Technical Brief
December 2016

QIX engine memory management and CPU utilization

Introduction

This document from the Scalability Center describes how the QIX engine uses system resources like RAM and CPU. Since the performance of the engine is related to the RAM and CPU usage, it is important to understand how the engine uses these resources.

The first part of this document discusses memory management and describes the Working set Low (QlikView) / Min memory usage (Qlik Sense) and Working set High (QlikView) / Max memory usage (Qlik Sense) settings. The second part discusses CPU usage and how the QIX engine scales over cores.

Note: The QIX engine memory usage is described without taking the risk for resource congestion or the influence from other services running in the same environment into consideration.

Memory management

The main memory RAM is the primary storage for all data to be analyzed by the QIX engine. The engine uses the RAM to store:

- The unaggregated dataset that is defined by the document data model.
- The aggregated data (that is, cached result sets) and the calculations defined by the user interface.
- The session state for each user of the document.

When a user requests a document, the QIX engine loads it into RAM, if it has not been loaded before. The dataset for a document is only loaded once and is not duplicated for multiple users who concurrently access and analyze it.

As the user makes selections in the document, the QIX engine performs the needed calculations in real time. To render a chart, the engine must first access the core unaggregated dataset that is based on the data model and then calculate and store the totals. The user session states and aggregates occupy RAM above and beyond the RAM used to store the core unaggregated dataset. Most of the session information is shared between sessions in the same state. Aggregates are shared across all users in a central cache.

The QIX engine is allowed to use a certain amount of the physically installed RAM. This can be configured as follows:

- QlikView: Use the Working set Low / Working set High settings in the QlikView Management Console.
- Qlik Sense: Use the Min memory usage / Max memory usage settings in the Qlik Management Console.

The Working set Low / Min memory usage setting is the memory allocation that the QIX engine will use. Prior to that point, the engine will not try to minimize its allocation of memory. For example, if the physical RAM on your server is 256 GB and Working set Low / Min memory usage is set to 70%, the engine will not try to minimize the allocated memory before 179.2 GB of RAM is used. On the other hand, the engine will not use any memory if it is not used for a beneficial purpose.

The Working set Max / Max memory usage setting is the point above which the QIX engine cannot allocate any memory. Obviously, Working set Low / Min memory usage must be lower than Working set Max / Max memory usage and leave enough room for handling of transients (that is, the amount of RAM temporarily allocated while the engine purges cached result sets) without reaching Working set Max / Max memory usage in an environment. For example, if the physical RAM on your server is 256 GB and Working set Max / Max memory usage is set to 90%, the engine cannot allocate any RAM above 230.4 GB.

It is recommended to leave these settings with their default values. However, on servers with large RAM (256 GB and upwards), the settings can be changed to allocate a couple of GBs of RAM for the operating system and allow the remaining RAM to be used by the QIX engine.

The QIX engine depends on the operating system to allocate RAM for it to use. When the engine starts, it attempts to reserve RAM based on the Working set Min / Min memory usage setting. The engine allocates all allowed memory with cached results sets as quickly as possible, but this does not mean that the engine will lack in performance once the allowed amount of memory is reached. When the allowed amount of RAM is exceeded, the engine starts to purge cached result sets to make place for new documents, new calculated aggregates, and session state information.

If the RAM becomes scarce, the operating system may, at its discretion, swap some of the QIX engine memory from physical RAM to Virtual Memory (that is, use the hard disk-based cache instead of RAM). When the engine is allocated Virtual Memory it may be orders of magnitude slower than when using 100% RAM. This is undesirable and may lead to poor user experience. Note that this is not unique to QlikView or Qlik Sense as the RAM is handled by the operating system.

