Snapshots without Penalty

How Maxta Beats VSAN Snapshots

A DeepStorage Technology Brief

DeepStorage.net

Prepared for
Maxta

Introduction

Server and storage administrators have long used storage system snapshots to rapidly protect their data and enable recovery in a minute or two, compared to the hours it can take to restore from a copy-based technology like backup, providing a much shorter RTO (Recovery Time Objective). As workloads have shifted from physical servers, with dedicated storage volumes, to hypervisors, which store the data from multiple virtual servers in common datastores, volume-level and storage-system snapshots have lost much of their value.

Problems with volume-level snapshots

Taking snapshots at the volume, or datastore, level introduces two significant problems. The obvious problem is that, since each snapshot contains the data from multiple virtual machines, snapshots taken to protect one or two mission-critical servers in a datastore include data from all the less important servers in the datastore. Therefore, the snapshots will take up more disk space.

> Taking a SAN snapshot to protect a single VM is like swatting flies with a bulldozer.

Less obvious but more significant is the consistency of the data in the snapshot. Since a snapshot preserves data at a specific point of time, it's important to ensure that the data being preserved is consistent. Modern applications, especially database servers, buffer some data in memory. In order to ensure application-consistent snapshots, the application should be quiesced, either through a script or the Windows Volume Shadowcopy Service (VSS), flushing these buffers to disk, therefore making the data on the disk consistent.

Unfortunately, most applications can only run in the quiesced state for just a few seconds while a snapshot is being taken. That means it's not possible to quiesce multiple VMs in a datastore before taking a snapshot, as the timing is too tight for seven VMs to all quiesce within a few seconds of each other.

As a result, vSphere administrators who want the fast recovery that application-consistent snapshots provide are forced to keep their mission-critical VMs in private datastores, complicating their storage management.

Enter the VM-level snapshot

One of VMware's primary claims about their newly released VSAN is that, unlike traditional SAN arrays, it takes snapshots at the VM, rather than datastore, level. The problem, of course, is that VMware's snapshots are, by any measure, inefficient.

vSphere snapshots use child disks, also called delta logs, to log the changes that are made to the disk while the snapshot exists. When the snapshot is created, the system creates a .delta.vmdk file that behaves very much like the redo log of a database engine.

The delta log file is formatted as a sparse disk that contains the new data which is written to the logical drive while the snapshot is in force. When new data is written to the snapshotted VM, vSphere writes that data to the delta log file, preserving the state of the disk at the time the snapshot was taken in the parent .VMDK. As with most snapshot mechanisms, the snapshot data—in this case, the delta log file—grows with the amount of new data written to the snapshotted disk.

Since the base .VMDK file is time-locked to the instant the snapshot was created, when an application reads data from the virtual disk, the system must first check to see if the snapshot has the newest version of the block it's trying to read, then read the data from the parent disk if it doesn't. As the delta file grows, the system has to do more work to satisfy each read request.

