

Zeitschrift: Geschichte und Informatik = Histoire et informatique
Herausgeber: Verein Geschichte und Informatik
Band: 11 (2000)

Artikel: Wie bringe ich Barzabel Facin auf den Bildschirm? Als Studierender oder Doktorand auf der Suche nach der geeigneten Datenbankarchitektur
Autor: Folini, Christian
DOI: <https://doi.org/10.5169/seals-8925>

Nutzungsbedingungen

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften auf E-Periodica. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. Das Veröffentlichen von Bildern in Print- und Online-Publikationen sowie auf Social Media-Kanälen oder Webseiten ist nur mit vorheriger Genehmigung der Rechteinhaber erlaubt. [Mehr erfahren](#)

Conditions d'utilisation

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. La reproduction d'images dans des publications imprimées ou en ligne ainsi que sur des canaux de médias sociaux ou des sites web n'est autorisée qu'avec l'accord préalable des détenteurs des droits. [En savoir plus](#)

Terms of use

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. Publishing images in print and online publications, as well as on social media channels or websites, is only permitted with the prior consent of the rights holders. [Find out more](#)

Download PDF: 12.04.2026

ETH-Bibliothek Zürich, E-Periodica, <https://www.e-periodica.ch>

Wie bringe ich Barzabal Facin auf den Bildschirm? Als Studierender oder Doktorand auf der Suche nach der geeigneten Datenbankarchitektur

Christian Folini

Einleitung¹

In einem einführenden Werk zur historischen Fachinformatik legte Manfred Thaller 1987 die spezifischen Eigenschaften historischer Software dar: «*Historical software deals with problems (a) not appearing in other disciplines, (b) on hardware as available to a typical historian and can (c) be controlled with a level of skills a historian can be expected to acquire without re-focussing his main research interests*». ²

In diesem Aufsatz sollen die Möglichkeiten vorgestellt werden, welche relationale Datenbanken der Forschung eröffnen. Dabei stehen die Bedürfnisse von Studenten und Doktoranden im Zentrum der Überlegungen. In Anlehnung an Thallers Zitat wendet sich die Arbeit an Historiker und Historikerinnen ohne vertiefte Vorkenntnisse, welche über die Benutzung von Textverarbeitung und ähnlicher Standardsoftware hinausgehen. Den theoretischen Betrachtungen stelle ich Beispiele aus einem Projekt zur Sozialgeschichte der Deutschen Mystik im 13. und 14. Jahrhundert zur Seite. ³

Planung des Projektes und Wahl der Architektur

Wenn man in einem Projekt den Computer als Hilfsmittel einsetzen möchte, tut man gut daran, seine Rolle von Beginn weg klar vorzugeben.

1 Der Vortrag wurde für die Drucklegung überarbeitet und in einigen Punkten erweitert.

2 Thaller, Manfred: «Methods and Techniques of Historical Computation». In: Denley, Peter; Hopkin, Deian (Hg.): *History and Computing*. Manchester 1987, S. 147 - 156.

3 Eine Vorstellung des Projektes und erste Ergebnisse finden sich bei Folini, Christian: «Möglichkeiten der Vernetzung von Quellentexten am Beispiel einer Datenbank zur Frauenmystik». In: *Das Mittelalter*, Nr. 4, 1999, S. 171 - 177; Ders.; Palaia, Alberto: «Die Dominikanerinnenklöster Töss und St. Katharinental. Sozialgeschichtliche Annäherungen». In: Signori, Gabriela (Hg.): *Lesen, Schreiben, Sticken und Erinnern. Beiträge zur Kultur- und Sozialgeschichte mittelalterlicher Frauenklöster, Kolloquium Kloster Ebsdorf* (= Religion in der Geschichte 7). Bielefeld 2000 (im Druck).

Soll der Rechner dabei für die Quellenarbeit eingesetzt werden, hängt die Wahl der Architektur und Software entscheidend von der Art dieser Quellen ab.

Zu unterscheiden sind dabei zwei Arten von Quellen: auf der einen Seite die listenartig strukturierten Quellen wie Memorialbücher des frühen und hohen Mittelalters oder die Quellen des modernen Verwaltungsstaates, welche dem Historiker mit ihren Listen und Tabellen ein reiches Feld bieten. Auf der anderen Seite die Quellen in Prosaform. Es ist bekannt, dass der Computer mit festen Strukturen besser zurecht kommt. Diese Ausrichtung erschwert den Geisteswissenschaftlern die Arbeit und geht auf die Interessen der Wirtschaft und der Naturwissenschaften zurück, welche im Bereich der Softwareentwicklung tonangebend sind.

Diese Ausrichtung bringt neben einem Schwergewicht der Software auch ein Schwergewicht der Literatur, des Kursangebots und des Supports vor Ort mit sich. Will man diese Möglichkeiten nutzen, dann sollte man sich heute für ein relationales Datenbanksystem entscheiden.⁴

Im Kern liegt die Ablage der Daten in verschiedenen Tabellen (Relationen), die mittels Schlüsselfeldern miteinander verknüpft werden. Diese Verknüpfungen erlauben den einfachen Zugriff auf den gesamten Datenpool oder Teilmengen desselben, was sich am einfachsten mit dem Modell der Mengenlehre erklären lässt: Aus der Menge der Personen sucht die Datenbank beispielsweise diejenigen heraus, welche bestimmte Kriterien erfüllen. Diese Entscheidung hat den Vorteil, dass man dann auf vergleichsweise gute Unterstützung zurückgreifen kann. Wie im Folgenden gezeigt wird, erschwert diese Wahl aber den Umgang mit der Gruppe der Prosaquellen.

