Query Optimization - Part 2

INFO-H417: Lab Session 6

2023-2024

Read through the PostgreSQL source code[1] to answer the following questions.

1. The default type analyze function computes different statistics based on the column type. What are the statistics that can be computed and how does the function choose which ones to compute? This standard type analyze function can be found in `src/backend/commands/analyze.c`.

2. What information does the distribution histogram store?

3. How is this data (the distribution histogram) used in estimating the selectivity of the less than '<' operator? The selectivity functions can be found in `src/backend/utils/adt/selfuncs.c`

4. Take a look at the `src/include/catalog/pg_operator.dat` file. This file contains the definitions for the different operators present in the database. Below is the definition of the equality operator between the int4 and int8 types.

   ```
   { oid => '15', descr => 'equal',
     oprname => '=', oprcanmerge => 't', oprcanhash => 't', oprleft => 'int4',
     oprright => 'int8', oprresult => 'bool', oprcom => '=(int8,int4)',
     oprnegate => '<>(int4,int8)', oprcode => 'int48eq', oprrest => 'eqsel',
     oprjoin => 'eqjoinsel' }
   ```

   Find the equivalent operator definition for the overlaps '&&' operator between two range types. What are the `oprcom` and `oprnegate` fields? Which functions are used for estimating the selectivity of restriction or join operators?

5. Taking a look at the code behind the join selectivity estimation for scalar and range types, what can you notice?

6. Similarly as for scalar types, the type analyze function for range types computes a distribution histogram. How is this histogram computed and what data does it store? The corresponding functions can be found in `src/backend/utils/adt/rangetypes_typanalyze.c`.

[1] `https://github.com/postgres/postgres`

7. How is the distribution histogram for range types used to estimate the selectivity of the overlaps '&&' operator? The corresponding functions can be found in `src/backend/utils/adt/rangetypes_selfuncs.c`.