Tuning Tableau and Your Database for Great Performance

PRESENTED BY

Matt Higgins, Tableau Software
Robert Morton, Tableau Software

Tuning Tableau and Your Database for Great Performance

- Understand Tableau's Query Generation
- Identify Database Tuning Opportunities
- Author Optimized Visualizations

Agenda: Technical* Guidelines for Performance

MARKETING

* Occasional marketing slides required

Understand Tableau's Query Generation

Motivation: "Direct connect & go!"

- No up-front, complex ETL
- Push the computation close to the data
- Leverage customer investment in fast databases

Topics covered:

- Filters
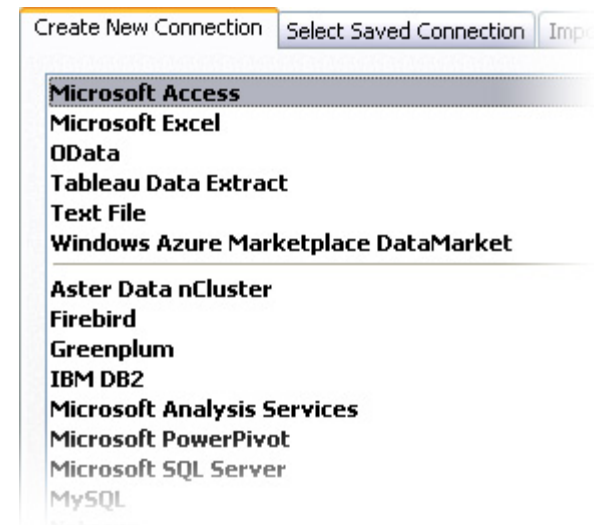- Subqueries and temporary tables
- Custom SQL
- Discovery queries

Create New Connection | Select Saved Connection | Imp

Microsoft Access
Microsoft Excel
OData
Tableau Data Extract
Text File
Windows Azure Marketplace DataMarket

Aster Data nCluster
Firebird
Greenplum
IBM DB2
Microsoft Analysis Services
Microsoft PowerPivot
Microsoft SQL Server
MySQL

Tableau Architecture

Databases Data Marts Files Cubes

Tableau Native Query Technology

+ Interactive queries to the database
+ Leverages IT policies for security
+ Avoids data silos
+ Ensures fresh data results
+ Values the EDW strategy

