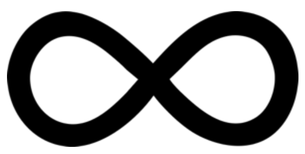
One Loop Does Not Fit All

Styliani Pantela, Stratos Idreos

Motivation

Just In Time compilation is a key technology in column stores

+



Infinite Possibilities

+

One solution does not fit all

=

No clear conclusion given the various setups

Our Approach

Query 1 Select Query

```
SELECT R.d FROM R
WHERE R.a < v1 AND R.b < v2
AND R.c < v3 AND R.d < v4
```

Algorithm 1 *4 Loops Strategy* for Query 1

```
1. inter = select(R.a, v1)
2. inter = select_fetch(R.b, v2, inter)
3. inter = select_fetch(R.c, v3, inter)
4. dest  = select_fetch(R.d, v4, inter)
```

Algorithm 2 *1 Loop Strategy* for Query 1

```
1. dest = select(R.a, v1, R.b, v2,
                 R.c, v3, R.d, v4)
```

Experimental Setup

- Multi Threaded Execution: 8 available cores
- Evaluation with branching and branchless execution
- Loop unrolling to optimize the branchless version
- Vectorized Processing
- 4-way Intel Xeon E7-4820 configuration with 64 hardware threads
- 1 TB of main memory
- GCC 4.7.2 (Debian 4.7.2-5).

Intuition

- Using separate loops for each predicate results in scanning fewer cache lines when the first filter selects very few tuples.
- For low selectivities it is the case that in the 1 Loop Strategy we are reading data that is not going to be used.

Results

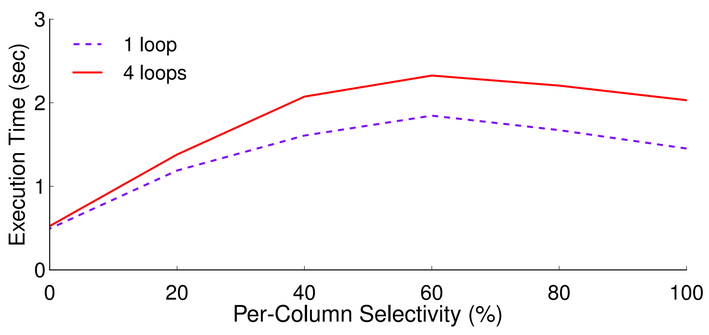


Figure 1: With Branches

Motivation: Optimizing for performance?
Method: Branchless Execution
High Selectivity: 1 Loop Strategy
Low Selectivity: 1 Loop Strategy

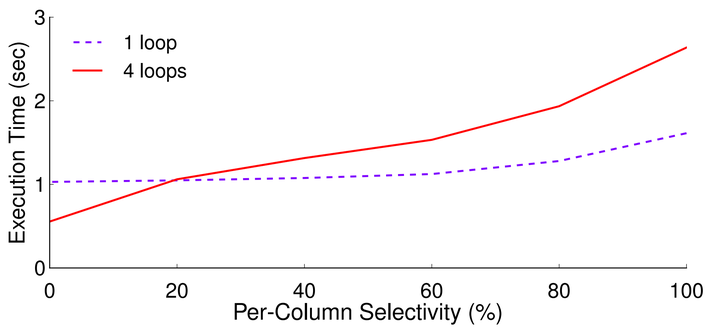
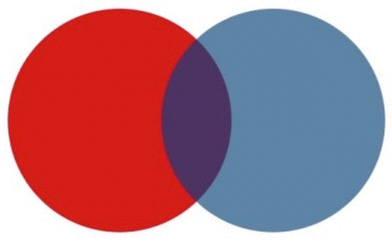


Figure 2: Branchless

Motivation: Optimizing for robustness?
Method: Branchless Execution
High Selectivity: 1 Loop Strategy
Low Selectivity: 4 Loop Strategy

Future Work

Disjunctive Selects



Alternate Distributions

Correlations