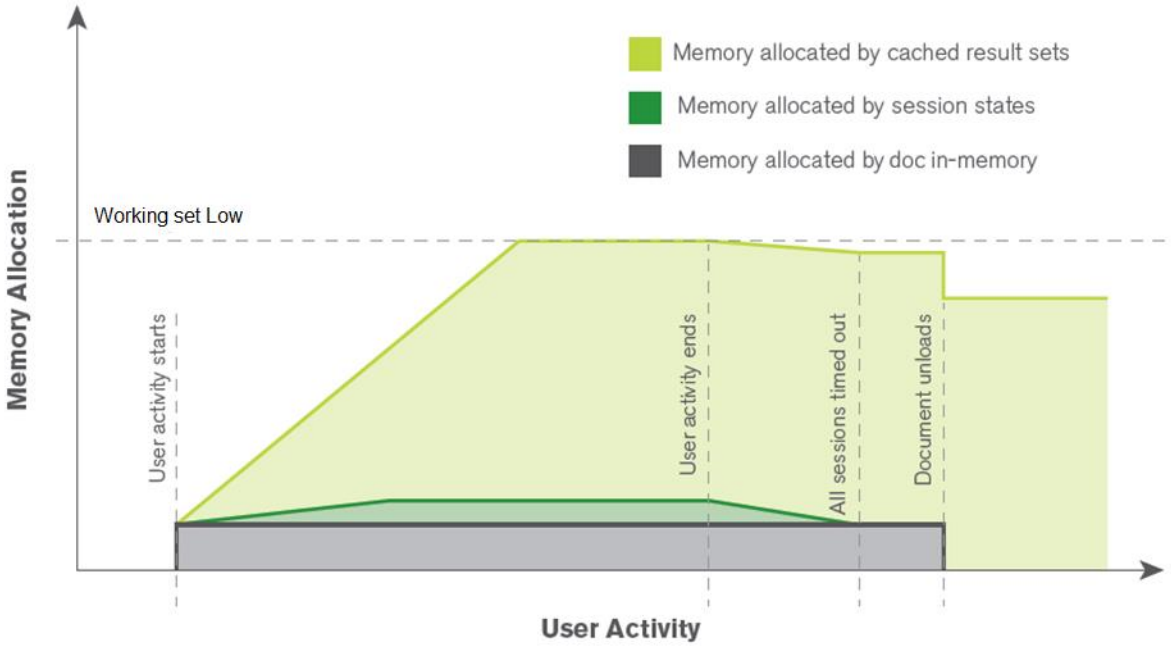

Fig. 1 QIX engine memory allocation when a single document is used

Figure 1 shows an example of the memory allocation by the QIX engine over time when a clean server is started and users begin to interact with a document. The document is first loaded into memory, which corresponds to a peak in memory consumption. Whilst the users continue to interact with the document, result sets from requested calculations are stored in RAM. Additional requests for already cached result sets can then be served without any additional calculations. The engine must also keep track of the state of each active user session, but the portion of RAM allocated for that is small in comparison to the memory allocated for the document and its cached result sets.

The QIX engine does not allow persistent allocation of more memory than specified by the Working set Low / Min memory usage setting. When the total amount of allocated RAM goes beyond that setting, previously cached result sets are purged to make room for new ones. The prioritization of which result sets to purge is based on the age, size, and time of calculation of the result sets currently in the cache.

When the document is unloaded from memory, the total amount of allocated memory drops by the same amount as was originally allocated by the document. If there are no requests to use the allocated memory, the cached result sets will stay in memory since there is no reason to remove result sets that might be useful later on.

Note the “User activity ends” and “All sessions timed out” entries in Figure 1. In QlikView, a session ends a configurable amount of time after the user closes the browser tab where the session is running (that is, at “All sessions timed out”). In Qlik Sense, which uses WebSocket, the session ends when the user closes the browser tab (that is, at “User activity ends”).

Figure 2 shows how multiple documents can fit into RAM, even when the total amount of allocated memory touches the Working set Low / Min memory usage limit. This is achieved by purging cached result sets, so that memory is released to load new documents. The amount of RAM that can be used for the cached result sets can be seen as a floating amount between the Working set Low / Min memory usage setting and the amount consumed by the documents and session state information.

