Computation of Large Sparse Aggregated Areas for Analytic Database Queries

Steffen Wittmer • Tobias Lauer
Jedox AG

Collaborators: Zurab Khadikov • Alexander Haberstroh • Peter Strohm
Business Intelligence and Corporate Planning
Jedox BI Workflow

Data Integration
- Extract
- Transform
- Load

Jedox ETL
- Extract
- Transform
- Load

OLAP Calculation
- Aggregation
- Enterprise rules
- Write-back

Jedox OLAP
- Aggregation
- Enterprise rules
- Write-back

User Frontend
- Reporting
- Analysis
- Planning
- Simulation

Jedox for Excel
Jedox Web
Jedox Mobile

Database
ERP
Text Files

Data Integration
Extract
Transform
Load

Jedox ETL
Extract
Transform
Load

Jedox OLAP
Aggregation
Enterprise rules
Write-back

User Frontend
Reporting
Analysis
Planning
Simulation

Jedox for Excel
Jedox Web
Jedox Mobile

jedox.
Online Analytical Processing (OLAP)

• Data modeled as multidimensional “cube”
  – Operations:
    • Analysis
    • Reporting
    • Planning
    • Simulation

  – Dimensions are structured hierarchically:
    • Consolidated elements
    • Base elements
In-memory OLAP storage model

- All data stored in main memory
In-memory OLAP storage model

- All data stored in main memory
- Only store base cells
In-memory OLAP storage model

- All data stored in main memory
- Only store base cells
- Do not store zero-value cells

→ Memory saving, data consistency

Represent cells as (key, value) pairs, e.g.

(2, 1, 0), 4.0

Note: Values are double precision!
In-memory OLAP storage model

- All data stored in **GPU** memory
- Only store base cells
- Do not store zero-value cells

→ Memory saving, data consistency

- Compute other cells *on the fly* when needed

→ Use GPU to accelerate
GPU aggregation solutions

Target driven aggregation
- Parallel reductions
- Very fast for small/dense target areas

Source driven aggregation
- Exploits OLAP characteristic of sparsity
- Well suited for large aggregated areas
Target driven aggregation

- Fast parallel aggregation step
- Utilizes shared, global and constant memory
- Coalesced memory access
- Almost no thread divergence

Multi-GPU solution
Performance optimized

bulk aggregations
prefiltering
Large sparse aggregated areas

- Large: up to millions of target cells
  
  Top 10 products for each customer

- Sparse: most target values are zero
  
  □ = zero
  ■ = non-zero
Handling large sparse areas

• Requirements
  Performance and memory consumption that correlate with number of non-zero target cells

• Solution
  Source driven approach
  - Serialized aggregation with atomics
  - Utilize hash tables on GPU
Source driven aggregation

Jan, 2011 sold units

source cells

Q1, all years sold units

target cell area

Year, all years sold units

parent map

Jan, Q1, Year, Q2, Year

atomic add

h(x)

target cell hash table

atomic add
Atomics: Contention

thread serialization

atomic add

Great improvement in Fermi and Kepler over CC 1.1
Reducing contention

-warp preaggregation

_ballot: merge with first?

warp-wise different hash functions

h₁(x)  h₂(x)
Speedup: Small target areas (1-11 cells)

GPU Target Driven

GPU Source Driven

Database: 1B records (filled cube cells)
GPU: 3x Tesla C2070 (18 GB RAM)
Larger areas

Calculation times

<table>
<thead>
<tr>
<th>Target Cells</th>
<th>CPU Time (seconds)</th>
<th>GPU Time (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1547</td>
<td>15.6</td>
<td>3.2</td>
</tr>
<tr>
<td>38012</td>
<td>35.0</td>
<td>3.6</td>
</tr>
</tbody>
</table>

Speedup

<table>
<thead>
<tr>
<th>Target Cells</th>
<th>Speedup over CPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>1547</td>
<td>4.9</td>
</tr>
<tr>
<td>38012</td>
<td>9.7</td>
</tr>
</tbody>
</table>

Database: 40M records (filled cube cells)
GPU: 2x Tesla K20 (10 GB RAM)
Comparison: aggregation algorithms

- CPU algorithm good for low aggregation
- Target driven algorithm: good for small and/or dense target areas
- Source driven algorithm: optimized for large and sparse areas
Future challenges

• Decision mechanism
  – When GPU, when CPU?
  – Which GPU algorithm in which situation?

• Find suitable thresholds

• What about large and dense target areas?
  → GPU memory problem
Come visit us!

Exhibit Hall: Booth 718

Online: www.jedox.com

tobias.lauer@jedox.com
steffen.wittmer@jedox.com