



PGX.D/Async

A Scalable Distributed Pattern Matching Engine for Property Graphs

Nicholas Roth, Vasileios Trigonakis, Sungpack Hong, Hassan Chafi

Oracle Labs

Anthony Potter, Boris Motik, Ian Horrocks

Department of Computer Science, University of Oxford

May 19, 2017

Oracle Labs

ORACLE

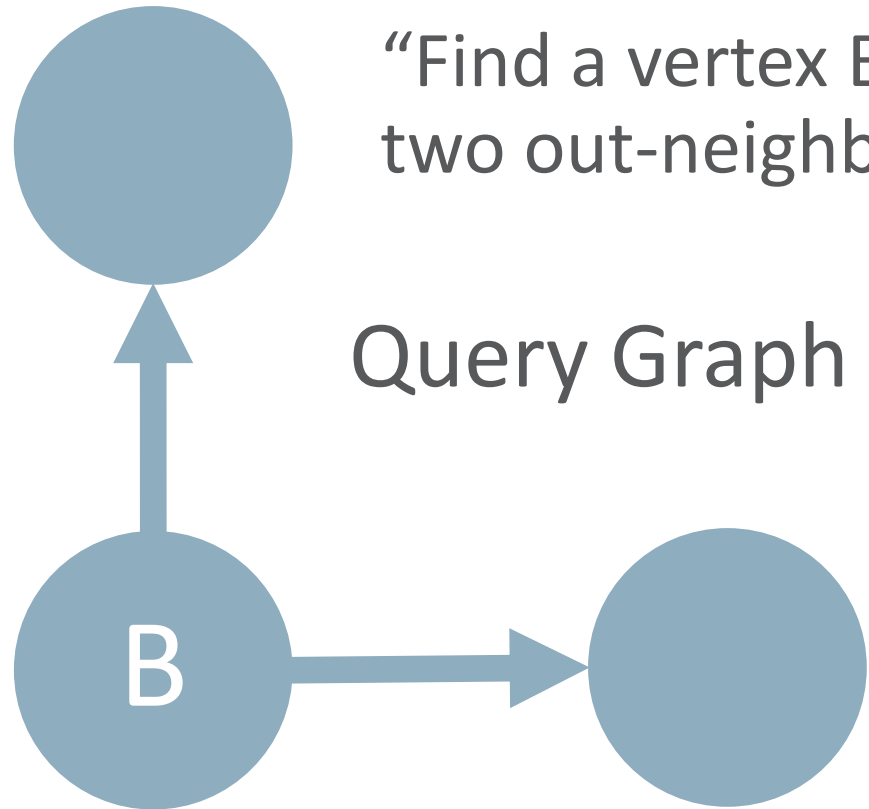
Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Program Agenda

- 1 Motivation
- 2 PGX.D/Async
- 3 Evaluation
- 4 Conclusion

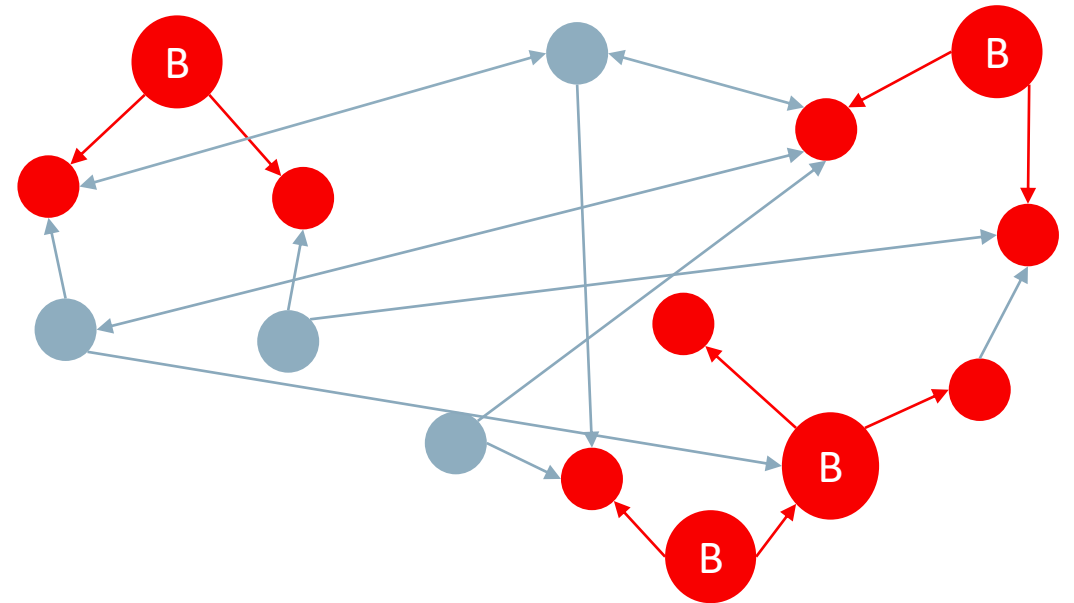
Graph Pattern Matching



“Find a vertex B with two out-neighbors”

