Felix Naumann

Informationsintegration

Antrittsvorlesung

5. Mai 2004

Humboldt-Universität zu Berlin
Mathematisch-Naturwissenschaftliche Fakultät II
Institut für Informatik

Die digitalen Ausgaben der *Öffentlichen Vorlesungen* sind abrufbar über den Dokumenten- und Publikationsserver der Humboldt-Universität unter: http://edoc.hu-berlin.de

Herausgeber:
Der Präsident der Humboldt-Universität zu Berlin

Copyright: Alle Rechte liegen beim Verfasser
Berlin 2004

Redaktion:
Birgit Eggert
Forschungsabteilung der Humboldt-Universität zu Berlin
Unter den Linden 6
D–10099 Berlin

Herstellung:
Forschungsabteilung der Humboldt-Universität zu Berlin
Unter den Linden 6
D–10099 Berlin

Heft 134

ISSN 1618-4858 (Printausgabe)
ISSN 1618-4866 (Onlineausgabe)
ISBN 3-86004-180-0
Gedruckt auf 100 % chlorfrei gebleichtem Papier

Sehr geehrte Damen, sehr geehrte Herren,

willkommen zu meiner Antrittsvorlesung am Tag der Informatik und zum Jahrestag meiner Berufung als Juniorprofessor für Informationsintegration am Institut für Informatik hier in Adlershof.

Das Forschungsthema und der Anwendungsbereich der Informationsintegration sind weit verbreitet. Informationsintegration verbirgt sich hinter vielen Synonymen wie etwa Informationsfusion, Datenkonsolidierung, Datenreinigung (*data cleansing*), Anwendungsintegration (*application integration*) oder Datenverschmelzung. Es gibt heute praktisch kein kommerziell verfügbares Informationssystem, das nicht zumindest einige Grundfähigkeiten der Informationsintegration beherrschen muss. Die seit vielen Jahren erfolgreich in Unternehmen eingesetzten *Data Warehouses* sind ein klassisches Beispiel großer integrierter Informationssysteme. Dieser Umstand ergibt sich in erster Linie aus dem Bedarf aller und insbesondere der international agierenden Unternehmen, ihre oft weltweit verteilten Datenbestände zu integrieren und zu konsolidieren.

Der Aufbau eines integrierten Informationssystems ist fast immer der gleiche: Eine zentrale, integrierende Komponente stellt Anwendern und Anwendungen eine einheitliche Schnittstelle zu den verteilten Daten zur Verfügung [SL90]. Für Anwender besteht die Schnittstelle meist aus einer deklarativen Anfragesprache wie SQL und gegebenenfalls Softwaretools, die den Umgang mit der Sprache erleichtern. Anwendungen greifen auf das integrierte System mittels vordefinierter Programmierschnittstellen oder auch mittels einer Anfragesprache zu. In dem System sind vielfältige Informationsquellen integriert: Klassische relationale Datenbanksysteme, Dateisysteme, Web Services, HTML Formulare, daten-produzierende Anwendungen und möglicherweise wiederum andere integrierte Informationssysteme. Diesen Systemen ist gemein, dass sie oft nichts gemein haben: Anfragen müssen in unterschiedlichsten Sprachen gestellt werden, Daten werden in heterogenen Formaten geliefert, die Struktur der Daten ist

unterschiedlich, usw. Anfragen an das zentrale integrierte Informationssystem müssen geeignet übersetzt und zerlegt und an die jeweiligen Informationsquellen weitergeleitet werden. Diese reichen ihre Antworten zurück an das integrierende System, welche die Resultate transformiert, integriert und dem Nutzer einheitlich darstellt. Der Vorteil dieses Verfahrens ist die Tatsache, dass der Nutzer bzw. die Anwendung nicht auf die Informationsquellen einzeln zugreifen muss und dabei die Ergebnisse gleichsam per Hand integrieren muss. Integrierte Informationssysteme stellen also eine einheitliche Sicht auf eine Vielzahl heterogener Datenquellen zur Verfügung.

Besonders einfache Beispiele solcher Systeme sind die im Internet weit verbreiteten Metasuchmaschinen wie der MetaCrawler[1]. Metasuchmaschinen speichern keine eigenen Informationen über Webseiten, sondern reichen Anfragen an „echte“ Suchmaschinen weiter, die tatsächlich große Mengen des World Wide Web (WWW) untersucht und indiziert haben. Die Ergebnislisten der einzelnen Suchmaschinen werden von der Metasuchmaschine zusammengestellt und dem Nutzer einheitlich präsentiert. Dabei ist es für den Nutzer idR. unerheblich, welche Suchmaschine welche Teilergebnisse lieferte. Wichtig ist, dass er oder sie nur eine einzige WWW Adresse anwählen musste, nämlich die Adresse der Metasuchmaschine, und nur ein einziges Mal die Suchanfrage formulieren musste.

Neben der Vereinfachung für Nutzer und Anwendungen sind die Vorteile solcher Systeme der Informationsintegration mannigfaltig. Da sie mehrere, unabhängige Informationsquellen der gleichen Art integrieren, können sich Nutzer größere bzw. bessere Ergebnisse erhoffen. Bei der Suche nach einer bestimmten Webseite können Metasuchmaschinen auf die kombinierten Datenmengen aller Suchmaschinen zugreifen – was die eine Suchmaschine nicht kennt, kann bei einer der anderen Suchmaschinen erfragt werden. Des Weiteren können sich Nutzer und Anwendungen detailreichere Informationen erhoffen: Verschiedene Informationsquellen speichern verschiedene Eigenschaften von Objekten der wirklichen Welt. Beispielsweise kennt die eine Suchmaschine die Da-

teigrößen aller Webseiten, eine andere kennt die Sprache in der die Seite geschrieben wurde. Zusammengenommen kann eine Metasuchmaschine also detailreichere Informationen bieten als dies einzelne Suchmaschinen können. Schließlich bieten integrierte Informationssysteme eine größere Genauigkeit und Zuverlässigkeit von Informationen. Gerade bei der Integration von autonomen und potentiell unzuverlässigen Quellen im Internet ist die Verifizierbarkeit von Informationen wichtig. Ein integriertes System kann Daten verschiedener Quellen vergleichen und auf diese Weise Konflikte erkennen. Die Auflösung solcher Datenkonflikte ist allerdings nicht einfach und bedarf meist des Eingriffs eines Experten. Bei unterschiedlichen Titeln der gleichen Webseite etwa, kann ein Experte festlegen, dass stets der längere Titel im integrierten Ergebnis verwendet werden soll.

Die Informationsintegration bietet offensichtlich viele Vorteile, deren technische Umsetzung in tatsächlichen Systemen ist jedoch sehr schwierig und für die Forschung von größtem Interesse.

Eine erste Schwierigkeit ist bereits die Erstellung eines globalen Datenschemas für das integrierte System. Ein Datenschema (kurz: Schema) gibt an, welche Struktur die Daten haben. Das Schema einer Suchmaschine besagt, dass die Daten als Tabelle zur Verfügung gestellt werden, und dass jede Zeile der Tabelle eine indizierte Webseite repräsentiert und eine bestimmte Menge von Spalten hat, die beispielsweise die Namen „URL“, „Titel“, „Größe“ usw. tragen. Wenn eine andere Suchmaschine ein anderes Schema benutzt, muss entschieden werden, welche Elemente in dem gemeinsamen, globalen Schema verwendet werden sollen. Zusätzlich zu der Entwicklung des globalen Schemas müssen Beziehungen zwischen dem globalen Schema und den lokalen Schemata bei den Quellen definiert werden. Diese beiden Schwierigkeiten sind respektive die Probleme der *Schemaintegration* [BLN86] und des *Schema Mapping* [MHH00]. Ein globales Schema spiegelt die Struktur der Daten in den einzelnen Informationsquellen wider, muss aber zugleich Gemeinsamkeiten dieser Struktur erkennen und Konflikte zwischen den Strukturen auflösen. Konflikte reichen von einfachen Dingen wie un-

terschiedliche Formate um ein Datum darzustellen (29. Mai 2003 vs. 5/29/03) bis zu komplexen Problemen wie die Unterscheidung von Männern und Frauen anhand eines einzelnen Attributs (`Personen (Name, Adresse, Mann, Frau)`), anhand von einzelnen Tabellen (`Männer (Name, Adresse)`), (`Frauen (Name, Adresse)`) oder anhand von einzelnen Attributwerten (`Personen (Name, Adresse, Geschlecht (m/w))`).

