Datenunvollständigkeit aufgrund der mangelnden Modellierungsmächtigkeit aktuell dominierender Datenmodelle

Fabian Panse

31. März 2009

Zusammenfassung

In den am weitest verbreiteten Datenmodellen, speziell dem relationalen Datenmodell, werden Informationen über die Ausprägungen einzelner Objekteigenschaften in Attributen gespeichert. In vielen Fällen (z.B. bei partiellen Informationen) ist eine Darstellung durch einzelne Elemente des dem Attribut zugehörigen Wertebereichs jedoch nicht möglich und erfordert die Anwendung spezieller Konzepte (z.B. Nullwerte). In aktuell verwendeten Modellen sind diese Konzepte jedoch nur unzureichend auf die notwendigen Erfordernisse ausgelegt. Die ursprünglich vorliegenden Informationen lassen sich daher oft nicht wieder aus den gespeicherten Daten zurück gewinnen. Bisherige Ansätze zur Behebung dieses Problems haben sich aus unterschiedlichen Gründen nicht durchsetzen können. Die hier beschriebene Arbeit enthält daher einen entsprechenden Vorschlag, der sowohl den Informationsverlust während der Datenspeicherung verringern als auch die Schwächen der bisherigen Lösungsansätze hinsichtlich eines fehlenden Durchsetzungsvermögen vermeiden soll. Ersteres wird durch eine Verwendung mehreren Nullwerten ermöglicht, letzteres beruht hauptsächlich auf der Vermeidung gravierender Abweichungen von den aktuell vorherrschenden Modellen. Da dies wiederum eine Beibehaltung wichtiger und fundamentaler Konzepte erfordert, muss die Auswertung der verschiedenen Nullwerte in der drei-wertigen Logik erfolgen. Neben einer Kompatibilität zu den vorherrschenden Modellen, bietet dieses Vorgehen zudem die Vorteile einer geringen Modellkomplexität und ermöglicht eine intuitive Handhabung der auf dem neu entworfenen Modell basierenden Systeme.

1 Einleitung

Die Qualität der in Datenbanken gespeicherten operationalen Daten ist in den letzten Jahren zu einem wichtigen und viel beachteten Faktor geworden. Dies betrifft unter anderem auch das Qualitätskriterium der Datenvollständigkeit. Ausgehend von der Annahme, dass eine dauerhafte und korrekte Speicherung der Daten durch das verwendete Datenbanksystem gewährleistet werden kann, lässt sich unvollständige Datenbasis auf die Minderwertigkeit zweier wesentlicher Vorgänge, der Informationserhebung und der Datenspeicherung, zurückführen. Während die Folgen einer unzureichenden Informationserhebung für die Vollständigkeit einer Datenbasis offensichtlich sind, spielt die Datenspeicherung eine weitere wichtige, heutzutage aber leider weitgehend vernachlässigte Rolle. Die in den heutigen dominierenden Datenbanksystemen vorhandenen Konzepte reichen nicht aus um verschiedenartige Informationen immer durch verschiedenartige Datenwerte oder Datensätze darstellen zu können. Der daraus entstehende Informationsverlust ist in einer minderwertigen Darstellungsfähigkeit des verwendeten Datenbankschemas begründet. Diese lässt sich wiederum auf eine mangelnde Modellierungsmächtigkeit des dem Schemaentwurf zugrundeliegenden logischen Datenmodells zurückzuführen. Bisherige Lösungsansätze konnten zwar den bei der Datenspeicherung auftretenden Informationsverlust verringern, aufgrund der fehlenden Akzeptanz seitens der Datenbank-Gemeinde (Anbieter wie auch Nutzer), konnten sie sich bis dato aber nicht bewähren. Dies zeigt, dass neben der Behebung des Informationsverlustes noch weitere, nicht datenbankspezifische Aspekte berücksichtigt werden müssen. Der hier vorgestellte

Lösungsansatz basiert daher auf dem Entwurf eines Modells, dessen Modellierungsmächtigkeit zum Einen ausreicht, um eine möglichst verlustfreie Datenspeicherung zu ermöglichen, zum Anderen aber auch mögliche Akzeptanzprobleme sowohl von den Systemanbietern als auch den -nutzern umgehen kann.

