



Power BI Parallel Data Loading options' impact on Server Resources with large dataset using TMVGate

Technical Note: TA2019007



Contents

1.0	Overview.....	2
2.0	Power BI Parallel Loading of Tables and Background Data Options	3
3.0	TMVGate MaxRequests parameter	5
4.0	Power BI Gateway	5
5.0	Run TMVGate on a separate server.....	5
6.0	Summary.....	6
6.0	References	7



1.0 Overview

In a typical Power BI report development process - from defining the data source using TMVGate URL (PSV format) till the report design screen is displayed, Power BI issues a total of 4-5 calls to TMVGate based on the built Version: 2.51.4885.701 (at the time the article was written). Some of these calls are fired off concurrently from Power BI.

As TMVGate is multithreaded, it would handle these calls as concurrent requests and process them in parallel.

For data sets that are relatively large (e.g. in millions of data cells), this may cause a server resource constraint (specifically for memory usage) if TMVGate is running on the same server as the TM1/PA instance.

This technical document explores a few options to tackle the behaviour of Power BI when dealing with large datasets. For customers running TMVGate on the same server as TM1/PA instance, do review this document, evaluate and apply the suggestions accordingly.

Note that the behaviour is not unique to TMVGate but applies in general to all data sources supported by Power BI.

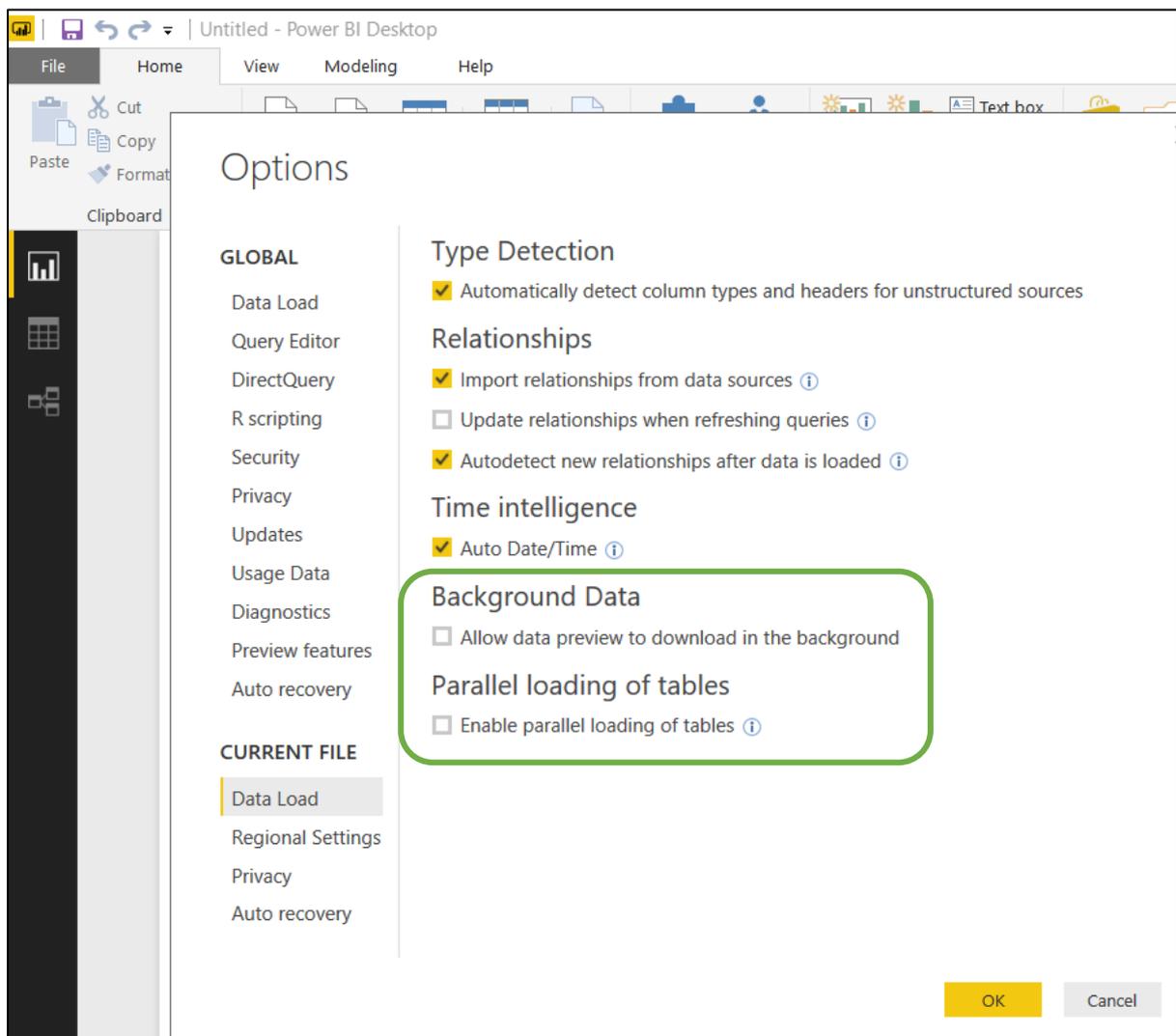
2.0 Power BI Parallel Loading of Tables and Background Data Options