Eine weitere Schwierigkeit bei der Einrichtung eines integrierten Informationssystems ist die Übersetzung und das Weiterleiten von Anfragen vom integrierten System zu den einzelnen Informationsquellen. Dies ist das Problem der *Anfragebearbeitung*. Verschiedene Quellen sprechen verschiedene und verschieden mächtige Sprachen. Während beispielsweise eine Suchmaschine lediglich ein HTML Formular zur Verfügung stellt, kann eine andere Suchmaschine einen komplexen Web Service als Schnittstelle anbieten. Darüber hinaus könnte beispielsweise die erste Suchmaschine optional eine Einschränkung der Ergebnisse auf Deutsche Webseiten anbieten, eine andere Suchmaschine könnte solche Selektionen nicht beherrschen. Eine Metasuchmaschine muss also eine Suchanfrage zum einen automatisch in ein HTML Formular eintragen und zum anderen einen Web Service mit einer in der Bedeutung gleichen Anfrage aufrufen. Hat der Nutzer eine Selektionsmöglichkeit ausgenutzt, muss auch diese automatisch an die eine Suchmaschine weitergereicht werden. Da die andere Suchmaschine Selektion nicht beherrscht, muss das integrierte System diesen Mangel kompensieren und selbst die Selektion für deren Ergebnisse durchführen. Diese Probleme lassen sich relativ leicht formalisieren, deren automatische Lösung (worauf wir hier nicht eingehen) ist jedoch allgemein von großer Schwierigkeit.

Nach der Lösung dieser Probleme bei der Einrichtung eines integrierten Informationssystems stellen sich zur Laufzeit des Systems zwei weitere, wichtige Probleme: Erstens muss automatisch erkannt werden, *welche* der jeweiligen Ergebnisse der einzelnen Informationsquellen integriert werden sollen, und es muss bestimmt werden, *wie* die Ergebnisse integriert werden sollen. Eine Metasuchmaschine muss also erkennen, dass zwei

Suchmaschinen Ergebnisse über die gleiche Webseite liefern. In der Regel fällt dies leicht, da Webseiten anhand ihrer Webadresse (URL) eindeutig identifiziert werden können. In anderen Anwendungsdomänen, wie etwa bei Adressdaten, ist eine solch eindeutige Identifikation nicht immer möglich. Dies ist das Problem der *Duplikaterkennung*. Zweitens muss spezifiziert werden, wie Daten integriert werden, die zuvor als zusammengehörig (als Duplikate) erkannt wurden. Insbesondere müssen Widersprüche in den Daten aufgelöst und das Fehlen von Daten erkannt werden. Dies ist das Problem der *Datenfusion*.

Diese Vorgänge, die zur Laufzeit des Systems durchgeführt werden, müssen zudem so optimiert werden, dass zwischen dem Stellen der Anfrage und dem Anzeigen des integrierten Ergebnisses möglichst wenig Zeit vergeht. Eine Metasuchmaschine ist nur dann erfolgreich, wenn sie Ergebnisse innerhalb weniger Sekunden, möglichst in weniger als einer Sekunde, anzeigen kann. Dies ist das Problem der *Anfrageoptimierung*.

Zuletzt in diesen einleitenden Worten möchte ich noch auf das Problem der *Visualisierung* aufmerksam machen: Ohne spezialisierte Visualisierungstechniken sieht man Informationen nicht an, dass sie von verschiedenen Quellen zusammengetragen und integriert wurden, dass sie mit verschiedenen Zuverlässigkeitsmerkmalen behaftet sind und dass sie möglicherweise das Ergebnis eines gelösten Datenkonfliktes sind. Während diese Zusatzinformationen für gewisse Anwendungen nicht relevant sind (der typische Nutzer einer Metasuchmaschine ist gegenüber der Herkunft der Daten unkritisch), ist es in vielen Fällen nötig, solche Metadaten zu sammeln und geeignet darzustellen.

In dieser Vorlesung beschreibe ich – in einer dem Datenfluss von den Informationsquellen zum Nutzer entsprechenden Reihenfolge – verschiedene Lösungen für die Probleme des *Schema Mappings*, die die strukturelle Heterogenität der Quellen überwinden, der *Duplikaterkennung*, um Daten über gleiche Dinge (*real-world objects*) als solche zu erkennen, und der *Datenfusion*, um zu spezifizieren, wie Widersprüche in den Daten gelöst wer-

den sollen. Ideen zur *Anfrageoptimierung* und *Visualisierung* werden gegen Ende nur kurz erwähnt.

Schemata beschreiben die Struktur von Datenmengen. In einem relationalen Datenbankschema wird definiert, welche Tabellen in der Datenbank gespeichert sind, welche Spalten die jeweiligen Tabellen umfassen, und welche Beziehungen zwischen den Tabellen herrschen. Ein *Schema Mapping* ist eine Abbildung von einem solchen Schema, dem Quellschema, zu einem anderen Schema, dem Zielschema [MHH00, Levy01]. Die Abbildung umfasst eine Menge von so genannten *Korrespondenzen*, die einzelne Attribute der Schemata miteinander verbinden. Bildlich kann man sich die Korrespondenzen als Pfeile von Attributen im Quellschema zu Attributen im Zielschema vorstellen; tatsächlich bedienen sich viele Softwaretools dieser Metapher und visualisieren Mappings als eine Menge von Pfeilen. Mappings und ihre Korrespondenzen werden verwendet, um Daten einer Datenquelle so zu transformieren, dass ihre Struktur dem Zielschema entspricht. In einem integrierten Informationssystem stellen die Schemata der Informationsquellen die Quellschemata dar. Zielschema ist das globale Schema im integrierten System selbst. Stellt ein Nutzer eine Anfrage an das globale Schema, müssen die in den Quellen vorliegenden Daten gemäß der Schema Mappings transformiert werden.

Um Schema Mappings einsetzen zu können, bedarf es zweier Schritte: Dem *Erstellen* des Mappings durch einen Experten und der *Interpretation* des Mappings als Datentransformation. Während der erste Schritt wegen seiner schwierigen Semantik vermutlich niemals ohne menschliche Hilfe getan werden kann, ist die Automatisierung gerade des zweiten Schrittes eine enorme Erleichterung für Entwickler: Oft sind die Datentransformationen sehr komplexe Anfragen oder Programme, die viele Seiten umfassen und von Hand kaum fehlerfrei entwickelt werden können.

Auch wenn es in der Forschung einige Bestrebungen gibt, die Erstellung eines Mappings zu automatisieren (wir kommen später zu diesem *Schema Matching* genannten Thema), ist sie in der Re-

gel Domänenexperten vorbehalten. Unterstützt durch Softwaretools ziehen Experten Linien vom Quellschema zum Zielschema und erstellen so einzelne Korrespondenzen. So würde beispielsweise der Experte, der ein Mapping zwischen einem in englischer Sprache verfassten Schema einer Suchmaschine und dem Schema einer deutschen Metasuchmaschine erstellt, eine Line zwischen dem „`description`“-Feld der Suchmaschine und dem „`Kurzfassung`“-Feld ziehen und so definieren, dass `description`-Daten in das `Kurzfassung`-Feld übernommen werden sollen. Komplexere Mappings können mehrere Felder kombinieren (`vorname` und `nachname` in der Quelle zu Name in einem Zielschema für Personendaten) und Funktionen auf Korrespondenzen spezifizieren (Umrechnung von `Dollar` auf `Euro` in der Finanzdomäne).