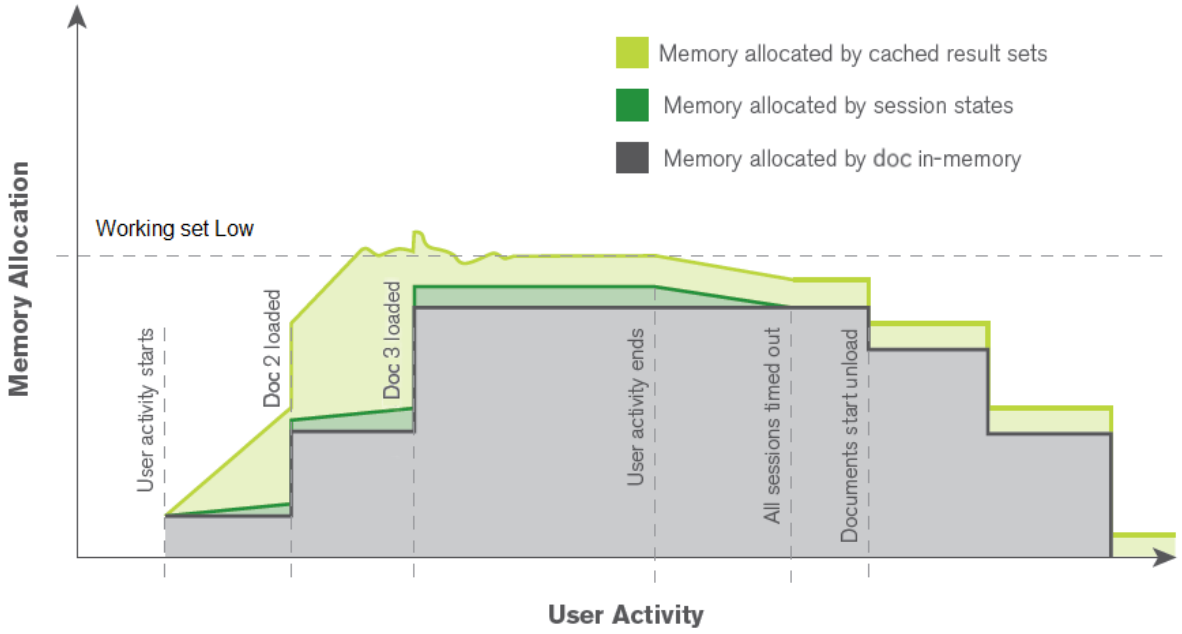

Fig. 2 QIX engine memory allocation when multiple documents are used

It is good practice to investigate how the QIX engine uses memory. When the memory curve fluctuates a lot, it usually means that the engine needs to allocate extra memory during a calculation. The memory is released when the result set is cached. Jitter on the memory curve might indicate poor document design that may be worth investigating as jitter often means slow response times.

