Accelerating Storage Performance in Virtualized Servers using SR-IOV and MR-IOV

SNIA Solid State Storage Initiative

LUCA BERT
SNIA SSSI member, LSI

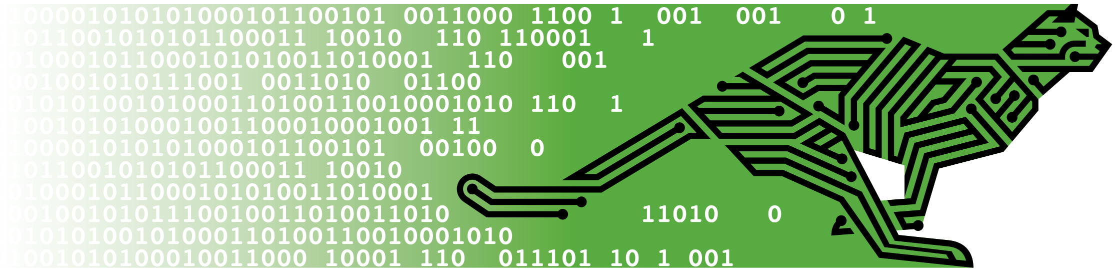
One aspect that is at the center of attention these days is how to get the most performance out of the storage subsystem. This has always been an issue, but has become much more prominent in the last couple of years for two primary reasons:

server virtualization has provided a means to scale up the number of operating systems sharing the same system and therefore taxing the storage subsystem proportionally
the advent of Solid State Drives (SSDs) has created a completely new storage segment that did not existent just 3 years ago. While these two technologies add significant value by themselves, jointly they actually create a new set of opportunities and challenges for storage, most of all in the performance arena. However, before we look at them we need to better understand what we really mean with the word "performance" as this can be misleading.

Most people are used to looking at IOPS (input/output operations per second) as the key performance indicator for storage. While this was a good indicator for mechanical hard disk drives (HDDs), it is no longer useful (or as useful) in an era dominated by SSDs. The main reason is that what the user really cares about is how fast can the storage subsystem return data upon receiving a request and, in mechanical hard drives, almost the entire time is spent by the mechanical parts which makes the response time of everything else (mainly the software stack) negligible.

Things are different with SSDs, as they respond so fast (one can safely assume that SSDs are up to 100x times faster than their mechanical HDD counterparts) that all the other components can no longer be ignored, which makes optimizing the access time of the software stack extremely important. While IOPS are still used to express a capability of the SSD, they do not tell the user anything about how much time is lost in the software stack. The optimization of the software stack has become essential to deliver performance in an SSD dominated world.

This is very different from the original approach used for server virtualization when mainstream adoption began several years ago. At that time, I/O was left alone and the main concern was the virtualization of CPU and system memory, by far the most expensive and underutilized resources in the computer. To ensure I/O was really left untouched, extra software layers were added to the storage path, mainly to mask the CPU/ memory virtualization layers. Each I/O card works as if it is connected to only one CPU, while in reality it is serving multiple virtualized CPUs. Each virtual machine (VM) assumes that it fully owns the I/O card that it is presented, while in reality, multiple VM are sharing the same resource.



The result is the addition of software layers that add both latency and increased CPU utilization. However, if we look at it in the optic of few years ago, that was fine as the big storage bottleneck is in the mechanical latency, not the software stack, so adding more software layers did not visibly impact performance.

Fast forward to 201x: SSDs are 100x faster than HDDs and software is no longer 1% of latency, but more in the 20% range. All this additional latency actually brings total latency up to 50% or higher. What was a fast device is now severely impaired by the virtualization stack that is wrapping it.

If we look it in the proper context, depending on configurations, these limitations are only visible when the I/O load is high, probably in the 50,000+ IOPS range, which is not necessarily an every day and every installation experience. (As a reference point, if normal HDDs were used, this would have required well over 100 hard drives to show up… instead of only 3 SSD's).

While this may not be your average server need, there are second level factors that are being affected. One of them is DRAM size: why is memory so big in servers today? The answer is generally simpler than people might think: performance (of course!). However, this is not because so much RAM is needed, but more because of the need to create very large cache to decouple the CPU (getting faster and faster) from hard disks (improving performance at a much slower rate). The problem though is that RAM is expensive, consumes a lot of power and can't scale above a certain size.

How about using SSDs to build such caches? Each transaction is by itself slower than DRAM, but the size can easily be 10- to-100x what could be implemented with DRAM, and the higher capacity compensates for the single I/O latency by improving the cache hit rate.

This is driving a shift in how servers are built as it helps in better distributing memory elements across DRAM, SSD and HDD in a way that is more efficient and economical.

User can't forget though about all the software layers that server virtualization built to separate I/O from the CPU complex. Now this is really in the way as using SSD as caching grows the I/O demand very steeply and the added latency is often the self-limiting performance factor.

All these issues were anticipated as part of server virtualization plans, though most O/Ss did not implement them. First and foremost, single root I/O virtualization (SR-IOV), part of PCI-SIG standard, was intended to address this aspect.

What SR-IOV does is turn the problem upside down: instead of keeping the I/O separate from the virtualized world wrapping it in software, let's invest in I/O so that it is aware of the virtualization environment and can directly communicate with all the VMs without any additional software layer in between.

Under the SR-IOV standard, each I/O card exposes itself as a standard PCIe card and a number of Virtual Functions (VF) that look like a set of virtual copies of the same device and can be used to connect directly with all the virtual machines. The advantage of this method is that it really brings virtualization to the I/O world so that they can actively participate to this new model. Most of all, they do not require any extra software layers on the data path so that I/O performance may scale linearly, unhindered by the extra layers of software currently present.

Benchmarks using SR-IOV devices can clearly show the differences. Lab tests with next generation storage controllers show performance improvements of up to 3X, with up to 4X less CPU utilization, demonstrating that the impact of the software layer, per I/O, is improved by almost an order of magnitude.

On the negative side, managing SR-IOV devices is complex and requires further investments in the management stack that may often be painful. But like every other investments, it is a matter of ROI: the more value it adds, the more likely somebody will be willing to invest in it. SSDs become the perfect catalyzer to help the technology reach the tipping point of viability.

SR-IOV is also getting significant help from a completely different angle. While SR-IOV is designed to address mechanism to share I/O within VM in a virtualized server, the same approach can also be used to solve the completely different problem of sharing the same I/O resource across multiple physical servers. For the most part, this is a completely different problem with almost the exact same solution. This is what is mostly known as a multi-root I/O virtualization (MR-IOV) environment. The reason MR-IOV is significant for the virtualization world is that the pain points addressed by SR-IOV (performance/ latency under very heavy I/O loads) are only affecting a limited set of solutions and are quite expensive to address.

At the same time, the pain points for MR-IOV (sharing I/O across multiple servers) are affecting the solution price point and are also less intrusive and risky, making it a much more viable solution in the short term. SR-IOV and MR-IOV often go hand in hand: same type of solution, addressing very different types of deployment models. However, both address the common pain points … that in a virtualized world keeping I/O secluded is not going to pay off in the long term.

Changes take time, but the pressure to get a better efficiency from I/O, be it in performance (SR-IOV) or connectivity (MR-IOV), is mounting and more solutions are being made available. Over time, I/O virtualization, in any form or fashion, seem like the only sustainable solution.

To learn more about SNIA's Solid State Storage Initiative, visit www.snia-europe.org/en/technology-topics/solid-state-storage/index.cfm