Schema Matching Verfahren unterstützen Experten bei dieser Arbeit. Sie ermitteln unter allen Möglichkeiten die wahrscheinlichsten Korrespondenzen und schlagen diese vor. Der Experte kann nun diese Vorschläge annehmen oder ablehnen. Schema Matching Verfahren stehen mehrere Ressourcen zur Verfügung um Vorschläge zu bearbeiten [RB01]. Schema-basierte Verfahren vergleichen die Namen von Attributen und schlagen Korrespondenzen vor, wenn sich die Namen in Quell- und Zielschema ähneln. So würde beispielsweise eine Korrespondenz zwischen einem englischen „title“- und dem deutschen „Titel“-Attribut automatisch entdeckt. Die Ähnlichkeit von „birthday“ und „Geburtstag“ würde so jedoch nicht erkannt. Hier helfen Instanz-basierte Schema Matching Verfahren, die die Daten der Schemata analysieren. So würde erkannt, dass die Datenwerte unter „birthday“ und „Geburtstag“ ähnliche Eigenschaften haben (viele Zahlen, die mit 19… beginnen; ähnliche Länge der Einträge; usw.). Struktur-basierte Schema Matching Verfahren hingegen analysieren den Aufbau der Schemata und schlagen Korrespondenzen zwischen Attributen vor, die von ähnlichen Attributen und Tabellen „umgeben“ sind. Hybrid- und Mischverfahren schließlich kombinieren die oben genannten Verfahren und erzielen meist die besten Ergebnisse. Die Effektivität der Schema Matching Verfahren wird daran gemessen, wie oft Vorschläge nicht abge-

lehnt wurden (*precision*) und welcher Anteil aller wahren Korrespondenzen überhaupt gefunden wird (*recall*). In günstigen Fällen liegen diese Zahlen für neuere Verfahren bei ca. 80%. D.h. vier von fünf Vorschlägen sind korrekt, und nur 20% aller Korrespondenzen müssen von Hand gefunden und spezifiziert werden. Im Übrigen wurde noch kein Verfahren entwickelt, das komplexere als die einfachsten 1:1 Korrespondenzen entdeckt.

Ist ein Schema Mapping zwischen zwei Schemata erstellt, gilt es, entsprechend dieses Mappings die Daten der Informationsquelle zu transformieren [Popa+02]. In der Regel wird als Transformationssprache eine Anfragesprache wie SQL, XSLT oder XQuery gewählt. Diese Sprachen haben den Vorteil, dass sie von vielen Datenbankmanagementsystemen verstanden und optimiert werden. Die Umwandlung eines Schema Mappings in eine Datentransformationsanfrage nennt man die *Interpretation* des Mappings. Wie die Bezeichnung schon vermuten lässt, gibt es zu einem Mapping mehr als eine Interpretation. Das einfache Erstellen einiger Korrespondenzen lässt viele Möglichkeiten der Datentransformation offen. Stellen Sie sich in der Informationsquelle zwei Tabellen vor: Eine Tabelle über „`Professoren`“ mit ihren Namen und eine andere Tabelle über „`Kurse`“ mit dem Titel des Kurses und einem Verweis auf den lehrenden Professor. Stellen Sie sich des Weiteren eine Tabelle namens „`Lehrt`“ im Zielschema vor, in der Professorennamen und Kursnamen gespeichert werden können. Zwischen Quelle und Ziel habe ein Experte die beiden offensichtlichen Korrespondenzen spezifiziert, nämlich:

1. <`Professoren.Name` → `Lehrt.Professorenname`>
2. <`Kurse.Titel` → `Lehrt.Kursname`>.

Eine naive Interpretation diesen Mappings würde alle Professorennamen in die Zieltabelle übertragen, und unabhängig davon alle Kurstitel in die gleiche Tabelle schreiben. Allerdings würde so die Beziehung zwischen Professoren und den Kursen verloren. Man könnte der Zieltabelle nicht entnehmen, welcher Professor welchen Kurs lehrt. Eine vermeintlich bessere Interpreta-

tion würde die Verweise in der Informationsquelle von der Kurse-Tabelle auf die Professoren-Tabelle ausnutzen und so entsprechende Professor-Kurs Paare in die Zieltabelle eintragen. Was geschieht jedoch mit Professoren, die zurzeit keine Kurse lehren? Eine Interpretation des Mappings würde solche Professoren ignorieren (denn schließlich geht es ja um Kurse), eine andere Interpretation würde auch Professoren im Freisemester berücksichtigen (denn schließlich sollen keine Daten verloren gehen). Die gleichen Fragen gelten auch für Kurse, die nicht von Professoren (sondern von Assistenten) gelehrt werden.

Aus einem denkbar einfachen Szenario mit insgesamt drei Tabellen und nur zwei Korrespondenzen konnten wir bereits fünf verschiedene Interpretationen herleiten. Diese Anzahl steigt exponentiell mit der Zahl der beteiligten Tabellen und der Beziehungen zwischen ihnen. Um dieser Menge Herr zu werden, verwenden Algorithmen Heuristiken, um die wahrscheinlichsten Interpretationen auszuwählen und anhand von Beispielen dem Experten anzubieten. Diese Heuristiken verwerfen beispielsweise solche Interpretationen, die leere Ergebnisse im Zielschema erzeugen, oder die in der Datenquelle vorhandene Beziehungen zwischen Daten bei der Transformation zerstören [MHH00].

Schema Mappings und deren Interpretation werden beim Aufbau eines integrierten Informationssystems verwendet. Wir wenden uns nun einem bereits aufgebauten und laufenden System zu, das Nutzeranfragen an das globale Schema beantwortet, indem Daten der Datenquelle abgerufen, transformiert und integriert werden. Dabei kommt es oft vor, dass mehrere Quellen Informationen über gleiche Realwelt-Objekte speichern. Man sagt in solchen Fällen, dass die Quellen sich überlappen. Die Überlappung kann sowohl extensional als auch intensional sein. Zwei Quellen überlappen sich extensional, wenn sie Daten über gleiche Dinge speichern. Sie überlappen sich intensional, wenn sie gleiche Daten über die Dinge speichern. Zwei Suchmaschinen, die beide Daten über die Webseite `www.ibm.com` gespeichert haben, überlappen sich extensional. Die Überlappung der Suchmaschinen besteht also aus dem „Ding“ `www.ibm.com` und

vermutlich vielen anderen Webseiten. Die Überlappung ist nicht vollständig, wenn beide Suchmaschinen auch Seiten speichern, die die jeweils andere Suchmaschine nicht speichert. Wenn beide Suchmaschinen zudem beispielsweise den Titel und die Größe der Webseite speichern, überlappen sie sich auch intensional. Die Überlappung besteht aus den beiden Attributen Titel und Größe. Die Suchmaschinen können darüber hinaus jeweils andere Eigenschaften von Webseiten speichern. Die wesentliche Aufgabe der Informationsintegration ist es, mit diesen Arten der Überlappung umzugehen. Methoden der Duplikaterkennung entdecken die extensionale Überlappung und Methoden der Datenfusion lösen Probleme, die sich durch die intensionale Überlappung ergeben.

Bei einer gegebenen Datenmenge (in der Regel eine Tabelle), ist es die Kernaufgabe der Duplikaterkennung, alle Duplikate innerhalb dieser Tabelle zu finden. Dabei müssen zwei wesentliche Hürden überwunden werden: Die erste Hürde ist die automatische Erkennung von Duplikaten [HS98]. Deshalb wird größte Sorgfalt in die Entwicklung eines Ähnlichkeitsmaßes gesteckt, welches die Ähnlichkeit zweier Zeilen einer Tabelle bemisst. Sind sich zwei Zeilen hinreichend ähnlich, gelten sie als Duplikate. Ein einfaches Verfahren zur Duplikaterkennung wäre es also, jede Zeile mit jeder anderen zu vergleichen und hinreichend ähnliche Paare als Duplikate zu markieren. Dieses Verfahren macht die zweite Hürde deutlich: Bei typischen Tabellen mit mehr als 1 Million Einträgen wären über 1 Billion Vergleiche nötig, um alle Duplikate zu erkennen. Erschwerend kommt hinzu, dass schon ein einziger Vergleich mittels des Ähnlichkeitsmaßes oft eine komplexe Berechnung darstellt. Deswegen werden Algorithmen entwickelt, die durch geschickte Partitionierung der Daten die Anzahl der Vergleiche drastisch reduzieren.

