

Computation Reuse in Analytics Job Service at Microsoft

<u>Alekh Jindal</u>, Shi Qiao, Hiren Patel, Zhicheng Yin, Jieming Di, Malay Bag, Marc Friedman, Yifung Lin, Konstantinos Karanasos, Sriram Rao

Microsoft



Materialized Views Reinvented Yet Again

<u>Alekh Jindal</u>, Shi Qiao, Hiren Patel, Zhicheng Yin, Jieming Di, Malay Bag, Marc Friedman, Yifung Lin, Konstantinos Karanasos, Sriram Rao

Microsoft



A brief history of Views



Typical Materialized View Assumptions:

- Tuning <u>few databases</u>
- Relatively static data with <u>some updates</u>
- Views materialized <u>a priori and offline</u>
- <u>Accurate estimates</u> of utility/cost of view materialization



What's new: Analytics-as-a-Service!

Also, Job Service or Serverless Analytics:

- Not require users to manage h/w or s/w
- Only provide SQL queries over stored data
- Service provider takes care of the execution
- Users only pay for the processing cost

SCOPE Job Service at Microsoft:

- ~10⁵ number of machines
- ~10⁵ number of analytical jobs
- ~10³ developers across Microsoft
- ~EBs data processed per day





Reassigning Passengers to Planes in Mid-Air





CloudViews Overview





CloudViews Overview





Recurring Workloads





Recurring Workloads

- Periodic queries with different inputs and parameters
- Structured/unstructured data; custom user code





Reuse over Recurring Workloads

- Problem: detect/reuse common subexpressions when new data arrives in each recurring interval
- Solution: *precise/normalized* query signatures





Metadata Service





Metadata Service

- Materialized view lookup
- Consistent view materialization
- Quick view discovery



Query Rewriting / Online Materialization

Microsoft





Query Rewriting / Online Materialization





Analyzing Production Workloads

- Cluster-wide overlaps:
 - 45% jobs
 - 65% users
 - 80% subgraphs
- Operator-wise overlaps:
 - Up to 1000s of overlaps







Performance Impact

- Workload: 32 queries
- Latency:
 - Improvements depend on the critical path
 - Some queries slower due to materialization
- Processing time:
 - Additional processing time for read/write
 - Savings in general
- Overheads:
 - Workload analysis in an hour
 - ~10ms metadata service lookup
 - Optimization time higher/lower when creating/using views





Lessons Learned

- Discovering hidden redundancies, static computations
- Important to get the view physical design right in big data systems
- Interesting side effects: failure recovery, cost estimates
- User expectations: automatic, debuggability, privacy regulations
- Even classic database concepts take a lot of time to bake in industry
- Challenge: some of the assumptions may not hold
- Industrial research is fun! \bigcirc



Thanks!

See you at: Poster Session 1, Wednesday 16:00-18:00, Houston 567

Coming up:

Selecting Subexpressions to Materialize at Datacenter Scale Alekh Jindal, Konstantinos Karanasos, Sriram Rao, Hiren Patel VLDB 2018/PVLDB, Rio de Janeiro, Brazil

Computation Reuse in Analytics Job Service at Microsoft Tue, 11-12:30

Alekh Jindal (Microsoft), Shi Qiao (Microsoft), Hiren Patel (Microsoft), Zhicheng Yin (Microsoft), Jieming Di (Microsoft), Malay Bag (Microsoft), Marc Friedman (Microsoft), Yifung Lin (Microsoft), Konstantinos Karanasos (Microsoft), Sriram Rao (Microsoft)

• What do we mean by computation reuse?

S

uestion

- What is a "job service"? How is it different from "databases"?
- How does a job service look like at Microsoft?
- Why is computation reuse challenging in a job service?
- What is our solution, key insights, and takeaways?



Key Ingredients

Materialized views over recurring workloads CloudViews Analyzer ✓ Feedback Loop ✓ View Selection ✓ Physical Design ✓ View Expiry ✓ CloudViews Runtime ✓ Metadata Service Online Materialization ✓ Query Rewriting ✓ Synchronization Job Coordination