabelle* (CC-Tabelle) [10] oder in ähnlicher Form als *Attribute-Value-Class-Group* [21] bezeichnet. Für das Beispiel könnte sie etwa das in Tabelle 3 dargestellte Aussehen haben.

Eine solche Tabelle kann mit einer Anfrage der folgenden Form erzeugt werden [10]:

```

select 'A1' as attrib_name, A1 as attrib_value,
       C as class, count(*)
from BaseRelation
where <condition>
group by A1, C
union all
select 'A2' as attrib_name, A2 as attrib_value,
       C as class, count(*)
from BaseRelation
where <condition>
group by A2, C
union all
...

```

Die meisten Optimierer von aktuellen DBMS sind nicht in der Lage, einen Ausführungsplan zu generieren, der diese Anfrage mit einem einzigen Scan über die Basisrelation verarbeitet. Somit stellt diese Form von Anfragen einen geeigneten Kandidaten für eine Erweiterung der Fähigkeiten von Optimierern dar.

Neben der Berechnung der Häufigkeiten von Attributkombinationen wurde oben auch die Bestimmung der Tupel, die zu den einzelnen Partitionen gehören als gemeinsamer Teil aller Entscheidungsbaumverfahren genannt. Dies wird typischerweise mittels sogenannter *Partial-Match*-Anfragen durchgeführt. Dies sind Anfragen, die eine Bedingung der Form $P_1 \wedge P_2 \wedge \dots \wedge P_m$ enthalten, wobei P_i ein Prädikat $A_j \theta v$, $\theta \in \{<, \leq, >, \geq, =, \neq\}$ und $v \in \text{dom}(A_j)$ ist. Um beispielsweise die Partition des linken Knotens für den Entscheidungsbaum in Abbildung 1 zu bestimmen, ist folgende Anfrage zu formulieren:

```
select *
from BaseRelation
where giro.volumen ≤ 500 and alter ≤ 65
```

Hier kann mit einer speziellen Zugriffsstruktur, die Partial-Match-Anfragen auf mehrdimensionalen Daten besonders gut unterstützt, ein Vorteil gegenüber den herkömmlichen Indexstrukturen erzielt werden. In [41] wurde das multidimensionale Hashing (MDH) als eine solche Zugriffsstruktur identifiziert.

Neben der Unterstützung für Entscheidungsbaumverfahren ist die Untersuchung weiterer Data Mining-Algorithmen auf der Basis von SQL notwendig. Hierzu zählen Algorithmen zur Aufdeckung von Assoziationsregeln und Clustering-Verfahren.

4.5 Visualisierung

Das Verständnis großer, meist multidimensionaler Datenbestände und die Extraktion von Informationen daraus setzt ihre umfassende Exploration voraus. Das Erkennen von Mustern und Trends, die Navigation im Datenbestand sowie das Auffinden von Beziehungen zwischen einzelnen Datensätzen erweist sich hierbei schnell als schwierig, oftmals als unmöglich [17]. Es ist somit Aufgabe einer die Exploration unterstützenden Visualisierung, große Datenmengen handhabbar zu gestalten, die Betrachtung sowie Manipulation von Objekten in der Datenbasis zu ermöglichen und die Darstellung so aufzubereiten, dass Information durch den Benutzer leichter auffindbar wird.

Methodisch stehen zu Visualisierungszwecken neben der klassischen Tabellendarstellung relationaler Daten sämtliche Methoden der wissenschaftlichen Visualisierung zur Verfügung. Ein umfassender Überblick über einen derartigen Methodenkatalog wird in [44] gegeben. Im Folgenden sei exemplarisch eine Scatterplotdarstellung betrachtet.

Bei einem Scatterplot werden so genannte Glyphs als Repräsentationen für einzelne Eingabedatensätze in einem kartesischen Koordinatensystem angeordnet. Verschiedene Eingabedimensionen können dabei auf die verfügbaren Präsentationsvariablen abgebildet werden. In klassischer Darstellung stehen als solche zur Verfügung: Position (x, y, z), Materialeigenschaften (Farbe, Textur, Transparenz) sowie die geometrische Form des Glyphs. Zur Erweiterung dieses Ausdrucksrepertoires sei die Verwendung von Objektbewegungen

als eigenständige Präsentationsdimension motiviert [6,31]. Adäquat angewendet, bieten Bewegungen die Möglichkeit, mit geringem kognitiven Aufwand ausdrucksstarke Darstellungsmuster zu erstellen [36].

