Property Graph Query Language Landscape

Alastair Green  Neo4j
openCypher Implementers Group meeting 3, 27 July 2017
LDBC, openCypher, ISO/INCITS

There’s a lot going on!

April 2015 Linked Data Benchmark Council (LDBC) set up a Query Language TF
SQL used as the lingua franca of TPC benchmark definitions, where is the standard graph query language. Now getting to the point of proposing new features.

openCypher announced autumn 2015: started with 2016 work on software artefacts
February 2017 first oCIM → oCIGs → implementer consensus model (no Neo4j veto)

INCITS Ad Hoc on SQL Extensions Property Graphs April 2017
“Project split” to set up formal ISO SQL/PGQ project with a four-year horizon June 2017
What’s being discussed? LDBC

**LDBC Graph QL TF**

- Mix of vendors and researchers
- Started with examination of existing languages (including Cypher, PGQL)
- Composable language (closure over the PG data model, “returning graphs”)
- Extended Property Graph Data Model “paths as first class citizens”
- Complexity and its implications for path queries (relates to “morphism”)
- Has reached the point where the group feels they are ready to propose syntax
- Change an existing language, or create a new one?
The openCypher community: towards an open standard

In late 2015 Neo announced the openCypher initiative

- Apache-licensed grammar, ANTLR parser, TCK
- Open Cypher Improvements process based on Github issues/discussions
- Work has started on a formal specification of Cypher (denotational semantics) by University of Edinburgh
Governed by the openCypher Implementers Group

In 2017 two face-to-face openCypher Implementers Meetings have taken place

Regular openCypher Implementers Group virtual meetings scheduled through to October

Consensus-based governance: open to all, but implementers “at the heart of the consensus”
Cypher implementations

Cypher is used as the graph query language of four commercial/OSS databases

- Neo4j Enterprise Server, SAP HANA Graph, AgensGraph/Postgres, and RedisGraph

There are other databases/query engines in gestation or in the research community

- Memgraph, Ingraph, Scott Tiger, Cypher for Apache Spark, Graphflow ...

There are several other projects or tools that use Cypher

- IDEA plugin from Neueda, language parsers, editors, GraphQL Cypher directives, ...
What’s being discussed? openCypher

openCypher

2016 Oracle paper on PGX, PGQL and Green Marl criticised Cypher
Path expressions weak PGQL “path patterns” are very nice concept
“Morphism”
No graph construction (composition in another form)
No views (composition in another form)
Meshed with pre-existing discussions in Neo4j Cypher Language Group
Other issues like sub-queries, SQL interactions, aggregation, … many live topics
What’s being discussed? INCITS/ISO

INCITS Ad Hoc, SQL/PGQ

Graph objects in SQL and relationship to relational data model
Should SQL be extended to incorporate graph data model concepts?
Should SQL interoperate with graph query engines with their own language(s)
Over to Jan Michels (Oracle), Ad-Hoc Group Chair
SQL and Cypher (and PGQL)

Alastair Green  Neo4j
openCypher Implementers Group meeting 3, 27 July 2017
SQL

SQL/PGQ

Cypher 2016

Cypher 2017

PGQL 1.0

LDBC QL TF desired features
**SQL/JSON → SQL/Cypher**

- Context item
- Path Specification
- Passing clause
- SQL/JSON Sequence
- Path Engine

~ input graph
~ Cypher query
~ params
~ result set
~ Cypher engine

**Figure 4 — Architecture of SQL/JSON path language usage**
Cypher 2017 + PGQL → “CyQL”

SQL

SQL/PGQ

SQL Graph Representations + Graph Functions

Cypher 2016

PGQL 1.0

Cypher 2017

LDBC QL TF

desired features