2 Problemstellung

2.1 Datenvollständigkeit

Die Datenvollständigkeit ist ein Qualitätskriterium, welches den Vollständigkeitsgrad des durch die Daten gespeicherten Wissen im Verhältnis zu dem maximal vorhandenen Wissens der betroffenen Anwendungswelt wiedergibt. Die Datenvollständigkeit kann dabei in zwei Dimensionen unterteilt werden ([8]). Die Abdeckung der Daten gibt an, wieviele der vorhandenen Objekte und Beziehungen der zu modellierenden Anwendungswelt tatsächlich in der Datenbasis durch einen Datensatz gespeichert sind. Die Dichte der Daten spezifiziert, wieviele Eigenschaften der gespeicherten Objekte und Beziehungen durch geeignete Datenwerte in den zugehörigen Attributen informationserhaltend repräsentiert werden. Bezüglich der Daten-Dichte kann eine Unvollständigkeit in genau zwei Fällen auftreten. Entweder liegen bezüglich der zu speichernden Eigenschaft keine vollständigen Informationen vor, und/oder die vorliegenden Informationen lassen sich nicht verlustlos durch Daten speichern.

2.2 Informationsverlust während der Datenspeicherung

Informationen werden während der Datenspeicherung durch schemakonforme Datensätze dargestellt. Handelt es sich dabei um ein Datenbankschema, bei welchem zur Speicherung von Objekteigenschaften Attribute verwendet werden, so stehen zur Darstellung der Ausprägung einer Objekteigenschaft lediglich die Elemente des dem zugehörigen Attribut zugewiesenen Wertebereichs zur Verfügung. In vielen Datenbanksystemen soll zudem noch die Atomarität der Attributwerte gelten (z.B. aufgrund der 1. Normalform). Dies bedeutet, dass sich jede der zu speichernden Informationen durch einen einzelnen Wert aus dem zugehörigen Wertebereich darstellen lassen muss. Ist dies nicht möglich, muss auf andere Darstellungskonzepte zurückgegriffen werden. Das am meisten gebräuchlichste dieser Konzepte ist die Verwendung von Nullwerten. Enthält die Menge der Ausprägungsinformationen jedoch verschiedenartige Informationen, die sich nur durch einen gleichen Nullwert darstellen lassen, bildet die Datenspeicherung Informationen mit durchaus unterschiedlichen Informationsgehalten auf einen gleichen Datenwert ab und ist folglich nicht injektiv. Da eine nicht injektive Abbildung keine inverse Abbildung besitzt, können die ursprünglich vorhandenen, aus der zu modellierenden Welt erhobenen Informationen nicht wieder aus den gespeicherten Datensätzen hergeleitet werden. Ein irreperabler Informationsverlust während der Datenspeicherung ist demnach die Folge.

Welche Mittel zur Datenspeicherung zur Verfügung stehen, hängt also von dem verwendeten Datenbankschema und somit u.a. von der Modellierungsmächtigkeit des beim Schemaentwurf verwendeten Datenmodells ab. Die am weitest verbreiteten und von allen großen relationalen Datenbanksystemen verwendeten Variationen des relationalen Datenmodell besitzten lediglich den einzigen Nullwert *null*. Als ein diese Variationen stellvertretendes Datenmodell wird im Folgenden das als DM_L^{SQL} bezeichnete Modell des aktuellen SQL-Standards betrachtet. Eine Verwendung dieses Modells bedeutet, dass zur Darstellung aller verschiedenartiger, nicht durch einen atomaren Wert darstellbarer Informationen nur ein einziger Nullwert vorhanden ist. Da dieser Nullwert auch für den Fall, dass keinerlei Informationen über die betreffende Objekteigenschaft vorliegen verwendet wird und während der Informationsgewinnung keine falschen Annahmen getroffen werden dürfen, besitzt der Nullwert *null* eine *no-information*-Interpretation. Wird also eine Information auf *null* abgebildet, so suggeriert der aus der Speicherung dieses Objektes

