Qlik Q

Virtualization

Virtualization has become so widely accepted that for many companies it is a strategic mandate. The use of dedicated physical servers requires an exception if they are allowed at all. Qlik products have been shown to run very well on virtualized environments. Both Amazon and VMware virtualized environments have been tested with good performance. As with any additional layer between software and hardware, some performance degrade is introduced. Measured values show that average response times are from a few percent to about twenty percent slower in general for virtualized, whitelisted, well performing hardware. Not adhering to best practices will likely increase any degrade in performance.

Considerations

As with physical servers, it is vital to utilize hardware that is whitelisted in order to get expected performance. Virtualizing non-whitelisted servers will not improve performance.

BEST PRACTISES

- Always run Qlik products on whitelisted, well-performing hardware.

- If possible, place Qlik products on a physical hardware without virtualization.

- If possible, allocate an entire physical server to a single guest OS.

- If many guests are running on the same server, then be wary of resource saturation and whenever possible limit the resources available to non-Qlik Sense or QlikView guests.

- If many guests are running on the same server, dedicate a NUMA node for the Qlik instance.

- Oversubscription of CPU is beneficial for all Qlik Sense and Qlik View guests, but always reserve a minimum amount of CPU per guest OS.

- Always allocate all guest memory for guests.

- Make minimal required settings to Hypervisor (as per above) as they are often very adept at tuning for performance.

- Although Qlik Sense and QlikView are fully supported, it is important but not surprising to note that support of any interactions or issues that arise at the hardware or operating system layer as a result of the use of virtualization is the responsibility of the customer and/or the hypervisor vendor.

AVOID

- Never use poorly performing hardware (i.e. non whitelisted)

- Never allow floating allocations of memory to avoid ballooning.