In [31] wird die Verwendung von Bewegung für Visualisierungszwecke vorgestellt. Die möglichen Bewegungsarten teilen sie ein in drei Basiskategorien:

Translation (T) beschreibt eine andauernde Veränderung einer Objektposition. Da verschiedene Szenenobjekte in einem räumlichen Zusammenhang zueinander stehen, helfen oszillierende Bewegungen dabei, eine Invalidierung der Szenenkohärenz zu verhindern.

Rotation (R) erfolgt um variabel wählbare Rotationszentren. Drehungen um das Objektzentrum verursachen nur geringfügige Szeneninkohärenzen, sind allerdings auch kognitiv schwach und erfordern somit eine hohe Wahrnehmungsleistung des Anwenders. In Analogie zur Translation können auch Rotationen oszillierend erfolgen.

Formveränderung (D) drückt sich entweder in wiederholenden Skalierungen oder in morphenden Formwechseln aus. Während hierbei die Darstellung jedes einzelnen Objektes manipuliert werden kann, wird der räumliche Bezug zwischen verschiedenen Objekten nicht modifiziert. Ordnet sich die Bild- der Szenenkohärenz unter, bieten Formveränderungen ein ausdrucksstarkes Mittel der präemptiven Objekthervorhebung.

Der Bewegungskatalog ergibt sich somit als $cat = \{T, R, D\}$. Zur Bestimmung einer konkreten Bewegungsmethode ist zum einen die Kombination der einzelnen Bewegungsarten des Kataloges möglich und zum anderen eine Belegung des Parameterraumes $P = \{p_1, \dots, p_m\}$ zu definieren. Eine konkrete Methode zur Bewegungsdarstellung ergibt sich somit als $method = \{C, P\}$ mit $C = \{c_1, \dots, c_n\}$; $c_i \in cat, 1 \leq i \leq n$. Der Satz verfügbarer Präsentationsvariablen wird durch den Einsatz von Objektbewegungen somit erweitert. In Abhängigkeit der möglichen Parameterisierung einzelner Bewegungsmethoden können mehr Dimensionen der zu visualisierenden Datensätze repräsentiert werden, als es bei statischen Darstellungen möglich ist.

4.6 Temporale Aspekte der Visualisierung

Eine der zentralen Herausforderungen bei der Erstellung von Systemen, die Verhalten einer (realen oder konstruierten) Welt wiedergeben, ist die Repräsentation von Zeit. So ist beispielsweise die Verwendung jeder einzelnen Bewegungsmethode parameterisierbar und zeitlich zu beschränken. Zum einen erlaubt dies, die Anzahl gleichzeitig verwendeter Bewegungsdarstellungen zu limitieren. Zum anderen wird hierdurch die zeitlich beschränkte Hervorhebung bestimmter Auswirkungen der Ausführung des Fusionsprozesses ermöglicht. Nicht zuletzt dient eine temporale Beschränkung von Objektbewegungen der Interaktionsunterstützung, da eine Selektion bewegter Objekte mit Schwächen verbunden ist und eine höhere kognitive Leistung erfordert als die Selektion statischer Objekte.

Die Definition des Fusionsprozesses spiegelt den zeitlichen Ablauf unterschiedlicher Methoden und ihrer Ergebnisse wider. Verschiedene Operatoren zur Datenaufbereitung

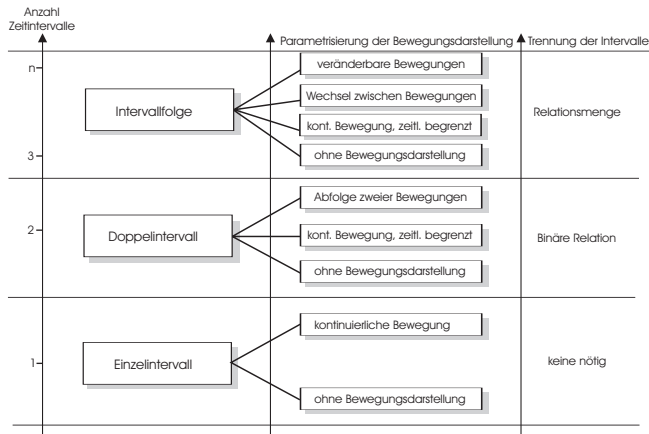


Abb. 4. Ebenenmodell für zeitabhängige Bewegungskarte

und -analyse, wie sie in den vorangegangenen Abschnitten diskutiert wurden, bedingen einander oder werden jeweils unabhängig in zeitlicher Folge ausgeführt. Eine visuelle Repräsentation dieses Prozesses sowie insbesondere seiner Ausführungsergebnisse sollte somit um temporale Aspekte angereichert werden, um zeitliche Abhängigkeiten dem Benutzer zugänglich machen zu können. Anderweitig verdeckte Strukturen des Fusionsprozesses können somit offensichtlich gemacht werden.

