RelaX

Interaktive Relationale Algebra in der Lehre

Günther Specht¹ · Johannes Kessler¹ · Maximilian Mayerl¹ · Michael Tschuggnall¹

Eingegangen: 15. Oktober 2020 / Angenommen: 5. Januar 2021 / Online publiziert: 24. Januar 2021
© Der/die Autor(en) 2021

Zusammenfassung

Das relationale Modell und insbesondere die relationale Algebra bilden die Grundlage jedes relationalen Datenbanksystems. Daher ist es in der Lehre wichtig, den Studierenden eine fundierte Einführung in die relationale Algebra zu geben. Nur so erhalten sie ein vertieftes Verständnis für die interne Ausführung einer Anfrage. Während es viele Möglichkeiten gibt, SQL zu üben, fehlen bisher größtenteils solche Möglichkeiten für die relationale Algebra. Sie wird meist nur theoretisch unterrichtet. Darum hat die Forschungsgruppe DBIS an der Universität Innsbruck ein rein webbasiertes Tool entwickelt, das die Lehre der relationalen Algebra erleichtern und verbessern soll: RelaX. RelaX ist unter <http://dbis-uibk.github.io/relax/> frei verfügbar.

Schlüsselwörter Relational Algebra · Database Systems · Learning Tools

1 Einführung

Die relationale Algebra [1] bildet nach wie vor eines der Fundamente relationaler Datenbanksysteme. Jede SQL-Anfrage wird intern in eine erweiterte Form der relationalen Algebra übersetzt, optimiert und als Operatorgraph ausgeführt. Für das Verständnis der internen Ausführung einer Anfrage und insbesondere für das Verständnis des Operatorgraph-Optimierers ist es daher wesentlich, die relationale Algebra zu verstehen und zu beherrschen. Das macht es nötig, der relationalen Algebra einen wichtigen Platz in der Vorlesung und in den Übungen zu Datenbanksystemen einzuräumen. Meist wird sie sogar noch vor SQL, spätestens aber bei der Optimierung und beim Tuning unterrichtet. Dies geschieht traditionellerweise in der Vorlesung mit (animierten) Folien und in den Übungen mit Bleistift und Papier. Während für SQL zahlreiche Datenbanksysteme und Werkzeuge zur Verfügung stehen, mit denen Studierende ihre Anfragen interaktiv ausführen können, und die im Feh-

lerfall hilfreiche Fehlermeldungen ausgeben, gibt es für die relationale Algebra nur sehr wenige Projekte die eine meist eingeschränkte Ausführung relationaler Algebra Ausdrücke erlauben (z. B. IRA [5], Relational [6], radb [7]). IRA erlaubt nur Anfragen auf einem vordefinierten Datenset. Relational und radb benötigen Installationen, RelaX kann dagegen direkt im Browser ausgeführt werden. Keines der Tools unterstützt den GROUP BY Operator. Zudem werden die Relationalen Algebra Ausdrücke bei allen anderen Systemen nicht direkt interpretiert, sondern zurück nach SQL übersetzt und dann in einer SQL-Datenbank ausgeführt.

Aber auch das Studium der Operatorgraphen, wie es manche Datenbanksysteme über EXPLAIN zulassen, ist nur ein unzureichender Ersatz, da darin der relationale Algebra Ausdruck nicht manipuliert werden kann um nach anderen Lösungen zu suchen. Das Arbeiten mit Papier und Stift verlangsamt nicht nur das Lernen bei den Studierenden, es stellt auch eine unnötige Erschwernis bei der Korrektur durch die Lehrenden dar.