Wie die anderen Beiträge dieses Bandes aufzeigen, gibt es Alternativen zu relationalen Datenbanken. So existieren verschiedene Ansätze, welche die Quellentexte nicht in das relationale Tabellenmodell einpassen wollen, sondern darauf abzielen, die Quellen in ihrer eigenen Struktur zu belassen. Dieser Beitrag konzentriert sich hingegen auf die relationalen Systeme.

4 Die Grundlagen des relationalen Ansatzes diskutieren die folgenden Arbeiten: Codd, Edgar F.: *The Relational Model for Database Management* (Version 2). Reading Mass., Wokingham etc. 1990; Fleming, Candace C.; Von Halle, Barbara: *Handbook of Relational Database Design*. Reading Mass., Menlo Park Calif. etc. 1989.

Der grösste Vorteil dieser Systeme ist die einfache Abfrage von Daten. Dies gelingt dank der standardisierten Abfragesprache SQL.⁵ Das Konzept ist leicht verständlich, wie das folgende Beispiel beweist:

```
SELECT Vorname, Nachname FROM Tabelle_aller_Personen WHERE  
Vorname="Barzabal"
```

Dieser Befehl fragt nach den Vor- und Nachnamen aller Personen des Datenbestandes, welche mit Vornamen Barzabal heissen. Es ist klar, dass das etwas schneller geht, als wenn man eigenhändig einen Zettelkasten nach einem Vornamen durchforsten muss. Wirklich erheblich wird die Zeitersparnis mit Anfragen folgender Art: «Ich möchte aus dem Quellmaterial alle Urkunden des 14. Jahrhunderts, welche das Kloster Ötenbach in Zürich betreffen und in welchen ein Laienbruder mit Namen genannt wird.» Dies ist mit einem relationalen System und einer überlegten Architektur leicht machbar. Ein wichtiger Vorteil eines solchen Systems liegt demzufolge im leichten Zugang zu den Daten: Der Computer enthebt einem von der mühseligen und fehleranfälligen Sucharbeit.

Prosaquellen haben im Computer einen schweren Stand. Es ist möglich, die Quellen gesamthaft zu erfassen, also abzutippen, aber damit ist für den Zugang noch nicht viel gewonnen. Der Computer erkennt in einer Urkunde von sich aus nicht, was beispielsweise ein Name ist und was ein Verb, was ein Datum und was eine Zinszahlung. Um damit arbeiten zu können, muss man es ihm erst von Hand bekanntgeben, und das verlangt einiges an Arbeitsaufwand. Immerhin stellt es schon einen Fortschritt dar, längere Quellentexte überhaupt in eine relationale Datenbank eingeben zu können, denn bis Mitte der neunziger Jahre gab es noch kaum ein relationales System, das bei Texten über zwei, drei Zeilen hinauskam. Diese Hürde ist inzwischen überwunden, so dass man heute leicht Texte erfassen kann. Die Unterstützung wurde so weit verbessert, dass die Benutzung von relationalen Datenbanken für Anfänger und für kleinere Projekte sogar ausdrücklich empfohlen werden kann. Diese Aussage stützt sich auch auf eine Umfrage, welche ich im Internet in einer Mailingliste unternommen habe.⁶ Eine

5 SQL bedeutet Structured Query Language. Das Konzept der Sprache ist eng mit dem relationalen Ansatz verwoben und wurde überhaupt erst durch relationale Datenbanken und ihre konsequente mathematische Basis denkbar. Vgl. Codd, Relational Model.

6 Die Umfrage lief auf der Mailingliste der International Association for History and Computing (H-AHC). Nähere Informationen zu dieser Liste finden sich im Internet bei: <http://www2.h-net.msu.edu/~ahc/> (Standort der Homepage im Mai 2000). Im Zentrum standen die Fragen: «Was ist heute <State of the Art>? Wie soll man heute als Historiker Datenbanken anlegen und welche Systeme werden in der Forschung überhaupt eingesetzt?» Das Echo war überraschend gut, und es

eindeutige Antwort auf die Frage nach der Wahl des Systems kann natürlich nicht gegeben werden, aber folgende Thesen stellen die Quintessenz der Antworten dar. Zunächst Aussagen zu relationalen Datenbanksystemen:

- Relationale Datenbanken eignen sich besonders für Quellenbestände in Listenform.
- Relationale Datenbanken eignen sich besser für Einsteiger und Benutzer mit wenig Vorkenntnissen.
- Die exakte Transkription von Quellen in eine relationale Datenbank ist nach wie vor nur mit gewissen Vorbehalten möglich.
- Relationale Datenbanken sind von Vorteil, wenn man mit einem eng vernetzten Bestand arbeitet.
- Relationale Datenbanken bieten dank der Abfragesprache SQL einen leichten Zugang zu den Daten.
- Die fortgeschrittene Integration relationaler Datenbanken in die grafische Benutzeroberfläche machen ein angenehmes Arbeiten mit der Maus möglich.
- Diese Integration lässt einen ohne grossen Zeitaufwand mit der Anwendung vertraut werden.