Als interaktives System ist die Workbench auch selbst durch temporale Parameter charakterisiert. Nutzerinteraktion drückt sich in Ereignissen aus, die nacheinander vom System aufgenommen und bearbeitet werden. Jedes Interaktionsergebnis ist ein potenzieller Aktionsauslöser und somit in der Lage, den Systemzustand zu modifizieren. Diese Änderung bleibt gültig bis zu einem weiteren Nutzerereignis oder bis sie durch Systemvoranschritt veraltet. Daraus resultiert, dass der Effekt jedes nutzerbasierten Ereignisses für ein spezifisches Zeitintervall gültig ist.

Die Verwaltung der Intervalle erfolgt auf der Basis der temporalen Relationenalgebra von Allen [3]. Anlehnend an Frekzas Betrachtungen in [20] können die zeitlichen Abhängigkeiten allerdings auch mit einer eingeschränkten Relationenmenge vollständig modelliert werden. Die Beziehung zwischen zwei gegebenen Intervallen I_i und I_j lässt sich somit beschreiben durch $I_i \circ I_j, \circ \in \{<, \leq, =, \geq, >\}$.

Zur Illustration sei das Ebenenmodell aus Abbildung 4 betrachtet. In Abhängigkeit der modellierten Zeitintervalle schildert es die Möglichkeit, verschiedene Darstellungen zur Zeitrepräsentation zu verwenden. Auf Grund ihrer zeitlichen Variabilität sind die in Abschnitt 4.5 eingeführte Bewegungen gut geeignet, um temporale Charakteristika von Objekten zu repräsentieren. Der Einsatz von Bewegungen ist abhängig von der Anzahl der für spezifische Objektbeziehungen bestehenden Zeitintervalle. Beispielhaft sei das Doppelintervall betrachtet. Zwei verschiedene Möglichkeiten zur Bewegungskarte bieten sich hier an: zum einen die zeitlich begrenzte Darstellung einer kontinuierlichen Bewegung, zum anderen die Abfolge zweier unterschiedlicher Bewegungskarten. Ihre Dauer ergibt sich aus den Intervallgrenzen. Diese werden durch die drei Zeitpunkte t_1 (Beginn des ersten Intervalls), t_2 (Ende des ersten und Beginn des zweiten Intervalls) sowie t_3 (Ende des dritten Intervalls) festgelegt. Die

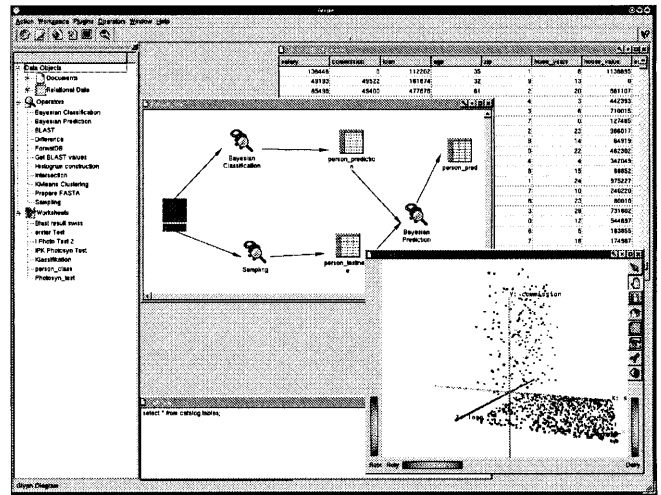


Abb. 5. Prototyp mit Darstellung eines einfachen Prozessgraphen sowie verschiedenen Ansichten auf einen exemplarischen Datensatz

Dauer der zeitlichen Begrenzung einer kontinuierlichen Darstellung beträgt somit $t_2 - t_1$. Da Intervallrelationen transitiv sind, gilt $t_2 \leq t_3: t_2 - t_1 \leq t_3 - t_1$. Das verbleibende Intervall $t_3 - t_2$ kann entweder zur Darstellung des zweiten Bewegungsmusters oder aber zur Darstellung mit Hilfe klassischer Präsentationsvariablen [34] eingesetzt werden.