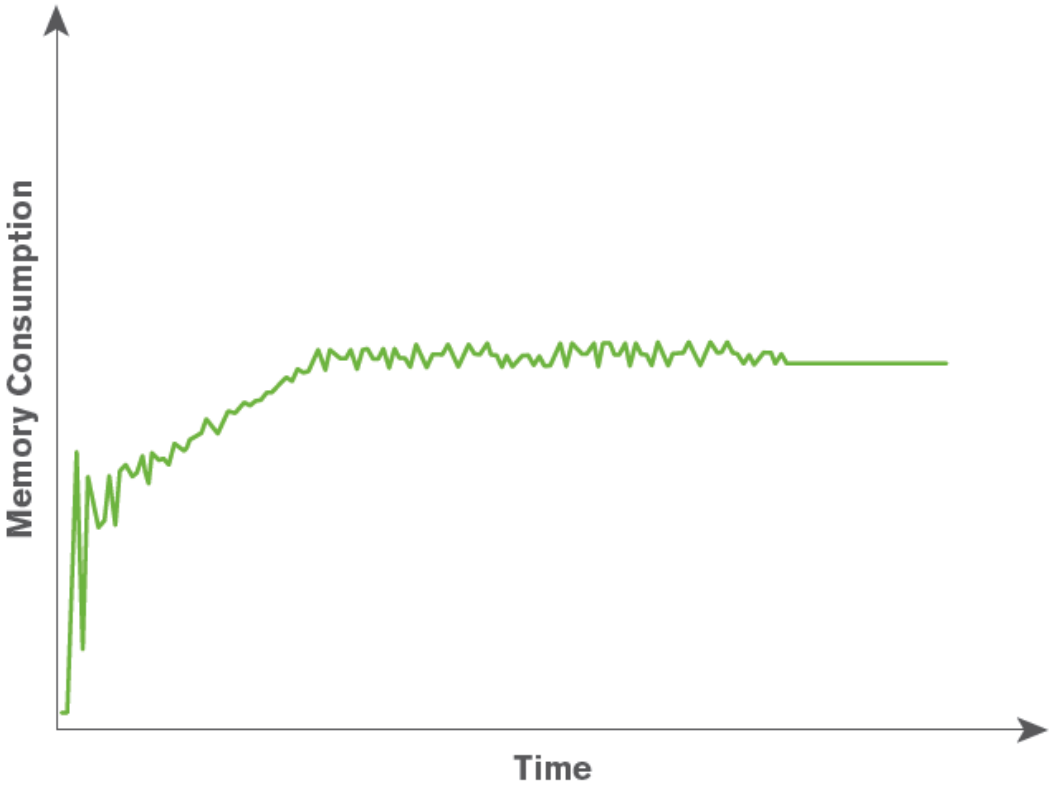

Fig. 3 Analyzing the memory curve fluctuation

Summary

The following is important to consider when it comes to memory management:

- The QIX engine caches all result sets as long as there is RAM available for allocation.
- The QIX engine will only release memory when unloading documents. When a document is unloaded from memory, the total amount of allocated memory drops by the same amount as originally allocated by the document. If there are no requests to use the allocated memory, the cached result sets will stay in memory since there is no reason to remove result sets that might be useful later on.
- When the Working set Low / Min memory usage limit is reached, old sessions and cached results are purged to make room for new values.
- The age, size, and time of calculation are factors in the prioritization of which values to purge.
- The QIX engine purges old sessions when the “maximum inactive session time” value is reached.
- High memory usage is usually the result of many cached results. As long as paging does not occur, high memory usage is a good thing.

CPU utilization and scaling over cores

The QIX engine leverages the processor to dynamically create aggregations as needed in real time, which results in a fast, flexible, and intuitive user experience. Note that the data stored in RAM is the unaggregated granular data. Typically, no pre-aggregation is done when the data is reloaded or a script is executed for a document. When the user interface requires aggregates (for example, to display a chart object or to recalculate after a selection has been made), the aggregation is done in real time, which requires CPU processing power.

The QIX engine is multi-threaded and optimized to take advantage of multiple processor cores. All available cores are used almost linearly when calculating charts. During calculations, the engine makes a short burst of intense CPU usage in real time.

It is good if the CPU utilization is high during peaks over time (see Figure 4). This indicates that the document is designed for good scaling over cores. A certain selection or calculation can be assumed to require a certain amount of processing capacity (that is, clock cycles from a certain chip), and a peak of high utilization results in faster response times as all available cores can cooperate to complete the calculation. The QIX engine has a central cache function, which means that chart calculations only need to be done once, which results in better user experience (that is, faster response times) and lower CPU utilization.