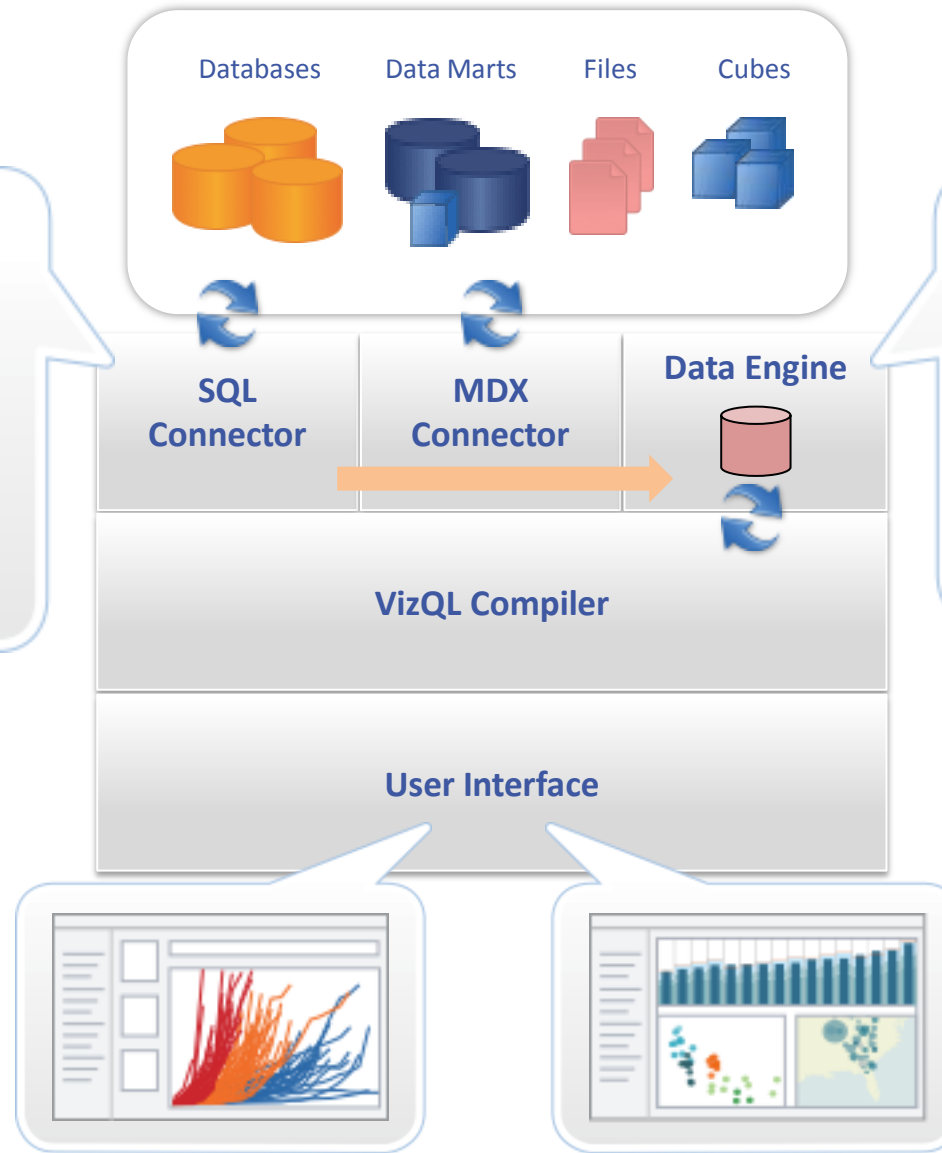**SQL Connector**

MDX Connector

Data Engine

Tableau Data Engine Technology

+ Memory efficient
+ Column store
+ Highly compressed
+ Optimized API specific for Tableau
+ 64-bit (32-bit version as well)

VizQL Compiler

User Interface

Use this option:

+ When the source DB query performance is slow
+ To offload iterative query workload from the source DB
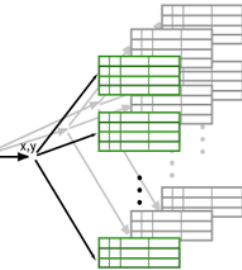+ To work offline from the network
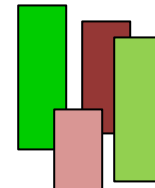+ To keep an archive of the data

Understand Tableau's Query Generation
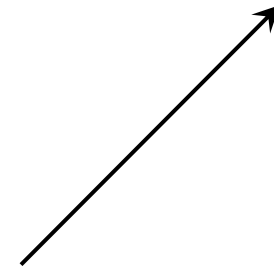

Transform into visual marks
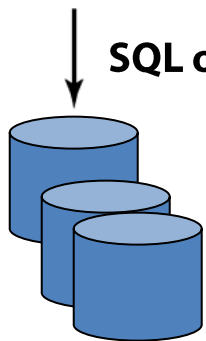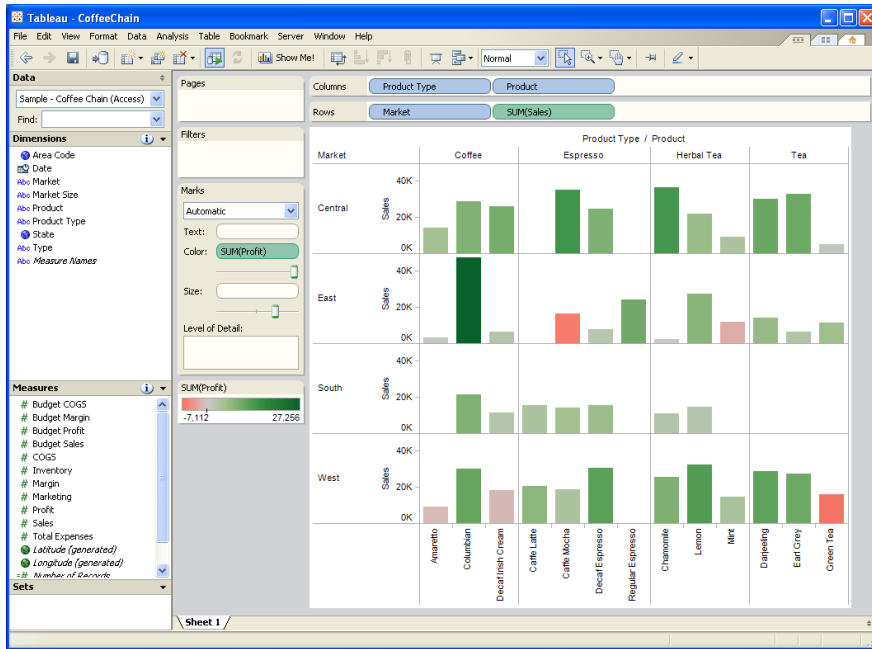
SQL or MDX query