Aus diesem Grund hat die Forschungsgruppe Datenbanken und Informationssysteme (DBIS) an der Universität Innsbruck ein Werkzeug entwickelt, das die Lehre der relationalen Algebra sowohl für Studierende wie auch für Lehrende einfacher und besser machen soll: RelaX. RelaX ist frei, kann von jedem verwendet werden und wird quell-

✉ Günther Specht
guenther.specht@uibk.ac.at

¹ FG Datenbanken und Informationssysteme (DBIS), Institut für Informatik, Universität Innsbruck, Technikerstr. 21a, 6020 Innsbruck, Österreich

offen zur Verfügung gestellt¹. Es läuft gänzlich im Browser ab. Anfragen können in einem Editor in der gewohnten relationale Algebra-Syntax formuliert und anschließend auf frei wählbaren oder definierbaren Relationen (Datenset) interaktiv ausgeführt werden. Führt die Anfrage zu einem Fehler, gibt RelaX sinnvolle Fehlermeldungen aus. War die Ausführung der Anfrage erfolgreich, zeigt RelaX nicht nur ihr Ergebnis an, sondern stellt auch eine grafische Darstellung des Operatorbaumes zur Verfügung, anhand dessen die Ausführung der Anfrage Schritt für Schritt nachvollzogen werden kann.

RelaX wird erfolgreich in der Datenbankvorlesung und den Übungen an der Universität Innsbruck eingesetzt. Es ist frei zugänglich und wird inzwischen auch an mehreren deutschen und internationalen Hochschulen eingesetzt. Die Oberfläche und Beispiele sind in Deutsch, Englisch, Spanisch, Portugiesisch und sogar Koreanisch verfügbar.

Im Folgenden werden wir zuerst auf den genauen Funktionsumfang von RelaX eingehen und anschließend berichten, wie das Werkzeug in der Lehre an der Universität Innsbruck eingesetzt wird.

2 Funktionsumfang von RelaX

RelaX ist ein webbasiertes Werkzeug, das eine Umgebung zum Arbeiten und Üben mit der relationalen Algebra be-

reitstellt. Eine lauffähige Version kann direkt über unsere GitHub Page² aufgerufen werden und benötigt zur Verwendung ausschließlich einen Browser. Es ist während der Ausführung keine Serververbindung notwendig. Abb. 1 zeigt die Benutzeroberfläche von RelaX, wie sie sich darstellt, wenn man den Editor öffnet.

Auf der linken Seite wird dem Benutzer eine Liste der im Datenset vorhandenen Relationen und ihrer Attribute angezeigt. Rechts davon befindet sich der eigentliche Editor, in dem die Anfragen formuliert werden. Der Editor bietet Syntax-Highlighting an, wodurch die Lesbarkeit verbessert wird.

2.1 Anfragesprache

RelaX unterstützt alle gängigen Operatoren der relationalen Algebra. Dies beinhaltet:

- Selektionen (σ), Projektionen (π) und Umbenennungen (ρ)
- alle Varianten von Joins, inklusive Kreuzprodukte, Inner Joins, Outer Joins, Semijoins und Antijoins, wobei hier alle Varianten auch als Natural Joins unterstützt werden
- Vereinigung ($\cup$), Schnittmenge ($\cap$) und Subtraktion ($-$)
- Division ($\div$)
- Gruppierung (γ)

Abb. 1 Der Anfrageeditor von RelaX

¹ <https://github.com/dbis-uibk/relax>.

² <http://dbis-uibk.github.io/relax/>.

Anfragen können mit den in der relationalen Algebra üblichen Symbolen formuliert werden. Alternativ können auch Schlüsselwörter verwendet werden, die leicht mit einer herkömmlichen Tastatur eingegeben werden können. So sind zum Beispiel die folgenden beiden Anfragen äquivalent:

```

π Name (
σ Titel = 'Logik' (
  ρ p Professoren
  ⋈ p.PersNr = v.gelesenVon
  ρ v Vorlesungen
))

```

```

pi Name (
sigma Titel = 'Logik' (
  rho p Professoren
  inner join p.PersNr = v.gelesenVon
  rho v Vorlesungen
))

```

Hier werden die Namen aller Professoren ausgegeben, die Vorlesungen mit dem Titel „Logik“ unterrichten. Um auch die symbolische Variante leicht eingeben zu können, befindet sich über dem Anfrageeditor eine Symbolleiste, auf der alle verwendbaren Symbole vorhanden sind. Durch einen Klick auf ein Symbol wird dieses an die aktuelle Cursorposition im Anfrageeditor eingefügt.