Demgegenüber bieten Alternativen wie zum Beispiel Volltextsysteme andere Vorteile:

- Volltextsysteme vermögen leichter mit variablen Strukturen umzugehen.
- Die Unterstützung für Benutzer von Volltextsystemen ist weniger gut gewährleistet. Man sollte sich selbst zu helfen wissen.
- Die Transkription von Quellen im Hinblick auf eine mögliche Edition ist in Volltextsystemen leichter möglich.
- Diese Systeme bieten weniger Hilfsmittel, wenn ein Bestand eng verzahnt ist.
- Die Suche nach Daten in Volltextsystemen ist zum Teil komplexer, andernteils eröffnen sich einem die sehr mächtigen Hilfsmittel der Volltextsuche, welche relationalen Datenbanken weitgehend fehlen.
- Die rasche Navigation im Bestand ist schwieriger, da die meisten Systeme nicht so sehr in die grafische Benutzeroberfläche integriert sind wie relationale Systeme.
- Befehle oder Sequenzen müssen oft von Hand eingegeben werden.

hat auch meine Meinung bestätigt: Innerhalb weniger Tage liefen Antworten von gegen zwanzig Wissenschaftlern ein. Geografischer Schwerpunkt war dabei Westeuropa, wobei mehrere Beiträge aus den Vereinigten Staaten und auch einer aus Südostasien einging. Eine grundsätzlich andere Sichtweise der Antworten nach Herkunftsland konnte ich nicht ausmachen.

Was bedeutet das konkret? Wenn man für sich sagt, ich will keine Urkunden abtippen, mir reicht es für meine Datenbank, zu einem Kloster die wichtigsten Daten einer Stiftung am Bildschirm präsent zu haben, dann ist ein relationales System die richtige Wahl, denn diese Daten passen gut in eine fixe Tabellenstruktur. Wenn ein Projekt davon ausgeht, dass die Quellen einmal ediert werden sollen, dann spricht hingegen viel für ein Volltextsystem. Ebenso wenn man vermeiden möchte, die Quelle in Unterabschnitte zu zerschneiden (um die Schnipsel dann in einer Tabelle abzulegen), oder wenn ein Bestand vorliegt, der eher aus Einzelstücken denn aus eng verzahnten Quellen besteht. In all diesen Fällen kann eine relationale Datenbank nur bedingt empfohlen werden. Schwerer können aber je nach konkreter Situation andere Punkte wiegen: Der rasche Einstieg in das System und die bessere Unterstützung durch versierte Informatiker im Bekanntenkreis oder in benachbarten Projekten.

Aber auch ein weiterer Aspekt hat sich in der Umfrage herauskristallisiert: Langjährige Erfahrung hat viele Forschende dazu gebracht, dass sie lieber mit Volltextsystemen arbeiten, da sie damit ihre Quellen besser im Griff haben, denn es macht einen Unterschied, ob man die Daten einer komplexen Software zur Verwaltung übergibt, oder ob man sie in Texten abspeichert, welche auch ohne diese Software oder mit anderen, zum Teil selbstprogrammierten Hilfsmitteln, weiterhin lesbar sind, was bei einer relationalen Datenbank ausgeschlossen ist. Da dies aber alles eher speziellere Anwendungsfälle sind, Quellen meist nicht transkribiert werden sollen, der verzahnte Bestand die Regel darstellt und den meisten Forschungsvorhaben eine strikte Gliederung weiterhilft, darf man behaupten, dass die Mehrzahl der informatikgestützten Untersuchungen dieser Tage auf relationalen Systemen ablaufen.

Zwei Forscher haben ferner berichtet, dass sie zum Teil auf sehr alten Systemen weiterarbeiten und sogar weiterarbeiten müssen, weil die Software es nicht zulässt, dass die Daten in ein anderes System übernommen werden können. Darauf wird später zurückzukommen sein.

Die Umfrage hat ergeben, dass man als Einsteiger eine gute Wahl trifft, wenn man mit einem strikten Tabellenansatz, also einem relationalen Ansatz, arbeitet. Nicht vergessen werden sollte dabei Folgendes: Nicht jede Lizentiandin und jeder Magister bleibt der Forschung treu, und da relationale Systeme die Wirtschaft und öffentliche Verwaltung monopolartig beherrschen, ist es von Vorteil, sich hier Fertigkeiten anzueignen, die als Schlüsselkompetenzen immer wieder gefragt sein werden, selbst dann, wenn man der Forschung nach Abschluss der Arbeit den Rücken kehrt.

Zusammenfassend kann man also sagen, dass die Entscheidung für ein System auf mehreren Ebenen abläuft, die von Fall zu Fall anders gewichtet werden. Wichtig sind erwartungsgemäss Forschungsmethode und Forschungsziel, aber auch Vorwissen, Unterstützung und die Einarbeitungszeit spielen meist eine Rolle.

Auch wir haben uns für ein relationales System entschieden. Wir waren uns bewusst, dass wir Mühe haben würden, unsere Quellen – chronikähnliche Texte im Prosastil, gepaart mit einer grossen Zahl von Urkunden – in die Datenbank eingeben zu können. Entscheidend waren für uns aber andere Faktoren: Das Projekt ist durch diese Wahl weniger einem einzelnen Programmierer ausgeliefert; ein anderer Datenbankbetreuer könnte innert nützlicher Frist gefunden werden. Dies kann bei einem spezialisierten Ansatz ein schwerwiegendes Handicap sein. Das sogenannte «Look & Feel» der Anwendung spielte eine weitere Rolle, denn mit dem Entscheid, die Standardsoftware Access einzusetzen, bleibt die Handhabung ähnlich zur gewohnten Textverarbeitung und anderer vertrauter Software. Des Weiteren gab es für die Entscheidung einen zusätzlichen Faktor: Wir wollten Personennetze in der Datenbank abbilden und in diesen Strukturen schnell navigieren können. Wir sind überzeugt, dass uns ein anderer Ansatz dabei weniger gut zur Hand ginge.

