Benchmarking GraphDB with SNB & SPB

The experiences from passing SNB with a SPARQL engine and parallelizing SPB workloads at AWS

Tomas Kovatchev and Atanas Kiryakov
LDBC TUC Meeting, June 2023
Presentation Outline

- Social Network Benchmark
- Semantic Publishing Benchmark
Extending SNB compatibility to RDF & SPARQL

- Implementing SNB Interactive driver
  - No imperative language or stored procedures based query execution

- GraphDB Path-search extension used for traversal
  - Graph path search/traversal is very clumsy to implement in vanilla SPARQL
  - GraphDB’s Path search extension is compliant with SPARQL 1.1 syntax, unlike other triplestores

- Data loading with SNB Hadoop data generator
  - Audited dataset generator with no modifications to data model
SNB Interactive Challenges

- Complex query plans with multiple JOINs and OPTIONAL clauses required
- Numerous aggregation queries and path traversals
- Multiple-hop queries matched with joins of related metadata
- Frequent data update queries
- Combined complex analytical with lightweight throughput queries
Ontotext’s Approach

- Optimized All-path traversal memory utilization by leveraging our global entity pool
- Optimized Shortest-path traversal by implementing a greedy approach to iterate adjacency lists
- Used inference to materialize “shortcuts” in the graph

<table>
<thead>
<tr>
<th>Queries SF10</th>
<th>An LPG engine (ms)</th>
<th>GraphDB (ms)</th>
<th>AVG reads Δ base</th>
</tr>
</thead>
<tbody>
<tr>
<td>6: all path search</td>
<td>4,303.25</td>
<td>1,631.12</td>
<td>-62.1%</td>
</tr>
<tr>
<td>14: shortest path + weight</td>
<td>2,037.14</td>
<td>812.40</td>
<td>-60.1%</td>
</tr>
</tbody>
</table>
Optimizing query performance with inference

**Shortest Path Extract - Q11 without Inference**

```
SERVICE path:search {
  <urn:path> path:findPath onto:shortestPath ;
  path:sourceNode ?source ;
  path:destinationNode ?destination ;
  path:maxPathLength 2;
  path:startNode ?start ;
  path:endNode ?fr .
  SERVICE <urn:path> {
  }
}
```

**Shortest Path Extract - Q11 with Inference (25x times faster)**

```
SERVICE path:search {
  <urn:path> path:findPath onto:shortestPath ;
  path:sourceNode ?source ;
  path:destinationNode ?destination ;
  path:maxPathLength 2;
  path:startNode ?start ;
  path:endNode ?fr .
  SERVICE <urn:path> {
  }
}
```
GraphDB: The First RDF Engine to Pass SNB

Audited results:

- Scale factor 30 (SF30) – a graph of 1.5 billion edges
- Workload: Interactive (14 queries)
- Hardware: AWS r6id.8xlarge server (256GiB RAM, Intel Xeon 8375C)
- 12 ops./sec. on a driver configured with 4 read and 4 write threads
  - Linear scalability - the result with single agent is 3 ops./sec.

The first audited result for system with declarative query language!

Further reading:
SNB main page: https://ldbcouncil.org/benchmarks/snb/
Audited results are published at https://ldbcouncil.org/benchmarks/snb-interactive/
Presentation Outline

- Social Network Benchmark
- Semantic Publishing Benchmark
Semantic Publishing Benchmark

- Replicates BBC’s Dynamic **Semantic Publishing approach** through
  - BBC implemented this first for their FIFA World Cup website in 2010
  - Large volume of streaming content, e.g. creative works and media assets
  - Enriching content with metadata that describes it and links it to reference knowledge - information about entities: players, teams, groups, matches
  - Regular updates to the metadata and less often updates to the reference knowledge
  - Aggregation queries, that retrieve content according to various criteria

- Challenges multiple possible bottlenecks in engine performance (full scans etc.)

- Combines frequent updates with inference, geospatial constraints and FTS
GraphDB combines high-availability and scalability on SPB

Audited benchmarks runs:

- Two scale factors:
  - Scale factor 5 (SF5, **SPB 1B**) – a graph of 1.4B edges, after inference materialization
  - Scale factor 3 (**SPB 256M**) - a graph of 400M edges, after inference materialization
- Workload: **Aggregation agents** (12 queries) + **Editorial agents** (2 updates queries)
- Hardware: **AWS r6id.8xlarge** server (256GiB RAM, 32 vCPUs, Intel Xeon 8375C)
- Two configurations: **single server** and high-availability replication **cluster of 3 nodes**

**Further reading**: SPB main page with audited results: [https://ldbcouncil.org/benchmarks/spb/](https://ldbcouncil.org/benchmarks/spb/)
**Cloud-ready graph database within minutes**

- **Quick & easy to set up and replicate benchmark results**
  - Available helm chart and docker images both for single instance and clustered setups
  - No parameter tunings required to match performance from audited results

- **Scaling performance with cloud hardware**

<table>
<thead>
<tr>
<th>AWS Cloud Instance</th>
<th>Cost/hour</th>
<th>SPB 1B 16 read agents (QPS)</th>
<th>Read/Write Agents</th>
<th>SPB 256M R/W Ops</th>
<th>SPB 1B R/W Ops</th>
</tr>
</thead>
<tbody>
<tr>
<td>i4i.4xlarge</td>
<td>$0.89</td>
<td>72</td>
<td>0/4</td>
<td>0/38</td>
<td>0/17</td>
</tr>
<tr>
<td>m6id.8xlarge</td>
<td>$1.20</td>
<td>104</td>
<td>8/4</td>
<td>217/31</td>
<td>69/13</td>
</tr>
<tr>
<td>r6id.8xlarge</td>
<td>$1.52</td>
<td>130</td>
<td>16/4</td>
<td>335/26</td>
<td>106/10</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>24/0</td>
<td>413/0</td>
<td>158/0</td>
</tr>
</tbody>
</table>

**Notes on the table:**
- Unaudited data in the table above
- GraphDB Single instance with 80 GiB heap
- 1Yr Reserved as of June 2023

Hardware: r6id.8xlarge
Enterprise grade graph database within minutes

Scaling throughput with # of concurrent users in a cluster

<table>
<thead>
<tr>
<th>Read/Write agents</th>
<th>SPB 256M QPS (Queries per Second)</th>
<th>SPB 1B QPS (Queries per Second)</th>
</tr>
</thead>
<tbody>
<tr>
<td>16/0</td>
<td>467</td>
<td>181</td>
</tr>
<tr>
<td>32/0</td>
<td>755</td>
<td>305</td>
</tr>
<tr>
<td>64/0</td>
<td>986</td>
<td>409</td>
</tr>
</tbody>
</table>

* GraphDB 3-node high-availability cluster, r6id.8xlarge instances with 32 vCPUs

- Serving ~1000 QPS to 64 clients with high availability cluster
- Effective query load balancing across the nodes in the cluster
- Sublinear query performance for growing datasets
Summary of Results & Findings

GraphDB is versatile and capable to handle diverse workloads:
- Transaction and analytical workloads
- Graph-analytics, logical reasoning, FTS, geo-spatial … all at once
- High-availability and vertical scalability

Social Network Benchmark (SNB):
- **GraphDB is the first RDF engine to pass graph analytics-heavy benchmark**
- **The first audited results for system with declarative query language (not C++)**

Semantic Publishing Benchmark (SPB):
- **Read throughput scales well given stronger AWS instance**
- **Great horizontal scalability in a cluster: over 1000 QPS!**
Thank you!