Query Graph

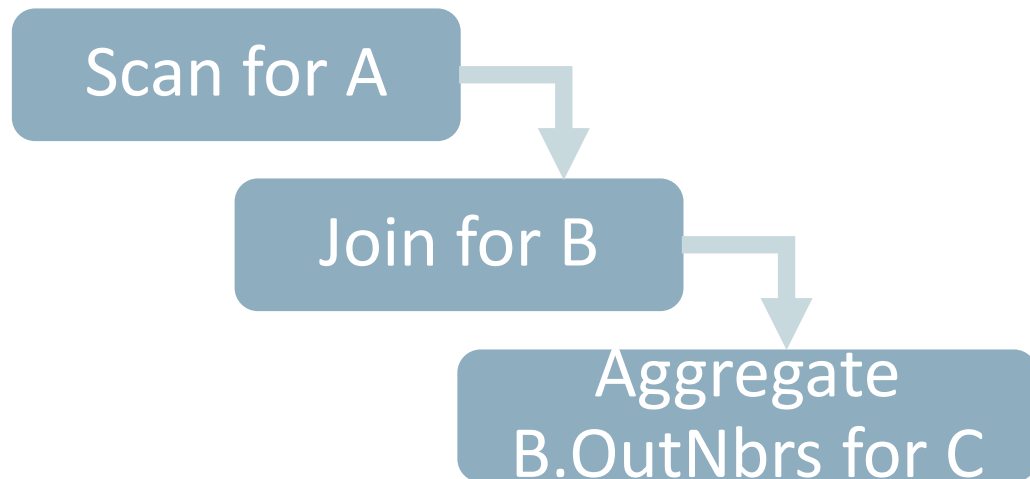
Data Graph



Naïve Approaches

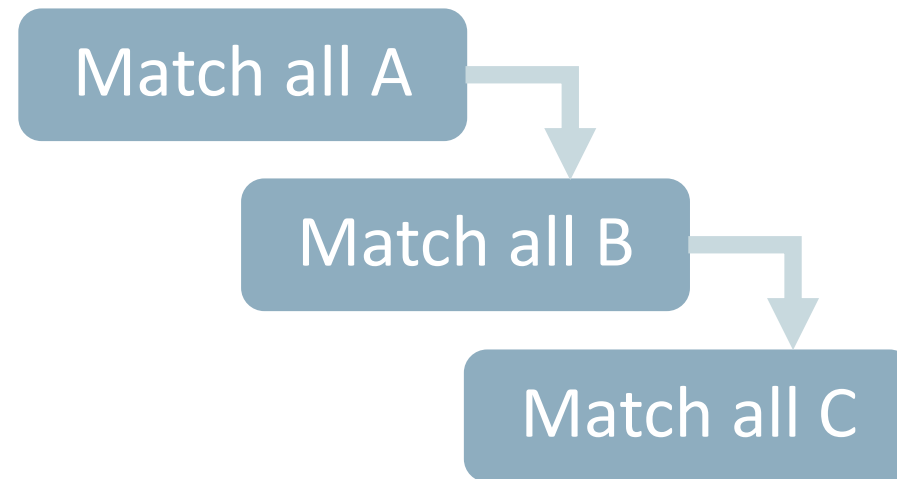
Joins

1. Scan to find A
2. Join A.InNbrs with Vertices on ID => B
3. Everything in B.OutNbrs is C



Breadth-first traversal (multi-source)