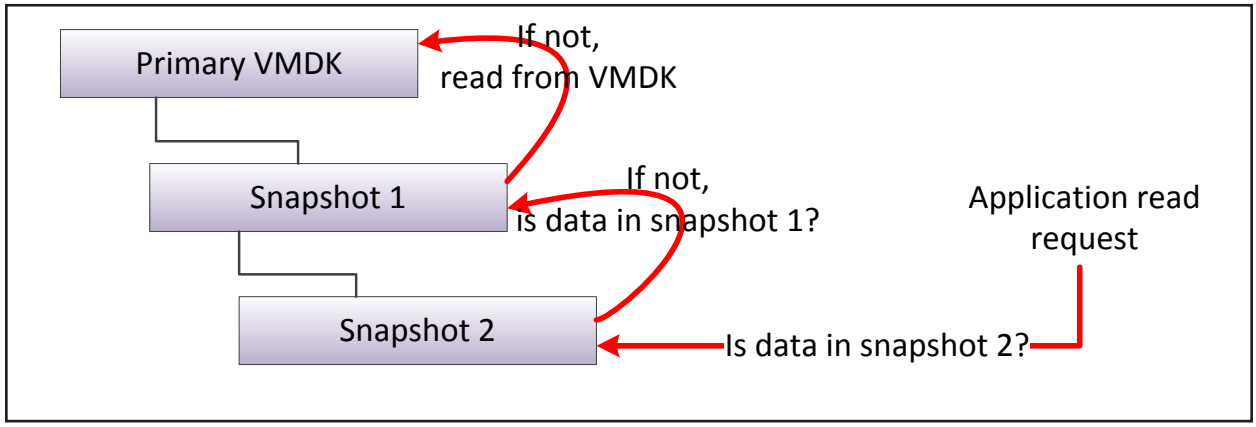

Figure 1. Read I/O amplification with VMware snapshots.

This read I/O amplification, and the resulting performance loss, is even worse if the user wants to maintain multiple snapshots of the same VM. In that situation, the system has to check in the delta file for the latest snapshot, then the file for the next newest, and so on. The more snapshots in the chain, the bigger the performance impact. We've seen VMs with multiple snapshots get less than half the storage performance they had without any snapshots.

To add insult to injury, when the snapshot is deleted, the system has to merge the snapshot data back into the parent disk. Not only does this create a substantial amount of I/O, placing a load on the underlying storage system, but it paradoxically requires that there be additional free disk space, as the system uses what is essentially an additional snapshot to buffer the delta data as it's written back to the parent disk.

As a result, VMware administrators really only use snapshots for those tasks where the snapshot remains in existence for a short period of time, especially to enable image backups via the vStorage API for data protection. A VMware administrator who tried to use snapshots as a more frequent data-protection technique to maintain multiple restore points would soon see their storage performance fall off so dramatically they would quickly abandon the attempt.

Better snapshots through metadata

Like many modern hybrid storage systems, Maxta's Storage Platform manages the mapping of the logical blocks within the files it holds to physical blocks on the underlying storage through the use of metadata. Each file stored on an MSP cluster has a metadata structure that defines which blocks in the storage hold each logical block in the file. When a user creates an MSP snapshot, all the system has to do is create a copy of the metadata for the virtual machine's files.

When new data is written, the system writes to available space in the file system and updates the metadata for the parent copy of the virtual machine. When an application on the VM reads data, the system determines which data block to read by looking it up in the file's metadata, just as it would if the snapshot didn't exist.

While creating a snapshot does prevent the system from overwriting any of the blocks that make up the snapshot, meaning that even metadata snapshots grow as applications write new data to the virtual disk, metadata snapshots have no impact on the performance of the virtual machine they protect.

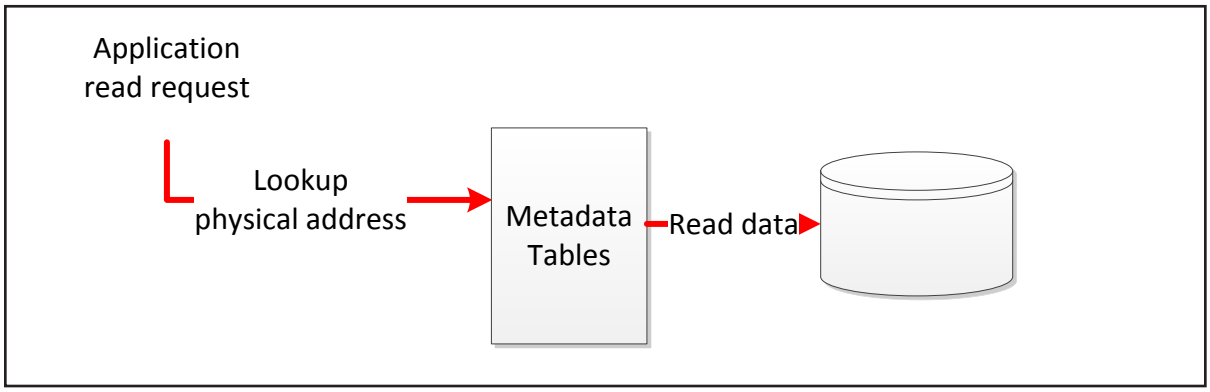

Figure 2. Read process with metadata is the same regardless of snapshots.

