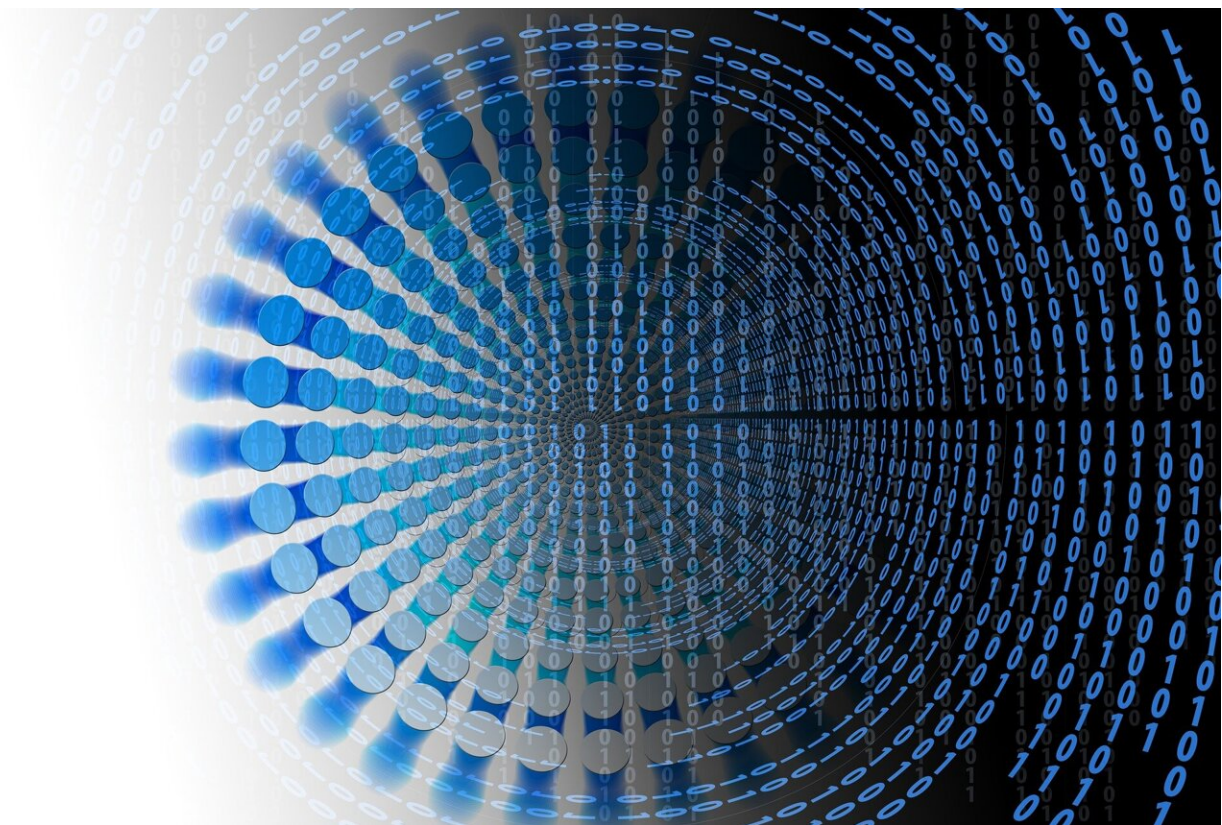# 'POLAR' lowers the adoption barrier for adaptive query processing in database systems

April 25 2024, by Jean-Paul Olivier

Credit: Pixabay/CC0 Public Domain

The preprint "[POLAR: Adaptive and Non-invasive Join Order Selection via Plans of Least Resistance](#)" introduces an adaptive query processing technique that lowers the adoption barrier for existing database systems while decreasing the risk of performance cliffs from ill-performing query plans.

BIFOLD researcher David Justen started the POLAR project as a collaboration of academia and industry with researchers from TU Berlin, Hasso Plattner Institute Potsdam, Ecole Polytechnique Paris, SAP, Google, Snowflake, and InfluxData. The preprint will be presented at the VLDB 2024.

Current database systems often encounter so-called performance cliffs—a huge slowdown in how fast a database can retrieve information. They may occur even through slight changes in the workload or data.

In user-facing applications, these cliffs lead to unexpectedly long loading times, increasing the probability that users will disengage from the application. The reason for these cliffs lies in the separation of concerns in database systems. The systems accept declarative queries (what to compute) and compile these into a query plan, which is an executable sequence of instructions that produces the result (how to compute).

Because of the huge amount of options, compiling the optimal plan for a given query is an [unsolved problem](#), in which wrong plan choices can be multiple orders of magnitude slower to process compared to the optimal plan.

The order of join operations, which can be found in everyday use cases like online shopping or travel booking, are especially impactful in terms of performance. Even slight changes in the queries or the data may

trigger the compiler to produce different query plans. These new plans may reduce the execution time, but oftentimes, they result in a performance cliff.

This problem has led researchers to introduce an approach called adaptive query processing (AQP). Instead of regarding the compilation and execution of a query as two separate phases, AQP intertwines these steps and uses live statistics to repeatedly recompile the query plan during execution. However, despite two decades of research, these strategies have not been implemented by commonly used database systems in practice.

In order to investigate this gap between academic research and applications in the industry, BIFOLD researcher David Justen started the POLAR project as a collaboration of academia and industry with researchers from TU Berlin, Hasso Plattner Institute Potsdam, Ecole Polytechnique Paris, SAP, Google, Snowflake, and InfluxData.

The group found two major hurdles for the adoption of AQP in practice: 1) As the intertwining of compilation and execution phases breaks with fundamental paradigms of database systems design, AQP approaches can be difficult to integrate into existing database systems as large parts of the code must be rewritten. 2) Prior AQP approaches often produce significant performance overheads and are not competitive with existing non-adaptive systems whenever those systems compile good query plans.

To reduce the adoption barrier for AQP, the research group introduced POLAR, an adaptive join reordering technique specifically focused on non-invasive integration and low overhead. For simpler integration, POLAR keeps the compilation and execution phases separate and only generates a small set of plan options during the compilation phase.

For overhead mitigation, POLAR selects the plan options in a way that

always allows it to fall back on the database system's original plan choice without recomputing any of the input data. Moreover, it introduces a probabilistic regret bound in the execution phase to decide which plan options to use.

As a testbed for a set of performance benchmarks, the researchers integrated a POLAR prototype into DuckDB, a modern, state-of-the-art, open-source database system. In a benchmark on a real database from the Internet Movie Database (IMDb), POLAR could improve the execution time of some queries by up to nine times without any noticeable overhead on queries, for which DuckDB already compiled well-performing plans. Moreover, the POLAR prototype in DuckDB outperformed state-of-the-art AQP systems by up to 15 times for entire workloads.

With POLAR, the research group introduces an adaptive query processing technique that lowers the adoption barrier for existing database systems while decreasing the risk of performance cliffs from ill-performing query plans. All of the group's code contributions are open-source.

barrier-query-database.html