Query results   **Sort into panes**

Query Generation: Filters

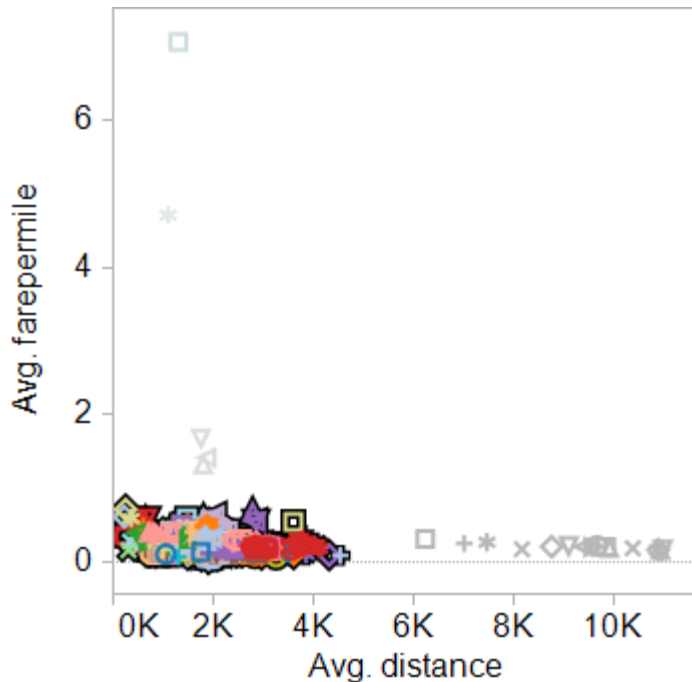- Filters are very expressive

- Different performance characteristics
  - Range filter vs. itemized list
  - Context filter
  - Quick filters with "show relevant values"
  - Slicing filter

Filters: Range vs. Itemized List

Several ways to filter to the selection:

- Keep-only the selection
- Exclude the inverse selection
- Filter by range:
  - Avg. FarePerMile < 1.0, *and*
  - Avg. Distance < 5,000

Filters: Range vs. Itemized List
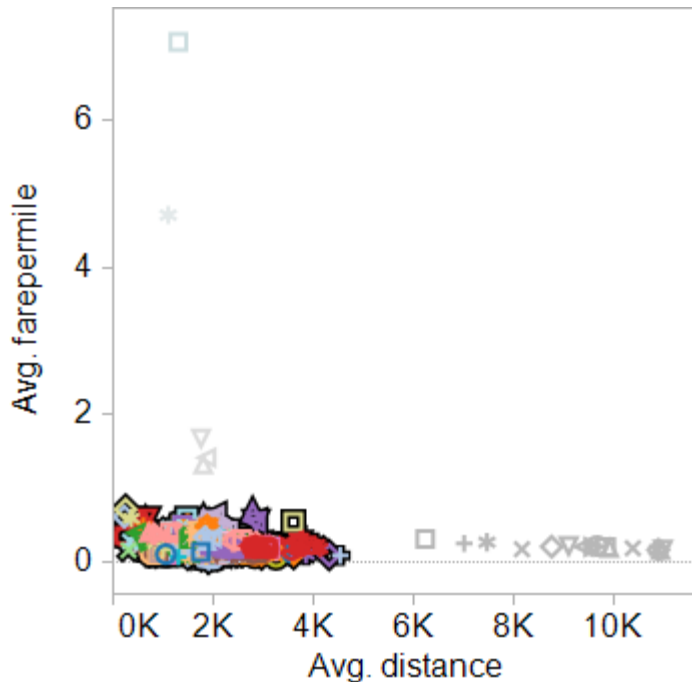
Performance:

- 10.0s: *Unfiltered*
- 19.5s: Keep-only the selection
- 15.9s: Exclude the inverse selection
- 9.5s: Filter by range

Filters: Range vs. Itemized List

Explanation:

- Discrete lists are expensive to evaluate

- "Keep-only" can easily result in large, discrete lists.

Filters: Add to Context

Purpose of Context Filters
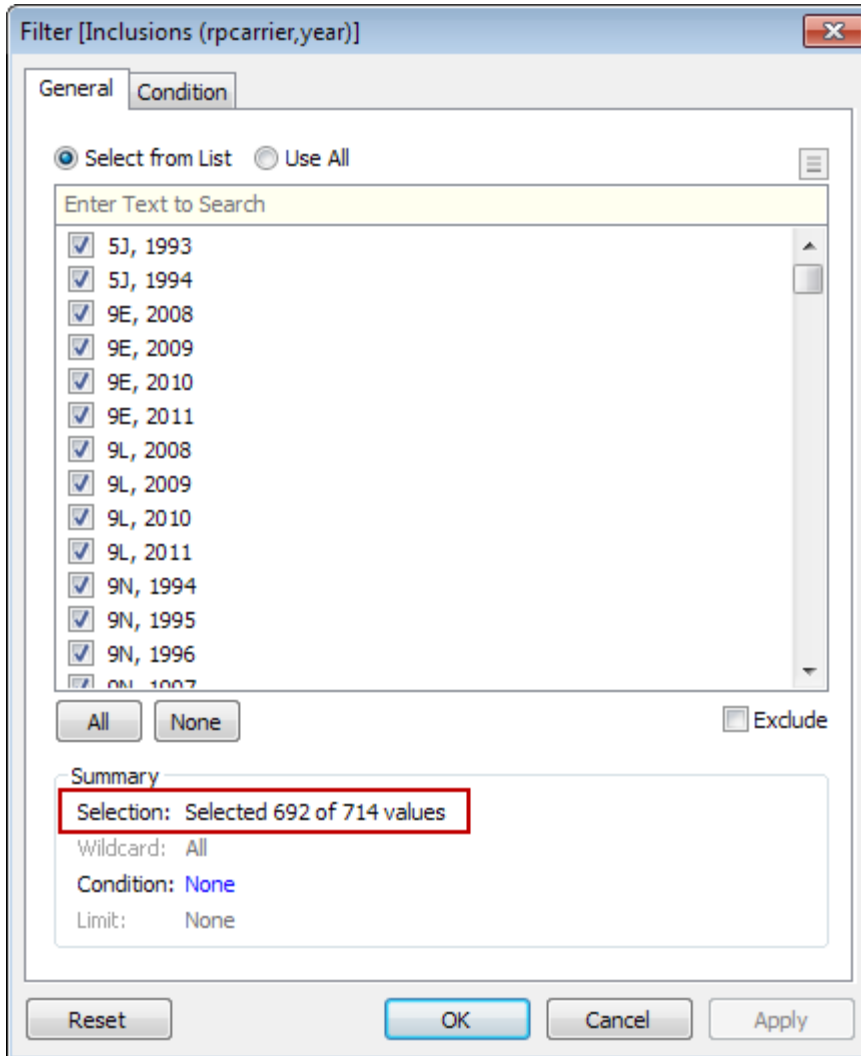- Establish a baseline criteria which is evaluated first
- All subsequent filters are performed on this resultset

How does Tableau process this?
- Instantiate context filter resultset into temp table, *or*
- Express the context filter resultset as a subquery