Als Ähnlichkeitsmaß für Zeichenketten und Attributwerte hat sich die *edit-distance* bewährt [Wag74]. Sie bezeichnet die minimale Anzahl an Einfüge-, Lösch- und Änderungsoperationen, die nötig sind, um den Text der einen Zeile in den Text der anderen Zeile umzuwandeln. Um beispielsweise das Wort „hase“ in

das Wort „rasen“ umzuwandeln, sind mindestens zwei Operationen nötig (das „h“ in „r“ ändern und am Ende ein „n“ einfügen), die edit-distance der beiden Worte ist mithin 2. Je kleiner die edit-distance, desto ähnlicher sind die Worte. Die edit-distance fängt auf hervorragende Weise eine der typischen Ursachen für Duplikate ein: Tippfehler. Werden beispielsweise die Daten eines Kunden in ein Datenbanksystem eingetragen und vertippt sich der Angestellte bei der Eingabe (er tippt aus Versehen „Heonz“ statt „Heinz“ oder aus Unwissenheit „Meier“ statt „Mayer“), kann das Datenbanksystem nicht feststellen, ob der Kunde bereits im System eingetragen war. Wurde Herr Heinz Mayer schon einmal erfasst, ist die edit-distance der beiden Einträge „Heinz Mayer“ und „Heonz Meier“ gering (nämlich 3) und sie würden als Duplikate erkannt. Andere Duplikate sind schwieriger zu erkennen: Fehlende Daten verfälschen die Ergebnisse, ein Umzug einer Person ändert den Tabelleneintrag an vielen Stellen, usw. Um solche Duplikate zu erkennen, muss man das allgemeine edit-distance Maß mit Wissen über die Eigenschaften der Anwendungsdomäne anreichern. Eine gängige Methode ist es, eine Menge von Regeln aufzustellen, die Ähnlichkeiten einzelner Felder kombinieren. Eine Regel könnte etwa lauten: „Wenn Nachname, Geburtstag und Geburtsort gleich sind, und zusätzlich die Vornamen hinreichend ähnlich sind, handelt es sich um Duplikate.“ Solchermaßen aufgestellte Regeln werden reihum geprüft bis eine greift, oder bis alle probiert wurden und keine den Hinweis auf Duplikate lieferte. Für bestimmte Datentypen sind andere Ähnlichkeitsmaße erfolgreicher: Für längere Textdaten werden Maße basierend auf TFIDF Berechnungen entwickelt [BR99], für numerische Daten werden numerische Abstandsmaße verwendet.

Seit einigen Jahren gewinnt das *XML Datenmodell*[2] an großer Popularität, insbesondere für Daten, die über das Internet ausgetauscht werden. XML Daten können im Unterschied zu relationalen Daten geschachtelt werden. Z.B. können die Kurse eines Professors unterhalb eines Strukturelements „Professor“ geschachtelt werden. Diese Schachtelung erschwert die Duplikaterkennung, da zunächst einmal entschieden werden muss, auf

welcher Schachtelungsebene überhaupt nach Duplikaten gesucht werden soll. Andererseits können tiefer geschachtelte Kindelemente gute Hinweise auf Duplikate liefern.

Wir wenden uns nun der zweiten Hürde der Duplikaterkennung zu, nämlich der potentiell sehr großen Anzahl von Vergleichen, die nötig sind, um alle Duplikate in einer Datenmenge zu finden. Wie eingangs bereits erwähnt, ist es eine übliche Methode, die Daten so zu partitionieren, dass Duplikate nur noch innerhalb einer Partition gesucht werden müssen. Die so genannte *sorted-neighborhood* Methode ist ein gutes und einfaches Beispiel eines solchen Algorithmus [HS98]: Bevor die eigentliche Duplikatsuche beginnt, werden die Daten sortiert. Die Wahl des Sortierschlüssels ist hierbei von großer Bedeutung, denn er soll gewährleisten, dass nach der Sortierung Duplikate nah beieinander liegen. Der Sortierschlüssel besteht nicht notwendigerweise aus den Daten selbst, sondern kann nach bestimmten, domänenspezifischen Regeln abgeleitet sein (z.B. die ersten drei Konsonanten des Nachnamen und die letzten 5 Ziffern der Telefonnummer). Anschließend wird gleichsam ein Fenster über die sortierten Daten geschoben. Duplikate werden nur noch innerhalb dieses Fensters gesucht. Sind alle Paare in einem Fenster verglichen, wird das Fenster um eine Stelle weiter geschoben und die neu hinzugekommene Zeile wird mit allen im Fenster verbliebenen Zeilen verglichen. Falls nach geeigneter Sortierung Duplikate nah beieinander (mindestens innerhalb der Fenstergröße) liegen, werden so alle Duplikate gefunden. Duplikate, deren Sortierschlüssel unähnlich ist, werden also nicht gefunden. Eine Verbesserung des Verfahrens wird erzielt, indem mehrere Durchläufe mit jeweils verschiedenen Sortierschlüsseln durchgeführt werden. Es besteht so die (begründete) Hoffnung, dass Duplikate, die mit dem einen Schlüssel nicht nah zueinander sortiert werden, in einem der späteren Durchläufe nah beieinander liegen. Der Vorteil der sorted-neighborhood Methode ist die Tatsache, dass sie mit nur wenigen Läufen über die Daten auskommt. Bei großen Datenmengen ist die Anzahl der Datenläufe, in denen alle Daten einmal in den Hauptspeicher geladen werden, die entscheidende Kennzahl, da sie ein Indikator für die benötigte Zeit ist.

Eine Adaption dieser und anderer Methoden der effizienten Duplikaterkennung für relationale Daten auf XML Daten ist gegenwärtiges Forschungsthema unserer Arbeitsgruppe. Insbesondere sollen hier Daten der geschachtelten Kinder im Ähnlichkeitsmaß berücksichtigt werden. So werden Daten, die nur bedingt ähnlich sind, dennoch korrekt als Duplikate erkannt, wenn sie wenigstens ähnliche XML Kinder haben.

Wir kommen nun zum zweiten Thema der Informationsintegration in einem laufenden System, nämlich dem Zusammenfügen von Daten verschiedener Quellen, die sich gegenseitig ergänzen aber auch widersprechen können. Zur Darstellung der Ziele greifen wir das vorige Beispiel der Suchmaschinen wieder auf. Wir stellen uns eine Suchmaschine A vor, die als Ergebnis einer Suchanfrage eine Liste mit Verweisen auf Webseiten liefert, die die Attribute `URL`, `Titel`, `Kurzfassung`, Datum und `Sprache` umfasst. Suchmaschine A kann nicht für alle Webseiten Kurzfassungen erstellen und so bleibt das Kurzfassung-Attribut von Zeit zu Zeit leer. In der Datenbanktheorie und der Informationsintegration nennt man solch fehlende Informationen `null`-Werte. Suchmaschine B hingegen gibt die Attribute `URL`, `Titel`, `Kurzfassung` und `Größe` in der Ergebnisliste zurück. Die intensionale Überlappung der beiden Informationsquellen umfasst also die jeweils ersten drei Attribute, die extensionale Überlappung umfasst eine gewisse Menge von Webseiten, die von beiden Suchmaschinen indiziert wurden. Die extensionale Überlappung kann leicht anhand der eindeutigen URL der Webseiten erkannt werden. Ist ein solcher Identifikator nicht vorhanden, werden Duplikaterkennungsmethoden eingesetzt.

Zur Einrichtung einer Metasuchmaschine muss zunächst entschieden werden, welche Attribute als Ergebnis dem Nutzer präsentiert werden sollen. Wählte man die größte gemeinsame Untermenge, enthielte das Ergebnis nur die drei Attribute `URL`, `Titel` und `Kurzfassung`. Interessante Informationen über `Größe` oder `Sprache` würden verworfen. Nähme man diese zusätzlichen Attribute hinzu, müssten Nutzer hinnehmen, dass dafür nicht immer Werte erhältlich sind. Für Webseiten, die bei-

spielsweise nur von Suchmaschine A genannt werden, enthielte das Größe-Attribut null-Werte. Wir nehmen im Folgenden an, dass die Metasuchmaschine keine Daten verwirft, also alle erhältlichen Attribute anzeigt und sie notfalls als null-Werte anzeigt.