1. Visit each vertex A
2. Visit each vertex B
3. Visit each vertex C





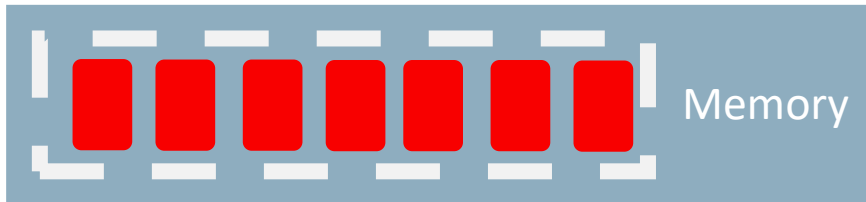
Key Principles

- Asynchronous
- Strict, precise flow control

Requirements for a Good Solution

Requirements

- Bounded memory consumption
 - Limit intermediate results



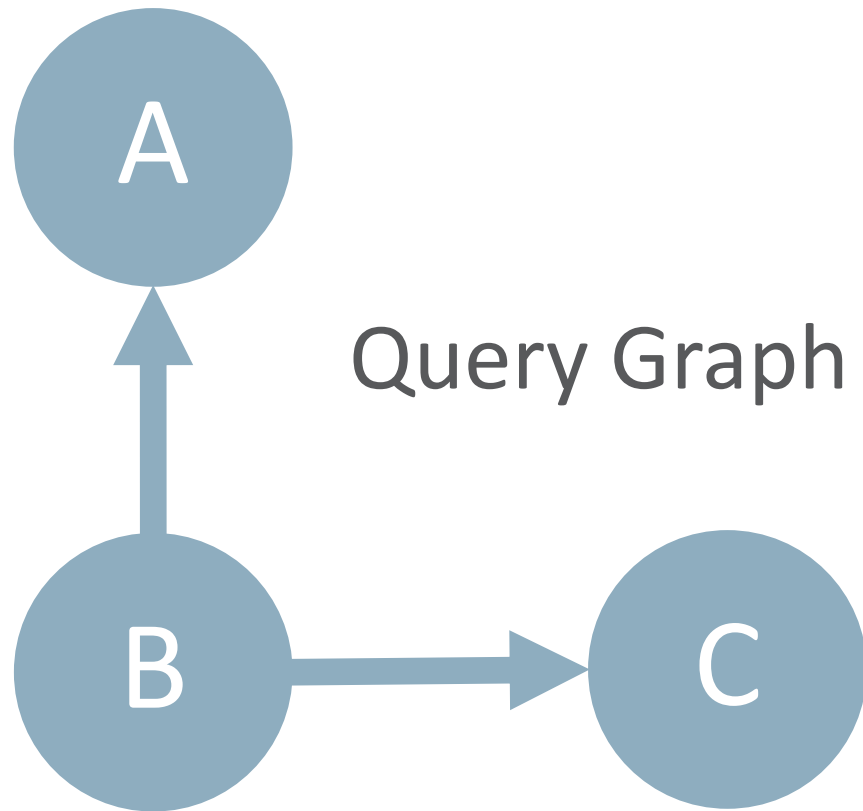
- Messaging should not block computation

Approaches

- Asynchronous DFT
 - Fixed memory requirement
 - Everything is concurrent

- Strict, precise flow control
 - Balanced memory allocation
 - Minimal performance impact

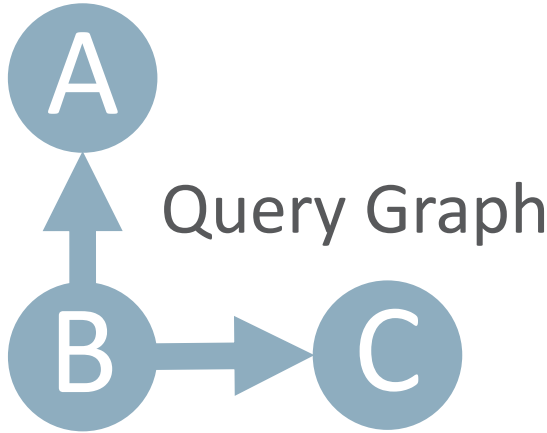
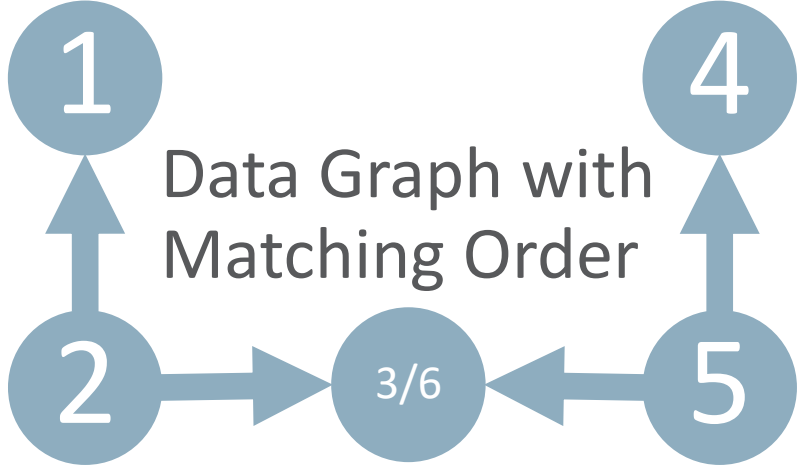
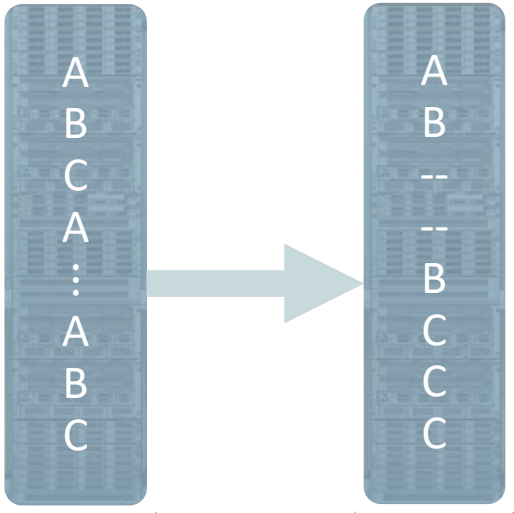
Graph Pattern Matching



