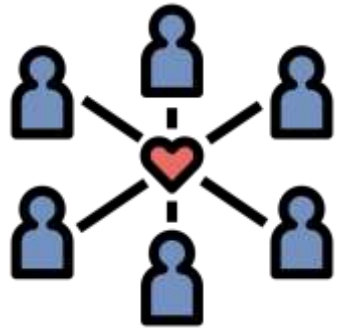
Raqlet: Cross-Paradigm Compilation for Recursive Queries

CIDR, Chaminade, January 2026

Amir Shaikhha, Youning Xia, Meisam Tarabkhah, Jazal Saleem, **Anna Herlihy**

Recursive queries power advanced data analysis

- **Graph Analytics**: Reachability, path queries



- **AI Knowledge Systems**: Rule-based reasoning on graphs

- **Program analysis** and dataflow frameworks

- **Declarative networking** and distributed protocols

Key enabler for rich logical inference in modern applications

Modern Recursive Query Engines

Relational DB

- SQL `WITH RECURSIVE`
- Relations + joins

```
WITH RECURSIVE path AS (
  SELECT * FROM edge
    UNION ALL
  SELECT path.x, edge.y
  FROM path, edge
  WHERE path.y = edge.x)
SELECT * FROM path
```

Deductive DB

- Datalog
- Facts + logic rules

```
path(x, y) :- edge(x, y)
path(x, y) :- path(x, z),
              edge(z, y)
```

Graph DB

- Cypher, GQL, PGQ
- PG nodes + edges, RDF triples

```
MATCH
   (x:Node)-[:EDGE*] ->
     (y:Node)
RETURN DISTINCT
   x.id AS x, y.id AS y
```

The recursive landscape is fragmented

Query model

- unfriendly
+/- standardized!
- less expressive

+ friendly, concise
+/- GQL standard
- least expressive

+ friendly/variants vary
+ strong formalization
+ most expressive

Performance

MySQL > ... and Neo4J for point-lookup queries but < fo...

Datalog > GDBMS + RDBMS for classic recursive queries [Brass et. al. 2019]

Neo4J > M... et.

DuckDB > Soufflé [Herlihy et al., 2025]

RDBMS ... workloads with heavy use of group by, sort, and aggregate [Cheng et al., 2019]

Data model

+rigid schema

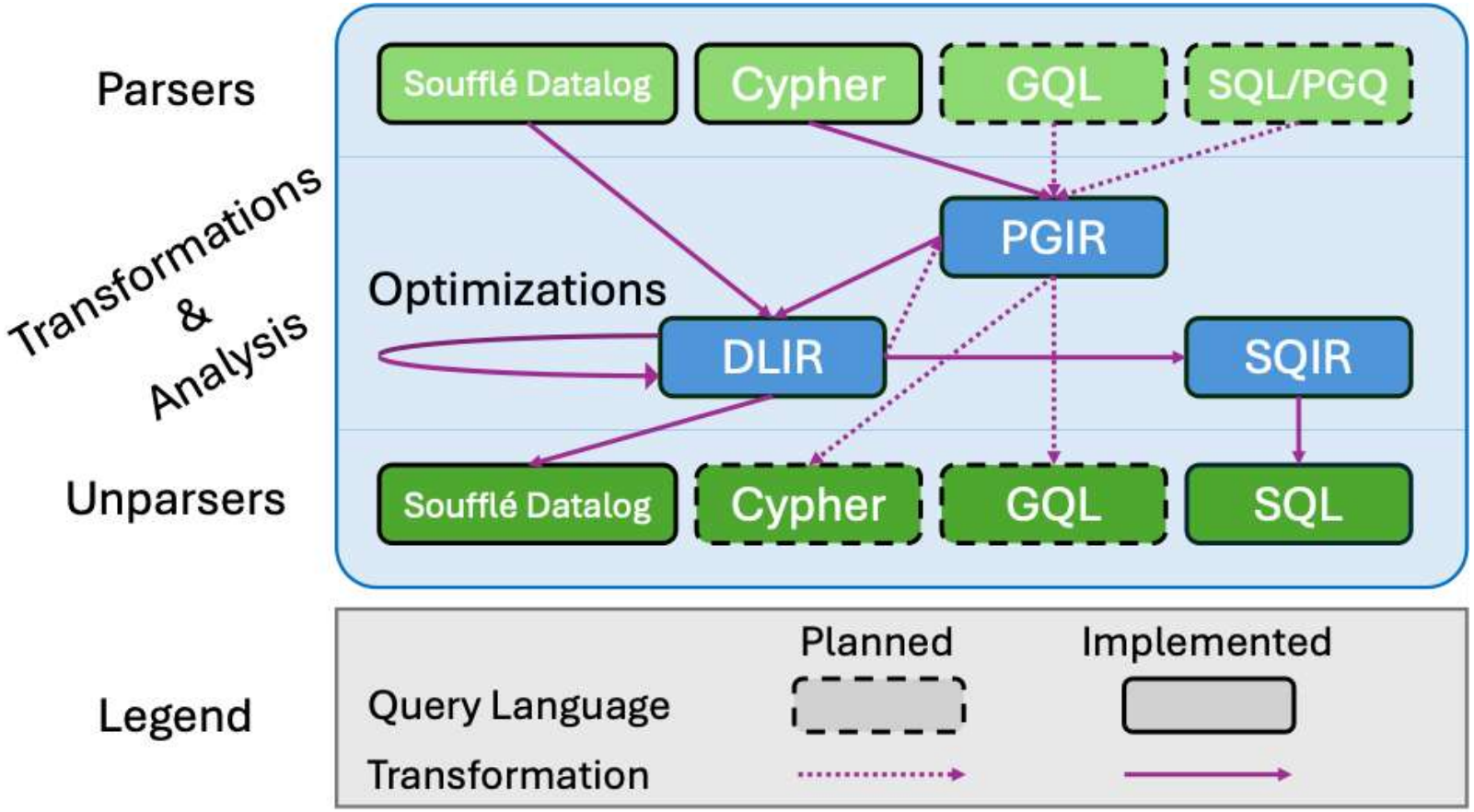
-join-heavy queries

- schema-optional (+pg-schema)
- node-and-edges assumption

Paradigms are distinct and incompatible

Raqlet: Cross-Paradigm Recursive Query Compilation

Compilation-based framework to unify the fragmented recursive ecosystem



Key insight: statically verified cross-compilation with *tiered intermediate representations*