Unlike vSphere's—and, therefore, VSAN's—log-based snapshots, Maxta's metadata snapshots are independent. Each snapshot is a metadata list of the blocks that made up the virtual machine at the time the snapshot was taken. Neither reads nor rights to the active disk require accessing the snapshot metadata; they proceed by accessing the primary file's metadata exactly the same as if there were no snapshots. Administrators can create as many snapshots of a single VM as they like without the performance penalty nested snapshots create on VSAN.

Bring in the clones

As we've already seen, a snapshot in a modern storage system like Maxta's MSP is simply a copy of the metadata that described the volume—or, in MSP's case, the virtual machine—that was the target of the snapshot. When an administrator clones a VM through the Maxta vCenter plugin MSP, rather than creating a read-only copy of the VM's metadata, as it would to create a snapshot, it creates a writeable copy of the metadata to create the new VM.

If, for example, an administrator needed to create four web servers, he could create a master VM and clone it to create the three additional web servers in just a few minutes through the MSP vCenter plug-in. The three clones will only take a few megabytes of disk space for the identities and other differences between the four machines. Technically, this is similar to the linked clones used for VDI, but since it's performed at the storage layer, it's transparent to vSphere.

Were that same administrator using VSAN, they could use vCenter to create the clones just as easily, but to create each clone, vSphere must actually copy the full VM or template. Creating four full copies of the VM, rather than MSP's metadata clones, will not only take four times as much disk space but will also create a significant amount of disk and network traffic to create the clones, which may significantly impact the performance of other applications accessing the VSAN datastore. Since copying the whole VM will take many times as long as making a metadata copy, the whole process will take hours, not minutes.

The last big advantage of metadata clones is that they make MSP's flash layer more effective. When 20, or 200, Windows servers are running in a vCenter cluster, several of them will be accessing the same DLLs; since these servers were created from the same master, this data is only stored in the MSP datastore once. In a system using full copies, there may be several copies of the same data hot enough to be in flash. Since MSP only has one copy of that data, it will only have one copy in flash, leaving more flash available for caching other data.

Conclusion

The time has come for the IT industry to shift the logical unit of data management from the logical volume, or LUN, to the virtual machine. Providing data services like snapshots, clones, and replication at the VM level allows system administrators to tailor their data protection levels to the requirements of individual workloads.

Let us, however, not forget that all data services are not created equal. Inefficient snapshots, like VMware's log-based snapshots, have such a large impact on performance that they're really only useful as a backup enabler. Highly efficient snapshots that have a minimal impact on performance can be used as a much longer-term constituent of a data-protection policy.

Maxta's MSP offers highly optimized metadata snapshots and clones on a per-VM basis, on top of a distributed, deduplicated, hybrid datastore. It runs as a virtual machine on each VMware ESXi host that offers storage to the datastore pooling all the local storage, including both flash and spinning disks, replicating data across multiple nodes for resiliency.

About DeepStorage

DeepStorage, LLC. is dedicated to revealing the deeper truth about storage, networking and related data center technologies to help information technology professionals deliver superior services to their users and still get home at a reasonable hour.

DeepStorage Reports are based on our hands-on testing and over 30 years of experience making technology work in the real world.

Our philosophy of real world testing means we configure systems as we expect most customers will use them thereby avoiding "Lab Queen" configurations designed to maximize benchmark performance.

This report was sponsored by our client. However, DeepStorage always retains final editorial control over our publications.