The Linked Data Benchmark Council (LDBC): Driving competition and collaboration in the graph data management space

Gábor Szárnyas

TPCTC | 2023-08-28 | Vancouver

Co-authors: Brad Bebee, Altan Birler, Alin Deutsch, George Fletcher, Henry A. Gabb, Denise Gosnell, Alastair Green, Zhihui Guo, Keith W. Hare, Jan Hidders, Alexandru Iosup, Atanas Kiryakov, Tomas Kovatchev, Xinsheng Li, Leonid Libkin, Heng Lin, Xiaojian Luo, Arnau Prat-Pérez, David Püroja, Shipeng Qi, Oskar van Rest, Benjamin A. Steer, Dávid Szakállas, Bing Tong, Jack Waudby, Mingxi Wu, Bin Yang, Wenyuan Yu, Chen Zhang, Jason Zhang, Yan Zhou, Peter Boncz
LDBC: Linked Data Benchmark Council

- Non-profit company
- Mission: Accelerate progress in graph data management
- Designs graph benchmarks & governs their use
- Fosters collaboration between researchers & practitioners

ldbcouncil.org  github.com/ldbc
My involvement in LDBC

2017  Joined a benchmark task force

2020  Started working at CWI in Amsterdam (Database Architectures group)

Tasks  Benchmarks and their auditing process
       Organizational restructuring
       Running board and community meetings
LDBC's history
LDBC timeline

EU FP7 project
EU FP7 project

Benchmark papers

LDBC timeline

TPC-H analyzed
Datagen
Parameter curation
Interactive
TPB
Graphalytics
ACID tests
SNB BI
VLDB
Interactive v2
LDBC timeline

EU FP7 project

Benchmark papers

Language and schema papers

Technical User Community meetings
Benchmark overview
## Similarities to TPC benchmarks

<table>
<thead>
<tr>
<th>Application-level benchmarks</th>
<th>Scale factors: SF30 = 30GiB CSV</th>
<th>Few dozen query templates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Third-party auditors</td>
<td>FDRs with metrics, e.g. throughput@SF</td>
<td>Benchmark approval and renewal</td>
</tr>
</tbody>
</table>
Similarities to TPC benchmarks

- **application-level benchmarks**
- **third-party auditors**
- **scale factors:** SF30 = 30GiB CSV
- **FDRs with metrics, e.g. throughput@SF**
- **few dozen query templates**
- **benchmark approval and renewal**
Similarities to TPC benchmarks

- Application-level benchmarks
- Third-party auditors
- Scale factors: $SF_{30} = 30$GiB CSV
- FDRs with metrics, e.g. throughput@SF
- Few dozen query templates
- Benchmark approval and renewal
## Similarities to TPC benchmarks

<table>
<thead>
<tr>
<th>Application-level benchmarks</th>
<th>Scale factors: SF30 = 30GiB CSV</th>
<th>Third-party auditors</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FDRs with metrics, e.g. throughput@SF</td>
<td>Benchmark approval and renewal</td>
</tr>
</tbody>
</table>
## Similarities to TPC benchmarks

<table>
<thead>
<tr>
<th>Application-level benchmarks</th>
<th>Scale factors: SF30 = 30GiB CSV</th>
<th>Few dozen query templates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Third-party auditors</td>
<td>FDRs with metrics, e.g. throughput@SF</td>
<td>Benchmark approval and renewal</td>
</tr>
</tbody>
</table>
Similarities to TPC benchmarks

- Application-level benchmarks
- Scale factors: SF30 = 30GiB CSV
- Few dozen query templates
- Third-party auditors
- FDRs with metrics, e.g. throughput@SF
- Benchmark approval and renewal
Similarities to TPC benchmarks

- application-level benchmarks
- scale factors: SF30 = 30GiB CSV
- few dozen query templates
- third-party auditors
- FDRs with metrics, e.g. throughput@SF
- benchmark approval and renewal
The Social Network Benchmark (SNB) suite
Data set and queries
<table>
<thead>
<tr>
<th>Data set</th>
<th>Queries</th>
<th>Updates</th>
</tr>
</thead>
</table>

Q9($name, $day)

Pa knows *1..2
name = $name
creation date < $day

M

author

Updates

Data set

Queries
Q9("Bob", "Sat")

Data set

Bob
Ada
Carl
Dan
Eve
Gia
Finn

author
knows
reply

Queries

M1 Mon
M2 Tue
M3 Sun
M4 Tue
M5 Fri

Updates

Q9("Bob", "Sat")

Pa

knows *1..2
name = "Bob"

Pb

author
creation date < "Sat"
**Data set**

- Ada
- Finn
- Bob
- Dan
- Gia
- Carl
- Eve

**Queries**

- M1 Mon: author
- M2 Tue
- M3 Sun
- M4 Tue
- M5 Fri

**Updates**

- Q9(“Bob”, “Sat”)
  - Pa knows *1..2
  - Pb
  - name = “Bob”
  - M
  - creation date < “Sat”
**Data set**