Für den Fall des Doppelintervalls erfolgt eine Bewegungskarte somit zeitlich begrenzt und wird nur einmalig verwendet. Wiederholte Darstellungen einzelner Muster sowie die fließende Überführung verschiedener Bewegungsmuster ineinander erfordert eine Intervallfolge.

Mit dieser Abbildung temporaler Parameter auf Bewegungskarten ist es möglich, den zeitlichen Aspekt der Metadaten direkt in die Visualisierung der zu bearbeitenden Daten zu integrieren. Eine Antwort auf die Frage der zeitbezogenen Validität ausgewählter Datensätze kann dem Anwender unmittelbar präsentiert werden. Alternativ bleibt natürlich die Verwendung der Präsentationsvariablen Bewegung als normales Visualisierungsmittel für Daten, die im Rahmen des Fusionsprozesses gewonnen wurden. In diesem Fall dient die temporale Parameterisierung primär dem Ziel der Interaktionsunterstützung. Bewegungen werden abgeschwächt oder gar aufgelöst, um dem Nutzer Zugriff auf Repräsentationsobjekte gewährleisten zu können.

4.7 Prototyp

Die Umsetzung der geschilderten Konzepte erfolgt auf Basis der in Abschnitt 4.1 skizzierten Architektur. Die Verwendung einer standardisierten Datenbank-API, des Visualization Toolkits [44] sowie Open InventorsTM als Basis der Visualisierungskomponente sowie von Qt als Grundlage für die Benutzungsschnittstelle gewährleistet eine weit reichende Plattformunabhängigkeit. Der Prototyp realisiert den Zugriff auf heterogene Quellen und Integrationsoperationen über das Anfragesystem FRAQL.

Abbildung 5 zeigt eine Beispielsitzung mit der Workbench. Der dargestellte Prozessgraph zur Bestimmung der Abfolge einzelner Fusionschritte wird interaktiv aufgebaut. Eine Übersicht über die verfügbaren Datenquellen (Relationen,

Dateien, Views, et cetera) sowie Operatoren liefert der Metadaten-Browser, im Bild links dargestellt. Exemplarisch sind als zwei Sichten auf die Relation mit den Trainingsdaten eine Tabelle und ein einfacher Scatterplot abgebildet.

Die einzelnen Bestandteile der Architektur gemäß Abb. 3 sind prototypisch bereits umgesetzt worden. Zur Vervollständigung des Prototyps wird die Integration weiterer Visualisierungsmodule sowie Fusionsoperatoren aus verschiedenen Anwendungsbereichen durchgeführt. Neben datenbankorientierten Optimierungs- und Analysefunktionen sind hier insbesondere interaktive Data-Mining-Verfahren sowie Methoden zum automatischen Lernen interessant. Erstere ermöglichen beispielsweise mittels Methoden des interaktiven Clusterings eine weiterführende Benutzerunterstützung, während letztere die Analyse dahingehend unterstützen, dass sie wiederkehrende Verhaltens- und Abhängigkeitsmuster in größeren Datensätzen zu entdecken helfen.

5 Verwandte Arbeiten

In den letzten Jahren wurden in der Literatur verschiedene Vorschläge zur Integration von heterogenen Datenquellen gegeben. Dabei lag zunächst der Schwerpunkt auf der Schemaintegration [7]. Derzeit, hervorgerufen durch die starke Verbreitung des Data-Warehouse-Konzeptes, wird stärker auf die Integration und Aufbereitung der Inhalte Wert gelegt [24]. Die vorliegende Arbeit ordnet sich im Schnittpunkt beider Bereiche ein, da zum einen die Schemakonflikte aufgelöst werden können und zum anderen, auch im Rahmen von Materialisierungen, die Datenqualität erhöht werden kann.

Einen Überblick über die Data Warehouse-Architektur und den Ablauf des Prozesses von der Extraktion aus den lokalen Quellen bis zur Auswertung der Daten wird in [9] gegeben. [1] zeigt eine Übersicht über verschiedene kommerzielle Werkzeuge, die zur Extraktion, zur Transformation und zum Laden (ETL) der Daten benutzt werden. Beispiele für grafische ETL-Systeme sind die Microsoft Data Transformation Services und die Oracle DataMart Suite.

