**Intellicache – What, How and When**

For today’s post I’d like to introduce the first guest blogger to post on speakvirtual.com, Jamie Lin! Jamie has been working in the IT industry for a long time and has a ton of knowledge across a broad spectrum of technologies. Jamie and I co-wrote the below post and I anticipate him contributing more content in the future.

**What is it?**
With the advent of XenServer 5.6 SP2 and XenDesktop 5 SP1, Intellicache became a configurable and supported feature for the Citrix VDI stack. You can use Intellicache with the combination of XenServer and XenDesktop Machine Creation Services (MCS). The basic idea behind Intellicache is that it allows you take some of the pressure off of your shared storage by offloading IO onto host local storage. As discussed before on [this site](http://speakvirtual.com/2011/02/11/vdi-io-characteristics/), IO in VDI environments has historically been one of the most overlooked and biggest technical challenges to any VDI rollout. With Intellicache, Citrix has sought to help alleviate this issue. See below for more on how this works and some additional considerations you won’t find in the documentation.

**How does it work?**
The folks over at [Citrix blogs](http://blogs.citrix.com/2011/06/22/xendesktop-and-local-storage-intellicache/) have already done an excellent job explaining how Intellicache works so we’ll try not to repeat too much here. However, at a fairly basic level, the offloading of IO is achieved by caching blocks of data accessed from shared storage by virtual desktops onto host local storage. So if Intellicache is enabled and a Windows 7 VM boots from a particular XenServer host, it will cache the roughly ~200MB accessed by the Operating System during the boot process on the hosts local storage. Subsequent VMs that boot up on that host will then access these blocks from local storage instead of the SAN. In addition, if you are using non-persistent images, your writes will occur exclusively on local storage as well. Persistent (aka “Dedicated”) images will write to local and shared storage. I think this image from the Citrix blog sums it up nicely:



You might also be wondering about storage space and what happens when you run out of room on your local storage. With both read and write caches happening on local storage, this is bound to happen from time to time. Luckily, Intellicache has taken this into account and will seamlessly fail back to shared storage in the event the local storage runs out of space. For more on “how it works”, see the link above or read more [here](http://virtualizationandstorage.wordpress.com/2011/06/03/xenserver-and-intellicahce-reducing-your-xendestop-storage-costs-by-50-80/).

**How to enable Intellicache**This [CTX article](http://support.citrix.com/article/CTX129052) explains the process of enabling Intellicache quite nicely. Basically it’s a two-step process. The first step occurs during the installation of XenServer itself, where you select “Enable thin provisioning (Optimized storage for XenDesktop)”. This option will change the default local storage type from LVM to EXT3. The next step occurs after the installation of XenDesktop itself where you create a connection to your host infrastructure. There is a checkbox that says “Use IntelliCache to reduce load on the shared storage device”. Selecting this checkbox will change the virtual disk parameter “allow-caching ( RW):” to “true” for any virtual desktop created that uses that particular catalog. You can verify this by issuing the command “*xe vdi-param-list uuid=<VDI\_UUID>*”:



You can also use the performance graphs to see Intellicache in action as well. In the performance tab, verify that “Intellicache Hits”, “Intellicache Misses” and “Intellicache Size” are all selected. If they are all selected, you will be able to monitor its usage as shown below:



While we’re uncertain as to if Citrix will support this or not, it is also possible to enable or disable Intellicache on a per VM basis. You do this with the following command, “*xe vdi-param-set uuid=VDI\_UUID allow-caching=true*”. You can then use the *param-list* command to view the parameters of that virtual disk to see that “allow-caching” is set to true. As it starts to utilize Intellicache, you’ll see Intellicache hits and misses for the VM start to appear in the performance tab.

While this may appear a bit complicated, it is important to note that the only thing necessary to implement Intellicache is selecting the Thin Provisioning option during XenServer install and selecting the checkbox when creating the catalog in XenDesktop. These command line options merely allow you more granular control on configuring Intellicache and allow you to see what it’s doing “under the hood”.

**Final Considerations**According to the [XenServer Installation guide](http://support.citrix.com/article/CTX130421), when you use Intellicache, *“The load on the storage array is reduced and performance is enhanced”*. Given that VDI IO is such a concern for most deployments, shouldn’t we just be enabling Intellicache all the time? Our answer is “no”. For while Intellicache does take IO pressure off of your shared storage array, you now have another IO concern to consider, IO on local storage. Remember what we said earlier about Intellicache failing back to shared storage if you run out of disk space on local storage? Well, what happens if your local storage can’t handle the *IO* being generated by the virtual desktops on your host, will it fall back to shared storage? The answer is no! There is no built in safeguard to prevent your VMs from using too much IO on local storage and thus, creating bad performance on any VM utilizing that hosts cache for reads and writes. This all but makes local storage SSDs an absolute necessity, particularly in blade environments where most vendors provide only two slots for local storage per blade. Given that most environments use N+1 redundancy for their host infrastructure, this means your local disks need to be able to handle the IO for the amount of VMs that can reside on two hosts!

There is another concern here as well that, as far as we can tell, is completely undocumented by Citrix. **When you use Intellicache, non-persistent VMs will be unable to XenMotion!** This makes complete sense when you think about it. How could a VM live migrate to another host when its write differentials reside on a separate host (the “Migrate to Server” option isn’t even present on these VMs)? What makes this so interesting is that it appears not to be mentioned by Citrix anywhere. It’s not in the installation guides, we couldn’t find it on edocs, and their blog on Intellicache only mentions XenMotioning in regards to *dedicated* desktops! This means you cannot perform any type of host maintenance that requires downtime while there are running non-persistent (aka “pooled”) desktops present on the host. Notice that we said “running”, not “in-use”, for a VM can still be running even though no one is using it. This caveat alone will be a deal-breaker for many considering the use of Intellicache and is something Citrix should have more openly documented.

With this post we wanted to give a broad overview on how Intellicache works and some general considerations before deploying XenDesktop with Intellicache. As we’ve seen, local host IO capability planning becomes paramount with the use of Intellicache and VM mobility is reduced. As it stands now, Intellicache use-case scenerios will be fairly limited and more features and configurable granularity needs to be built into the system before broader adoption can occur. We’ll dig deeper into Intellicache in future posts, in the meantime, let us know what you think!