Marp

Introduction to Query Engines

Scope

Just the **Data Query Language** aspects - that's more or less the bit that handles `SELECT` statements.

Will cover generic aspects of implementation, but will include detail relating to Opteryx.

Query Language Interpretation

Query Planning and Optimization

Execution Engine

Files / Storage

SQL

Query Language Interpretation

Abstract Syntax Tree

Query Planning and Optimization

Query Plan

Execution Engine

Resource Access

Files / Storage

Key Components

Parser / Lexer Interprets SQL into a semantic representation (AST)
Abstract Syntax Tree (AST) First machine processable representation of the query (we can rewrite the query here)
Query Plan Describes the steps to take to fulfil the request
Optimizer Reworks the Query Plan to improve performance
Executor Runs the Query Plan

Planner Steps

Reader
Selection
Projection
Join
Distinct

Fixed Query Plan

The order items are processed before optimizations.
Has implications, e.g. can’t GROUP BY aliases in the SELECT clause.
Optimizations have to create the same result.

FROM
JOIN
WHERE
GROUP BY
HAVING
SELECT
DISTINCT
ORDER BY

Plan Optimization

Get rid of data (rows and columns) as quickly as possible
Selection and Projection Push-Downs
Selection to eliminate records quickly
Use HASH or SORT MERGE JOINS

Execution Models

Row Processing (Volcano) Mabel
Block Processing Opteryx
Column Processing (Vectorized) Opteryx