18/01/2023

Mohana Fathollahi
Marcel Kuhlmann

Data Warehouse Project – Part 3 – Query Optimization

First Query:

```
SELECT t.monthid AS Month, SUM(a.FlightHours) AS FH, SUM(a.FlightCycles) AS FC
FROM AircraftUtilization a, TemporalDimension t,AircraftDimension d
WHERE t.ID = a.timeid AND d.id=a.aircraftid AND d.model='777'
GROUP BY t.monthid;
```

****cost:169 and extra 32 blocks for indexes

Second Query:

```
SELECT m.y AS Year, SUM(a.scheduledoutofservice) AS ADOSS,
SUM(a.unscheduledoutofservice)
AS ADOSU
FROM AircraftUtilization a, TemporalDimension t, Months m
WHERE a.timeid=t.id AND t.monthid=m.id AND a.aircraftid ='XY-WTR'
GROUP BY m.y;
```

****cost: 112, 24 blocks for indexes

To answer queries we used bitmap join index because in some cases it is more efficient than other algorithms. For example in a pipeline we need to bring a fact table in memory and it is a very expensive operation.

On the other side join index is very similar to row nested loop but with some improvements. first: it does not bring external table in memory, B_R . Second: instead of going through the indexes in the internal table for the whole row of the external table we check indexes for constant values that are desired in the query. Therefore, we can reach a more efficient execution plan.

In this algorithm the value exists in the dimension table, AircraftDimension, and the address exists in the fact table, AircraftUtilization. Additionally, by defining bitmap over aircraftid in AircraftUtilization we facilitate searching for addresses.

To find which indexes can be used in AircraftUtilization we should consider that Btree can be used on attributes with low selectivity factor like primary key, while Bitmap can be used on attributes with approximately 100 repetitions. Therefore, in this case bitmap is a better option.

```
ALTER TABLE AircraftDimension ADD PRIMARY KEY (id) USING INDEX PCTFREE 33;
ALTER TABLE PeopleDimension ADD PRIMARY KEY (id) USING INDEX PCTFREE 33;
CREATE BITMAP INDEX a_model ON AircraftUtilization(AircraftDimension.model)
FROM AircraftUtilization, AircraftDimension WHERE AircraftDimension.id =
AircraftUtilization.aircraftid PCTFREE 0;
CREATE BITMAP INDEX a_aircraft ON AircraftUtilization(aircraftid) PCTFREE 0;
```

Third Query:

```
SELECT I.month AS Month,
1000*SUM(I.pirep+I.marep)/SUM(a.FH) AS RRh, 100*SUM(I.pirep+I.marep)/SUM(a.FC) AS
RRc, 1000*SUM(I.pirep)/SUM(a.FH) AS PRRh, 1000*SUM(I.marep)/SUM(a.FH) AS MRRh,
100*SUM(I.pirep)/SUM(a.FC) AS PRRc, 100*SUM(I.marep)/SUM(a.FC) AS MRRc
FROM (SELECT I.aircraftid AS aircraft, I.monthid AS Month,
SUM(CASE WHEN p.ROLE = 'P' THEN counter ELSE 0 END) AS PIREP,
SUM(CASE WHEN p.ROLE = 'M' THEN counter ELSE 0 END) AS MAREP
FROM LogBookReporting l
INNER JOIN PeopleDimension p ON l.personid = p.id
INNER JOIN AircraftDimension d ON l.aircraftid = d.id
WHERE d.model = '777'
GROUP BY l.aircraftid, l.monthid) l
INNER JOIN (
SELECT a.aircraftid AS aircraft, t.monthid AS Month,
SUM(a.FlightHours) AS FH, SUM(a.FlightCycles) AS FC
FROM AircraftUtilization a
INNER JOIN TemporalDimension t ON a.timeid = t.id
GROUP BY a.aircraftid, t.monthid
) a ON l.aircraft = a.aircraft AND l.month = a.month
GROUP BY l.month;
----COST: 2261,
-----COST: 1312, AFTER INDEX OF L_MODEL(16)
-----COST: 1312, AFTER INDEX OF L_AIRPORT(16)
```

Fourth query:

```
SELECT d.model AS Model,
1000*SUM(l.marep)/SUM(a.FH) AS MRRh, 100*SUM(l.marep)/SUM(a.FC) AS MRRc
FROM (SELECT l.AircraftID AS aircraft, l.monthID AS Month, p.airport AS airport,
SUM(CASE WHEN p.ROLE = 'M' THEN counter ELSE 0 END) AS MAREP
FROM LogBookReporting l
INNER JOIN PeopleDimension p ON l.personid = p.id
WHERE p.airport = 'KRS'
GROUP BY l.aircraftID, l.monthID, p.airport) l
INNER JOIN (SELECT a.aircraftid AS aircraft, t.monthid AS Month,
SUM(a.FlightHours) AS FH, SUM(a.FlightCycles) AS FC
FROM AircraftUtilization a
INNER JOIN TemporalDimension t ON a.timeid = t.id
GROUP BY a.aircraftid, t.monthid) a ON l.aircraft = a.aircraft AND l.month = a.month
INNER JOIN AircraftDimension d ON l.aircraft = d.id
GROUP BY d.model;
---COST: 1764  INDEX L-MODEL
---COST:798 INDEX L-AIRPORT AND L-MODEL
```

To answer the last two queries we used the same strategy that described before. But in this part we need to consider two dimensions, AircraftDimension and PeopleDimension, we defined two bitmap join indexes that keys exist in these two dimensions and address exist in LogbookReporting.

```
CREATE BITMAP INDEX l_model ON LogbookReporting(AircraftDimension.model) FROM
LogbookReporting, AircraftDimension WHERE AircraftDimension.id =
LogbookReporting.aircraftid PCTFREE 0;
CREATE BITMAP INDEX l_airport ON LogbookReporting(PeopleDimension.airport) FROM
LogbookReporting, PeopleDimension WHERE
PeopleDimension.id=LogbookReporting.personid PCTFREE 0;
```