Nach der Entscheidung für ein System wartet die konkrete Umsetzung. Spätestens in diesem Moment gibt es ohne Handbücher und ohne konkrete Hilfe kein Weiterkommen. Wenn man sich bei den Informatikdiensten der Schweizer Universitäten umhört, muss man konstatieren, dass es um den Support, also die Unterstützung der Computerbenutzer, im Bereich der Datenbanken nicht gut steht.⁷ Die Software ist zwar in den meisten Fällen vorhanden, allerdings stehen nur Lizenzen für das Universitätspersonal und nur zur Installation auf universitätseigenen Geräten zur Verfügung. Der Support ist an fast allen Orten ziemlich mager und beschränkt. Falls Kurse für Universitätspersonal angeboten werden, dann sind sie kaum auf die speziellen Bedürfnisse von Geisteswissenschaftlern ausgerichtet. Offen für Studenten sind oft nur die Veranstaltungen der Wirtschaftsinformatik, welche sich sehr häufig an Anfänger im Datenbankbereich wenden. Hier ist allerdings Vorsicht geboten, denn was für die Kurse für Universitätspersonal gilt, trifft hier ebenfalls zu: Die Veranstalter dieser Kurse sind kaum in der Lage, Forschende zu beraten, denen es um unscharfe Daten geht, um in

7 Ich habe die Homepages der verschiedenen Schweizer Universitäten nach einschlägigen Angeboten abgesucht und zur Absicherung via E-Mail bei den Verantwortlichen nachgehakt, um den Eindruck zu überprüfen, den ich auch schon während meines Studiums gewonnen hatte.

wichtigen Punkten unvollständige Quellen, um komplizierte Transkription von Handschriften und all das, was für die Historikerin Alltag und für den Informatiker Alptraum ist.

Computerkurse für Geisteswissenschaftler oder speziell für Historiker stellen eine fortschrittliche Ausnahme dar. Solche Veranstaltungen werden beispielsweise an der Universität Bern seit Jahren regelmässig und mit gutem Echo durchgeführt. Zwar sind diese Kurse auf ein breiteres Gebiet als nur Datenbanken ausgerichtet; trotzdem stellen sie einen guten Einstieg in die Materie dar.

Genauso wie mit den fachunspezifischen Datenbankeinführungen an Universitäten ist es mit der bunten und zahlreichen Handbuchliteratur, die von wohlmeinenden Buchhändlern an ahnungslose Historiker verkauft werden. Zwar finden sich in der unüberblickbaren Menge gängiger Computerliteratur auch einige empfehlenswerte Werke, allerdings bekunden Einsteiger Mühe, die angebotenen Lösungsvorschläge auf historische Problemstellungen zu übertragen. Man ist deshalb besser beraten, sich an die Literatur der historischen Fachinformatik zu halten. Diese Literatur geht auf spezifische Probleme historischer Untersuchungen ein und arbeitet mit aktuellen Forschungsbeispielen, welche oftmals eine willkommene Anregung bieten. Für den Einstieg besonders geeignet scheinen mir zur Zeit die folgenden beiden Werke:

- Harvey, Charles; Press, Jon: *Databases in Historical Research. Theory, Methods and Applications*. Basingstoke, Hampshire etc. 1996.
- Lewis, Myrddin John; Lloyd-Jones, Roger: *Using Computers in History. A Practical Guide*. London, New York 1996.

Harvey und Press führen in ihrem Buch Schritt für Schritt in die Konzeption und Anwendung von historischen Datenbanken ein. Neben vielen Projektpräsentationen widmen sie SQL und den Alternativen zu relationalen Datenbanken jeweils eigene Kapitel. Die Einführung von Lewis und Lloyd-Jones beschränkt sich nicht nur auf Datenbanken. Trotzdem räumen die Autoren diesem Thema den breitesten Raum ein und gehen in ihrer Anleitung, basierend auf Access, Schritt für Schritt vor und bieten so eine leicht verständliche Einführung.

Design und Umsetzung

Nachdem man sich mit der Literatur und dem gewählten System beschäftigt hat, folgt der Schritt der Umsetzung. Eigentlich müsste ich hierbei von Programmierung sprechen, aber da dieses Wort eher abschreckt, sei es Design und Umsetzung genannt. Natürlich können hier keine fertigen

Lösungen vorgestellt werden, aber es soll versucht werden, eine Idee zu vermitteln, wie dies konkret aussehen könnte und wie man diese Umsetzung der Bedürfnisse des Projektes anpacken sollte.

Das Beispielprojekt arbeitet wie bereits erwähnt mit einem grossen Bestand an Urkunden und Prosatexten. In der Datenbank des Projektes werden die Urkunden und übrigen Quellen des Bestandes fast vollständig eingegeben. Eine Verknüpfung verbindet sie mit den Personendaten und zusätzlich wird jede Verknüpfung durch eine Eigenschaft charakterisiert.⁸

