Query Optimization for Excel People

Mara Lemagie
Manager and Lead Data Engineer at Bazean Corp.

WHO IS BAZEAN

Bazean is a technology-enabled energy firm that builds and utilizes data and analysis to drive significant investment decisions in energy

Holistic Approach

Sub-surface + Surface
Data + Insights + Machine Learning

Technology

Open Source | Linux vs.
Closed Ecosystems | Microsoft

Team

Deep Energy and Oil & Gas + Technology

MOTIVATION

Realizing that you have (or someone you know has) more data than Excel can handle

- The sales team/admin/physical security/analyst walks in with an Excel workbook (or several!) that holds the keys to the kingdom
- A key process is slowed because multiple people need to modify the same file and versions are difficult to maintain
- Capturing the time a row was edited can be difficult in Excel
- Sometimes Excel IS the best solution, but it is an important skill to be able to identify when this is and is not the case
- Row limits

GOALS FOR THIS TALK

Whether you are an SQL expert looking to onboard or new to SQL

- Why it is important and useful to be able to talk about database technologies
- Provide an over-arching framework for optimizing queries, and helping others start with good SQL writing habits
- How to relate SQL to Excel
  - vlookup is to a subselect what index-match is to a join
- How to bring expertise that you develop in one domain with you as you learn new skills
- Like all good rules, there will be exceptions

EXCEL PEOPLE

Who are they

- Data entry
- Analysts/Finance
- Managers
- Board members
- Not necessarily programmers

PROMOTING TECHNICAL LITERACY

How this benefits everyone

- Convince stakeholders to invest in larger projects
- Help others understand the news and world around us
- Make recruitment easier
  - More good candidates overall
  - More confidence in being able to train new hires
- Increase the richness of the data available
- Build trust

ANALYST'S MENTAL MODEL

The reasons Excel is popular

- Excel is very visual
- Because of how Excel is built, it forces performance considerations very early in the process
- This is where people who are used to using Excel come from

EXCEL ANALYST'S PROCESS

Standard Workflow

- Pull the needed data into a workbook
- Filter out the data that doesn't meet criteria (maybe join data in from another tab)
- Put data into a pivot table or some other aggregation function (countif, sumif)
- Make another pass with the filter as needed
- Pull out the relevant columns
- Sort the results

SQL OPTIMIZER'S PROCESS

- Verify syntax
- FROM clause
- WHERE clause
- GROUP BY clause
- HAVING clause
- SELECT clause
- ORDER BY clause

SIDE-BY-SIDE

The Excel analyst and the SQL optimizer

- VERIFY SYNTAX
- Pull the needed data into a workbook (FROM)
- Filter out the data that doesn't meet criteria (WHERE)
- Put data into a pivot table or some other aggregation function (GROUP BY)
- Make another pass with the filter as needed (HAVING)
- Pull out the relevant columns (SELECT)
- Sort the results (ORDER)

THE DIAMOND RULE FOR FAST SQL

- The faster you trim down the data set size, the faster your query will run
  - If an Excel file is slow because there are 10 million rows in the first tab, the first thing an analyst will do is make the data smaller
- One of the largest differences between SQL and Excel is that in Excel you have to physically delete the data to get a performance benefit

SQL OPTIMIZER'S PROCESS - OPTIMIZED

Smaller, faster (sooner)

- VERIFY SYNTAX
- Pull the needed data into a workbook (FROM, JOINS)
- Filter out the data that doesn't meet criteria (WHERE)
- Put data into a pivot table or some other aggregation function (GROUP BY)
- Make another pass with the filter as needed (HAVING)
- Pull out the relevant columns (SELECT)
- Sort the results (ORDER)

CHECK IN

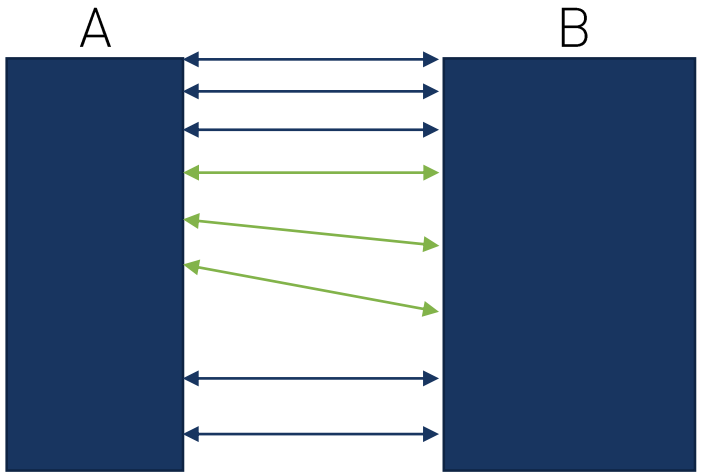
Important points

- Our mental model for how we would manually parse through data in Excel is not too different than how a SQL optimizer would work
  - In the case of the 10-million row workbook, if you had to handle it, you would consider splitting into several workbooks (partitions)
  - Using a join to filter your data can be much faster than the where clause because of its order in the SQL execution plan – you are reducing your overall data size sooner
- Now that we've merged our mental model for Excel into SQL, we will switch to talking about how to write faster SQL queries, leveraging Excel experiences