In logischen Ausdrücken, wie sie in Selektionen und in Join-Bedingungen eingesetzt werden, können alle gängigen logischen Operatoren verwendet werden ($\wedge$, $\vee$, $=$, $\neq$, $<$, $>$, $\leq$, $\geq$, $\neg$). Der $=$ Operator kann auch für NULL-Vergleiche verwendet werden. RelaX bietet die Möglichkeit, Variablen zu deklarieren, in denen Zwischenergebnisse von Anfragen gespeichert werden können. Dies ermöglicht es, Anfragen schrittweise aufzubauen und komplexe Anfragen in leichter verständliche, kleinere Einheiten zu zerlegen. Zudem ist es in RelaX möglich, Kommentare sowohl in einer einzeiligen Variante (mittels $\neg$) oder einer mehrzeiligen Variante (mittels $/*$ und $*/$) anzugeben. Die folgende Anfrage zeigt die Verwendung von Variablen und Kommentaren:

```

/* Ermitteln, welche Studenten an welchen
Vorlesungen teilgenommen haben. */
VorlesungenStudenten =
  Studenten ⋈
  hoeren ⋈
  Vorlesungen

/* Finde die Professoren, die diese
Vorlesungen gehalten haben. */
Professoren

```

```

⋈ Professoren.PersNr =
  Vorlesungen.gelesenVon
VorlesungenStudenten

```

Wie man an dem Beispiel sieht, machen Variablen keinen eigenen Namensraum auf, sie arbeiten intern wie Makros. Überall, wo der Variablenname steht, wird der ursprüngliche Abfragetext expandiert. Bei vollqualifizierten Attributen kann auf den ursprünglichen Relationennamen immer noch zugegriffen werden. Bei binären Mengenoperationen übernimmt RelaX immer die Attributnamen der linken Relation.

2.2 Ausführung von Anfragen

Eine im Anfrageeditor formulierte Anfrage kann mittels Klick auf die Schaltfläche „execute query“ vollständig ausgeführt werden. Sofern gewünscht, kann auch nur ein Teil der Anfrage ausgeführt werden, beispielsweise zu Demonstrationszwecken während man eine komplexere Anfrage erklärt. Dies funktioniert, indem man den gewünschten Teil der Anfrage markiert und anschließend auf die Schaltfläche „execute selection“ klickt.

Nach der Ausführung einer Anfrage werden dem Benutzer unterhalb des Anfrageeditors zwei Dinge angezeigt: Der Operatorbaum der eingegebenen Anfrage und das Ergebnis als Tabelle.

Bei realen SQL-Datenbanken liegt der Fokus auf Performanz, was bedeutet, dass eine Anfrage aufgrund vielseitiger Optimierungen praktisch nie so ausgeführt wird, wie ein Anfänger sich dies erwarten würde. Für die Akzeptanz als Lernwerkzeug spielt die Nachvollziehbarkeit der Ergebnisberechnung die entscheidende Rolle. RelaX verzichtet daher bewusst auf einige der üblichen Performance-Optimierungen:

- Der Operatorbaum wird nicht umsortiert, um die Ausführung zu beschleunigen, sondern genau so ausgeführt, wie der Benutzer die Anfrage eingibt und „von Hand“ berechnen würde.
- Alle Operationen wurden reihenfolgeerhaltend implementiert. Das bedeutet, dass die Ordnung der Tupel des Ergebnisses einer Operation der Ordnung in den Ausgangsrelationen entspricht. Die Ordnung wird also bei einer Projektion und Selektion beibehalten. Bei Joins entspricht die Ordnung der eines Nested-Loop-Joins und auch bei einer Gruppierung mit Aggregation wird sichergestellt, dass die resultierenden Tupel an einer nachvollziehbaren Position stehen.