resultierende Datensatz, unabhängig von dem tatsächlich darzustellenden Informationsgehalt, dass zu der betreffenden Objekteigenschaft keine Informationen vorhanden sind. Daraus wiederum folgt, dass jede Information, die durch *null* repräsentiert wird, verloren geht. Dies ist genau dann der Fall, wenn entweder die Ausprägung einer existierenden Objekteigenschaft nicht exakt bekannt ist (z.B. das Alter einer Person ist gar nicht oder nur begrenzt (Alter ≥ 18) bekannt), die Ausprägung zwar bekannt aber nicht konstant ist und innerhalb eines begrenzten Wertebereichs variiert (z.B. das schwankende Gewicht einer Person) oder diese Eigenschaft für das betreffende Objekt zumindest vorübergehend schlicht nicht vorhanden ist (z.B. eine Person hat kein Telefon). Das Informationen über die Nichtexistenz von Objekteigenschaften ebenfalls auf *null* abgebildet werden, zeigt, dass es sich bei den durch *null* repräsentierten Informationen nicht immer um unzureichendes Wissen und somit nicht zwangsweise um das Resultat einer mangelnden Informationserhebung handeln muss. Denn während die Unbekanntheit eines Wertes eine Unkenntnis darstellt, die durch vorhandene Teilinformationen lediglich eingeschränkt werden kann, ist die Nichtexistenz eines Wertes mit keinerlei Unwissen verbunden. Wie eine bekannte Ausprägung einer Objekteigenschaft stellt sie eine eindeutige und klare Information dar (in der Aussage „Die Person Müller hat kein Telefon“ ist keinerlei Unwissen enthalten). Diese exakte Erkenntnis (die Erkenntnis der Nichtexistenz) geht bei der Abbildung auf *null* jedoch wieder verloren und stellt aufgrund der fehlenden Unkenntnis den qualitativ schwerwiegendsten Informationsverlust dar. Als ein den bei der Datenspeicherung auftretenden Informationsverlust veranschaulichendes Beispiel soll im Folgenden eine Relation *Schüler_ belegt_ Fach(Schüler,Fach,Note)* dienen. In dieser Relation soll gespeichert werden, welcher Schüler (Name) welches Fach (Bezeichnung) belegt oder belegt hat. Desweiteren kann ein Schüler eine Prüfung über eben dieses Fach ablegen. Die resultierende Note ist für deutsche Oberstufen typischerweise eine Zahl von 0-15 und wird nach Abschluß der Prüfung in dem Attribut Note vermerkt. Nach der Informationserhebung ist bekannt, dass der Schüler Klaus im Fach Mathematik noch keine Prüfung abgelegt hat und dass der Schüler Hans im selben Fach bereits die Prüfung sowohl abgelegt als auch bestanden (Note 5-15) hat, seine genaue Note jedoch unbekannt ist. Vom Schüler Karl ist nur bekannt, dass er die Mathematikprüfung abgelegt hat, mit welcher Note er diese abgeschlossen hat, ist gänzlich unbekannt. Vom Schüler Gustav ist nur bekannt, dass er das Fach Mathematik belegt hat; über die Tatsache, ob er eine Prüfung abgelegt hat oder nicht, liegen keinerlei Informationen vor. Der aus einer idealen und informationserhaltenden Abbildung resultierende Datenbestand ist in Tabelle 1 dargestellt ('-' für nicht existent und '?' für keinerlei Informationen). Die Information, die über das Prüfungsergebniss des jeweiligen Schülers bekannt ist, lässt sich allerdings in keinem der vier Fälle durch eine der Zahlen von 0-15 darstellen. Demzufolge erfolgt bei der Verwendung eines auf dem Datenmodell DM_L^{SQL} basierenden Schemas jedesmal der Eintrag des Nullwertes *null* (siehe Tabelle 2).

Schüler	Fach	Note
...	...	...
Klaus	Mathematik	-
Hans	Mathematik	5-15
Karl	Mathematik	0-15
Gustav	Mathematik	?
...	...	...

Tabelle 1: Ergebnis einer idealen, verlustlosen Datenspeicherung

Schüler	Fach	Note
...	...	...
Klaus	Mathematik	*null*
Hans	Mathematik	*null*
Karl	Mathematik	*null*
Gustav	Mathematik	*null*
...	...	...

Tabelle 2: Ergebnis einer verlustbehafteten Datenspeicherung

Will nun ein Nutzer Informationen aus den Daten dieser Relation gewinnen, kann er zwischen den einzelnen Nullwert-Einträgen nicht mehr unterscheiden und muss um falsche Schlussfolgerungen zu vermeiden, für jeden Schüler annehmen, dass keine Informationen über dessen Prüfungsstand vorliegen. Demzufolge würden auf eine Anfrage, welche Schüler die Mathematikprüfung bestanden haben (Note ≥ 5), keiner der vier obigen Schüler als sicheres und alle als mögliches Ergebniss

resultieren. Dabei lagen anfangs zumindest für Klaus (Prüfung noch nicht bestanden) und Hans (Prüfung sicher bestanden) ausreichend Informationen vor, um sie bezüglich der gestellten Anfrage eindeutig und klar qualifizieren zu können.