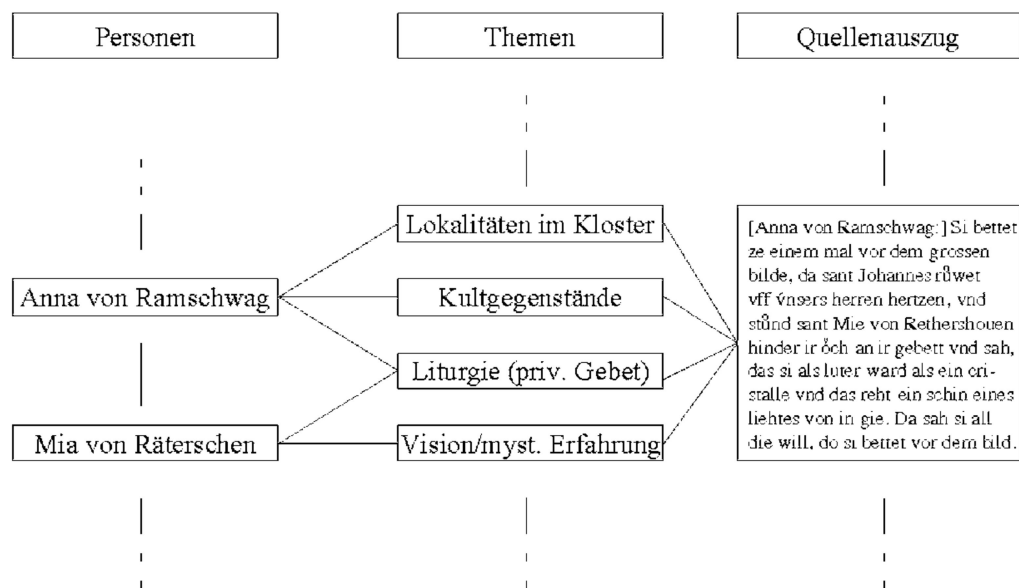


Abbildung 1: Schema der Datenablage

Dies bedeutet, dass im Kern drei Tabellen bestehen: eine für die Quellenauszüge, eine für die Personen und eine für das Thema, mit der die Verknüpfung charakterisiert wird. Beispiele für diese thematischen Zuweisungen sind: Gebet, Vision, Kultgegenstände, aber auch Stiftung, Kauf, Bürgerschaft, Schiedsspruch etc. Wir verknüpfen somit in einem konkreten Beispiel (Abbildung 1) die mutmassliche Visionärin Mia von Räterschens mit einem ihre Visionen betreffenden Quellenabschnitt und bezeichnen diese Verknüpfung mit den Themen «Vision» und «Gebet», denn zum Zeitpunkt der Vision betete sie ja. Dadurch wird die Konventualin aus St. Kathariental zur Visionärin und der Quellenabschnitt zum Beleg für eine Vision.

⁸ Die Verknüpfung stellt in der Datenbanksprache zwei m:n-Beziehungen zwischen den Tabellen «Personen», «Verknüpfungseigenschaft» und «Quellenauszüge» dar.

Die Verknüpfung der Tabellen geschieht über so genannte Schlüsselfelder, was in der angeführten Literatur genau beschrieben wird.⁹

