Reporting Technologies

Static and Dynamic Reporting

Michael Nissen
michaeln@diku.dk

Department of Computer Science, University of Copenhagen

Nov 18. 2008

1. Reporting

2. Technologies of Today
   - Materialized Views
   - OLAP
   - SIFT
   - Summary

3. Technologies of Tomorrow?
   - FunSETL
   - Map-Reduce

4. Summary

What is Reporting?

Definition (Report Function)

A *report function* is a function on transactional data.

Reporting is the discipline of

- *Applying* report functions, that is, executing their specification on actual data.
- *Expressing* report functions, that is, describe them in a specification- or programming language.

Note: Presentation of results is NOT included in the definition.

Static and Dynamic Report Functions

Concept (Static and Dynamic Report Functions)

- *A Static Report Function is a report function, which we know in advance that we want to compute at some point.*
- *A Dynamic Report Function is a report function, which we do NOT know in advance that we want to compute at some point.*

Reporting Today

- Report functions are usually expressed using fx. SQL, OLAP, SIFT (Microsoft NAV) or in a general purpose programming language (for instance, X++ or C/AL).
- ERP systems contain a lot of data.
- ERP systems primarily accumulate data.
- Many report functions are conceptually simple.
- Many report functions are computed from scratch.

What are the problems and what do we want?

- Computing report functions is time consuming.
- Expressing report functions can be hard in the existing specification- and programming languages.
- Real-time or near-real-time (dash-boarding) computations of report functions are preferable.
- The responsibility of efficient computation of report functions should be moved away from the developer.

What are the problems and what do we want?

- Computing report functions is time consuming.
- Expressing report functions can be hard in the existing specification- and programming languages.
- Real-time or near-real-time (dash-boarding) computations of report functions are preferable.
- The responsibility of efficient computation of report functions should be moved away from the developer.

Realized Technologies

- Materialized Views
- OLAP
- SIFT (Microsoft NAV)
- Google’s Map-Reduce.
- FunSETL

Realized Technologies

- Materialized Views
- OLAP
- SIFT (Microsoft NAV)
- Google’s Map-Reduce.
- FunSETL

Materialized Views

- *What?:* Storage of virtual relations.
- *Why?:* Faster access to virtual relations.

Bicycle Business - Example

Branch	Color	Time_Id	Price
Valby	Red	T1	1599
Frederiksberg	Red	T2	1799
Valby	Red	T3	1399
Frederiksberg	Blue	T4	2199
Valby	Red	T5	1299
Frederiksberg	Blue	T6	1299
Frederiksberg	Blue	T7	2399

Materialized Views - Example

Example

Declare a view *totalsales* that holds the sum of the sales for each branch.

```
create view totalsales(branch, amount) as
select Branch, sum(Price)
from sale
group by Branch
```

branch	amount
Frederiksberg	7696
Valby	4297

Materialized Views - Issues

- View Maintenance.
  - How should a materialized view be updated when the data it depends on is changed?
  - The example view can be updated incrementally.
- Purging unused views.
- Can in some cases be used to do real-time report function computation:
  - A materialized view can be declared to maintain results needed by a static report function.
  - We can get lucky and use a materialized view in the computation of a dynamic report function.

OLAP - OnLine Analytical Processing

- *What?*: Special kind of materialized views. (Union of GROUP BY SQL statements).
- *Why?*: Speedup computation time of queries that benefit from these kind of views.

OLAP - Issues

- OLAP cube relations can be as big (or even bigger) than the source tables they stem from.
- Updating OLAP cubes has the same problems as Materialized Views.
- Can in some cases be used to do real-time report function computation:
  - An OLAP cube can be declared to maintain results needed by a static report function.
  - We can get lucky and use an OLAP cube in the computation of a dynamic report function.

SIFT

- *What?*: Virtual fields on existing tables containing aggregate information.
- *Why?*: To speedup the computation of report functions.