Die Erhaltung der Tupel-Reihenfolge und der bewusste Verzicht auf z.B. Join-Reordering-Algorithmen mindert zwar die Gesamtperformance, dies ist jedoch in der Lehrpraxis aufgrund der üblicherweise kleinen Datenmengen

vernachlässigbar. RelaX stellt zusätzlich zum Ergebnis auch die auszuführende Anfrage als Operatorbaum dar, wie in Abb. 2 dargestellt. Dabei können mit Klick auf die Knoten sämtliche Zwischenergebnisse der entsprechenden Teilbäume ausgegeben werden, um die Ausführung schrittweise nachvollziehen zu können. Außerdem werden für jeden Knoten Zusatzinformationen angezeigt, wie z. B. das Schema des Zwischenergebnisses, oder auch welche Spalten für die Joinbedingung eines Natural Joins tatsächlich verwendet wurden. Im Beispiel in Abb. 2 wird das Zwischenergebnis des Knotens der Variable S2 dargestellt. Unterhalb des Baums (in der Abbildung nicht sichtbar) wird zusätzlich das gesamte Zwischenergebnis des angeklickten Knotens angezeigt.

2.3 Syntax

Die verwendete Syntax der relationalen Algebra orientiert sich an der klassischen Notation möglichst ohne Änderung. Auch die übliche Definition und Verwendung von Variablen ist möglich. Anfragen werden als interpretierbarer Quelltext verstanden. Zur leichteren Eingabe wird auf das Tiefstellen der Argumente verzichtet. Auch auf Klammerung kann in RelaX durch die Definition einer Präzedenz meist verzichtet werden.

Damit ist die verwendete Syntax nahezu identisch mit den Standardlehrbüchern (z. B. Kemper/Eickler [3] oder Gracia-Molina et al [2]). Intern werden Relationen als geordnete Mengen von Tupeln umgesetzt und Attribute haben sowohl einen innerhalb des Schemas eindeutigen Namen als auch eine feste Position. Die Duplikatelimination wird automatisch vorgenommen und Null-Werte werden dabei analog zu SQL behandelt. Auch ein Sortierungsoperator τ , wie von Gracia-Molina et al. vorgeschlagen wurde implementiert.

2.4 Datensets

Ein wichtiger Punkt bei der Verwendung als Lern- und Übungstool sind die Daten, auf denen die Anfragen ausgewertet werden. Eigene Datensets – also die Tabellen/Relationen, die etwa für eine Übung verwendet werden sollen – können in einem einfachen Klartextformat definiert, als Github-Gists³ gespeichert und über deren eindeutige IDs eingebunden werden. Dadurch können sehr leicht neue Datensets erstellt werden.

Zudem werden einige Datensets, wie z. B. die in Kemper/Eickler [3] verwendete Vorlesungsdatenbank, die beiden Beispieldatenbanken aus Gracia-Molina [2], die populäre KursDB über Kurse, Dozenten und Teilnehmer, ein generisches R,S,T-Beispiel und einige mehr, standardmäßig

eingebunden und zur Verfügung gestellt. Für die Erstellung und Modifikation von Datensets steht im Tool ein entsprechender Editor zur Verfügung. Damit können Relationen direkt oder über einen Tabellen-Editor eingegeben werden.

2.5 SQL

Zusätzlich zur relationalen Algebra unterstützt RelaX die Eingabe von Anfragen in SQL. Diese werden dann in die relationale Algebra übersetzt und können als solche angezeigt und ausgeführt werden. Damit soll einerseits das Erlernen von SQL erleichtert werden, wenn die Grundkonzepte der relationalen Algebra bereits verstanden wurden, andererseits kann umgekehrt auch das Erlernen der relationalen Algebra erleichtert werden, wenn auf SQL-Vorwissen zurückgegriffen werden kann. Studierende erhalten so einen ersten Einblick, wie eine SQL-Anfrage im Prinzip in einem Datenbanksystem berechnet wird.

Unterstützt werden einfache SELECT-Anfragen, deren Übersetzung in relationale Algebra leicht nachvollziehbar und einfach zu verstehen ist. Aus diesem Grund werden zur Zeit z. B. keine korrelierten Unteranfragen und keine rekursiven Anfragen unterstützt.