Key Benefits of Raqlet

1. Static analysis

- Determines if well-formed in both the source and target language

2. Query optimization

- Paradigm-agnostic optimizations
- Unlocks unsupported optimizations

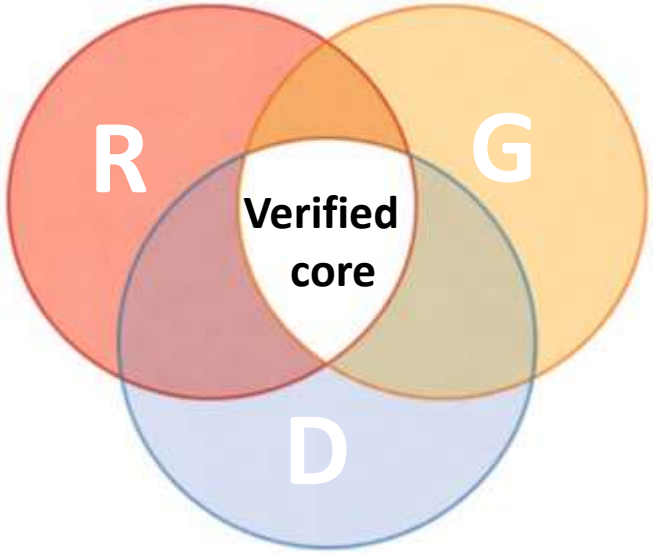
3. Towards a unified formal semantics

- Serve as a “golden” reference implementation
- DLIR serves as a formally-specified core

Execution time (ms) LDBC short query 1 (SQ1) and complex query 2 (CQ2) for SF10.

Query	Optimized	Neo4j	Soufflé	DuckDB	HyPer
SQ1	✗	72.17	0.05	24.25	0.89
	✓	-	0.02	1.78	0.78
CQ2	✗	87.85	11.70	33.18	215.85
	✓	-	11.31	4.01	168.16

AMD Ryzen 9 5950X 16-Core Processor operating at 3.4GHz, with 64GB of DDR3 RAM on Ubuntu 24.04.1 OS



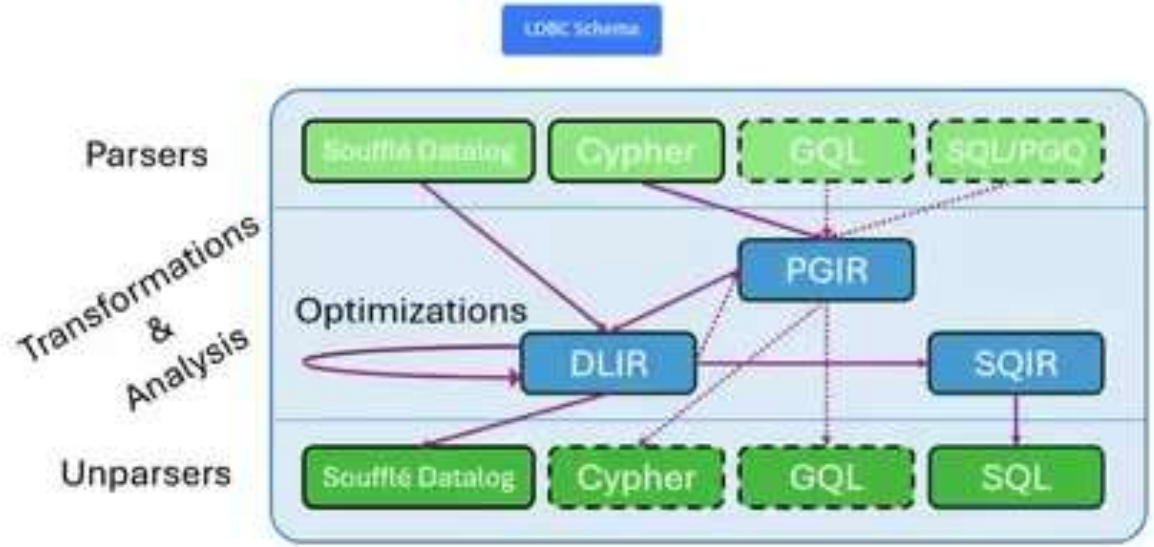
The vision: Raqlet as a machine-checked cross-paradigm semantic core for recursive queries

Demo: Dr. Meisam Tarabkhah

Raqlet TRANSLATION OPTIMIZATION EXECUTION

Raqlet: A Universal Framework for Querying Graphs

For this demonstration, we embed the LDBC Social Network Benchmark (SNB) dataset.



In this presentation, we present the Web interface of our proposed framework, Raqlet.

localhost:3000

Conclusion

Raqlet unlocks:

- Interop between relational, deductive, and graph systems
- Cross-system comparison and rapid prototyping
- Cross-paradigm formal reasoning and optimization

Thank you!

anna.herlihy@epfl.ch

amir.shaikhha@ed.ac.uk y.xia-32@sms.ed.ac.uk m.tarabkhah@ed.ac.uk j.saleem@sms.ed.ac.uk