Weitere Forschungsprojekte zur interaktiven Datenaufbereitung und -integration sind unter anderem Clio [23] und Potter's Wheel [37]. Diese haben das Ziel einer interaktiven, datenorientierten und iterativen Aufbereitung der Daten für die weitere Analyse. In [23] verwenden die Autoren hierfür als Datenbank-Middleware das Multidatenbanksystem Garlic. Potter's Wheel ist ein ähnliches Projekt für ein Framework zur Unterstützung der interaktiven Datenaufbereitung. Dieses verwendet eine grafische Benutzungsschnittstelle in Form eines skalierbaren Spreadsheets. Mit diesem kann der Anwender seine Aktionen zur Datenaufbereitung sofort auf einer Stichprobe der Daten ausführen und validieren.

Multidatenbanksysteme stellen mit ihren Möglichkeiten des Zugriffs auf heterogene Datenquellen und der Integration von Daten die Grundlage für eine virtuelle und interaktive Aufbereitung dar. Beispiele für solche Systeme sind u.a. MS-QL [22], SchemaSQL [32] oder auch das im hier vorgestellten Prototyp verwendete FRAQL [40].

Nach der Bereinigung und Integration der Daten erfolgen üblicherweise verschiedene numerische, statistische oder grafische Analysen zur Gewinnung von neuen Erkenntnissen

aus dem aufbereiteten Datenbestand. Im Data Warehouse-Bereich erfolgt diese zumeist durch OLAP-Werkzeuge, die oft gänzlich abgekoppelt von den oben genannten ETL-Werkzeugen vorliegen.

Zur Datenanalyse werden verschiedene Algorithmen und Methoden benutzt, deren Spektrum von Ad-hoc-Anfragen bis zu lang laufenden Data-Mining-Methoden reicht. Um eine effiziente Verarbeitung der Daten in einem Datenbanksystem zu ermöglichen, müssen diese Algorithmen in das DBMS integriert werden. Eine Untersuchung der verschiedenen Möglichkeiten der Integration dieser Algorithmen wurde am Beispiel der Ableitung von Assoziationsregeln in [39] durchgeführt. Das System DBMiner [25] integriert verschiedene Data-Mining-Algorithmen für On-Line Analytical Mining in großen Datenbanken bzw. Data Warehouses.

Bei der Exploration von Datenbankinhalten ist die Standardbenutzungsschnittstelle noch immer eine Tabellensicht [37]. Verschiedene Techniken wurden entwickelt, um multidimensionale Daten dem Anwender leichter zugänglich aufzubereiten [17,25,28]. Diese sind geprägt durch eine selektive Beschränkung der zu Grunde liegenden Dimensionalität zum Vorteil der besonders hervorgehobenen Darstellung von einzelnen Merkmalen der Ausgangsdaten. Zu ihrem besseren Verständnis werden große Pivottabellen somit auf mehrere kleine Tabellen aufgeteilt. Die Darstellung derselben erfolgt partiell auf visuell reichere, aber kognitiv weniger belastende Variationen. Beispiele hierfür sind Kombinationen aus Bubble Plots, parallelen Koordinaten sowie Boxengraphiken [17,25]. Alternativ werden zur Darstellung sehr großer Datensätze Abbildungen auf Volumendarstellungen eingesetzt. Dabei existiert Information im 3D-Raum und wird nicht nur in Form von 2D-Daten in den 3D-Raum abgebildet. Volumenrendering ist im Gegensatz zu herkömmlichen Renderingmethoden nicht an die Vorgabe von geometrischen Informationen gebunden. Ein weiterer Ansatz besteht in der Abbildung dreidimensionaler Würfelausschnitte im Cyberspace [4].

6 Zusammenfassung und Ausblick

Aufbauend auf den Grundkonzepten der Informationsfusion, die sich einzeln betrachtet inzwischen in einem weitgehend ausgereiften Stadium befinden, wird ein Rahmen entwickelt, der den gesamten Prozess der Informationsfusion von der Integration heterogener Datenquellen bis zur Ableitung neuer Informationen abdeckt. Dieser bietet die benötigten Basisdienste in einer einheitlichen Schnittstelle an, die es ermöglicht, dass zusätzliche Operatoren und Visualisierungsmethoden entwickelt und in das System eingebunden werden können.

Anhand von Beispielen und verschiedenen Anwendungsszenarien soll die praktische Relevanz der gebotenen Unterstützung evaluiert und erweitert werden. Somit wird der Satz von Präsentationsvariablen (Form, Farbe, Position, Transparenz) um Objektbewegungen angereichert. Außerdem werden nicht nur weitere Operatoren wie beispielsweise zusätzliche KDD-Verfahren implementiert, sondern auch die Basisdienste ergänzt und verbessert. Zur Zeit werden Datenbankprimitive erarbeitet, die Effizienzsteigerungen insbesondere von Data-Mining-Algorithmen erlauben. Weitere Teilprojekte befassen sich mit der Generierung von Samples und Zwischenergebnissen mit iterativ zunehmender Genauigkeit, um

den interaktiven Charakter der Fusionsaufgabe besser abbilden zu können.