3 Einsatz in der Lehre

RelaX läuft seit fünf Jahren erfolgreich bei uns in der Lehre. Wir verwenden es für Beispiele in der Vorlesung und stellen es den Studierenden das ganze Semester zur Verfügung. Insbesondere die Übungsaufgaben zur relationalen Algebra, Übersetzung von SQL in die relationale Algebra und zur algebraischen Optimierung sollten damit gelöst werden. Die Ergebnisse können direkt in unser Abgabetooll übernommen werden. Die Erfahrung zeigt deutlich bessere Ergebnisse als vor dem Einsatz. Dies bezieht sich sowohl auf einfache als auch auf komplexere Anfragen. Bei den Lehrevaluationen zur Vorlesung und zur Übung wurde der Einsatz von RelaX von den Studierenden sehr positiv beurteilt. In Summe haben wir in Zwischenprüfungen und in der Abschlussklausur bei den Aufgaben zur relationalen Algebra durchschnittlich eine Notenstufe bessere Ergebnisse. Dies lässt sich sicherlich auch auf das tiefere Verständnis durch den interaktiven Umgang mit dem Tool zurückführen. Darüber hinaus erleichtert sich die Korrektur der Übungsblätter nicht nur durch die gedruckte Lesbarkeit der relationalen Algebra Ausdrücke gegenüber den handschriftlichen Abgaben, sondern auch durch die nun möglichen, einfachen Ergebnismengenvergleiche. So kann man gegebenenfalls ganz dediziert auf den Denkfehler bei der studentischen Lösung eingehen.

Wir haben eine erste Version von RelaX als Poster auf der BTW 2019 vorgestellt [4] und daraufhin einige Kol-

³ <https://gist.github.com/>.

leginnen und Kollegen mit ihren Kursen als neue Nutzer gewonnen. Viele internationale Universitäten kamen durch Websuche auf uns und haben uns ihre Übersetzungen überlassen. So kommt es, dass wir RelaX inzwischen nicht nur in Deutsch und Englisch, sondern auch in Spanisch, Portugiesisch und sogar in Koreanisch anbieten können. Auf Anregung wurden auch heuer wieder weitere Datensätze standardmäßig eingebunden. Über weitere Nutzer freuen wir uns.

4 Zusammenfassung

Mit RelaX wurde ein webbasiertes Übungstool für relationale Algebra erstellt, das die Ausführung beliebiger Anfragen auf frei definierbaren Datensets ermöglicht. Konzipiert für den Einsatz in der Lehre soll es aufgrund vieler Funktionen und Hilfsmittel dazu beitragen, das relationale Modell und dessen Anfragesprache auf praktischer Basis verständlich zu machen. Nach einer aktiven Anwendung des Tools innerhalb mehrerer Semester in der Lehre am Institut für Informatik an der Universität Innsbruck zeigt sich in den Klausuren ein deutlich besseres Abschneiden der Studierenden in den Aufgaben zur relationalen Algebra.

Funding Open access funding provided by University of Innsbruck and Medical University of Innsbruck.

Open Access Dieser Artikel wird unter der Creative Commons Namensnennung 4.0 International Lizenz veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in

jedlichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Artikel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.

Weitere Details zur Lizenz entnehmen Sie bitte der Lizenzinformation auf <http://creativecommons.org/licenses/by/4.0/deed.de>.

Literatur

1. Codd EF (1970) A relational model of data for large shared data banks. *Commun ACM* 13(6):377–387
2. Garcia-Molina H, Ullman JD, Widom J (2008) Database systems - the complete book, 2. Aufl. Pearson Prentice Hall, New Jersey
3. Kemper A, Eickler A (2015) Datenbanksysteme - Eine Einführung, 10. Aufl. De Gruyter Oldenbourg, Berlin
4. Kessler J, Tschuggnall M, Specht G (2019) RelaX: a Webbased execution and learning tool for relational algebra. In: Proceedings of the 18th Fachtagung des GI-Fachbereichs Datenbanksysteme für Business, Technologie und Web (BTW), LNI GI, Rostock, S 503–506
5. Mühe H (2018) IRA - Interaktive Relationale Algebra. <http://db.in.tum.de/people/sites/muehe/ira/>. Zugegriffen: 15. Okt. 2020
6. Tomaselli S (2018) Educational tool for relational algebra. <https://github.com/ltworf/relational/>. Zugegriffen: 15. Okt. 2020
7. Yang J (2018) IRA - Interaktive Relationale Algebra. <https://users.cs.duke.edu/~junyang/radb/>. Zugegriffen: 15. Okt. 2020