Report date 09/11/2024

Paper report:

**Overview:**

The paper proposed an algebraic approach to incremental view maintenance (IVM) for multisets (bags, or databases that contain duplicates).

Most of the IVM approaches at the time were restricted to set (so without duplicates) and also used an algorithmic approach, which led to IVM systems to be not very general in the sense that they were not resistant to language changes (such as new primitives being added), and didn't allow optimizations instead of actually implementing a language for it.

The idea is that this new algebra should assign an equation to each primitive (SELECT, JOIN…), the equation allows us to see how the final result of that primitive would change when a change to the data is made.

The paper introduces all the necessary notation.

The paper formalized the IVM problem writing that a correct solution should equal the rematerialized view and be in the form: (S∸▽S)⊎△S, where S is the starting set.

The whole goal now is to find out the set of entries to remove from S (▽S) and the set of entries to later add (△S). Both the plus and minus operations are multiset operations and take into consideration the number of elements when adding or removing them.

Constraint are placed to ▽S and △S in order for the solution to be minimal (cheap and efficient). A weakly minimal solution being a solution where ▽S only contains tuples of S because it would make no sense to remove tuples that are not in S. A strongly minimal solution is a weakly minimal solution having disjoint ▽S and △S since it would make no sense to first remove an item to only add it back later.

The paper then show a table where operations on S with a second relation T and ▽S and △S are decomposed so that ▽S and △S components are by themselves so that they can be computed as part of the incremental maintenance and applied to the materialized view. Here the benefits of having this algebraic logic are clear and it's quite easy to see how these equations can be modeled.

The paper also does a complexity analysis on space and time since it's important to understand when it's more beneficial to recompute the view vs applying the IVM process.

The most interesting part of the paper is not the algebraic logic itself but rather the way to prove and explain that having an algebraic logic is beneficial and once it's done it makes the system much more flexible and optimizable.

**Experimental metrics and scenarios:**

It's a theoretical/mathematical paper and it did not include an implementation

**Benchmarked against:**

No actual benchmark has been done since there isn't an implementation but the performances have mathematically been compared against the cost and time of the whole materialization process.

**Limitations:**

Complex to initially come up with a complete and consistent algebra. Also, implementation is not discussed.

**Repo:**

None

**Authors:**

Timothy Griffin, Leonid Libkin

**Year:**

1995