Neben der Erweiterung der Basisdienste und der Entwicklung weiterer Operatoren ist geplant, Erkenntnisse verwandter Forschungsgebiete in die Workbench einfließen zu lassen. Denkbar sind hier Lernverfahren zur Optimierung einzelner Prozessschritte oder sogar des Gesamtprozesses, sowie Methoden der Wissensakquisition zur Einbindung natürlichsprachlicher Texte in den Fusionsprozess. Methoden des nichtfotorealistischen Renderings [46] können helfen, zum einen Unschärfe in gewonnenen Informationen auszudrücken und zum anderen das bestehende Ausdrucksrepertoire zur Darstellung temporaler Zusammenhänge zu erweitern.

Danksagung. Diese Arbeit wird gefördert von der DFG (FOR 345/1). Unser Dank gilt den anonymen Gutachten für wertvolle Kritik und Anregungen zur Verbesserung des Beitrages.

Literatur

1. Data Extraction, Transformation, and Loading Tools (ETL), <http://www.dwinfocenter.org/clean.html>, August 2000
2. Adriaans P, Zantinge D. Data Mining. Harlow 1996
3. Allen JF. Maintaining Knowledge about Temporal Intervals. Communications of the ACM, 26(11):832–843, 1983.
4. Ammoura A, Zaiane OR, Ji Y. Immersed Visual Data Mining: Walking the Walk. In: Read [38], pp. 202–218
5. Arabia HR, Zhu D (eds) Proc. of the Int. Conf. on Multisource-Multisensor Information Fusion – FUSION '98, Las Vegas, NV, 1998. CSREA Press
6. Bartram L. Enhancing Visualizations With Motion. In: Hot Topics: Information Visualization 1998, North Carolina, USA, 1998
7. Batini C, Lenzerini M, Navathe SB. A Comparative Analysis of Methodologies for Database Schema Integration. ACM Computing Surveys, 18(4):323–364, December 1986
8. Chaudhuri S. Data Mining and Database Systems: Where is the Intersection? Data Engineering Bulletin 21(1): 4–8 (1998)
9. Chaudhuri S, Dayal U. An Overview of Data Warehousing and OLAP Technology. SIGMOD Record 26(1), 1997
10. Chaudhuri S, Fayyad UM, Bernhardt J. Scalable Classification over SQL Databases. In: Proc. of the 15th Int. Conf. on Data Engineering, Sydney, Australia, pp. 470–479. IEEE Computer Society, 1999
11. Chaudhuri S, Motwani R, Narasayya VR. On Random Sampling over Joins. In: Delis A, Faloutsos C, Ghandeharizadeh S (eds.) SIGMOD 1999, Proc. ACM SIGMOD Int. Conf. on Management of Data, Philadelphia, Pennsylvania, USA, pp. 263–274. ACM Press, 1999
12. Clear J, Dunn D, Harvey B, Heytens ML, Lohman P, Mehta A, Melton M, Rohrberg L, Savasere A, Wehrmeister RM, Xu M. NonStop SQL/MX Primitives for knowledge Discovery. In: Proc. 5th ACM SIGKDD Int. Conference on Knowledge Discovery and Data Mining 1999, San Diego, CA USA, pp. 425–429, 1999
13. Conrad S. Föderierte Datenbanksysteme: Konzepte der Datenintegration. Berlin Heidelberg: Springer 1997
14. Conrad S, Saake G, Sattler K-U. Informationsfusion - Herausforderungen an die Datenbanktechnologie. In: Buchmann AP (ed.) Datenbanksysteme in Büro, Technik und Wissenschaft, BTW'99, GI-Fachtagung, Freiburg, März 1999, Informatik aktuell, pp. 307–316. Berlin: Springer 1999
15. Dunemann O. Unterstützung der Quantifizierung von Kreditrisiken mit Methoden der Informationsfusion. Präsentiert bei: 5. Internationale Tagung Wirtschaftsinformatik / 3. Tagung Informationssysteme in der Finanzwirtschaft. Augsburg, September 2001