In Power BI Options, there are two specific options that will affect how Power BI loads the data from respective data sources, and previews the data.

This can be set via File -> Options as shown below.

The two specific options are:

- **Background Data**
- **Parallel loading tables**





By default, when you create a new Power BI workbook (pbix), these two options are enabled. You will need to explicitly disable these two options before you start defining any data sources.

For existing workbooks, you can change the option and re-save the workbook.

For large datasets, we suggest disabling these two options as this will force Power BI to issue the calls to TMVGate in sequential manner instead of issuing concurrent calls. Note that this will have an impact on the response time, but will result in lower memory utilization.

With TMVGate Cache (enabled in the URL generator), if each request is completed within the 20 minutes cache expiration window, the impact to response time will be minimal. In fact, the performance will improve as caching is be utilized.

An example of the impact of these two options:

An extraction of 5 million records with the options enabled will result in approximate 4*5 million records to be processed by TMVGate concurrently. This will cause a heavy load on the server and will be made worse if the server is already low on memory running with TM1/PA.



3.0 TMVGate MaxRequests parameter

You can consider disabling the multithreading capability of TMVGate by changing the MaxRequests parameter to 1. This parameter is found in the TMVGate configuration file ***TMVGate.exe.config***.

```
<add key="MaxRequests" value="1" />
```

This would force TMVGate to process 1 request at any point in time. However, by changing this, you will need to increase “MaxQueue” and “QueueTimeout” parameters.

This is discussed in greater depth in the Technical Document: **Server Resources and Performance Tuning with TMVGate.pdf**

4.0 Power BI Gateway

At the time of writing, we have yet to find any settings for Power BI Gateway to disable parallel loading of tables. When the scheduled refresh kicks in, based on some Power BI users’ feedback, all requests were fired off concurrently.

For customers with a large dataset environment, we suggest setting TMVGate MaxRequests to 1 (see above) to force the extraction of data in TMVGate to run in sequential manner.

5.0 Run TMVGate on a separate server

For customers with big deployments with large datasets and many reports, we suggest deploying TMVGate on a separate server instead of having it run on the TM1/PA server. This will minimize server resource competition between TMVGate and TM1/PA. By far this will be the most effective configuration.



6.0 Summary

In summary, if customers are already running TMVGate on a separate server, they may not have an issue with TMVGate competing with TM1/PA on server resources. Otherwise, customers are advised to review the above options to optimize the deployment.

It is also good to note that the Power BI options discussed in this document are applicable to all other data sources and not just for TMVGate.



6.0 References

The following are some references found on the Web relating to Power BI handling of larger dataset.

- <http://community.powerbi.com/t5/Desktop/Queries-over-large-data-tables/td-p/26783>
- <http://community.powerbi.com/t5/Desktop/Power-BI-Desktop-automatic-refresh-queries-one-by-one/td-p/155889>
- <https://blog.crossjoin.co.uk/2016/12/05/power-query-power-bi-and-the-allow-data-preview-to-download-in-the-background-option/>
- <https://community.powerbi.com/t5/Integrations-with-Files-and/On-premises-gateway-ability-to-limit-data-source-connections/td-p/108644>