**DIAS: Data-Intensive Applications and Systems Laboratory**
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

# Instance-Optimal (Cloud) DB Through Adaptive Execution

**Keywords:** Instance-optimality, adaptive execution, learned predicate, block-based execution

**Problem:** Instance-optimality in OLAP processing is an "optimal" goal that every database wants to achieve. It means for every dataset given, we aim for the minimum overhead by processing only the necessary data, generate only the necessary intermediate results. For cloud databases with multiple workers and data movements between workers, we also want to minimize the data transfer overheads and make every worker do some work until all there's no work left.

This instance-optimality is now being achieved in a very limited way, just for acyclic join queries [2, 3, 4], where theoretical foundations were built many decades ago [1]. The focus of recent papers [2, 3, 4] is to implement the theoretical works in an efficient way. However, we see a lot of reinventions of wheels here, for example, the idea of postponing the generation unnecessary intermediate results was from 2016 in a slightly different domain [5].

We also see lots of disconnections, for example, learned predicate [6] deals more operations than just joins, yet it is not considered as an effort to achieve instance-optimality. This is just an example, and there are many unveiled efforts related to the instance-optimality.

Yet another problem is that the query optimization remains outdated and isolated, which does not consider the benefits and overheads of such recent techniques in query execution. Hence, we are missing a holistic query optimization here.

Therefore, we aim to develop a system that achieves a near instance-optimal performance. We define and simulate the instance-optimal performance with the minimum input data and intermediate results, measure the gap between current baselines (e.g., [7] for cloud setting, [2, 3, 6] for non-cloud, i.e., single-node setting) and the virtual instance-optimal performance. Then, we enhance recent techniques [2, 3, 4, 6] with innovative ideas of approximation, holistic optimization, and other previous experience/expertise from the lab, to further approach to the instance-optimality.

As this project gains high interests from industry and academy, we have potential collaborators and connections including Snowflake, Google, Microsoft, and other universities.

[1] Yannakakis algorithm
[2] Robust Join Processing with Diamond Hardened Joins
[3] Instance-Optimal Acyclic Join Processing Without Regret: Engineering the Yannakakis Algorithm in Column Stores
[4] Predicate Transfer: Efficient Pre-Filtering on Multi-Join Queries
[5] Efficient Subgraph Matching by Postponing Cartesian Products
[6] PLAQUE: Automated Predicate Learning at Query Time
[7] FlexpushdownDB: rethinking computation pushdown for cloud OLAP DBMSs
https://github.com/cloud-olap/FlexPushdownDB

**DIAS: Data-Intensive Applications and Systems Laboratory**
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

**Project:** In this project, the student will 1) run baseline databases such as [7], 2) implement virtual instance-optimal methods (the goal we want to achieve), 3) implement recent techniques into the baseline, and 4) further improve the performance with novel ideas of approximation, holistic optimization, or work sharing between multiple queries.

**Plan:**

1. Setup and run a baseline database, measure the initial performance.
2. Implement the virtual instance-optimal method and measure the optimal performance.
3. Implement recent techniques onto the baseline and measure the improvements.
4. Further improve the performance with novel techniques, connecting researches from 1981 to 2024.

**Supervisor:** Prof. Anastasia Ailamaki, anastasia.ailamaki@epfl.ch
**Responsible collaborator(s):** Kyoungmin Kim, kyoung-min.kim@epfl.ch

**Duration:** 3-6 months