Technical Perspective: Relational Diagrams and the Pattern Expressiveness of Relational Languages

Mirek Riedewald
Northeastern University
Boston, MA, USA
m.riedewald@northeastern.edu

ACM Reference Format:
Mirek Riedewald. 2018. Technical Perspective: Relational Diagrams and the Pattern Expressiveness of Relational Languages. *J. ACM* 37, 4, Article 111 (August 2018), 1 page. https://doi.org/XXXXXXX.XXXXXXX

When a database researcher or user hears the word "expressiveness," they most likely think about what kind of query some language or formalism supports. For instance, can a given version of SQL express recursion or computation of the transitive closure of a graph? For what fragment of relational calculus can we decide query containment? Can relational algebra express every relational-calculus query and vice versa? The following paper by Gatterbauer and Dunne may forever change the discourse on expressiveness in the context of relational languages by prompting the follow-up question: "Do you mean *logical* expressiveness or *pattern* expressiveness?"

For full disclosure, I am a co-author of Gatterbauer and Dunne's on QueryVis, a tool for visualizing complex SQL queries, which forms the foundation of the Relational Diagrams proposed in this new paper. However, the authors now are asking a bigger question that goes beyond visualization. They study whether there exist common patterns for expressing certain information needs in relational languages in general, and which specific languages support them.

One of the most surprising results concerns tuple relational calculus (TRC) and relational algebra (RA). It is well known that safe TRC and RA have the same logical expressiveness. However, it turns out that the former is more expressive in terms of patterns! While the examples in the paper may appear subtle—is it really so bad if the RA query has one more reference to a base table?—I am convinced that this is just the beginning of a new way of thinking about relational design patterns and how to teach and explain them to data analysts and database users.

On the technical side, it is exciting to see how the authors were able to formalize the notion of pattern expressiveness in a way that abstracts away the idiosyncrasies of the specific relational languages and visual formalisms. Essentially they boil the notion of a pattern down to the relation references, which sounds simple, but does require a substantial amount of technical machinery. And while a list of relation references may not look like a pattern, it represents a first and essential step in the direction of identifying and comparing practically useful relational design patterns.

The proposed Relational Diagrams do provide an effective visualization of relational patterns, as demonstrated by two applicability studies. For one, the authors went through 5 popular database textbooks to find all relational-calculus queries and determine if their patterns are supported by the different relational languages. Relational Diagrams can express 95% of the patterns found; relational algebra only 81%. The other is a carefully designed and executed, and anonymously preregistered, user study that shows with statistical precision how Relational Diagrams can help users distinguish patterns in queries faster and more accurately than SQL.

Overall, the paper is a must-read for anyone interested in the notion of design patterns in relational queries and in teaching how to write and understand non-trivial SQL queries. For a while, many people will probably ask for clarification of the meaning of "pattern expressiveness" and how it relates to its more familiar cousin. But I would not be surprised to one day see it appear in database textbooks (or whatever their future equivalents will be), right next to logical expressiveness. Gatterbauer and Dunne surely make a strong case for broadening one's view of expressiveness in the context of relational languages.

Author's Contact Information: Mirek Riedewald, Northeastern University, Boston, MA, USA, m.riedewald@northeastern.edu.

ACM 1557-735X/2018/8-ART111
https://doi.org/XXXXXXX.XXXXXXX