Performance side-effects of Context Filters
- May have no impact
- May reduce performance if the filter is not very selective
  - The resultset will be nearly as big as the original table
- May boost performance if the filter is highly selective
  - The resultset will be small, making all subsequent filters, aggregations, etc. much faster
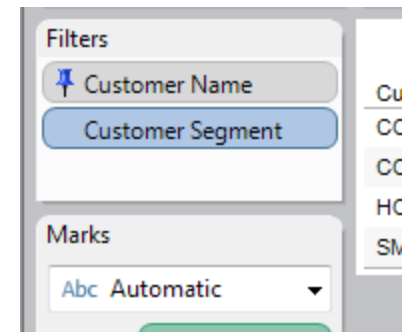
Quick Filters

- Easy-access UI for filtering
- Compact forms:
  - Wildcard Match
  - Compact List
  - Slider, …

- Only Relevant Values
  - Filter items reflect choices made for other filters
    - *e.g.* [State] should not list 'California' if [Region] excludes 'West'
  - Requires querying the database for the domain of *every* filter any time a user modifies any of the filters.
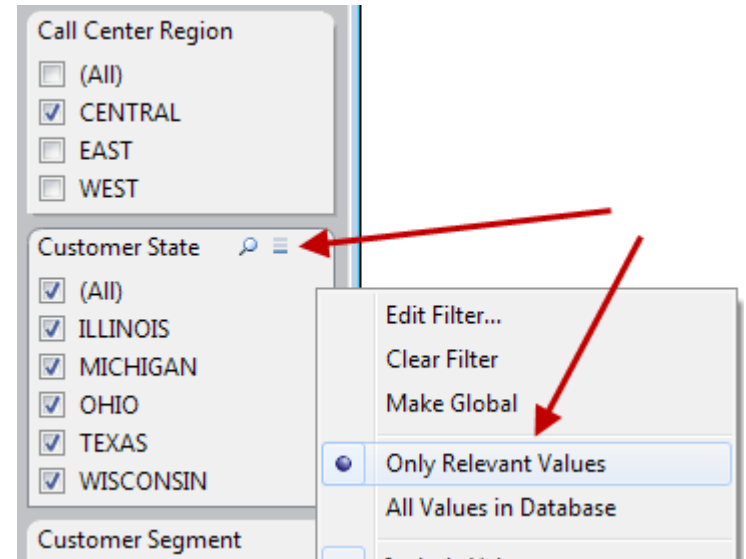
Slicing Filter

- Filter criteria is not part of the visual level of detail
    - Inspired by MDX slicers
    - Not trivial to express in SQL
    - Tableau must isolate the slicing filter level of detail
        - Temp table
        - Subquery

Subquery example

```
SELECT "Staples"."Prod Type1" AS "none:Prod Type1:nk",
   SUM("Staples"."Sales Total") AS "sum:Sales Total:qk"
FROM "TestV1"."Staples" "Staples"
  INNER JOIN (
  SELECT "Staples"."Prod Type4" AS "none:Prod Type4:nk",
    COUNT(1) AS "_Tableau_join_flag"
  FROM "TestV1"."Staples" "Staples"
  GROUP BY 1
  HAVING (SUM(CAST(1 AS BIGINT)) > 1000)
) "t0" ON ("Staples"."Prod Type4" = "t0"."none:Prod Type4:nk")
GROUP BY 1
```
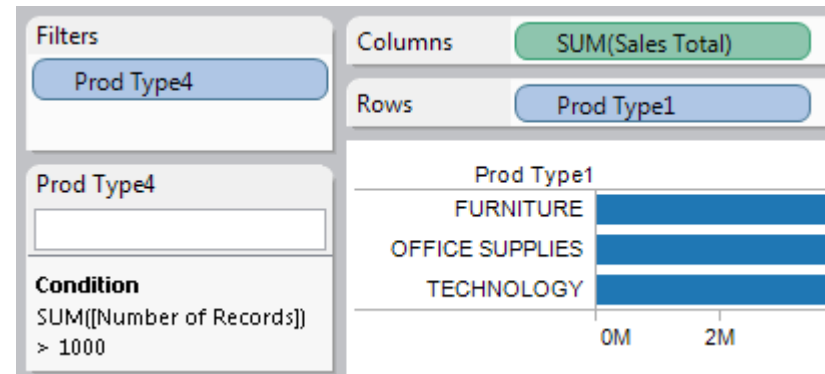
Query Generation: Custom SQL

- ## What is Custom SQL?

