The LDBC SNB Implementation in TuGraph

HENG LIN
Ant Group
6/23/2023
TuGraph Overview

TuGraph-DB
• Performance oriented graph database on single machine
• Full ACID support with serializable transactions
• Integrated with query / analytics / learning
• OpenCypher API (IOS GQL 2023.9)
• Stored procedure with C++/Python/Rust API
• Open source from 2022.9

--- Don Chamberlin, 49 Years of Queries, SIGMOD2023
Design

Query Impl.
- Materialized views
- Intra-query parallelism

Transactions
- Lock free read txn
- Multi-version B+ tree

Topology
- Adaptive continuous packing
- Prop. value based sorting

Properties
- Compact layout
- Compress with fixed and variable length data
Our observations:

1. More read than write (a.k.a 10:1 in LDBC SNB)
2. Access locality of certain vertex’s edges
3. Power-law distribution
## Adaptive Continuous Packing (2)

<table>
<thead>
<tr>
<th>Observations</th>
<th>Techniques</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. R/W = 10:1</td>
<td>Repack data while writing</td>
<td>A bit slow write but faster read</td>
</tr>
<tr>
<td>2. Access locality</td>
<td>Continuous pack edges in a block</td>
<td>Faster scan</td>
</tr>
<tr>
<td>3. Power-law</td>
<td>One block for small vertex, which more for hub</td>
<td>Avoid slow down for hub vertex</td>
</tr>
</tbody>
</table>

![Diagram of vertex and edges](image-url)
Adaptive Continuous Packing (3)

<table>
<thead>
<tr>
<th>Observations</th>
<th>Techniques</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. R/W = 10:1</td>
<td>Repack data while writing</td>
<td>A bit slow write but faster read</td>
</tr>
<tr>
<td>2. Access locality</td>
<td>Continuous pack edges in a block</td>
<td>Faster scan</td>
</tr>
<tr>
<td>3. Power-law</td>
<td>One block for small vertex, which more for hub</td>
<td>Avoid slow down for hub vertex</td>
</tr>
</tbody>
</table>

![Diagram](image)
### Observations

<table>
<thead>
<tr>
<th>Observations</th>
<th>Techniques</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. R/W = 10:1</td>
<td>Repack data while writing</td>
<td>A bit slow write but faster read</td>
</tr>
<tr>
<td>2. Access locality</td>
<td>Continuous pack edges in a block</td>
<td>Faster scan</td>
</tr>
<tr>
<td>3. Power-law</td>
<td>One block for small vertex, which more for hub</td>
<td>Avoid slow down for hub vertex</td>
</tr>
</tbody>
</table>

![Sorted edges by timestamp/label](image-url)
## Adaptive Continuous Packing (5)

<table>
<thead>
<tr>
<th>Observations</th>
<th>Techniques</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. R/W = 10:1</td>
<td>Repack data while writing</td>
<td>A bit slow write but faster read</td>
</tr>
<tr>
<td>2. Access locality</td>
<td>Continuous pack edges in a block</td>
<td>Faster scan</td>
</tr>
<tr>
<td>3. Power-law</td>
<td>One block for small vertex, which more for hub</td>
<td>Avoid slow down for hub vertex</td>
</tr>
</tbody>
</table>

![Diagram showing small vertices and hub vertices with their packing strategies](image-url)
Compact Layout

During traversal, one or more properties of vertices or edges are accessed.

--> Choose Compact Layout
LDBC SNB Interactive Audit

TuGraph ranks No.1 in the overall throughput.

LDBC SNB Interactive
- Formulated by LDBC
- Simulate social network scenario
- Workload including 29 R/W queries
- Transaction/Correctness/Throughput

SNB SF100 QPS, Higher is better
Roadmap

TuGraph-DB Repo:  https://github.com/TuGraph-family/tugraph-db

TuGraph-Analytics Repo:  https://github.com/TuGraph-family/tugraph-analytics

Free Trail:  https://tugraph-db.readthedocs.io/en/latest/5.developer-manual/1.installation/1.cloud-deployment.html
Thank You