Liefern nun beide Suchmaschinen in ihren Ergebnislisten zu der Suchanfrage einer Metasuchmaschine Informationen über eine gleiche Webseite, müssen diese zu einem Gesamtergebnis integriert werden. In einigen Fällen ist die Integration trivial: Die URL ist bei beiden Listeneinträgen gleich (dies ist ja der Anlass zur Integration). Die Werte für `Datum`, `Sprache` und `Größe` stammen jeweils exklusiv von der einen bzw. der anderen Suchmaschine. Insofern ergänzen sich die Einträge hervorragend. Allerdings liefern beide Suchmaschinen sich möglicherweise widersprechende Datenwerte für den `Titel` und die `Kurzfassung` einer Webseite. Sind die Werte gleich, kann man sie direkt ins Ergebnis übernehmen; besteht ein Konflikt, muss er gelöst werden. Bei der Konfliktlösung sind vielerlei Strategien, ausgedrückt durch Konfliktlösungsfunktionen, denkbar: Beispielsweise könnte immer der Wert der Suchmaschine A gewählt werden. Eine solche Strategie drückt ein größeres Vertrauen in die Richtigkeit der Daten in Suchmaschine A aus. Diese Präferenz kann man verfeinern, indem die Konfliktlösungsfunktion immer den Wert von Suchmaschine A wählt, es sei denn der Wert ist ein null-Wert. In dem Fall wird der Wert von Suchmaschine B als Ersatz gewählt und im Ergebnis angezeigt. Eine andere Strategie ist die Wahl des längeren Wertes. Intuitiv bietet ein längerer Titel mehr Information über eine Webseite. In anderen Anwendungsszenarien könnte etwa von mehreren Datumsangaben die jüngste gewählt werden, von unterschiedlichen Preisen der günstigste usw. Gerade das letzte Beispiel zeigt, dass es nicht nur wichtig ist, Konflikte möglichst sinnvoll zu lösen, sondern dass es ebenso wichtig ist, die Konfliktlösungsentscheidung nachvollziehen zu können: Der günstigste Preis eines Produktes ist nutzlos ohne zu wissen, welcher der Anbieter diesen Preis nannte. Diesen Aspekt der so genannten Datenherkunft (*data lineage*) beleuchten wir später und wenden uns nun integrierenden Operatoren der Datenbanktechnologie zu.

Wenn wir von der Fusion von Daten sprechen, können wir drei Fälle unterscheiden: 1. Die Fusion identischer Zeilen (Zeilen 1 und 2 im Beispiel unten). 2. Die Fusion subsumierter Zeilen. Eine Zeile subsumiert eine andere Zeile, wenn sie mit der anderen in allen Datenwerten übereinstimmt und weniger null-Werte enthält. Sie trägt gleichsam mehr Information (Zeilen 2 und 3). 3. Die Fusion sich komplementierender Zeilen. Zwei Zeilen komplementieren einander, wenn sie in allen Datenwerten übereinstimmen oder null-Werte enthalten (Zeilen 3 und 4). 4. Die Fusion widersprüchlicher Zeilen. Zwei Zeilen widersprechen einander, wenn sie mindestens einen sich widersprechenden Wert enthalten (Zeilen 4 und 5). In den Fällen 1.–3. bestehen keine Widersprüche in den Daten.Wir untersuchen nun, welche der bekannten Operatoren welche Art der Fusion beherrscht.

1.	A	B	C
2.	A	B	C
3.	A	B	null
4.	A	null	C
5.	A	B	D

Die einfachste Art der Fusion von Daten ist die Vereinigung, manifestiert durch den `Union` Operator in gängigen Anfragesprachen wie SQL und XQuery. Zwei Tabellen, vereint durch `Union`, ergeben eine größere Tabelle, die die Zeilen beider einzelnen Tabellen enthält. `Union` integriert lediglich Daten der Kategorie 1., d.h. es werden nur völlig identische Zeilen eliminiert.

Der `Minimum Union` Operator [GL94] geht einen Schritt weiter und integriert zusätzlich Daten der Kategorie 2.; es werden also subsumierte Zeilen entfernt. Ein Informationsverlust findet durch diese Elimination nicht statt, da die subsumierte Zeile weniger Information enthielt als die subsumierende Zeile. Der `Minimum Union` Operator ist zwar in der Literatur definiert, wird

jedoch noch von keinem kommerziellen Datenbankmanagementsystem angeboten.

Eine direkte Operator-Unterstützung der Fusion der Kategorien 3. und 4. gibt es nicht. Es gibt allerdings Konstrukte, die beide ermöglichen. Ich deute hier nur kurz an, dass die Gruppierung mit dem `Group By` Operator in Verbindung mit Aggregationsfunktionen das Potential hat, Konflikte zu lösen. Allerdings ist diese Funktionalität in gängigen Datenbankmanagementsystemen stark durch die enge Auswahl der Aggregationsfunktionen eingeschränkt. In der Regel werden lediglich numerische Aggregationen wie die Summen- oder Durchschnittsbildung angeboten. Komplexere Konfliktlösung z.B. durch die Auswahl des längeren Wertes ist nicht möglich. Unsere Forschungsgruppe ist zurzeit dabei, einen `Fuse By` genannten Operator zu entwickeln, der alle vier Kategorien der Fusion beherrscht.

Die Einführung eines solchen Operators in ein Datenbankmanagementsystem ist eine schwierige Aufgabe. Neben der Definition der Syntax des Operators, also wie er in einer Anfragesprache wie SQL verwendet wird, ist die Festlegung einer klaren Semantik von großer Bedeutung. Die Semantik eines Operators bestimmt, wie er sich bei bestimmten Eingaben verhält, d.h. welcher Datenoutput bei welchem Dateninput zu erwarten ist. Der `Fuse By` Operator hat eine ähnliche Syntax wie der `Group By` Operator. Die Semantik entspricht einem `Outer Union` Operator mit einer anschließenden benutzerdefinierten Gruppierung. `Outer Union` ist eine Variante des `Union` Operators, die es erlaubt, auch Tabellen mit unterschiedlichen Spalten zu vereinen: Das Ergebnis enthält die Menge aller Spalten beider Tabellen. Fehlende Werte für Attribute die nur in der einen, nicht aber in der anderen Tabelle vertreten waren, werden mit null-Werten ergänzt. Die Art der Gruppierung, also der Schlüssel nach dem gruppiert werden soll, wird dem `Fuse By` Operator als Parameter übergeben. Wird dem `Fuse By` kein Parameter übergeben, wird nach keinem Schlüssel gruppiert und es wird das Verhalten von `Minimum Union` angenommen. Genau wie beim `Group By` Operator müssen Attribute, nach denen nicht gruppiert wird,

mit einer Konfliktlösungsfunktion versehen werden. Die Default-Konfliktlösungsfunktion ist die im SQL Standard vorgesehene `Coalesce` Funktion, die den ersten nicht-null-Wert aus der Liste der Eingabewerte auswählt. Mittels des `Resolve`-Befehls können aber auch andere Konfliktlösungsfunktionen spezifiziert werden. Das intuitive Default-Verhalten des `Fuse By` Operators erleichtert die Nutzung in einfachen SQL Ausdrücken.

Zur Einführung eines neuen Operators zur Informationsintegration genügt dessen Definition in Syntax und Semantik nicht. Es muss auch die schnelle Ausführung von Anfragen, die den Operator verwenden, auf großen Datenmengen gesichert sein. Anfragen werden von kommerziellen Datenbankmanagementsystemen in einem internen Baummodell dargestellt. Dieser Baum stellt zugleich den Anfrageausführungsplan dar; er gibt insbesondere die Reihenfolge an, in der bestimmte Operationen durchgeführt werden. Zur Optimierung der Anfrage wird die Struktur dieses Baumes mittels Heuristiken verändert, so dass zwar das Ergebnis der Anfrage stets das gleiche bleibt, die Ausführung der Anfrage jedoch beschleunigt wird. Die Heuristiken, ausgedrückt in formalen Regeln oder Algorithmen, sind zum Teil das Ergebnis jahrzehntelanger Forschung und Erfahrung mit Datenbanken. Durch die Einführung eines neuen Operators mit neuen Eigenschaften müssen zunächst diese Heuristiken angepasst werden. Die aktuelle Arbeit in unserer Forschungsgruppe beschäftigt sich mit eben diesem Problem, nämlich der Anfrageoptimierung mit Integrationsoperatoren.

