SCHEDULING QUERY MIXES: THE CHALLENGES
BRITTANY FASY, HERODOTOS HERODOTOU
COMPUTER SCIENCE, DUKE UNIVERSITY

**Motivation**
- Database Systems handle concurrent execution of queries of many different types.
- But, standard methods of query optimization are based on evaluating the cost of each query in isolation from the rest.
- Query interactions impact the performance of the system significantly.
  - Negative Interactions: lock contention, CPU & I/O bound effects etc
  - Positive Interactions: cache hits, table scan sharing etc
- Taking Query interactions into consideration improves:
  - Query Scheduling
  - Load Balancing
  - Admission Control

**Background**
- QShuffler is the first step towards an interaction aware query scheduler.
- Uses a multidimensional linear regression model to predict performance of query mixes
- When a query completes, QShuffler selects the most expensive query to execute, that keeps the overall load closed to a predefined load threshold.
- QShuffler Setting:

**Performance Model**

**Linear Model**
- Less Accurate But Less Overhead
- Less Training Sampling
- \( \sum_{j=1}^{n} y_{ij} = \sum_{j=1}^{n} a_{ij} \)

**Quadratic Model**
- More Accurate But More Overhead
- More Training Sampling
- \( \sum_{j=1}^{n} y_{ij} = \sum_{j=1}^{n} a_{ij} \) (\( a_{ij} \) is the weight of query mix combination)

**Online Training**
- Online Updating of Regression Weights
  - Decrease Offline Training Data Collection Time
  - Online Calculation of NRO
  - Minimize the Error:
  - Update to Weights:

**Future Investigation**
- Load Threshold Selection
  - Fixed Vs Online Adaptation
- Further Investigation of Online Update Model
  - Exploration Vs Exploitation
- Integrate Scheduler into a Database System
  - Black Box → White Box