



Server Resources and Performance Tuning with TMVGate

Technical Note: TA2019002



Contents

1.0	Overview	2
2.0	Understanding the Performance related parameters	3
2.1	CellGroupCount	3
2.2	MaxRequests	4
2.3	MaxCells	4
3.0	Summary	5



1.0 Overview

TMVGate is a REST API Web Service application that acts as a gateway to IBM Cognos Planning Analytics/TM1 Servers. It uses the TM1 REST API to extract Dimension Subsets or TM1 Cube Views and transforms them into tabular data before returning them to the requesting application.

The transformation of TM1 data to a tabular form is performed in memory. If TMVGate is installed together with a TM1 Server on the same computer, you will need to monitor the server resources closely as TM1 servers rely heavily on memory for optimum performance.

This technical document aims to provide a general guideline on how to monitor and adjust the parameters in the TMVGate configuration file to achieve a balance between performance and server resources.



2.0 Understanding the Performance related parameters

With TMVGate, there are a few parameters which can be adjusted to obtain optimal performance while balancing server resources. These parameters are defined in the TMVGate configuration file “**TMVGate.exe.config**”, which can be found in the folder where TMVGate was installed.

Any changes to the parameters in the TMVGate configuration file will require a restart of the TMVGate service.

2.1 CellGroupCount

The “CellGroupCount” parameter defines the number of records to be extracted and processed from TM1. This will affect the number of requests to TM1 to extract the complete dataset.

For example, if the total record count for a TMVGate request is 1,000,000 (i.e. 1 million records) and the “CellGroupCount” is set to 50,000, this will result in 20 requests to TM1, with 50,000 records being processed for each TM1 request.

With the above example, TMVGate will only process a maximum of 50,000 records at a time. Hence, regardless of the total TMVGate records to be extracted, the server resources, in specific, the memory, will be maintained at a certain level.

TMVGate is designed to handle multiple concurrent requests. If TMVGate receives five concurrent user requests, it will require approximately five times the server resources. A “CellGroupCount” of 50,000 will use approximately $5 * 50,000$ equivalent server resources.

To achieve an optimal setting, you will have to understand the usage pattern of TMVGate users. Taking your average dataset size, number of concurrent requests, and current memory utilization into consideration, you can monitor and assess if TMVGate reaches a stage where its memory utilization is too high and it competes with other processes. If so, you can lower the “CellGroupCount” to adjust the memory utilization rate.



Do note that lowering the “CellGroupCount” can result in multiple TM1 requests for a TMVGate request which translate to a longer response time to the user.

2.2 MaxRequests

“MaxRequests” manages the number of concurrent requests that TMVGate will accept at any point in time. If you have a scheduled extract environment set up (e.g. if you use Power BI and have regular data requests scheduled), you may consider lowering the “MaxRequests” if real time performance is not crucial for such request.

Note that you may need to increase the “MaxQueue” and “QueueTimeout” parameters if you lower the “MaxRequests” parameter, especially if there are concurrent requests coming in.

“MaxQueue” defines the number of requests that can be queued in TMVGate if the requests exceed the “MaxRequests” setting.

“QueueTimeout” defines (in seconds) how long the request can wait in the queue before it is discarded.

2.3 MaxCells

For companies with high application interaction and concurrent user usage patterns, you can consider adjusting the “MaxCells” parameter, which defines the maximum dataset size that can be extracted per request.

By lowering “MaxCells”, the requested dataset size cannot exceed the specified value. This will result in better performance in interactive requests, as well as lower server memory utilization.

For scheduled extractions with a large dataset size, do make sure the “MaxCells” value is high enough to accommodate the largest request.



3.0 Summary

In summary, the first step to optimizing performance with TMVGate is to determine your company's usage patterns based on the parameters highlighted above.

You can browse through the TMVGate log files to understand your company's usage patterns before changing any of the parameters.

In general, for high volume extractions, you should consider setting TMVGate up on a separate server as not to compete the memory resources that TM1 server required.