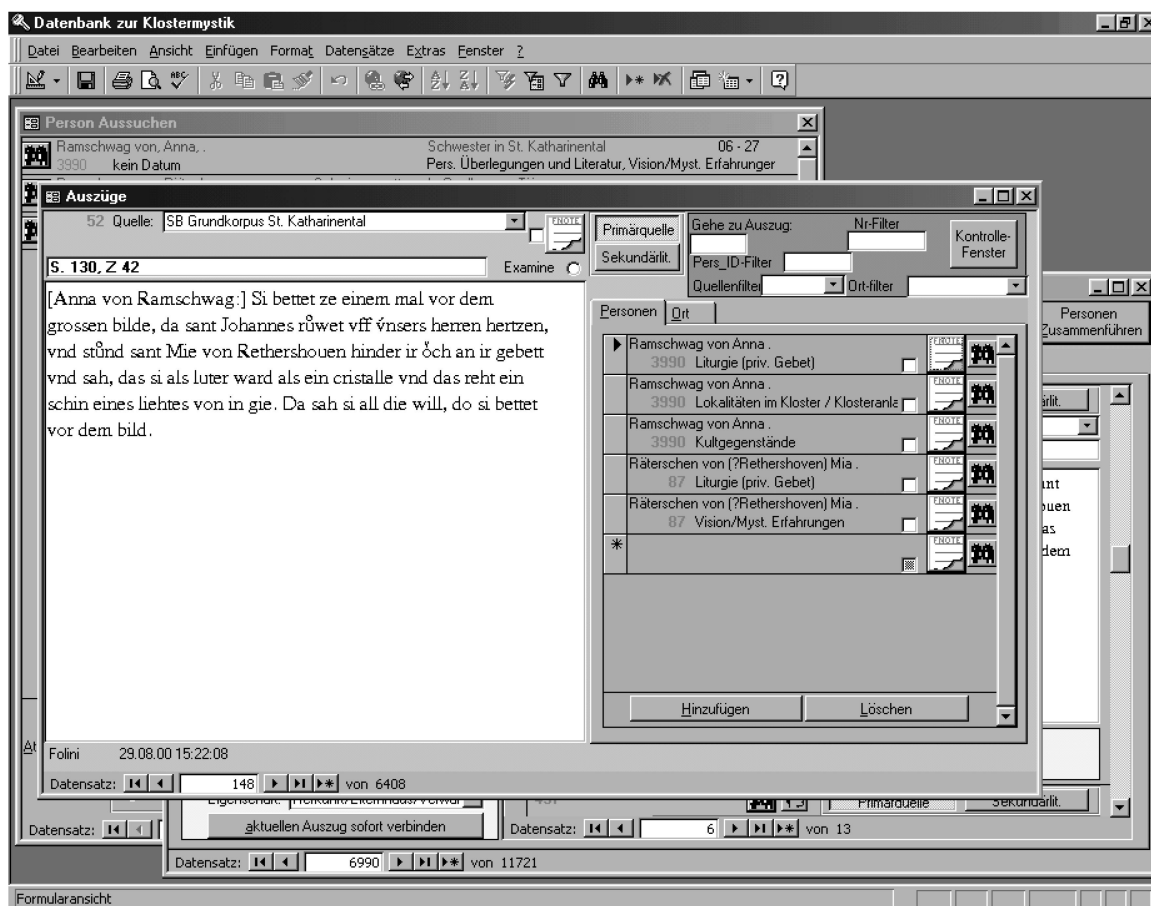


Abbildung 2: Illustration der Beziehung Quelle-Person in der Datenbank

Mit diesem Konzept startete das Projekt im Frühjahr 1997. Heute enthält die Datenbank gegen 2'000 Urkunden, einen Katalog von um die 40 Eigenschaften und über 10'000 Personen. Die drei Tabellen erlauben im Kern drei verschiedene Zugänge zu den Quellen: Man kann beispielsweise von Urkunde zu Urkunde gehen und überprüfen, wer darin vorkommt und was für ein Rechtsgeschäft behandelt wird. Ein anderer Ansatz ist es, alle Quellen und Personen ausgeben zu lassen, welche mit einer bestimmten Eigenschaft bezeichnet werden. So zum Beispiel alle Jahrzeitstiftungen eines bestimmten Klosters, oder sämtliche Hinweise auf die Ernährung. Als dritte Variante schliesslich lässt sich eine einzelne Person genauer untersuchen, und man wird innert kürzester Zeit feststellen, dass sie bei Stiftungen aus einer bestimmten Region jeweils als Zeuge fungiert. Für diese drei

⁹ Zum Beispiel Harvey; Press, Databases, S. 102 - 139.

Herangehensweisen wären drei herkömmliche Karteikästen nötig, welche mühsam parallel à jour gehalten werden müssten. Die eigentliche Stärke des Systems liegt deshalb darin, dass sehr schnell von einer Person zur nächsten und von Urkunde zu Urkunde gesprungen werden kann. Es besteht eine bewusste Analogie zum Internet-Surfen. Ferner bietet der Computer die Möglichkeit, sämtliche Informationen an einem Ort in einem Medium zu vereinigen. Dies ist eine grosse Arbeitserleichterung, welche es zudem erlaubt, diesen Datenbestand komplett einem grösseren Personenkreis zur Verfügung zu stellen.

Es ist möglich, die Arbeit mittels der Datenbank besser zu strukturieren. Verschiedene Arbeitsschritte können recht einfach aufgeteilt werden und somit lässt sich eine Trennung von Erfassung und Interpretation der Quellen leicht durchsetzen, was sonst in der Praxis gerne verschimmt. Abbildung 2 ist ein Schnappschuss der Datenbank mit dem bereits bekannten Quellenauszug. Auf der linken Seite ein Quellenausschnitt, rechts die mit dieser Quelle verbundenen Personen und die Themen, welche die Verbindung kennzeichnen. Die Zuweisung eines Themas zu einer Verknüpfung ist eine Interpretation der Quelle, denn damit qualifiziert man ja die Quelle als Stiftung, Lehen, Tausch oder Verkauf; eine Unterscheidung, welche für unseren Quellenbestand bisweilen knifflig und nicht immer eindeutig ist. Auf dem Bildschirm wird durch die räumliche Trennung von Quellen und Personen auch die Eingabe von der Interpretation räumlich getrennt. Die Interpretation lässt sich einfach auf einen späteren Zeitpunkt verschieben, womit beispielsweise erreicht werden kann, dass sämtliche Quellen mit demselben Kriterienkatalog interpretiert werden. Dieser Katalog verändert sich gerne im Lauf der Arbeit, wenn man ihn nicht explizit schriftlich fasst und, so wie in unserem Projekt, fest in der Datenbank einschreibt; allein schon die Tatsache, dass der Computer bei der Strukturierung helfen soll, zwingt einen, sich diesen Katalog gründlich zu überlegen. Der Einsatz des Computers besitzt also in diesem Punkt eine starke Methodenrelevanz. Dabei darf nicht vergessen werden, dass dieser Schritt, also die Kodierung der Quellen, eine hohe Anfälligkeit für die Konstruktion von Sachverhalten besitzt. Wer hier nicht aufpasst, wird alsbald eine unsaubere Kodierung für den Quellenbefund erhalten und ohne Überprüfung übernehmen.

Bei Anfragen an die Datenbank, der Auswertung der Daten also, zeigt sich, was die gewählte Architektur zu leisten vermag. Dies ist allerdings auch der Punkt, wo es am meisten Erfahrung im Umgang mit Datenbanken braucht, um ihre Informationsvielfalt überhaupt ausnutzen zu können. Dabei kann zwischen vorgefertigten und Ad-hoc-Abfragen unterschieden werden. Erstere werden nach der Erstellung abgespeichert und sind fortan

immer wieder verfügbar. Ein Beispiel für eine häufiger gebrauchte Abfrage ist die Liste aller Personen in der Datenbank. Oder diejenigen Quellen, in welchen Aussagen zur «Ernährung» gemacht werden. Diese Anfragen an die Datenbank sind vergleichsweise simpel und mit der oben vorgestellten Literatur gut zu bewältigen. Komplizierter wird es in dem Moment, in dem aus der Forschung heraus Fragen auftauchen, mit denen zu Beginn nicht gerechnet worden war. Zum Beispiel eine Liste aller Quellen, in welchen Adelige in den 1320er Jahren an einer Stiftung beteiligt waren. Oder die mutmassliche soziale Herkunftsschicht der Klosterfrauen eines bestimmten Zeitabschnittes nach Herkunftsort gegliedert. Hierfür braucht es mehr Erfahrung und vor allem den Willen, immer wieder Neues auszuprobieren und seine Kenntnisse zu erweitern. Das wird bei den ersten Gehversuchen sehr zeitintensiv sein. Bis man etwas Erfahrung gesammelt hat, wird es auch eine langwierigere Arbeit sein, als aus einem Zettelkasten die betreffenden Urkunden herauszusuchen. Von Beginn weg wird die Anfrage an die sauber erstellte Datenbank aber weniger fehleranfällig sein.

