



7 Power BI Performance Optimization Checklist

Visit ScaleupAlly.io to learn more.

 ScaleupAlly[®]

Power BI is a powerful tool for transforming data into actionable insights, but its performance can significantly impact your ability to make timely decisions. Performance optimization ensures that your reports are fast, reliable, and efficient, enhancing the overall user experience.

This checklist is designed to guide you through the key areas where you can optimize Power BI for better performance. Whether you're a beginner or an experienced user, this comprehensive guide will help you identify bottlenecks and implement solutions effectively.

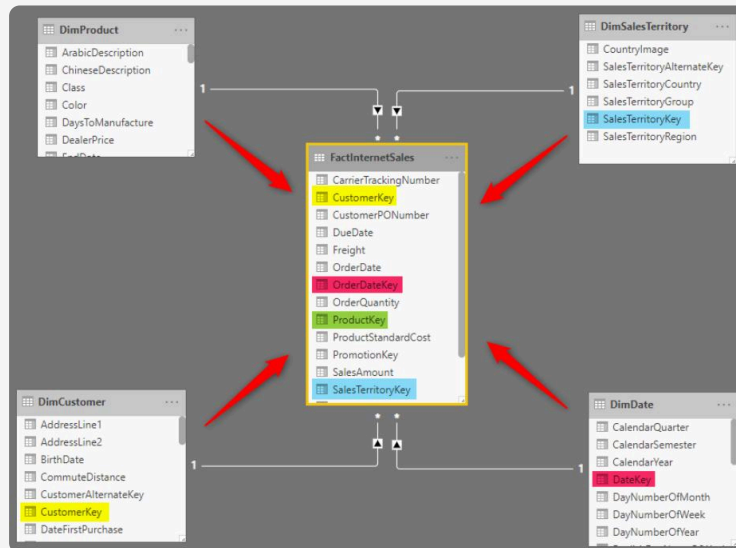
How to use this checklist

- ✓ **Understand Each Section**
Read the "**What to do?**" and "**How to implement**" parts for clear objectives and steps.
- ✓ **Prepare Your Workspace**
Gather necessary tools like Power BI Desktop and Power Query Editor. Have a way to track changes.
- ✓ **Assess Your Current Reports**
Evaluate existing reports against the checklist points to identify optimization areas.
- ✓ **Implement One Step at a Time**
Focus on one section at a time for accuracy and manageable changes.
- ✓ **Test Regularly**
Use tools like Power BI's Performance Analyzer to test performance after changes.
- ✓ **Iterate and Review**
Read the "**What to do?**" and "**How to implement**" parts for clear objectives and steps.
- ✓ **Understand Each Section**
Continuously revisit the checklist as datasets grow or requirements change.

Let's dive into the checklist now! 🏁

7 Power BI Performance Optimization Checklist ✔

1 Data Model Design



What to do ?

How to implement ✔

Use Star Schema for organizing data into facts and dimension tables.

Create relationships in Power BI's data model view where fact tables are connected to dimension tables via key fields (like product IDs or customer IDs).

Identify and remove unnecessary Data.

Use Power Query to filter out unnecessary rows and remove unneeded columns during data transformation.

Optimize Relationships: Use simple 1:1 or 1:* relationships.

In the Manage Relationships window, ensure that relationships are simple and ensure key fields (like product or customer ID) uniquely identify records in dimension tables.

Employ Power Query

Open Power Query Editor and apply transformations such as removing duplicates, changing data types, filtering rows, or merging tables.

2 Data Size Management

What to do



How to implement



Aggregate data at higher levels wherever possible.

In Power Query, use Group By operations to aggregate data before importing it into Power BI. For example, aggregate sales data by month rather than by day.

Incremental refresh, meaning refresh only the changed data.

In Power BI Desktop, go to the Modeling tab and set up incremental refresh on tables that contain large datasets (e.g., transaction logs).

Use pre-aggregated tables for performance.

In your data source, create summary tables (e.g., monthly sales) and load these instead of the detailed data.

3 Data Storage Mode

What to do



How to implement



Prefer import mode.

In Power BI Desktop, when connecting to a data source, choose Import instead of DirectQuery.

Combine imported and DirectQuery tables for flexibility.

In Power BI Desktop, use the Composite Model feature to mix storage modes by connecting different tables with either Import or DirectQuery.

Struggling with Power BI optimization? Let us handle it for you!

Connect with us for a free consultation at ally.success@scaleupally.io today for a better tomorrow!



4 DAX Optimization

What to do



How to implement



Keep DAX expressions simple.

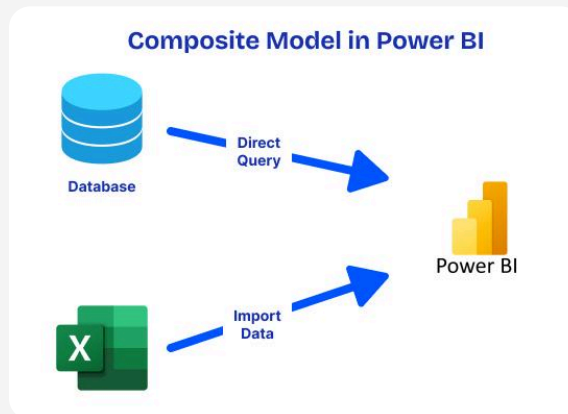
Use measures for calculations rather than calculated columns, as measures are computed only when needed.

