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/

Transforming Concurrency into Parallelism through Transaction Partitioning

Keywords: transaction execution plan generation, data-oriented transaction execution, DORA, transaction flow graph, Shore-MT

Problem: Transaction processing is one of the most data-intensive server applications and one of the most challenging to scale, even though the parallelism at the request level is ample. The traditional transaction processing execution paradigm has inherent scalability problems, stemming from the fact that any thread in the system is allowed to access any data structure. To overcome the inherent scalability problems of traditional designs, recently there have been proposals for partitioned transaction execution designs, with data-oriented transaction execution (DORA) as an example. All those research designs however require the submission of transactions in some special form which typically has hard-coded the execution plans. For example, DORA gets as input transaction flow graphs which are directed graphs of actions over disjoint sets of data and rendezvous points. The automatic derivation of the execution plan of the transaction which includes the decomposition of the transaction to map to a specific partitioning schema is an open research challenge. Equally challenging is the automatic derivation of the physical design (including the partitioning schema and the needed indexes) given a transactional workload for a partitioned transaction execution platform.

Project: This project is about designing and implementing algorithms that automate the generation of transaction execution plans as well as the physical design for partitioned transaction execution platforms. The first algorithm given as input a traditional transaction as well as a partitioning schema should be able to generate an execution graph represented as a transaction flow graph. The second algorithm given as input a transactional workload should be able to suggest a physical design (partitioning schema and indexes) for this workload. The implementation will take place in the current DORA prototype. The DORA prototype operates over Shore-MT. Shore-MT provides a C API which can be used to specify and populate database objects and execute transactions over them. However, it currently does not have a front-end and all the interaction with the database must be hard coded through the C API. Upon completion of the project the expected deliverables are the following:

- A specification of the various representations for transactions. There are at least 3 different representations needed, each one with more details about the execution plan.
- A specification of the description of the physical design of the database that includes the description of the employed partitioning schema and the indexes used.
- An algorithm that takes as input a transaction as well as the partitioning schema and outputs the transaction execution graph for that transaction.
- A demonstration of the automatic generation of the transaction flow graphs for some popular database benchmarks, spanning from simple, such as TATP, to fairly complicated, such as TPC?E, given a partitioning schema.

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/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

- An algorithm that takes as input a set of transactions and outputs suggestions for the physical design (indexes and partitioning schema) that should be used, along with the corresponding transaction execution graphs for each transaction in the set.
- A demonstration of the suggestion of a specific database physical design (partitioning schema and database objects created) given a transactional workload.
- A tool for automatically generating all the classes and functions needed to implement a transaction based on its definition inside the DORA prototype.

Supervisor:	Prof. Anastasia Ailamaki, anastasia.ailamaki@epfl.ch
Responsible collaborator(s):	Ippokratis Pandis (ippokratis.pandis@epfl.ch), Ryan Johnson (ryan.johnson@epfl.ch)
Duration:	3 months