3 Verwandte Arbeiten

Die unzureichende Modellierungsmächtigkeit der aktuell dominierenden relationalen Datenmodelle war bereits in den vergangenen Jahren und Jahrzehnten Bestandteil verschiedenartiger Lösungsansätze. Während eine Vielzahl an früheren Ansätzen das Nullwert-Konzept verwendeten (u.a. [10], [6], [7], [3], [5], [1], [2]), bauten andere Modelle, wie z.B. Wahrscheinlichkeitstheoretische- (u.a. [4]) oder Fuzzy-relationale Datenmodelle (u.a. [9]) auf völlig neuen Konzepten auf. Da sich nach der bisherigen Ansicht mehrere Nullwert-Arten nicht in der drei-wertigen Logik interpretieren lassen, basieren die auf dem Nullwert-Konzept beruhenden Ansätze auf vier- oder mehrwertigen Logiken. Alle oben genannten Ansätze haben gemeinsam, dass sie sich bereits auf fundamentaler Ebene von den aktuell dominierenden Modellen unterscheiden. Obgleich diese Ansätze mehr oder weniger zur Verringerung des Informationsverlusts während der Datenspeicherung gegeignet sind, hat sich bisher keines der aus diesen Ansätzen resultierenden Modelle durchsetzen können. Dies lässt sich auf zwei wesentliche Eigenschaften zurückführen. Zum Einem besitzen viele dieser Ansätze eine sehr hohe Komplexität (z.B. resultierend aus vier- oder mehrwertigen Logiken), welche die Verwendung für einfache Datenbanknutzer sehr erschwert. Zum Anderen führen die gravierenden Unterscheidungungen fundamentaler Modelleigenschaften wiederum zu einem stark veränderten Verhalten der resultierenden Datenbanksysteme. Aktuelle Datenbanksysteme sind jedoch seit Jahren etabliert, aktuell vorhandene Datenbank-Anwendungen sind auf diese ausgerichtet und sowohl Datenbankanbieter als auch -nutzer haben sich an deren Systemverhalten gewöhnt. Die Aussicht auf durch einen Wechsel des zugrundeliegenden Datenmodells entstehende Veränderungen dieses Systemverhaltens finden daher nur wenig Anklang in der breiten Masse der Datenbank-Gemeinde. Die fehlende Akzeptanz der bisherigen Lösungsansätze ist neben der bereits erwähnten Komplexität daher wohl hauptsächlich durch marktwirtschaftliche (Hohe Kosten bei Systemumstellung) und psychologische Faktoren (Umstellung bei der Systemnutzung) verschuldet.

4 Lösungsansatz

Die bei der Datenspeicherung in einen relationalen Datenbanksystem vollzogene Abbildung von Informationen auf operationale Datensätze ist nur dann verlustfrei, wenn sie injektiv und somit umkehrbar ist. Identische Abbildungen verschiedener Informationen und der damit verbundene Informationsverlust sollten daher auf ein Minimum reduziert werden. Um auch den aus den marktwirtschaftlichen und psychologischen Faktoren resultierenden Anforderungen zu genügen, erfolgte der Entwurf eines logischen Datenmodells, das den bestehenden Informationsverlust durch eine erhöhte Modellierungsmächtigkeit beschränken, zugleich aber auch die Schwächen der bisherigen Ansätze in Form einer mangelnden Akzeptanz vermeidet kann. Die erhöhte Modellierungsmächtigkeit ist, wie bereits aus anderen Ansätze bekannt, durch die Hinzunahme neuer Nullwerte erfolgt. Zur Vermeidung einer fehlenden Akzeptanz wurden neben einer geringen Komplexität und einer intuitiven Benutzung besonders weitreichende Abweichungen bezüglich des Datenmodells DM_L^{SQL} vermieden. Die sich daraus ergebende Modell-Kompatibilität zum Datenmodell DM_L^{SQL} schließt insbesondere den Erhalt der dreiwertigen Logik und andere an den bisherigen Nullwert *null* verknüpften Modellkomponenten mit ein. Der Lösungsansatz basiert dabei vornehmlich auf der Tatsache, dass eine sinnvolle Auswertung von relationalen Vergleichsoperatoren bezüglich verschiedener Nullwerte innerhalb der drei-wertigen Logik, entgegen weitläufiger Meinung, möglich ist. Da sich eine allgemeine Modellierung von partiellen Informa-