Vor dem Ende dieses Vortrages will ich noch ein letztes und vielleicht entscheidendes Problem der Informationsintegration ansprechen. In den meisten Integrationsszenarien werden früher oder später einem Nutzer (z.B. einem Manager) die integrierten Daten vorgelegt, auf deren Basis Entscheidungen getroffen werden. Zwar ist die Integration der Daten und damit das Verbergen der vielfältigen Herkunft der Daten ein Mehrwert, der die Arbeit des Nutzers erleichtert. Andererseits wird die Akzeptanz der Daten und somit der Integrationsleistung nur gewährleistet, wenn die Integration nachvollzogen werden kann. Gerade bei der Inte-

gration von potentiell unzuverlässigen Quellen des WWW müssen Nutzer die Herkunft der Daten (die *data lineage*) erfragen können, bzw. sollte die Herkunft durch eine geeignete *Visualisierung* erkennbar gemacht werden. Da sogar einzelne Datenwerte oft aus den Daten vieler Quellen zusammengesetzt sind, ist diese Aufgabe nicht leicht. Um sie zu lösen, werden entlang der Transformation und Integration der Daten, beginnend bei den ursprünglichen Daten und bis hin zum integrierten Ergebnis, entsprechende Metadaten gesammelt. Die Metadaten enthalten Informationen über die Herkunft der Daten, aber auch Informationen über die Weise wie sie erlangt wurden, die Art der Transformationen und andere Statistiken. Sie stehen dann auf Abruf bereit, entweder um den Informationsbedarf eines Nutzers zu befriedigen oder um graphischen Nutzerschnittstellen die Visualisierung zu ermöglichen. Ohne solche Techniken wird integrierten Daten nicht getraut, und insbesondere werden sie nicht verwendet um wichtige Entscheidungen zu treffen.

Zum Abschluss des Vortrags sei angemerkt, dass das Beispiel der Suchmaschinen und anderer im Vortrag genannter Szenarien nur die denkbar einfachsten sind. In typischen Anwendungen für integrierte Informationssysteme sollen bis zu 100 Informationsquellen integriert werden. Jede dieser Quellen speichert komplexe Datensammlungen mit heterogenen Strukturen über viele Tabellen mit jeweils vielen Attributen. Die Anforderungen zur Einrichtung und zum Betrieb solcher Anwendungen sind so hoch, dass große Teams aus Domänenexperten, Datenbankexperten und Programmentwicklern über Jahre hinweg zusammenarbeiten um sie zu entwerfen, planen und realisieren. Die Forschung in der Informationsintegration leistet einen Beitrag, indem Techniken und Werkzeuge entwickelt werden, die große Anteile der anfallenden Arbeit automatisieren oder zumindest Experten unterstützen. Eine vollautomatische Integration heterogener Quellen zu großen integrierten Informationssystemen liegt noch in ferner Zukunft.

Anmerkungen

1 www.metacrawler.com
2 http://www.w3.org/XML/

Literatur

[BLN86] *Carlo Batini, Maurizio Lenzerini, Shamkant B. Navathe*: A Comparative Analysis of Methodologies for Database Schema Integration. ACM Computing Surveys 18(4): 323–364 (1986).

[BR99] *Ricardo A. Baeza-Yates, Berthier A. Ribeiro-Neto*: Modern Information Retrieval ACM Press / Addison-Wesley 1999.

[GL94] *César Galindo-Legaria*: Outerjoins as Disjunctions. In: Proceedings of the ACM International Conference on Management of Data (SIGMOD), 348–358, 1994.

[HS98] *Mauricio A. Hernández and Salvatore J. Stolfo*: Real-world data is dirty: Data cleansing and the merge/purge problem. Data Mining and Knowledge Discovery, 2(1): 9–37, 1998.

[Levy01] *Alon Y. Halevy*: Answering queries using views: A survey. VLDB Journal 10(4): 270–294, 2001.

[MHH00] *Renée J. Miller, Laura M. Haas, Mauricio A. Hernández*: Schema Mapping as Query Discovery. In: Proceedings of the International Conference on Very Large Databases (VLDB): 77–88, 2000.

[Popa+02] *Lucian Popa, Yannis Velegrakis, Renée J. Miller, Mauricio A. Hernández, Ronald Fagin*: Translating Web Data. In: Proceedings of the International Conference on Very Large Databases (VLDB): 598–609, 2002.

[RB01] *Erhard Rahm, Philip A. Bernstein*: A survey of approaches to automatic schema matching. VLDB Journal 10(4): 334–350, 2001.

[SL90] *A.P. Sheth, J.A. Larson*: Federated database systems for managing distributed, heterogeneous, and autonomous databases, ACM Computing Surveys, 22(3): 183–236, 1990.

[Wag74] *R.A. Wagner*: Order-n correction for regular languages. Communications of the ACM 17(5): 265–268, 1974.

Felix Naumann

1971 geboren in Hamburg.
1990 Abitur in Hamburg.
1990–1997 Studium der Wirtschaftsmathematik an der Technischen Universität Berlin.
1997–2000 Promotion als DFG Stipendiat im Graduiertenkolleg „Verteilte Informationssysteme“ an der Humboldt-Universität zu Berlin. Auszeichnung der Dissertation mit dem GI Dissertationspreis 2000.
2001–2002 Visiting Scientist am IBM Almaden Research Center in San Jose, Kalifornien.
Seit 2003 Juniorprofessor für Informationsintegration am Institut für Informatik der Humboldt-Universität zu Berlin.

Ausgewählte Veröffentlichungen

- Quality-driven Integration of Heterogenous Information Systems,
  with Ulf Leser and Johann-Christoph Freytag, International Conference on Very Large Databases (VLDB), Edinburgh, Scotland, 1999.
- From Databases to Information Systems – Information Quality Makes the Difference,
  International Conference on Information Quality (IQ), Cambridge, MA, 2001.
- Quality-driven Query answering for Integrated Information System,
  Lecture Notes in Computer Sciences LNCS 2261, Springer Verlag, Heidelberg, 2002.
- Completeness of Integrated Information Sources,
  with Johann-Christoph Freytag and Ulf Leser, Information Systems (IS) 29(7):583-615, Elsevier, 2004.

In der Reihe **Öffentliche Vorlesungen** sind erschienen:

1 *Volker Gerhardt*
Zur philosophischen Tradition der Humboldt-Universität

2 *Hasso Hofmann*
Die versprochene Menschenwürde

3 *Heinrich August Winkler*
Von Weimar zu Hitler
Die Arbeiterbewegung und das Scheitern der ersten deutschen Demokratie

4 *Michael Borgolte*
„Totale Geschichte" des Mittelalters?
Das Beispiel der Stiftungen

5 *Wilfried Nippel*
Max Weber und die Althistorie seiner Zeit

6 *Heinz Schilling*
Am Anfang waren Luther, Loyola und Calvin – ein religionssoziologisch-entwickungsgeschichtlicher Vergleich

7 *Hartmut Harnisch*
Adel und Großgrundbesitz im ostelbischen Preußen 1800–1914

8 *Fritz Jost*
Selbststeuerung des Justizsystems durch richterliche Ordnungen

9 *Erwin J. Haeberle*
Berlin und die internationale Sexualwissenschaft
Magnus Hirschfeld-Kolloquium, Einführungsvortrag

10 *Herbert Schnädelbach*
Hegels Lehre von der Wahrheit

11 *Felix Herzog*
Über die Grenzen der Wirksamkeit des Strafrechts
Eine Hommage an Wilhelm von Humboldt

12 *Hans-Peter Müller*
Soziale Differenzierung und Individualität
Georg Simmels Gesellschafts- und Zeitdiagnose