Store intermediate results with variables to avoid recalculation.

In your DAX formulas, define variables with VAR and reference them in your calculations instead of repeating the same expression multiple times.

Minimize SUMX, CALCULATE in large datasets.

Use more efficient aggregation functions such as SUM or AVERAGE, or perform calculations in Power Query. Also, Minimize the use of iterators like SUMX and FILTER on large datasets.



5 Visualisations

What to do



How to implement



Reduce the number of visuals per report page.

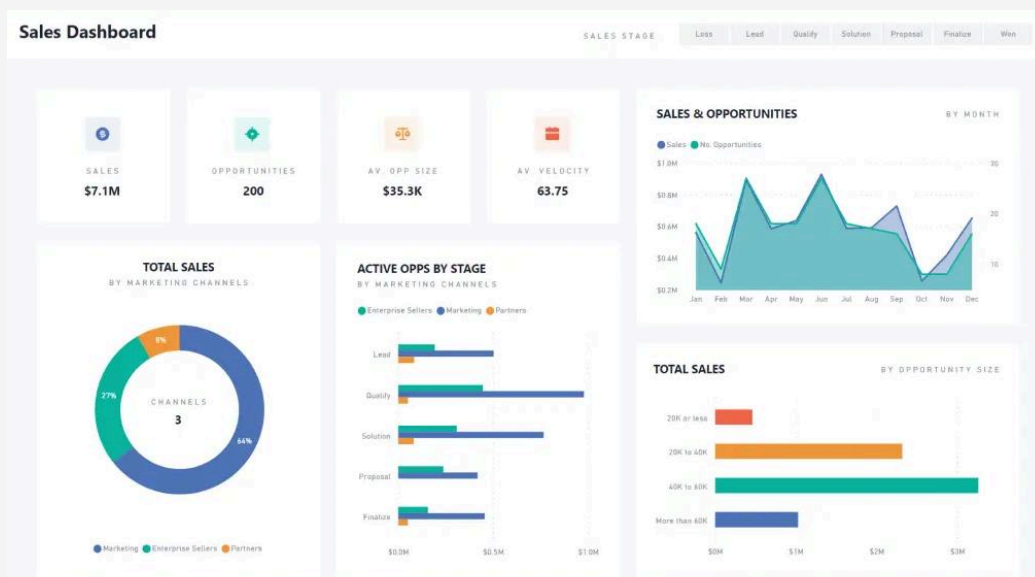
Review your report design and focus on the most essential visuals. Use bookmarks or tabs to split content across pages if necessary.

Avoid complex charts like TreeMaps or Sankey.

If performance is a concern, opt for bar, column, or line charts, which are more efficient.

Limit slicers with many unique values.

Limit slicers to small, manageable data sets, or use searchable slicers to minimize performance impact.



6 Query Performance

What to do



How to implement



Enable Query Folding.

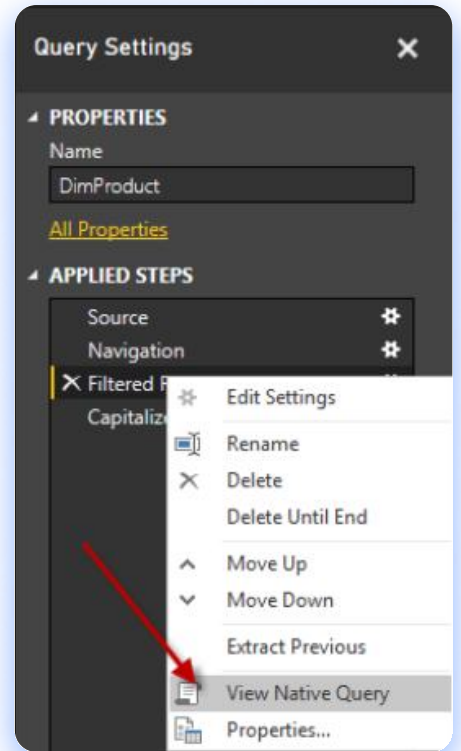
In Power Query, check the query folding status for each step (right-click the applied step and see if "View Native Query" is enabled).

Limit Joins in Power Query.


Simplify queries and avoid unnecessary merges in Power Query.


Cache data for performance when merging tables.

Add Table.Buffer() to your query steps before performing expensive operations like merges.



7 Power BI Services

What to do 

How to implement 

Talk to ScalupAlly’s experts. No, we are not kidding!

Free consultation at ally.success@scaleupally.io

To improve refresh performance in the service, enable Query caching

Enable this setting in the Power BI Service settings, particularly for large and frequently used datasets.

Schedule refreshes during off-peak hours.


Set the refresh schedule in the Power BI Service under Dataset Settings.

Avoid overcrowding a single workspace.

Optimize Your Power BI with Us

Consult Our Experts for Tailored Performance Solutions

 ally.success@scaleupally.io

 ScaleupAlly.io