16. Dzienziol J, Schroeder N, Wolf C. Kundenwertorientierte Unternehmenssteuerung. In: Buhl HU, Kreyer N, Steck W (eds.) e-Finance - Innovative Problemlösungen für Informationssysteme in der Finanzwirtschaft, pp. 63–86. Berlin Heidelberg New York: Springer 2001
17. Eick SG. Visualizing Multi-Dimensional Data. Computer Graphics, pp. 61–67. February 2000
18. Eickbusch J. Kundenabwanderungen in Kreditinstituten – Eine empirische Analyse mittels Data Mining-Methoden für das Privatkundengeschäft einer Großsparkasse. Inauguraldissertation, Gerhard Mercator-Universität Duisburg, Duisburg, 2000
19. Fayyad UM, Piatetsky-Shapiro G, Smyth P, Uthurusamy R (eds.) Advances in Knowledge Discovery and Data Mining. Menlo Park, CA: AAAI Press / The MIT Press 1996
20. Freksa C. Temporal Reasoning Based on Semi-Intervals. Artificial Intelligence, 52(1-2):199–227, 1992
21. Gehrke J, Ramakrishnan R, and Ganti V. Rainforest – a framework for fast decision tree construction of large datasets. In: Gupta A, Shmueli O, Widom J (eds) VLDB'98, Proc. of 24rd Int. Conf. on Very Large Data Bases, New York City, New York, USA, pp. 416–427. Morgan Kaufmann, 1998
22. Grant J, Litwin W, Roussopoulos N, Sellis TK. Query Languages for Relational Multidatabases. The VLDB Journal 2(2): 153–171 (1993)
23. Haas LM, Miller RJ, Niswonger B, Roth MT, Schwarz PM, Wimmers EL. Transforming heterogeneous data with database middleware: Beyond integration. IEEE Data Engineering Bulletin 22(1): 31–36 (1999)
24. Hammer K. Migrating Data from Legacy Systems: Challenges and Solutions. In: Barquin RC, Edelstein HA (eds.) Building, Using, and Managing the Data Warehouse, The Data Warehouse Institute Series, pp. 27–40. New Jersey, USA: Prentice Hall PTR 1997
25. Han J Towards On-Line Analytical Mining in Large Databases. ACM SIGMOD Record, (27):97–107, 1998
26. Han J, Fu Y, Koperski K, Wang W, Zaiane O. DMQL: A Data Mining Query Language for Relational Databases. In: SIGMOD'96 Workshop. on Research Issues on Data Mining and Knowledge Discovery (DMKD'96), Montreal, Canada, June 1996
27. Han J, Kamber M. Data Mining: Concepts and Techniques. San Francisco, CA: Morgan Kaufmann Publishers 2001
28. Hearst MA, Karadi C. Cat-a-Cone: An Interactive Interface for Specifying Searches and Viewing Retrieval Results using a Large Category Hierarchy. In: Proceedings of the 20th Annual International ACM/SIGIR Conference, Philadelphia, PA, July 1997
29. Imielinski T, Virmani A. MSQL: A Query Language for Database Mining. Data Mining and Knowledge Discovery 3(4): 373–408 (1999)
30. Jesse R, Geist I, Dunemann O. Konzeption einer datenbankbasierten Plattform für die Informationsfusion. Preprint 9, Fakultät für Informatik, Universität Magdeburg, Magdeburg 2001
31. Jesse R, Strothotte T. Motion Enhanced Visualization in Support of Information Fusion. In: Arabia HR (ed) Proceedings of International Conference on Imaging Science, Systems, and Technology (CISST'2001), pp. 492–497. CSREA Press, June 2001
32. Lakshmanan LVS, Sadri F, Subramanian IN. SchemaSQL – A Language for Interoperability in Relational Multidatabase Systems. In: Vijayaraman TM, Buchmann AP, Mohan C, Sarda

