Data modelling techniques for optimal Tableau performance

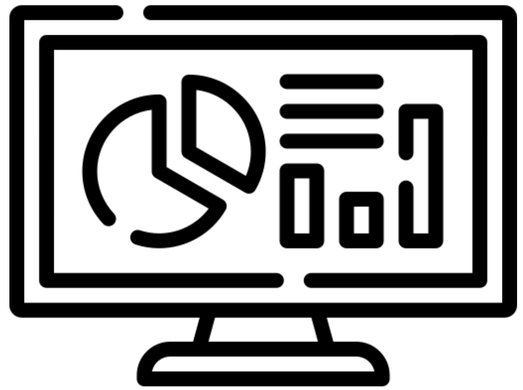
Anya Prosvetova
Senior Data Engineer @ Aimpoint Digital
Tableau Visionary & DataDev Ambassador

Why Tableau performance is important?

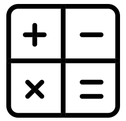
- Overall query execution speed
- Seamless user experience
- Scalability to handle growing data volumes
- Timely data refresh
- Optimal resource utilisation
- Cost efficiency

Load time benchmarks

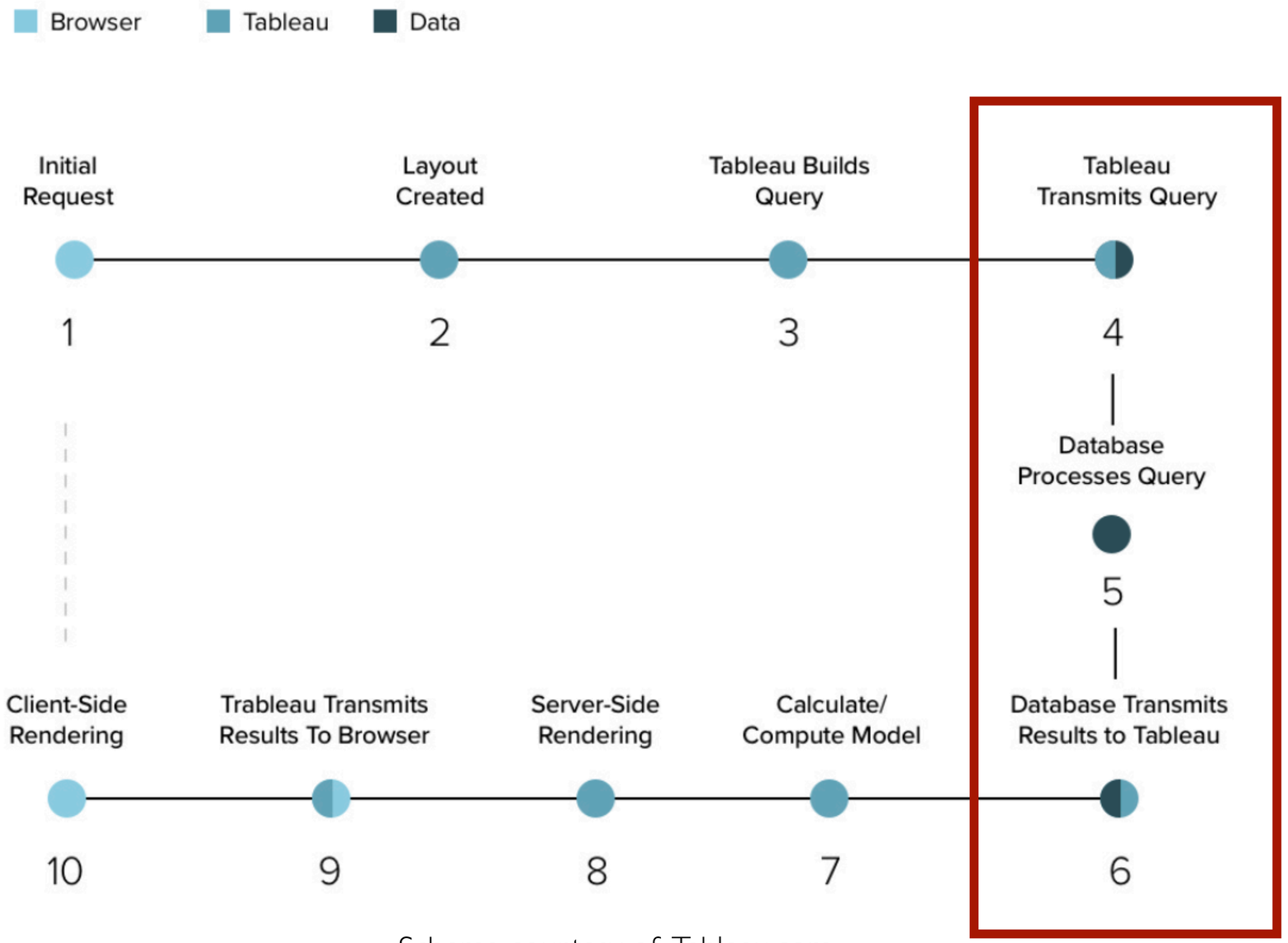
0-5 sec	General audience
10-30 sec	Invested audience
1 min	Specialised audience



