

TECH NOTE

Nutanix Clones and Snapshots

Copyright

Copyright 2022 Nutanix, Inc.

Nutanix, Inc.
1740 Technology Drive, Suite 150
San Jose, CA 95110

All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. Nutanix and the Nutanix logo are registered trademarks of Nutanix, Inc. in the United States and/or other jurisdictions. All other brand and product names mentioned herein are for identification purposes only and may be trademarks of their respective holders.

Contents

1. Introduction.....	4
Audience.....	4
Purpose.....	4
Document Version History.....	4
2. Nutanix Clones and Snapshots.....	5
Solutions.....	6
Clones.....	13
Clones and Snapshots.....	16
Cloning Multiple VMs from a Template.....	18
3. Nutanix Recommendations.....	23
About Nutanix.....	24
List of Figures.....	25

1. Introduction

Audience

This tech note is part of the Nutanix Solutions Library. We wrote it for everyone who wants to learn more about the Nutanix distributed storage cloning and snapshot features, how they work, and the Nutanix recommendations for using them.

Purpose

In this document, we cover the following topics:

- Introduction.
 - Overview.
 - Clones and snapshots in action.
 - Recommendations.
-

Document Version History

Version Number	Published	Notes
1.0	May 2020	Original publication.
1.1	June 2021	Refreshed content.
1.2	August 2022	Updated links.

2. Nutanix Clones and Snapshots

Nutanix AOS distributed storage provides elegant clone and snapshot functionality using a redirect-on-write algorithm. vDisks at the Nutanix layer back the files that AOS presents to VMs. Each vDisk in the system is hosted, or owned, by a Nutanix node's Controller VM (CVM). The CVM typically runs on the same Nutanix node as the VM using its vDisks. vDisks are made of blocks, which are 1 MB chunks of virtual address space.

Although each vDisk has multiple internal identifiers, we use only one of them—`vdisk_id`—throughout this document for simplicity.

A Nutanix vDisk can be in one of the following modes:

- Mutable (read/write). You can perform both reads and writes to and from the vDisk.
- Immutable (read-only). You can't make any changes to the vDisk. Effectively, it's in read-only mode.
- SnapshotImmutable (read-only). You can't make any changes to the vDisk. This mode is used during backup and disaster recovery situations.
- ImmutableShadow (read-only). You can't make any changes to the vDisk. This mode is used with VMware Horizon linked clones.

Pithos is the internal vDisk configuration manager that is responsible for vDisk (AOS file) configuration data.

During both clone and snapshot operations, the system puts the original vDisk in read-only mode and creates at least one new vDisk. The newly created vDisk and its metadata structure include internal references to the original vDisk, as the two disks have a parent-child relationship. After taking a snapshot, AOS can copy metadata from the original vDisk to the live vDisk if needed.

In the following examples, we write new data to vDisk blocks where no other data has been written. Overwrites in these examples use the same assumption:

that there is no other data in the same vDisk block. In these cases, the system doesn't copy metadata from the parent to the child vDisk block map.

For improved readability, in some of the following diagrams we've omitted steps where the logical entities created during these steps are cleaned up as quickly as possible in the background.

Snapshots

This section describes the data and metadata structure for VM snapshots. In this example, we have one VM (VM1) with one vDisk attached. This vDisk is one mount point in the VM1 OS; for example, / in a VM based on the Linux OS or c:\ in a VM based on the Windows OS.

Let's walk through how data and metadata evolve during a snapshot scenario.

At the starting point, the VM is running and its vDisk is in read/write mode (RW in the following figures).