- NL (eds.) VLDB'96, Proc. of 22nd Int. Conf. on Very Large Data Bases, Bombay, India, pp. 239–250. Morgan Kaufmann 1996
33. Luo RC, Kay MG (eds.) Multisensor Integration and Fusion for Intelligent Machines and Systems. Norwood, NJ: Ablex Publishing Corporation 1995
 34. Noik EG. A Space of Presentation Emphasis Techniques for Visualizing Graphs. In: *Proceedings of Graphics Interface '94*, pp. 225–233, 1994
 35. Olken F, Rotem D. Simple Random Sampling from Relational Databases. In: Chu WW, Gardarin G, Ohsuga S, Kambayashi Y (eds.) VLDB'86 Twelfth Int. Conf. on Very Large Data Bases, Kyoto, Japan, Proc., pp. 160–169. Morgan Kaufmann 1986
 36. Pylyshyn ZW, Burkell J, Fisher B, Sears C, Schmidt W, Trick L. Multiple parallel access in visual attention. *Canadian Journal of Experimental Psychology*, 1993
 37. Raman V, Hellerstein JM. Potter's Wheel: An Interactive Data Cleaning System. In: Apers PMG, Atzeni P, Ceri S, Paraboschi S, Ramamohanarao K, Snodgrass RT (eds) VLDB 2001, Proc. of 27th Int. Conf. on Very Large Data Bases, Roma, Italy, pp. 381–390. Morgan Kaufmann, 2001
 38. Read B (ed.) *Advances in Databases*, Proc. of the 18th Brit. Nat. Conf. on Databases (BNCOD 18), volume 2097 of *Lecture Notes in Computer Science*, Chilton, UK, July 2001. Berlin Heidelberg: Springer 2001
 39. Sarawagi S, Thomas S, Agrawal R. Integrating Mining with Relational Database Systems: Alternatives and Implications. In: Haas LM, Tiwary A (eds.) SIGMOD 1998, Proc. ACM SIGMOD Int. Conf. on Management of Data, Seattle, Washington, USA, pp. 343–354. ACM Press, 1998
 40. Sattler K-U, Conrad S, Saake G. Adding Conflict Resolution Features to a Query Language for Database Federations. *Australian Journal of Information Systems* 8(1): 116–125 (2000)
 41. Sattler K-U, Dunemann O. SQL Database Primitives for Decision Tree Classifiers. In: Proc. of the 10th ACM CIKM Int. Conf. on Information and Knowledge Management, November 5–10, 2001, Atlanta, Georgia, USA, 2001
 42. Sattler K-U, Dunemann O, Geist I, Saake G, Conrad S. Limiting Result Cardinalities for Multidatabase Queries using Histograms. In: Read [38], pp. 152–167
 43. Sattler K-U, Schallehn E. A Data Preparation Framework based on a Multidatabase Language. In: Adiba M, Collet C, Desai BP (eds.) Proc. of Int. Database Engineering and Applications Symposium (IDEAS 2001), pp. 219–228, Grenoble, France, 2001. IEEE Computer Society
 44. Schroeder W, Martin K, Lorenzen B. *The Visualization Toolkit – An Object-Oriented Approach to 3D Graphics*. Prentice Hall PTR, 2. edition, 1998
 45. Schwenkreis F. New Working Draft of SQL/MM Part 6: Data Mining based on BHX008 and BHX033-BHX039. Technical Report, International Organization for Standardization (ISO), Mai 2000
 46. Strothotte T, Schlechtweg S. Non-Realistic Rendering and Animation. Morgan Kaufmann Publishers Inc, to appear 2001
 47. Vitter JS. An Efficient Algorithm for Sequential Random Sampling. *ACM Transactions on Mathematical Software* 13(1): 58–67 (1987)



Oliver Dunemann studierte Wirtschaftsinformatik in Braunschweig. Nach dem Diplom im Jahr 1995 war er in der Softwareentwicklung und Beratung im Banken- und Sparkassensektor tätig. Von April 2000 bis März 2002 war er im Rahmen einer Forschergruppe an der Universität Magdeburg tätig. Seit April 2002 ist er im Bereich Systemintegration/Investmentbanking bei der Nord/LB beschäftigt.



Ingolf Geist studierte Informatik an der Otto-von-Guericke Universität in Magdeburg. Nach dem Abschluss des Diploms im Jahr 2000 ist er an der Universität Magdeburg innerhalb einer Forschungsgruppe tätig.



Roland Jesse hält einen B.Sc. in Computer Information Systems der University of Wisconsin – Stevens Point (U.S.A.) sowie ein Diplom der Informatik von der Otto-von-Guericke Universität Magdeburg. Seit Januar 2000 untersucht er Aspekte zur Interaktion und Visualisierung für die Informationsfusion im Rahmen einer Forschergruppe an der Universität Magdeburg.



Gunter Saake ist Professor für Datenbanken und Informationssysteme an der Uni Magdeburg und forscht unter anderem auf den Gebieten Datenbankintegration, digitale Bibliotheken, objektorientierte Informationssysteme und Informationsfusion.



Kai-Uwe Sattler ist wissenschaftlicher Assistent in der Arbeitsgruppe Datenbanken der Uni Magdeburg. Seine Arbeitsgebiete sind Datenbankintegration und -föderation sowie Anfragebearbeitung in heterogenen Datenbanksystemen.