- Ada
- Bob
- Dan
- Carl
- Eve
- Gia

**Queries**

- $\text{M1 Mon}$
- $\text{M2 Tue}$
- $\text{M3 Sun}$

- $\text{M4 Tue}$
- $\text{M5 Fri}$

$\text{author}$

$\text{reply}$

**Updates**

$Q9(\$name, \$day)$

$\text{Pa knows} *1..2 \text{ Pb}$

$name = \$name$

$creation date < \$day$
**Ada**

**Bob**

**Dan**

**Gia**

**Carl**

**Eve**

**Finn**

**M1**

**M2**

**M3**

**M4**

**M5**

**Pa**

**Pb**

**M**

**Q9(“Finn”, “Wed”)**

**Data set**

**Queries**

**Updates**

knows

knows

author

reply

author

name = “Finn”

creation date < “Wed”

*1..2
Q9(“Finn”, “Wed”)

Pa
knows *1..2

Pb
name = “Finn”

M
creation date < “Wed”
Data set

- Q9("Bob", "Sat"): 10 nodes
- Q9("Finn", "Wed"): 5 nodes

Queries

- M1 Mon
- M2 Tue
- M3 Sun
- M4 Tue
- M5 Fri

Updates

- Q9("Finn", "Wed")
- Pa knows *1..2
- name = "Finn"
- creation date < "Wed"

- M

- Q9("Bob", "Sat"): 10 nodes
- Q9("Finn", "Wed"): 5 nodes
Parameter selection

- *Uniform random parameters* $\rightarrow$ unstable distributions
Parameter curation

A. Gubichev, P. Boncz (TPCTC 2014)
Parameter selection

- **Uniform random parameters** $\rightarrow$ unstable distributions
- **Curated parameters** $\rightarrow$ tighter distributions, closer to bell curves
Updates
Data set

- Ada
- Finn
- Mon
- Bob
- Dan
- Carl
- Eve
- Fri
- Gia
- Tue

Queries

- author
- reply

Updates

- + knows("Eve", "Gia")
Data set

- Ada
- Bob
- Dan
- Carl
- Eve
- Finn
- Gia

Queries

- knows("Eve", "Gia")
- Comment("Gia", "M3")

Updates

- + knows("Eve", "Gia")
- + Comment("Gia", "M3")
Updates

+ knows("Eve", "Gia")
+ Comment("Gia", "M3")
- Person("Eve")

Heavy-hitting operation!

GRADES-NDA@SIGMOD 2020
SNB workloads

- OLTP: Interactive
- OLAP: Business Intelligence
SNB Interactive v1 (2015)

Queries start in 1–2 person nodes
14 complex reads, 7 short reads
8 insert operations run concurrently
Goal: High throughput (ops/s)

“Driving competition”
SNB Business Intelligence (2022)

Goal: High throughput & low query runtimes

Queries touch on large portions of the data
20 complex read queries, insert & delete ops
Both bulk and concurrent updates allowed

Audited results

Results for 100GB, 1TB, and 10TB

Scores for 10TB:

- Power@SF: 89,444
- Throughput@SF: 30,990

More results expected in late 2023
Financial Benchmark (2023)

**Target:** Distributed transactional systems
Financial Benchmark (FinBench)

Originally proposed by the Ant Group, developed with Create Link, Ultipa, etc.

Features:

- Strict latency requirements (P99 < 100 ms), relaxed consistency guarantees
- Truncation (sampling) on more recent edges
- Interesting queries, e.g. REM path queries (Regular Expression with Memory)
Benchmarking and auditing
Making benchmarks easy to use

For each workload:
- Specification
- Academic paper
- Data generator
- Pre-generated data sets
- Benchmark driver
- 2+ reference implementations

Guidelines:
- How to execute the benchmark correctly
- Validate the results
- Verify ACID-compliance
Auditing and trademark

Auditing process:

- Auditors are trained by the LDBC task forces and they take an auditor exam to get certified.
- Audits typically cost around 20-50k USD (plus infra costs) and take multiple weeks.