tionen hinsichtlich der beabsichtigten Modell-Kompatibilität als nicht trivial erweist, beschränkt sich dieses Modell zunächst auf eine verlustlose Darstellung von nichtexistenten und existenten aber unbekannten Objekteigenschaften. Eine allgemeine Modellierung partieller Informationen soll dann durch spätere Erweiterungen erfolgen, ohne dabei die Kompatibilität einzubüßen oder, falls nicht anders möglich, die auftretenden Abweichungen zumindest minimal zu halten.

Um das oben genannte Ziel zu erreichen, wurden drei Nullwerte eingeführt. Neben dem bereits bekannten Nullwert *null*, welcher zur Repräsentation des Falles, dass zu einem Wert keinerlei Informationen vorliegen verwendet wird, handelt es sich dabei um den Nullwert *nE* zur Darstellung nicht existender und den Nullwert *uk* zur Darstellung existierender aber unbekannter Objekteigenschaften. Um die durch die zusätzlichen Nullwerte dargestellten Informationen auch wieder aus den gespeicherten Daten gewinnen zu können, wurden neben dem aktuell vorhandenen und an die neuen Nullwerte angepaßten $isNull$-Prädikat noch die neu definierten Prädikate $isExistent$, $isKnown$ und $isSpecified$ benötigt. Anforderungen an Attributwerte, wie sie sich zum Beispiel aus der Notwendigkeit eindeutiger Primärschlüssel ergeben, erforderten Maßnahmen, die eine solche Einschränkungen der Attributsausprägungen auch in der Gegenwart mehrerer Nullwerte gewährleisten können, bzw. eine Umsetzung dieser ermöglichen. Dafür reichte die bisherige Spezifikation NOT NULL nicht mehr aus und musste um die weiteren Spezifikationen EXISTENT, KNOWN und SPECIFIED ergänzt werden. Weitere wichtige und zubeachtende Aspekte sind sowohl die Auswertungen der Nullwerte in arithmetischen Funktionen und Aggregationsfunktionen als auch deren Verwendung in weiteren Nullwert bezogenen Konzepten wie Outer Unions, Outer Joins oder in der Sortierung von Tupeln. Eingehendere Erläuterungen der hinzugefügten Nullwerte, deren Auswertung innerhalb der drei-wertigen Logik sowie die Gestaltung der anderen bereits kurz angesprochenen und für die Handhabung mehrerer Nullwerte notwendigen Konzepte sind Teil des Vortrags und werden dort genauer behandelt.

Literatur

[1] K. bun Yue. A More General Model For Handling Missing Information In Relational DataBases Using A 3-Valued Logic. *SIGMOD Record*, 20(3):43–49, 1991.

[2] K. S. Candan, J. Grant, and V. S. Subrahmanian. A Unified Treatment of Null Values Using Constraints. *Inf. Sci.*, 98(1-4):99–156, 1997.

[3] E. F. Codd. More Commentary on Missing Information in Relational Databases (Applicable and Inapplicable Information). *SIGMOD Record*, 16(1):42–50, 1987.

[4] D. Dey and S. Sarkar. A Probabilistic Relational Model and Algebra. *ACM Trans. Database Syst.*, 21(3):339–369, 1996.

[5] G. H. Gessert. Four valued logic for relational database systems. *SIGMOD Record*, 19(1):29–35, 1990.

[6] T. Imielinski and W. Lipski. Incomplete Information in Relational Databases. *J. ACM*, 31(4):761–791, 1984.

[7] A. M. Keller and M. Winslett. On the Use of an Extended Relational Model to Handle Changing Incomplete Information. *IEEE Trans. Software Eng.*, 11(7):620–633, 1985.

[8] F. Naumann et al. Completeness of integrated information sources. *Inf. Syst.*, 29(7):583–615, 2004.

[9] G. D. Tré et al. Null values in fuzzy databases. *J. Intell. Inf. Syst.*, 30(2):93–114, 2008.

[10] C. Zaniolo. Database relations with null values . In *PODS*, pages 27–33, 1982.