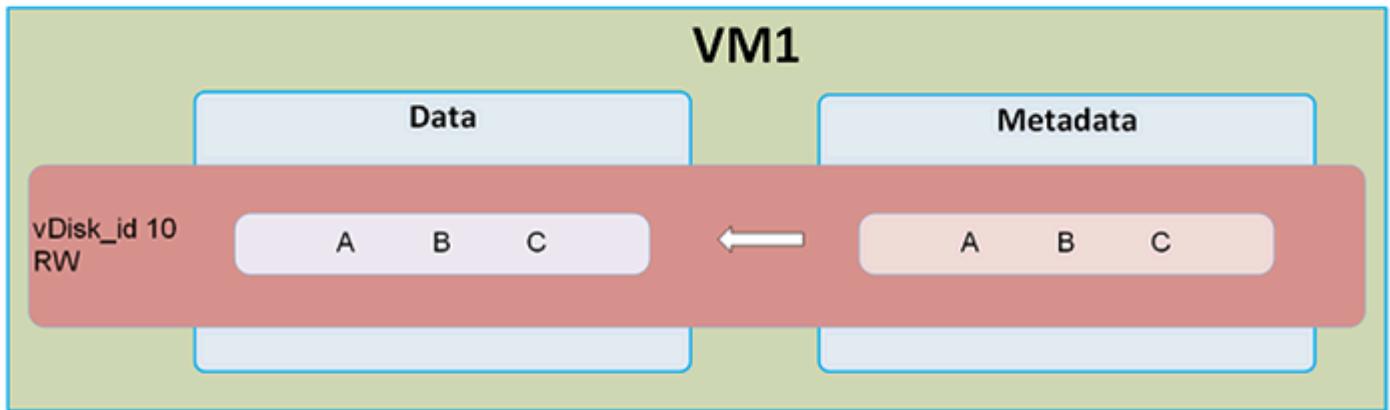


Figure 1: Snapshot: Starting Point

The system takes a snapshot, creating vDisk_id 22 in Pithos, the AOS vDisk configuration manager.