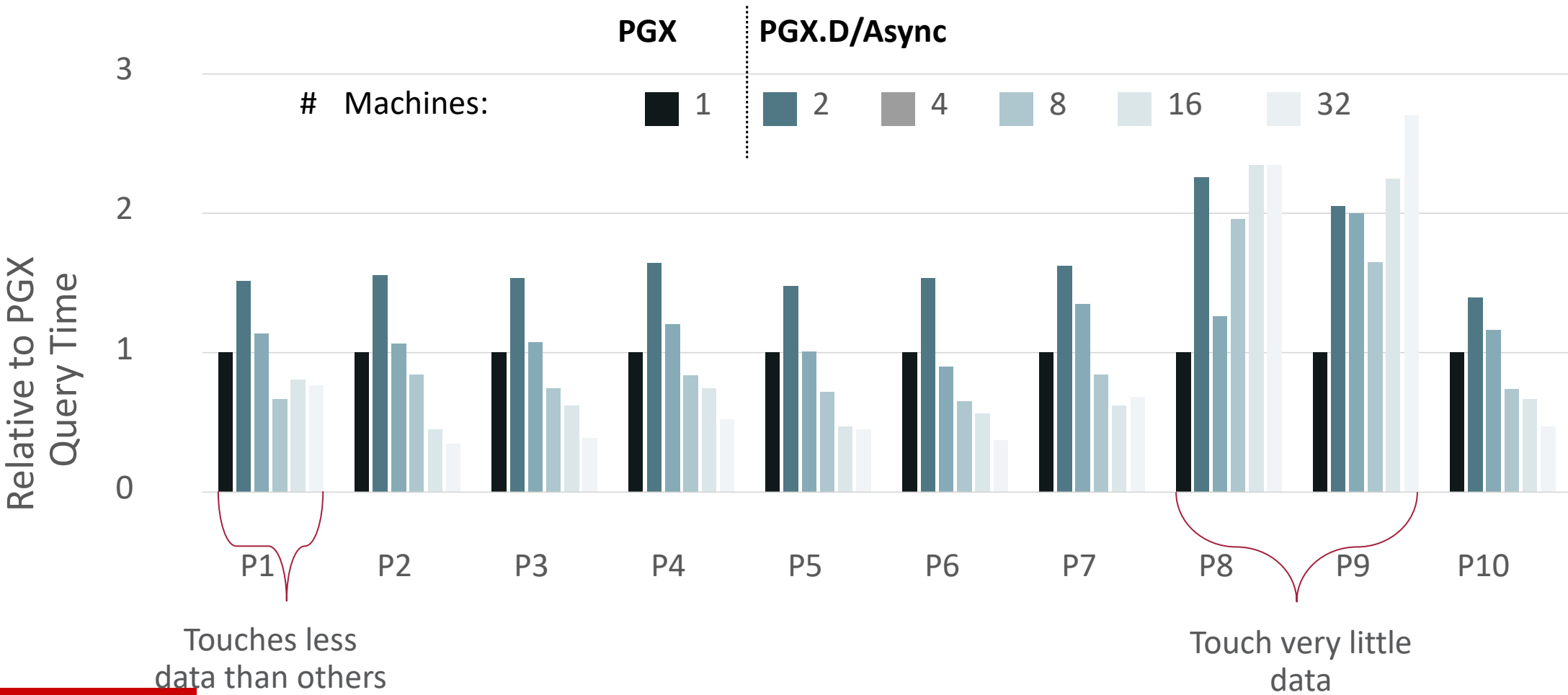
“Find a vertex B with two out-neighbors”

PGX.D/Async

- Depth-first traversal
 - Match A, then B, then C
- Asynchronous
 - Messaging in parallel with computation



Evaluation



Conclusions

- Asynchronous DFT is a good distributed solution
 - Parallelism and asynchrony hide local imbalances
 - Scales to larger queries
- More in paper
 - From PGQL (query) to execution plan
 - Details about runtime