REMINDER

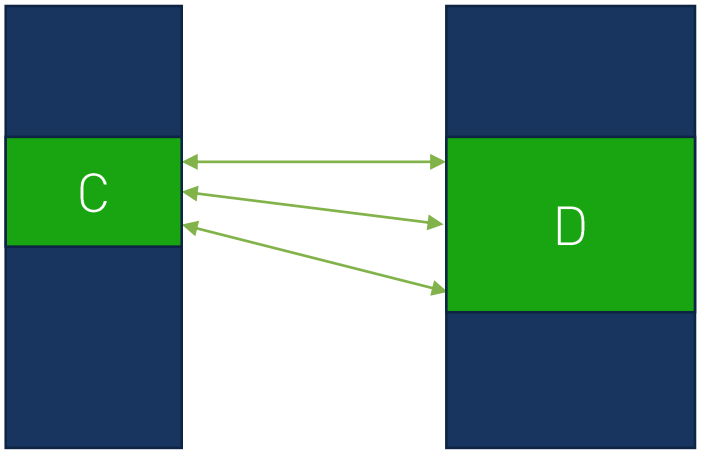
- The faster you trim down the data set size, the faster your query will run
- This is especially important when your data can go from not in memory to within memory
- Cross joins and extra loops are good counter examples to this rule
- Even at the end, if your query returns 100k rows of data instead of a single row, you'll get a performance benefit

MAKE IT SMALLER FASTER




```
SELECT * FROM
  A JOIN B
ON A.id = B.id
WHERE A.j < 6
AND B.k  <6
```

Vs.

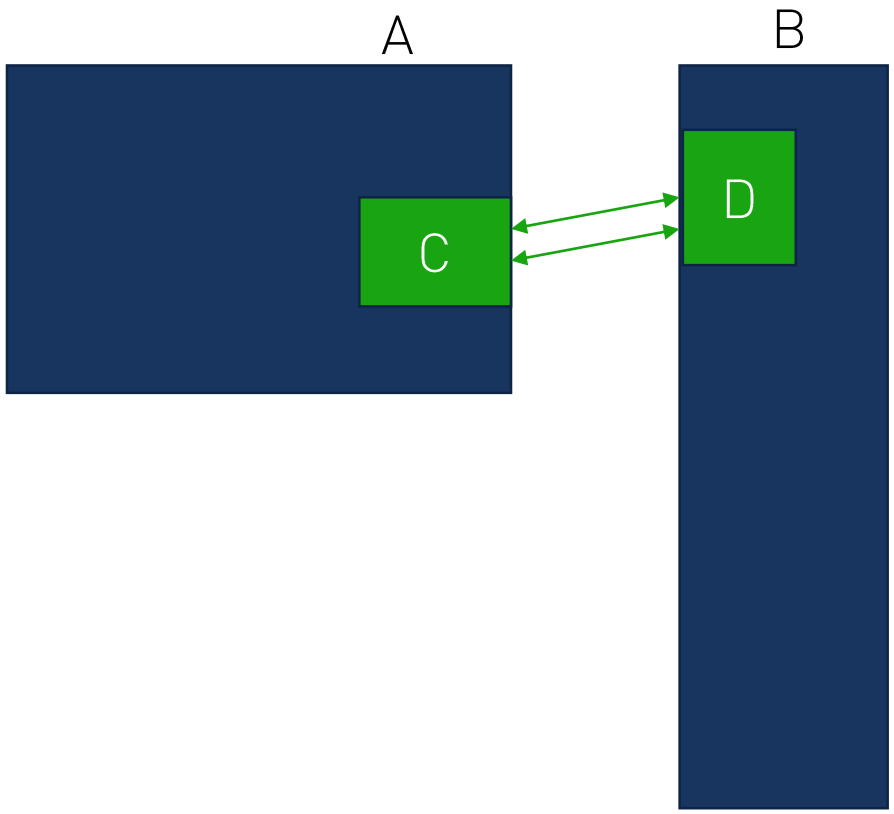



```
SELECT * FROM
  (SELECT * FROM A WHERE A.j <6) C
JOIN
  (SELECT * FROM B WHERE B.k < 6) D
ON C.id = D.id
```

IMPORTANT CAVEATS

- Filtering on the join column won't help – use the join!
- This is not a definitive guide to performance improvements, especially as optimizers have started to catch up with our habits

EVEN SMALLER




```
SELECT * FROM
  (SELECT id, j FROM A WHERE A.j <6) C
JOIN
  (SELECT id, k FROM B WHERE B.k < 6) D
ON C.id = D.id
```

GOING BACK TO EXCEL

- This rule helps write efficient queries as data sizes increase
- Learning to write SQL in the first place: start with determining what data you need (FROM), and work through each of the steps
- Don't discount Excel intuition

THANKS FOR YOUR TIME TODAY!

- Questions?
- Please email me at mlemagie@bazean.com
  - Or find me on LinkedIn!

APPENDIX

SUBQUERY VS. SUBSELECT

- vlookup is to a subselect what index-match is to a join
  - Select * from A where id in (select id from B) (subselect)
- Subquery is where either A or B becomes their own query
  - Select * from (select id from A where x >2) a join B on a.id = B.id (a is a subquery)
- Don't overcomplicate!

INDEXES

- For purposes of helping locate the physical location on disk, they are an addition to the diamond rule
- BUT, for the purposes of helping write faster queries, indexes can be thought of as reductions in the overall footprint of a given data set
  - The more indexes reduce the size of the data, the more they help improve performance

INDEXES VISUALIZED