13 *Thomas Raiser*
Aufgaben der Rechtssoziologie als Zweig der Rechtswissenschaft

14 *Ludolf Herbst*
Der Marshallplan als Herrschaftsinstrument?
Überlegungen zur Struktur amerikanischer Nachkriegspolitik

15 *Gert-Joachim Glaeßner*
Demokratie nach dem Ende des Kommunismus

16 *Arndt Sorge*
Arbeit, Organisation und Arbeitsbeziehungen in Ostdeutschland

17 *Achim Leube*
Semnonen, Burgunden, Alamannen
Archäologische Beiträge zur germanischen Frühgeschichte des 1. bis 5. Jahrhunderts

18 *Klaus-Peter Johne*
Von der Kolonenwirtschaft zum Kolonat
Ein römisches Abhängigkeitsverhältnis im Spiegel der Forschung

19 *Volker Gerhardt*
Die Politik und das Leben

20 *Clemens Wurm*
Großbritannien, Frankreich und die westeuropäische Integration

21 *Jürgen Kunze*
Verbfeldstrukturen

22 *Winfried Schich*
Die Havel als Wasserstraße im Mittelalter: Brücken, Dämme, Mühlen, Flutrinnen

23 *Herfried Münkler*
Zivilgesellschaft und Bürgertugend
Bedürfen demokratisch verfaßte Gemeinwesen einer sozio-moralischen Fundierung?

24 *Hildegard Maria Nickel*
Geschlechterverhältnis in der Wende
Individualisierung versus Solidarisierung?

25 *Christine Windbichler*
Arbeitsrechtler und andere Laien in der Baugrube des Gesellschaftsrechts
Rechtsanwendung und Rechtsfortbildung

26 *Ludmila Thomas*
Rußland im Jahre 1900
Die Gesellschaft vor der Revolution

27 *Wolfgang Reisig*
Verteiltes Rechnen: Im wesentlichen das Herkömmliche oder etwas grundlegend Neues?

28 *Ernst Osterkamp*
Die Seele des historischen Subjekts
Historische Portraitkunst in Friedrich Schillers „Geschichte des Abfalls der vereinigten Niederlande von der Spanischen Regierung"

29 *Rüdiger Steinlein*
Märchen als poetische Erziehungsform
Zum kinderliterarischen Status der Grimmschen „Kinder- und Hausmärchen"

30 *Hartmut Boockmann*
Bürgerkirchen im späteren Mittelalter

31 *Michael Kloepfer*
Verfassungsgebung als Zukunftsbewältigung aus Vergangenheitserfahrung
Zur Verfassungsgebung im vereinten Deutschland

32 *Dietrich Benner*
Über die Aufgaben der Pädagogik nach dem Ende der DDR

33 *Heinz-Elmar Tenorth*
„Reformpädagogik"
Erneuter Versuch, ein erstaunliches Phänomen zu verstehen

34 *Jürgen K. Schriewer*
Welt-System und Interrelations-Gefüge
Die Internationalisierung der Pädagogik als Problem Vergleichender Erziehungswissenschaft

35 *Friedrich Maier*
„Das Staatsschiff" auf der Fahrt von Griechenland über Rom nach Europa
Zu einer Metapher als Bildungsgegenstand in Text und Bild

36 *Michael Daxner*
Alma Mater Restituta oder Eine Universität für die Hauptstadt

37 *Konrad H. Jarausch*
Die Vertreibung der jüdischen Studenten und Professoren von der Berliner Universität unter dem NS-Regime

38 *Detlef Krauß*
Schuld im Strafrecht
Zurechnung der Tat oder Abrechnung mit dem Täter?

39 *Herbert Kitschelt*
Rationale Verfassungswahl?
Zum Design von Regierungssystemen in neuen Konkurrenzdemokratien

40 *Werner Röcke*
Liebe und Melancholie
Formen sozialer Kommunikation in der ‚Historie von Florio und Blanscheflur'

41 *Hubert Markl*
Wohin geht die Biologie?

42 *Hans Bertram*
Die Stadt, das Individuum und das Verschwinden der Familie

43 *Dieter Segert*
Diktatur und Demokratie in Osteuropa im 20. Jahrhundert

44 *Klaus R. Scherpe*
Beschreiben, nicht Erzählen!
Beispiele zu einer ästhetischen Opposition: Von Döblin und Musil bis zu Darstellungen des Holocaust

45 *Bernd Wegener*
Soziale Gerechtigkeitsforschung: Normativ oder deskriptiv?

46 *Horst Wenzel*
Hören und Sehen – Schrift und Bild
Zur mittelalterlichen Vorgeschichte audiovisueller Medien

47 *Hans-Peter Schwintowski*
Verteilungsdefizite durch Recht auf globalisierten Märkten
Grundstrukturen einer Nutzentheorie des Rechts

48 *Helmut Wiesenthal*
Die Krise holistischer Politikansätze und das Projekt der gesteuerten Systemtransformation

49 *Rainer Dietrich*
Wahrscheinlich regelhaft. Gedanken zur Natur der inneren Sprachverarbeitung

50 *Bernd Henningsen*
Der Norden: Eine Erfindung
Das europäische Projekt einer regionalen Identität

51 *Michael C. Burda*
Ist das Maß halb leer, halb voll oder einfach voll?
Die volkswirtschaftlichen Perspektiven der neuen Bundesländer

52 *Volker Neumann*
Menschenwürde und Existenzminimum

53 *Wolfgang Iser*
Das Großbritannien-Zentrum in kulturwissenschaftlicher Sicht
Vortrag anläßlich der Eröffnung des Großbritannien-Zentrums an der Humboldt-Universität zu Berlin

54 *Ulrich Battis*
Demokratie als Bauherrin

55 *Johannes Hager*
Grundrechte im Privatrecht

56 *Johannes Christes*
Cicero und der römische Humanismus

57 *Wolfgang Hardtwig*
Vom Elitebewußtsein zur Massenbewegung – Frühformen des Nationalismus in Deutschland 1500 – 1840

58 *Elard Klewitz*
Sachunterricht zwischen Wissenschaftsorientierung und Kindbezug

59 *Renate Valtin*
Die Welt mit den Augen der Kinder betrachten
Der Beitrag der Entwicklungstheorie Piagets zur Grundschulpädagogik

60 *Gerhard Werle*
Ohne Wahrheit keine Versöhnung!
Der südafrikanische Rechtsstaat und die Apartheid-Vergangenheit

61 *Bernhard Schlink*
Rechtsstaat und revolutionäre Gerechtigkeit. Vergangenheit als Zumutung?
(Zwei Vorlesungen)

62 *Wiltrud Gieseke*
Erfahrungen als behindernde und fördernde Momente im Lernprozeß Erwachsener

63 *Alexander Demandt*
Ranke unter den Weltweisen
Wolfgang Hardtwig
Die Geschichtserfahrung der Moderne und die Ästhetisierung der Geschichtsschreibung: Leopold von Ranke
(Zwei Vorträge anläßlich der 200. Wiederkehr des Geburtstages Leopold von Rankes)

64 *Axel Flessner*
Deutsche Juristenausbildung
Die kleine Reform und die europäische Perspektive

65 *Peter Brockmeier*
Seul dans mon lit glacé – Samuel Becketts Erzählungen vom Unbehagen in der Kultur

66 *Hartmut Böhme*
Das Licht als Medium der Kunst
Über Erfahrungsarmut und ästhetisches Gegenlicht in der technischen Zivilisation

67 *Sieglind Ellger-Rüttgardt*
Berliner Rehabilitationspädagogik: Eine pädagogische Disziplin auf der Suche nach neuer Identität

68 *Christoph G. Paulus*
Rechtsgeschichtliche und rechtsvergleichende Betrachtungen im Zusammenhang mit der Beweisvereitelung

69 *Eberhard Schwark*
Wirtschaftsordnung und Sozialstaatsprinzip

70 *Rosemarie Will*
Eigentumstransformation unter dem Grundgesetz

71 *Achim Leschinsky*
Freie Schulwahl und staatliche Steuerung
Neue Regelungen des Übergangs an weiterführende Schulen