Understanding Tableau Performance

Workbook element	Performance impact
 DATA	Query time
 CALCULATIONS	Calculation time
 WORKSHEETS	Rendering time
 DASHBOARD LAYOUT	Layout computation

Understanding Tableau Performance

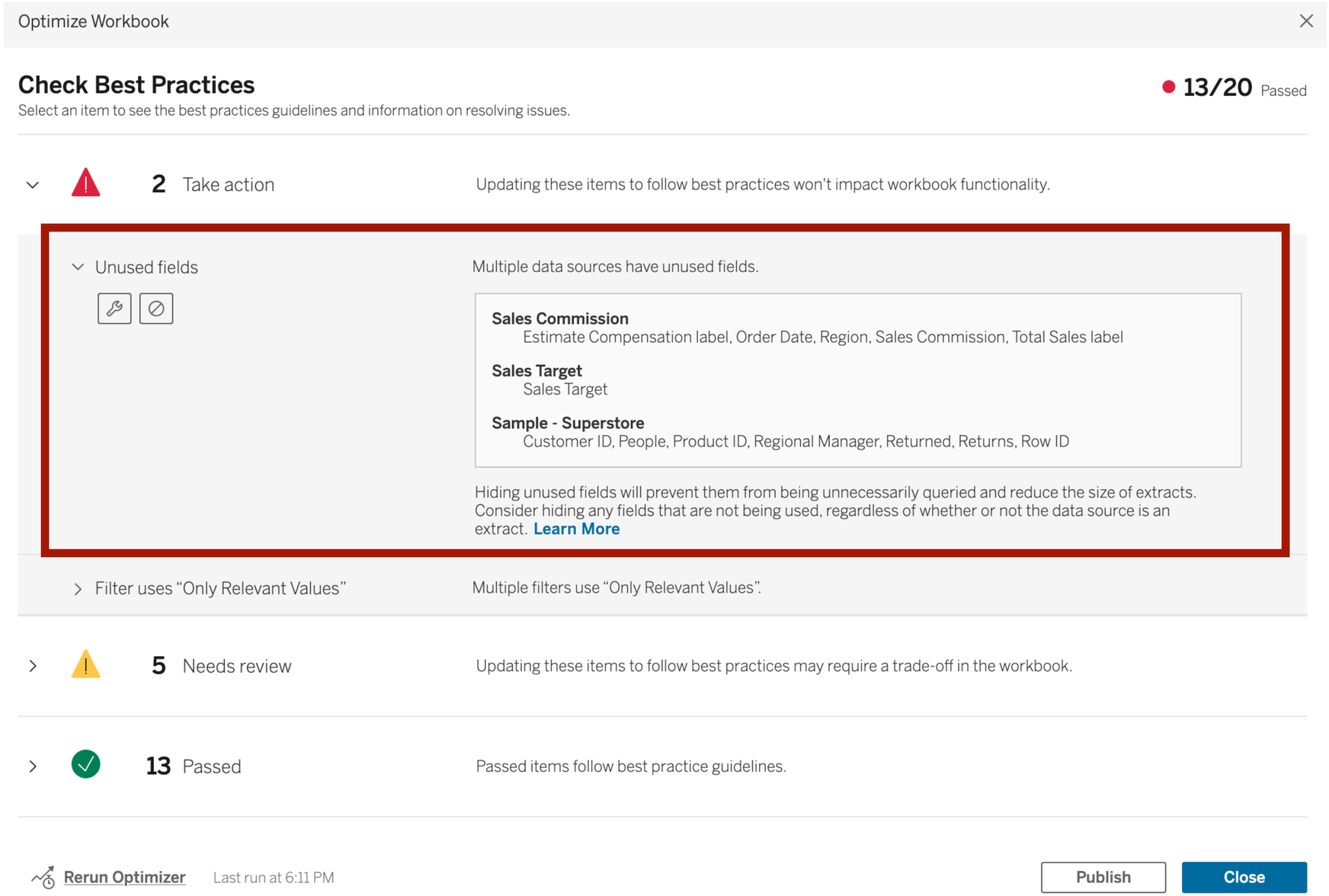


Schema courtesy of Tableau.com

Eternal Tableau optimisation tips

- If the data source is slow, it will be slow in Tableau
- If it is slow in Tableau Desktop, it will most likely be slow in Tableau Server / Tableau Cloud
- Newer is generally better
- Less is more

Where to start with performance optimisation?