Trademark:

- LDBC is trademarked worldwide. Only a result produced by a certified auditor is an “LDBC benchmark result”
- Unofficial benchmark results must come with a disclaimer: “This is NOT an official LDBC benchmark result”
LDBC's working groups: graph schema and query languages
Modern graph query languages

- neo4j: Cypher
- TigerGraph: GSQL
- Amazon Neptune: SPARQL
- Dgraph: DQL
- ArangoDB: AQL
- Vaticle TypeDB: TypeQL
- JanusGraph: Gremlin
- NebulaGraph: nGQL
- XTDB: Datalog

LDBC benchmarks define queries in plain text
New ISO standard query languages

- **SQL/PGQ** (Property Graph Queries), part of SQL:2023
- **GQL** (Graph Query Language), to be released in 2024

- LDBC has a **liaison with ISO** which allows its members to access to the standard drafts
**SQL:1992**

```
SELECT DISTINCT m.id
FROM ( 
    SELECT k.p2id AS id
    FROM person Pa,
        knows k
    WHERE Pa.name = $name
    AND Pa.id = k.p1id
    UNION
    SELECT k2.p2id AS id
    FROM person Pa,
        knows k1,
        knows k2
    WHERE Pa.name = $name
    AND Pa.id = k1.p1id
    AND k1.p2id = k2.p1id
    AND k1.p1id <> k2.p2id
) Pb,
    Message m
WHERE Pb.id = m.authorId
AND m.creationDate < $day
```

**SQL/PGQ (SQL:2023)**

```
SELECT id
FROM GRAPH_TABLE (socialNetwork
MATCH ANY ACYCLIC
(Pa:Person WHERE Pa.name = $name)
-[:knows]-{1,2} (Pb:Person)
-[:author]-> (m:Message)
WHERE m.creationDate < $day
RETURN DISTINCT m.id)
```

**Graph pattern matching language with visual graph syntax inspired by Cypher**

**GQL**

```
MATCH ANY ACYCLIC
(Pa:Person WHERE Pa.name = $name)
-[:knows]-{1,2} (Pb:Person)
-[:author]-> (m:Message)
WHERE m.creationDate < $day
RETURN DISTINCT m.id
```
Q13($src, $dst)

**SQL:1999**

WITH RECURSIVE ps(sp, ep, path, eR) AS (
    SELECT p1id AS sp, p2id AS ep, [p1id, p2id] AS path, (p2id = $dst) AS eR
    FROM knows WHERE sp = $src
    UNION ALL
    SELECT ps.sp AS sp, p2id AS ep, array_append(path, p2id) AS path, max(CASE WHEN p2id = $dst THEN 1 ELSE 0 END) OVER (ROWS BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING) AS eR
    FROM ps JOIN knows ON ps.ep = p1id WHERE NOT EXISTS (SELECT 1 FROM ps pps WHERE list_contains(pps.path, p2id)) AND ps.eR = 0)

SELECT min(length(path)) AS length FROM ps WHERE ep = $dst

**SQL/PGQ (SQL:2023)**

SELECT length FROM GRAPH_TABLE (sn
MATCH p = ANY SHORTEST
(Pa:Person WHERE Pa.name = $src)-[:knows]-*
(Pb:Person WHERE Pb.name = $dst)
COLUMNS (path_length(p) AS length))
LDBC working groups

**Graph schema:** Balancing expressive power, usability and tractability

- PG-Keys: Keys for Property Graphs (SIGMOD’21)
- PG-Schema: Schemas for Property Graphs (SIGMOD’23)

**Graph query languages:** Formalizing semantics, ensuring tractability

- G-CORE (SIGMOD’18)
- Graph Pattern Matching in GQL and SQL/PGQ (SIGMOD’23)
- GPC: A Pattern Calculus for Property Graphs (PODS’23)
LDBC organization
LDBC organization

LDBC is registered in the UK as a non-profit company

Annual membership fees (approx.):

- sponsors: 11,000 USD
- companies: 2,800 USD
- institutions: 1,400 USD

Approx. 100,000 USD per year revenue
Organizational structure

Old structure: member organizations delegate directors to the board

This made the company suspicious

- 100,000 USD per year revenue
- 20+ directors with different nationalities

→ we restructured
Organizational structure

**Voting Members** | individuals and organizations

- **Board of Directors (3–5)**
  - *Decides on Associate Member applications*

- **Members Policy Council (20+)**
  - *Decides on Voting Member applications*

**Associate Members**
- individuals

The membership form is 32 pages (patent declaration, CLA, etc.)
Summary
SNB Interactive v1

Financial Benchmark

Traversal with truncation

Strict latency bound (P99 < 100 ms)

SNB Business Intelligence

Graphalytics

Algorithms

BFS
PR
CDLP
SSSP
LCC
WCC

Data sets

DMC SNB
Graph500
Twitter
Friendster
Patents
wiki-Talk

Semantic Publishing Benchmark

Target: RDF/SPARQL

Domain: Media/publishing industry

Inferencing & continuous updates

SNB Interactive v2
LDBC’s main challenges

● Handling large-scale data sets is expensive:
  ○ Data generation – SF30k in AWS EMR – 120 instances, TODO hours, TODO USD
  ○ Data hosting
  ○ Transferring data

● Audits are complex
  ○ 8–20 weeks
  ○ Long and expensive

● Most audited results use imperative languages
  ○ GQL and SQL/PGQ may help

● Developing a benchmark takes 5+ person-years
  ○ No standard language, implementations take a long time
  ○ Hard to obtain a good baseline system