SIFT - Issues

- Updating FlowFields.
- Purging unused FlowFields.
- Some static report functions can be computed in real-time using FlowFields.

Summary

The technologies presented so far:

- Some static report functions can benefit from these technologies.
- Can maintain unnecessary information, which however gives some possibility of dynamic report function computation.
- Unclear when real-time computation can be performed (the developers responsibility to identify this).

Technologies of Tomorrow?

- Why only use Relational Database Technologies?
- Relational databases do not have a distinction of static and dynamic queries.
- Generally low support for real-time computation.

FunSETL

- Declarative specification of report functions.
- Automatic transformation to incremental specification (often real-time).
- Asymptotic improvement in many cases.
- Only maintaining the necessary information.
- Suited for static report functions.

Map-Reduce

- *What?*: C++ library.
- *Why?*: Automatic parallelization of computations.
- *How?*: Execute on many low price machines.

Map-Reduce - Example

Example

Compute the total number of bicycles sold of each color.

map and **reduce** functions declared as (written in pseudo code).

```
map (String branch, String color) :
    EmitIntermediate(color, 1);

reduce (String color, Iterator values) :
    int result = 0;
    foreach v in values :
        result += v;
    Emit(result);
```

Map-Reduce Comments

- Current Map-Reduce not suited for real-time computation (maybe it can be adapted).
- Suited for dynamic report functions.
- Removes responsibility of efficient computation away from the developer.

Summary

Relational Databases, Materialized Views, OLAP and SIFT does not provide good support for

- Real-time or near-real-time computation of report functions.

Idea

Split the specification of report functions in two classes:

- *Dynamic: Specification that guarantees parallelization of the computation.*
- *Static: Specification that guarantees that the results are maintained (incrementally) in real-time or near-real-time.*

OLAP - Example - Query

Example

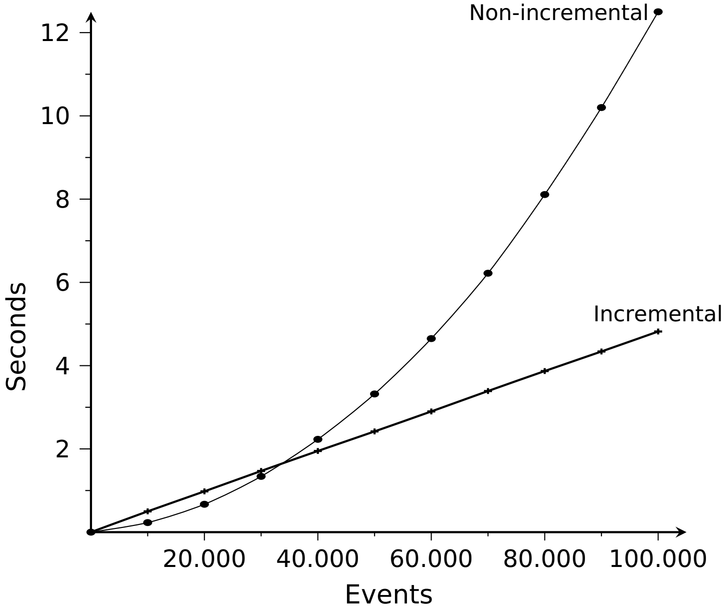
OLAP Cube with Color and Quarter and aggregate *Sum*.

select *sale*.*Color*, *time*.*Quarter*, **sum**(*sale*.*Price*)
from *sale*, *time*
where *sale*.*Time_id* = *time*.*Time_id*
group by cube(*sale*.*Color*, *time*.*Quarter*)

OLAP - Example - Result

Color	Quarter	sum(Price)
Red	1	4797
Blue	1	2199
Red	2	1299
Blue	2	3698
Blue	-	5897
Red	-	6096
-	1	6996
-	2	4997
-	-	11993

FunSETL - Financial Statement


Non-incremental
Incremental
Seconds
Events