Transform your data outside of Tableau: Aggregate / Filter

- Keep only relevant dimensions & measures
- Filter out irrelevant values
- Aggregate measures where possible
- Roll up date / date time fields

Transform your data outside of Tableau: Clean

- Cast data types in data source
- Replace misspelled / incorrect values
- Split / transform string values

Transform your data outside of Tableau: Row-level calculations

- Transformations of values at the data source level of detail (creating a new column)

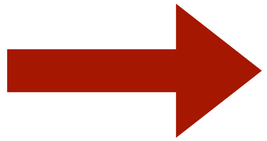
[Revenue] = [Quantity] * [Price]

- Cohorts or groups of values

Transform your data outside of Tableau: Pivot

- Rows to columns / Columns to rows

Country	01/01/23	01/02/23	01/03/23
France	125	563	246
Netherlands	234	876	689



Country	Date	Sales
France	01/01/23	125
France	01/02/23	563
France	01/03/23	246
Netherlands	01/01/23	234
Netherlands	01/02/23	876
Netherlands	01/03/23	689

Data transformation outside of Tableau: Scaffold

- Expanding the data with new rows

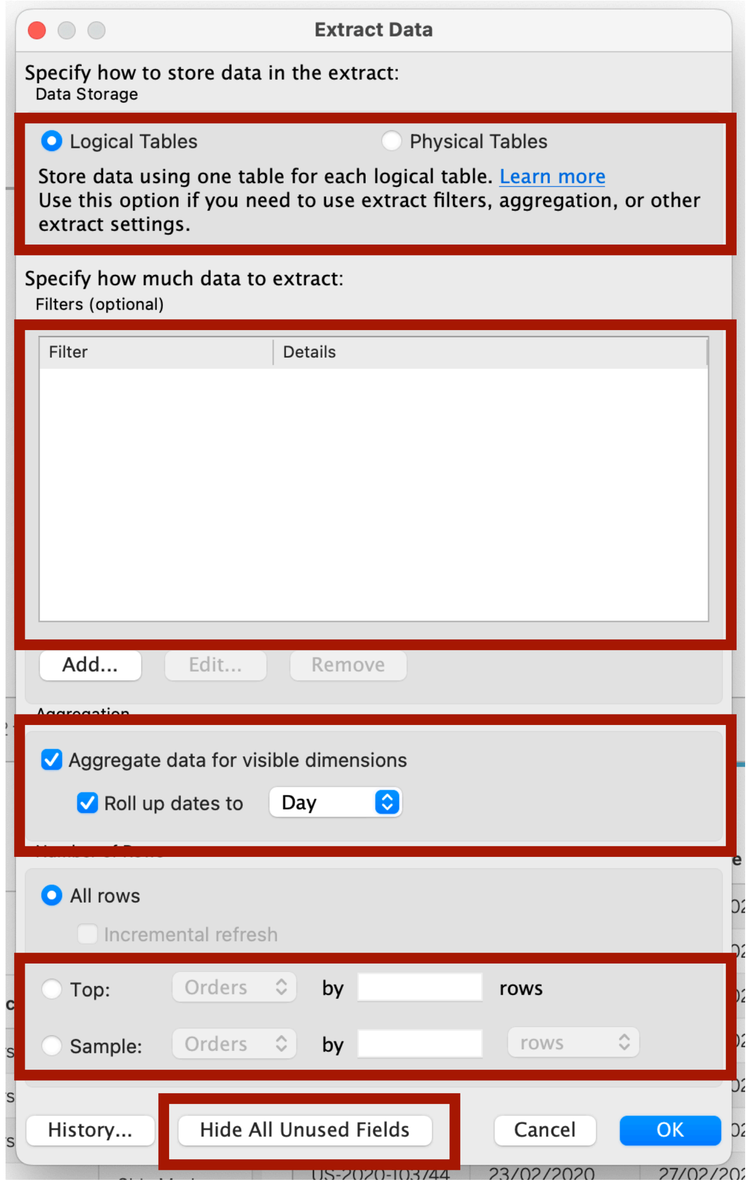
Employee id	Name	Start date	Leave date
1245	John Doe	14/03/2022	30/03/2023
5876	Mary Smith	02/07/2020	
7364	John Smith	08/08/2020	
8374	Alice Tate	28/05/2019	09/04/2022

Data modelling in Tableau: Joins vs Relationships

Joins	Relationships
Physical layer	Logical Layer
One SQL statement for all views	Unique SQL statements for every view
One combined table	Individual tables

Connecting to data in Tableau: Hyper Extracts

- Use Hyper extracts when possible
- Choose between physical vs logical tables for storage
- Use data source filters
- Aggregate data and / or roll up dates
- Keep only a sample of rows
- Hide unused columns
- Materialise calculations
- Consider embedding an extract



Connecting to data in Tableau: Live Connections

- Use referential integrity
- Leverage the relational data model
- Consider using multiple data sources
- Avoid using Custom SQL in production
- Optimise the database

Connecting to data in Tableau: Optimise the database

- Set indexes on join dimensions
- Use a star schema when possible
- Set appropriate primary and foreign keys
- Set Index on filtering dimensions

Let's recap

- Transform your data outside of Tableau
- Choose relationships instead of joins
- Use Hyper Extracts when possible
- Avoid using Custom SQL for live connections in production
- Fine-tune your database

Let’s recap

Useful materials

- Designing efficient workbooks, whitepaper by Tableau
- Tableau performance optimisation flowchart
- dbt and Tableau Integration demo
- Plan your Tableau datasource, help page by Tableau
- Tableau Logshark

Questions?

Website: prosvetova.com

Twitter: Anyalitica