Schwierigkeiten und Probleme

Wo liegen konkret die Schwierigkeiten der Arbeit mit Datenbanken in Projekten dieser Grösse? Es bedeutet eine grosse Unsicherheit, dass der Arbeitsaufwand von Einsteigern nur sehr schwer abgeschätzt werden kann.¹⁰ Ebenso fehlen meist realistische Vorstellungen bezüglich der Umsetzbarkeit einzelner Wünsche. Abhilfe bringt es allein, wenn man sich gründlich einliest und sich dadurch ein besseres Bild verschafft, was innert nützlicher Frist durchführbar ist. Danach sollte eine Prioritätenliste erstellt werden, anhand der die Ideen realisiert werden. Aufgabe dieser Liste ist es, möglichst rasch ein lauffähiges Basissystem, einen Prototypen, herzustellen.¹¹ Damit kann man erste Schritte wagen, entdeckt grundlegende

10 Dies ist allerdings ein gängiges Problem der meisten Informatikprojekte, wie ein Blick in die Literatur zeigt. Vgl. dazu: Greenstein, Daniel I.: *A Historian's Guide to Computing*. Oxford 1994, S. 2; Jarausch, Konrad H.; Armingier, Gerhard; Thaller, Manfred: *Quantitative Methoden in der Geschichtswissenschaft. Eine Einführung in die Forschung, Datenverarbeitung und Statistik*. Darmstadt 1985, S. 3f.

11 Diese Vorgehensweise steht einer sauberen Planung von Softwareprojekten, wie sie die Informatikliteratur fordert, entgegen. Sie ist für grössere Projekte als die hier bedachten auch nicht geeignet. Hingegen bietet sie in dem Moment Vorteile, in dem keiner der Beteiligten über vertiefte Programmierkenntnisse verfügt und die von der Informatik geforderte saubere Gliederung in zeitlich getrennte Stufen der Konzeption, Realisierung und Testung nicht durchführbar ist. (Zwar ist das sogenannte Evolutionäre oder Rapid Prototyping inzwischen auch in der gängigen Software-

Mängel rasch und zudem fördert es die Motivation, wenn man schnell erste Resultate vorweisen kann. Man darf damit rechnen, dass man innerhalb von ein, zwei Tagen erste Daten probeweise eingeben kann.

Zweifelsohne stellt es eine Schwierigkeit dar, dass der grösste Zeitaufwand beim Einstieg in die Welt der Datenbanken meist mit dem Projektstart zusammenfällt und damit in einer Phase liegt, in der bereits genügend andere Probleme anstehen. Demgegenüber trägt die in diesem Moment getätigte Zeitinvestition die grössten Früchte. Eine wohlüberlegte Architektur erspart bei der Auswertung der Daten Stunden und Tage der Arbeit. Etwas entschärft werden kann dieser Aufwand, wenn man möglichst früh, also schon während dem Studium, erste Schritte in Richtung Datenbanken wagt. Zwar ist die Zeitersparnis für die Datenmengen einer Seminararbeit marginal, aber man wird ein grösseres Projekt mit der gewonnenen Erfahrung bedeutend ruhiger in Angriff nehmen können.

Es ist wichtig, sich anfangs die nötige Zeit zu nehmen. Denn während man einen Text nochmals umschreiben kann, ist man der Architektur der eigenen Datenbank vollumfänglich ausgeliefert; dazu kommt, dass Veränderungen am Aufbau der Tabellen nur noch mit grossem Zeitaufwand realisiert werden können und diese Eingriffe gewöhnlich Folgeprobleme mit sich bringen.

Die fachmännische Unterstützung ist wie bereits erwähnt nur schwer verfügbar. Sie beschränkt sich an den Universitäten häufig auf Basisdienste und man tut gut daran, sich nach Alternativen umzusehen. Der professionelle Support dürfte kaum im Bereich des finanziell Möglichen liegen, so dass man entweder auf Bekannte zurückgreifen oder sich durch die Literatur arbeiten muss und allein noch darauf hoffen darf, im Internet auf Unterstützung zu treffen.

Ein weiterer Negativpunkt ist die Geschäftspolitik von Anbietern kommerzieller Datenbanksysteme. Allen voran die Hersteller von Access und Filemaker, auf welche die Mehrzahl der Forschenden setzen. Je länger ein Projekt läuft, desto heftiger wird man von den Entwicklungsschüben der Software erfasst. Während eine Lizentiats- oder Magisterarbeit mit grob gerechnet einem Jahr auskommt, dauert eine Dissertation bis zu fünf Jahre und grössere Forschungsvorhaben noch bedeutend länger. Es ist ausgesprochen mühselig, wenn man alle zwei Jahre angehalten wird, seine Software auszuwechseln. Man muss auf jeden Fall befürchten, dass die Datenbank auf der neuen Softwarebasis nicht fehlerfrei weiterläuft. Daher sollte für

entwicklung verbreitet. Allerdings wird dabei in der Regel noch nicht mit Produktionsdaten gearbeitet.)

den Transfer ein hoher Anpassungsaufwand eingerechnet werden. Natürlich kann man auch weiterhin mit der alten Software weiterarbeiten, aber man muss in diese Entscheidung einbeziehen, dass die anderen Programme eines Tages die Zusammenarbeit mit der Datenbank verweigern und es zusehends schwieriger werden wird, fachmännische Hilfe für das veraltete Produkt zu erhalten.