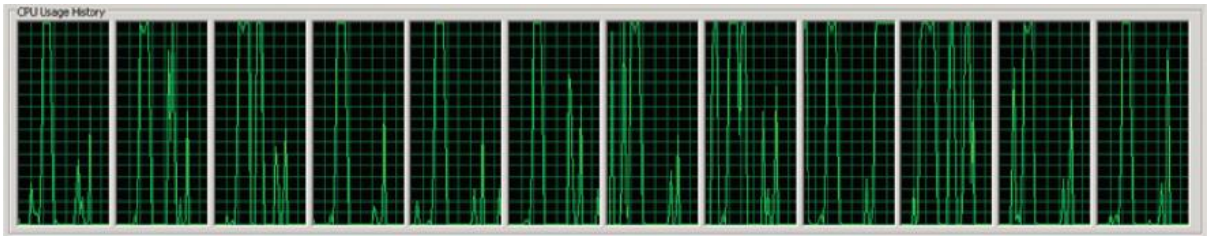

Fig. 4 Example of high CPU utilization during peaks over time

If a server has high CPU utilization on average (>70%), incoming selections have to be queued prior to being calculated as there is no processing capacity immediately available (see Figure 5). This is an indication of poor performance. The cases where the QIX engine will not scale well over cores include:

- A single user triggers single-threaded operations.
- The underlying hardware does not allow for good scaling (for example, when the memory bus is saturated).

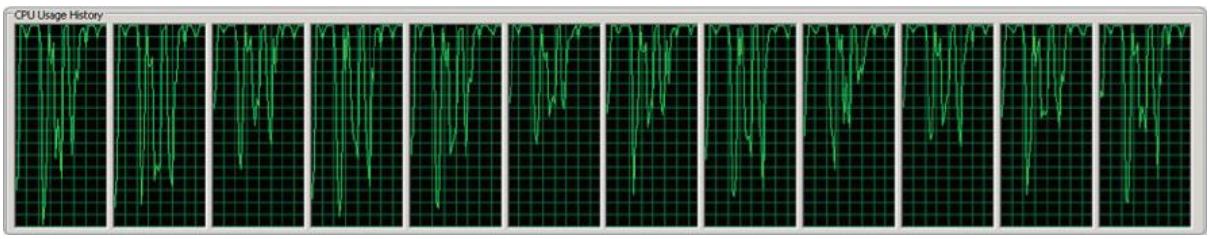

Fig. 5 Example of high CPU utilization on average (>70%)

Cores: Performance test

The processing capacity of the QIX engine can be increased by adding cores. If a user scenario scales well over cores, the processing capacity can be increased and calculations completed faster by adding additional cores. However, if the user scenario does not scale well over cores, it may not be beneficial to add more cores. In many cases the user experience is better with fewer, but faster, cores than with many, but slower, cores.

The test results below illustrate this point. Two different types of servers were used during the tests:

- Fast server – 12 cores @ 3.33 GHz, 144 GB RAM
- Wide server – 32 cores @ 2.27 GHz, 256 GB RAM

Performance results for a single user

The user-perceived performance results for a single user were as follows:

- Large, well-designed document: The wide server provided better performance as it had more clock cycles.
- Document with a diverse set of calculations: The servers performed the same.
- Document with less demanding calculations: The fast server performed better as it had higher clock frequency.
- Less than optimal document: The fast server performed better as it had higher clock frequency.

Performance results for many concurrent users

The user-perceived test results were similar to the ones above with the exception that the fast server saturated in CPU much earlier than the wide server. This was because the wide server had more clock cycles (32 cores * 2.27 GHz > 12 cores * 3.33 GHz) and more RAM, which resulted in a larger cache and less calculations being required.

Summary

The following is important to consider when it comes to how the QIX engine utilizes the CPU:

- Peaks with 100% CPU utilization are good as they indicate that the QIX engine utilizes all available capacity to deliver the responses as fast as possible.
- High average CPU utilization (>70%) is bad as it means that the system saturates and incoming selections in documents have to be queued prior to being served.
- The QIX engine processing capacity can be increased by adding more cores or by increasing the clock frequency. More processing capacity makes the engine handle load peaks in a robust manner.

References

For additional information on QlikView and Qlik Sense, refer to:

http://www.qlik.com/us/resource-library

150 N. Radnor Chester Road
Suite E120
Radnor, PA 19087
Phone: +1 (888) 828-9768
Fax: +1 (610) 975-5987

qlik.com