72 *Harry Dettenborn*
Hang und Zwang zur sozialkognitiven Komplexitätsreduzierung: Ein Aspekt moralischer Urteilsprozesse bei Kindern und Jugendlichen

73 *Inge Frohburg*
Blickrichtung Psychotherapie: Potenzen – Realitäten – Folgerungen

74 *Johann Adrian*
Patentrecht im Spannungsfeld von Innovationsschutz und Allgemeininteresse

75 *Monika Doherty*
Verständigung trotz allem. Probleme aus und mit der Wissenschaft vom Übersetzen

76 *Jürgen van Buer*
Pädagogische Freiheit, pädagogische Freiräume und berufliche Situation von Lehrern an Wirtschaftsschulen in den neuen Bundesländern

77 *Flora Veit-Wild*
Karneval und Kakerlaken
Postkolonialismus in der afrikanischen Literatur

78 *Jürgen Diederich*
Was lernt man, wenn man nicht lernt? Etwas Didaktik „jenseits von Gut und Böse“ (Nietzsche)

79 *Wolf Krötke*
Was ist ‚wirklich‘?
Der notwendige Beitrag der Theologie zum Wirklichkeitsverständnis unserer Zeit

80 *Matthias Jerusalem*
Die Entwicklung von Selbstkonzepten und ihre Bedeutung für Motivationsprozesse im Lern- und Leistungsbereich

81 *Dieter Klein*
Globalisierung und Fragen an die Sozialwissenschaften: Richtungsbestimmter Handlungszwang oder Anstoß zu einschneidendem Wandel?

82 *Barbara Kunzmann-Müller*
Typologisch relevante Variation in der Slavia

83 *Michael Parmentier*
Sehen Sehen
Ein bildungstheoretischer Versuch über Chardins ‚L’enfant au toton‘

84 *Engelbert Plassmann*
Bibliotheksgeschichte und Verfassungsgeschichte

85 *Ruth Tesmar*
Das dritte Auge
Imagination und Einsicht

86 *Ortfried Schäffter*
Perspektiven erwachsenenpädagogischer Organisationsforschung

87 *Kurt-Victor Selge, Reimer Hansen, Christof Gestrich*
Philipp Melanchthon 1497 – 1997

88 *Karla Horstmann-Hegel*
Integrativer Sachunterricht – Möglichkeiten und Grenzen

89 *Karin Hirdina*
Belichten. Beleuchten. Erhellen
Licht in den zwanziger Jahren

90 *Marion Bergk*
Schreibinteraktionen: Verändertes Sprachlernen in der Grundschule

91 *Christina von Braun*
Architektur der Denkräume
James E. Young
Daniel Libeskind’s Jewish Museum in Berlin: The Uncanny Art of Memorial Architecture
Daniel Libeskind
Beyond the Wall
Vorträge anläßlich der Verleihung der Ehrendoktorwürde an Daniel Libeskind

92 *Christina von Braun*
Warum Gender-Studies?

93 *Ernst Vogt, Axel Horstmann*
August Boeckh (1785 – 1867). Leben und Werk
Zwei Vorträge

94 *Engelbert Plassmann*
Eine „Reichsbibliothek“?

95 *Renate Reschke*
Die Asymmetrie des Ästhetischen
Asymmetrie als Denkfigur historisch-ästhetischer Dimension

96 *Günter de Bruyn*
Altersbetrachtungen über den alten Fontane
Festvortrag anläßlich der Verleihung der Ehrendoktorwürde

97 *Detlef Krauß*
Gift im Strafrecht

98 *Wolfgang Thierse, Renate Reschke, Achim Trebeß, Claudia Salchow*
Das Wolfgang-Heise-Archiv. Plädoyers für seine Zukunft
Vorträge

99 *Elke Lehnert, Annette Vogt, Ulla Ruschhaupt, Marianne Kriszio*
Frauen an der Humboldt-Universität 1908 – 1998
Vier Vorträge

100 *Bernhard Schlink*
Evaluierte Freiheit?
Zu den Bemühungen um eine Verbesserung der wissenschaftlichen Lehre

101 *Heinz Ohme*
Das Kosovo und die Serbische Orthodoxe Kirche

102 *Gerhard A. Ritter*
Der Berliner Reichstag in der politischen Kultur der Kaiserzeit
Festvortrag anläßlich der Verleihung der Ehrendoktorwürde mit einer Laudatio von Wolfgang Hardtwig

103 *Cornelius Frömmel*
Das Flair der unendlichen Vielfalt

104 *Verena Olejniczak Lobsien*
„Is this the promised end?“ Die Apokalypse des King Lear, oder: Fängt Literatur mit dem Ende an?

105 *Ingolf Pernice*
Kompetenzabgrenzung im Europäischen Verfassungsverbund

106 *Gerd Irrlitz*
Das Bild des Weges in der Philosophie

107 *Helmut Schmidt*
Die Selbstbehauptung Europas im neuen Jahrhundert. Mit einer Replik von Horst Teltschik

108 *Peter Diepold*
Internet und Pädagogik
Rückblick und Ausblick

109 *Artur-Axel Wandtke*
Copyright und virtueller Markt oder Das Verschwinden des Urhebers im Nebel der Postmoderne?

110 *Jürgen Mittelstraß*
Konstruktion und Deutung
Über Wissenschaft in einer Leonardo- und Leibniz-Welt

111 *Göran Persson*
European Challenges. A Swedish Perspective. Mit einer Replik von Janusz Reiter

112 *Hasso Hofmann*
Vom Wesen der Verfassung

113 *Stefanie von Schnurbein*
Kampf um Subjektivität
Nation, Religion und Geschlecht in zwei dänischen Romanen um 1850

114 *Ferenc Mádl*
Europäischer Integrationsprozess. Ungarische Erwartungen. Mit einer Replik von Dietrich von Kyaw

115 *Ernst Maug*
Konzerne im Kontext der Kapitalmärkte

116 *Herbert Schnädelbach*
Das Gespräch der Philosophie

117 *Axel Flessner*
Juristische Methode und europäisches Privatrecht

118 *Sigrid Jacobeit*
KZ-Gedenkstätten als nationale Erinnerungsorte
Zwischen Ritualisierung und Musealisierung

119 *Vincent J.H. Houben*
Südostasien. Eine andere Geschichte

120 *Étienne Balibar, Friedrich A. Kittler, Martin van Creveld*
Vom Krieg zum Terrorismus?
Mosse-Lectures 2002/2003

121 *Hans Meyer*
Versuch über die Demokratie in Deutschland

122 *Joachim Kallinich*
Keine Atempause – Geschichte wird gemacht
Museen in der Erlebnis- und Mediengesellschaft

123 *Anusch Taraz*
Zufällige Beweise

124 *Carlo Azeglio Ciampi*
L'amicizia italo-tedesca al servizio dell'integrazione europea. Die italienisch-deutsche Freundschaft im Dienste der europäischen Integration
Johannes Rau
Deutschland, Italien und die europäische Integration

125 *Theodor Schilling*
Der Schutz der Menschenrechte gegen den Sicherheitsrat und seine Mitglieder
Möglichkeiten und Grenzen

126 *Wolfgang Ernst*
Medienwissen(schaft) zeitkritisch
Ein Programm aus der Sophienstraße

127 *Hilmar Schröder*
Klimaerwärmung und Naturkatastrophen im Hochgebirge
Desaster oder Stabilität im 21. Jahrhundert?

128 *Kiran Klaus Patel*
Nach der Nationalfixiertheit
Perspektiven einer transnationalen Geschichte

129 *Susanne Frank*
Stadtplanung im Geschlechterkampf
Ebenezer Howard und Le Corbusier

130 *Matthias Langensiepen*
Modellierung pflanzlicher Systeme
Perspektiven eines neuen Forschungs- und Lehrgebietes

131 *Michael Borgolte*
Königsberg – Deutschland – Europa
Heinrich August Winkler und die Einheit der Geschichte. Festvortrag anlässlich des 65. Geburtstages

132 *Guy Verhofstadt*
The new European Constitution – from Laeken to Rome

133 *Elke Hartmann*
Zur Geschichte der Matriarchatsidee

134 *Felix Naumann*
Informationsintegration