Hinzu kommt, dass die Software, wie man sie kauft oder vom Informatikdienst der Universität installieren lässt, als Basissystem daherkommt. Bei ihr stösst man schnell an Grenzen des Systems. Grenzen, welche durch Erweiterungen der Software an sich einfach zu überwinden wären. Allerdings kommen diese Zusatzpakete schnell teuer zu stehen, denn Studentenrabatte sind dafür kaum mehr zu haben und Support wird allein von privaten Anbietern offeriert. Rückblickend betrachtet erscheint es fragwürdig, sich auf einen kommerziellen Anbieter zu verlassen. Vielmehr sollte man die Option prüfen, ob sich der Aufwand nicht lohnt, sich auf eines der vielen frei verfügbaren Systeme einzulassen, wie sie für Unixbetriebsysteme entwickelt wurden, inzwischen aber mehr und mehr auch auf Windows-Rechnern zugänglich sind. Diese Datenbanksysteme bieten den Vorteil, dass sie oft von unabhängigen Programmierern betreut werden, denen mehr an den Stabilität und den Interessen der Nutzer denn an Gewinnmaximierung gelegen ist. Dies bedeutet zwar einen zusätzlichen Aufwand, der für eine kleinere Arbeit sicher übertrieben wäre, aber ein Projekt mit einer geplanten Dauer von mehreren Jahren sollte sich diese Zeitinvestition durchaus überlegen, denn sie erspart einem im Nachhinein viel Mühsal. Ein Problem älterer Software kann beispielweise sein, dass die Daten nur noch sehr schlecht wieder aus der Datenbank herausgeholt werden können. Zwar bieten Access und Filemaker vielfältige Exportfunktionen, aber andere, ältere kommerzielle Produkte lassen es nicht zu, die Daten in eine andere Software zu transferieren, so dass Forscher, welche sich für diese Systeme entschieden haben, auch dann noch auf dieser Basis weiterarbeiten müssen, wenn sie längst überholt ist. Sich zu überlegen, ob die Daten aus dem gewählten System auch wieder komplett herausgeholt werden können, um sie mit anderer Software weiter zu bearbeiten, kann also gar nicht früh genug geschehen. Im konkreten Beispielfall unseres Projektes könnten wir zwar leicht die Daten in eine andere Datenbank übernehmen. Hingegen wären die erstellten Verarbeitungsmasken (ein Beispiel ist in Abbildung 2 zu sehen) verloren, denn diese sind fix an Access gebunden. Ein Wechsel der Datenbank wäre also nur mit grossem Arbeitsaufwand realisierbar.

Desiderata

Bei der kleinen E-Mail-Umfrage war ein Resultat, dass es durchaus eine Fraktion von Forschenden gibt, welche mehrere Ansätze nebeneinander favorisiert; nämlich den Einsatz von relationalen Datenbanken, wie er in diesem Artikel im Zentrum steht, und die gleichzeitige Arbeit mit verschiedenen Textsystemen. In diese Richtung weisen denn auch die Entwicklungslinien der Software. Es ist wünschenswert, dass Textsysteme und relationale Datenbanken enger zusammenrücken. Dies scheint in greifbare Nähe zu rücken, denn auf der Seite der Textsysteme ist XML auf dem Vormarsch.¹² Es würde zu weit führen, hier genauer zu erklären, welche Möglichkeiten sich damit eröffnen. Gesagt sei lediglich soviel, dass es mit diesem Standard, der auf den elektronischen Austausch von Daten im Internet zugeschnitten ist, möglich wird, Information leicht aus dem eigenen System zu exportieren und fremde Daten bedeutend einfacher als bislang möglich zu integrieren. Ferner eröffnet XML eine einfachere Möglichkeit einer Kombination von Volltextsystemen und der Abfragesprache SQL, denn auf der einen Seite bleiben die Quellen in ihrer eigenen Struktur erhalten und auf der anderen Seite sind die Daten doch sehr genau inhaltlich gekennzeichnet (marked-up), was die Voraussetzung für strukturierte Abfragen bietet.

Fazit

Auch kleine Forschungsprojekte können heute Datenbanken einsetzen. Dazu sind keine besonderen Vorkenntnisse nötig. Die Datenbanken dienen als Werkzeug und erleichtern die Arbeit in vielen Bereichen. Ihre Hauptstärke liegt im raschen Zugang zu den Quellen und in der Verwaltung von grossen Beständen. Nach einer klaren Projektierung, während der die Anforderungen an eine Datenbank möglichst klar formuliert werden sollten, sollte ein Prototyp erstellt werden. Für Historiker verfasste Informatik-Literatur leistet dabei Hilfe. Der Prototyp wird dann parallel zu der mit ihm ausgeführten Arbeit verbessert und weiter ausgebaut. Das Haupthindernis für Benutzer und Ersteller von Datenbanken bleibt die geringe universitäre Unterstützung für Geisteswissenschaftler, deren spezifischen Problemen kaum Aufmerksamkeit zukommt.

12 XML steht für «eXtended Markup Language». Die Seite <http://www.xml.org> verhilft zu einem leichten Einstieg. Der vergleichsweise neue Standard wird zur Zeit heftig diskutiert. Man ist sich unter Informatikern in den Geisteswissenschaften einig, dass das Konzept grosses Potential besitzt. Hingegen sind noch kaum konkrete Anwendungsbeispiele in den Geisteswissenschaften verfügbar.