  - ### A way to define a connection based on complex join conditions, filtering and pre-aggregation

- ## Why does Tableau wrap Custom SQL? *e.g.:*

```
SELECT *
 FROM (
   SELECT
     `starbucks`.`Market` AS `Market`,
     `starbucks`.`Product Type` AS `Product Type`,
     SUM(`starbucks`.`Sales`) AS `Sum of Sales`
   FROM `starbucks`
   WHERE `starbucks`.`State` <> 'California'
   GROUP BY 1, 2
 ) `TableauSQL`
```

Query Generation: Custom SQL

- Custom SQL represents a resultset
- Tableau visualizations build upon the resultset
  - Queries are not part of the resultset
  - Instead layered *on top of* the resultset

```
SELECT `TableauSQL`.`Market` AS `none_Market_nk`,
       SUM(1) AS `sum_Number of Records_qk`
FROM (
  SELECT
    `starbucks`.`Market` AS `Market`,
    `starbucks`.`Product Type` AS `Product Type`,
    SUM(`starbucks`.`Sales`) AS `Sum of Sales`
  FROM `starbucks`
  WHERE `starbucks`.`State` <> 'California'
  GROUP BY 1, 2
) `TableauSQL`
 WHERE (`TableauSQL`.`Product Type` = 'Coffee)
 GROUP BY 1
```

Query Generation: Discovery Queries

- Connect-time probes
    - Validate joins / custom SQL
        - LIMIT 1
    - Metadata discovery
        - LIMIT 0
        - WHERE 1=0
- Side-band queries
    - Domain size checks
    - Contains NULL?

Identify Database Tuning Opportunities

- Directly used by Tableau
  - PK / FK (join culling)
  - NOT NULL constraint (filtering)
  - Temp table permissions (filtering)
- Indirectly used by Tableau
  - Indexes
    - hash index for grouping
    - range index for dates
  - Partitioning
- Alternatives to Custom SQL
  - Database view (possibly materialized)
  - Initial SQL

Author Optimized Visualizations

- Visualizations are targeted to the human brain
    - Must be a digestible quantity of information
    - Challenge: condensing insight from large data sets

- Working with Big Data
    - Aggregation
    - Filtering
    - Extracts
    - Avoiding joins
    - Advanced techniques

Authoring: Aggregation

- Tableau aggregates by Default
- Pre-computing aggregations can lead to huge performance gains
  - Certain things not preserved (Count Distinct)
- More than one aggregated extract can be used to great advantage

Authoring: Filtering

- Filter first, then viz

- Filtering High-Cardinality Fields (Skip)
  - Wildcard match
  - Type-in List

- Dashboard actions
  - Can replace Quick Filters
  - Edit your Action to exclude on deselect
  - Create 'None' filter first

Authoring: Avoiding Joins

- Aliases
  - Rather than look up table
  - Potential drawbacks:
    - Filters are done on original field
    - Good for values that rarely change
- Blending
  - Create Primary Group
  - Edit Primary Aliases
  - Connect to same DB, separate connection for each table
- Ad-hoc groups
  - On the fly, visually

Authoring: Extracts

- Extract creation (Demo)
  - Filter
  - Aggregate (summary view)
  - Incremental
  - Sampled
  - Hide unused fields
  - Data types/characteristics matter
    - Integer better than String
    - Date better than Timestamp (if detail not needed)

- Optimize extracts

Authoring: Advanced Techniques

- Initial SQL
  - Can be used to prep data for faster querying

- Query Banding
  - Teradata Only
  - Can set query priority

Please evaluate this session (TCC11 202)

Tuning Tableau and Your Database

★ Text to **32075**

★ In the body of the message, type: **TCC11<space>202** then letters from the table below to indicate each response.

★ Provide additional comments after an asterisk "*"

★ Sample text: **TCC11 202aho*That was great!**

Please give your response to the following:	Excellent	Great	Good	Average	Poor	Bad	Very Bad
What was the value of this session to you?	a	b	c	d	e	f	g
What are the chances you will apply what you learned in this session in your work?	h	i	j	k	l	m	n
What are the chances you would recommend this session to a colleague?	o	p	q	r	s	t	u

Each text evaluation you send enters you into a drawing for an iPad!