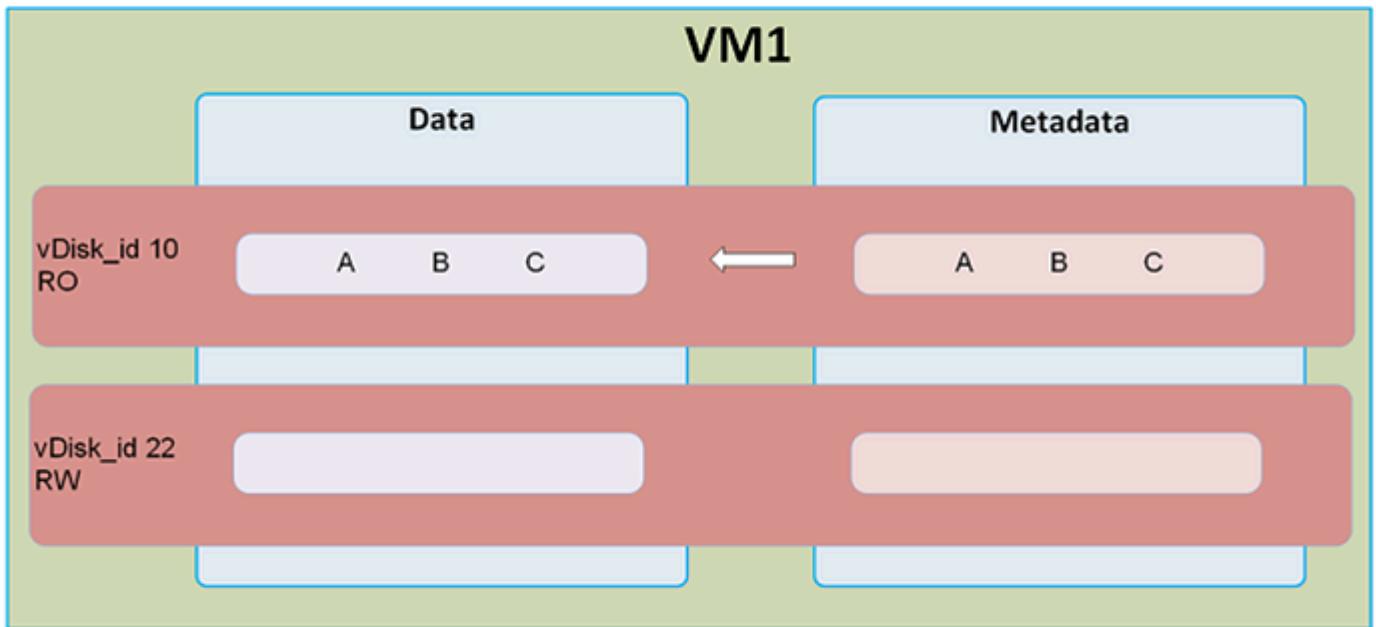


Figure 2: Snapshot: Initial Snapshot

All read operations are issued in the context of vDisk_id 22, but they're served from the metadata of vDisk_id 10.

When the VM writes new data, D, it places this data in vDisk_id 22, which is now allocated on disk.

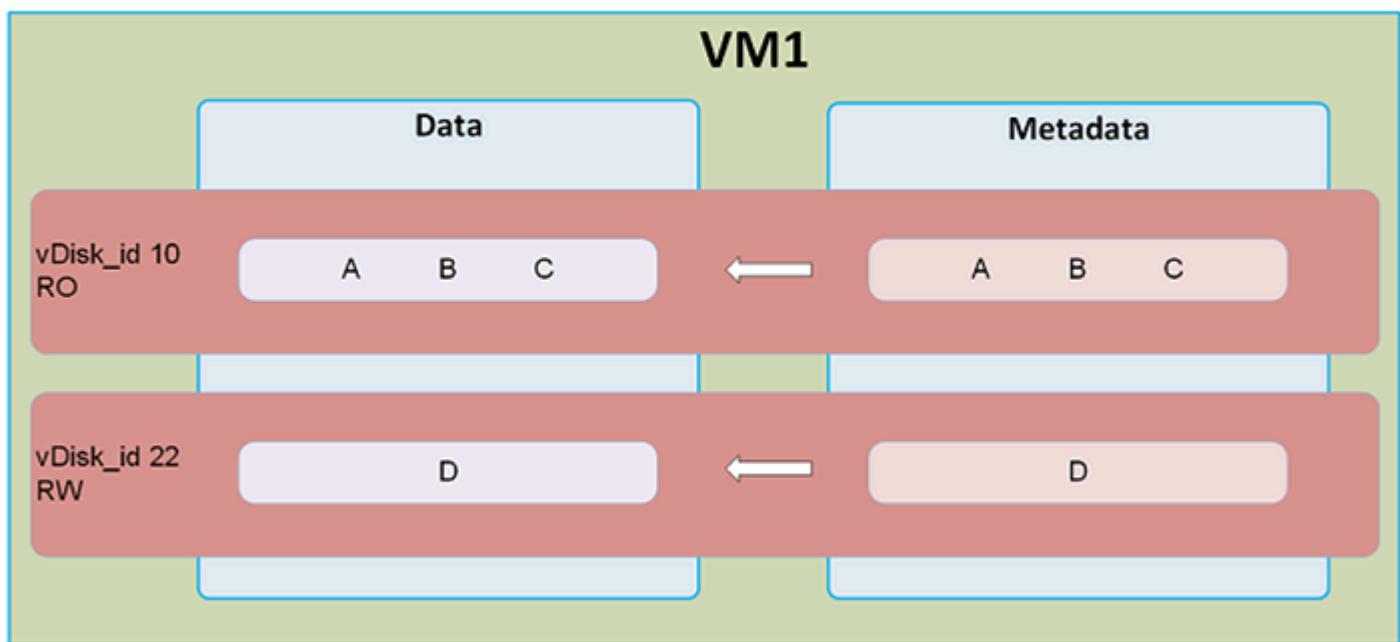


Figure 3: Snapshot: First Write

When the VM overwrites data A with new data, the most recent copy of A goes to vDisk_id 22 with data D.

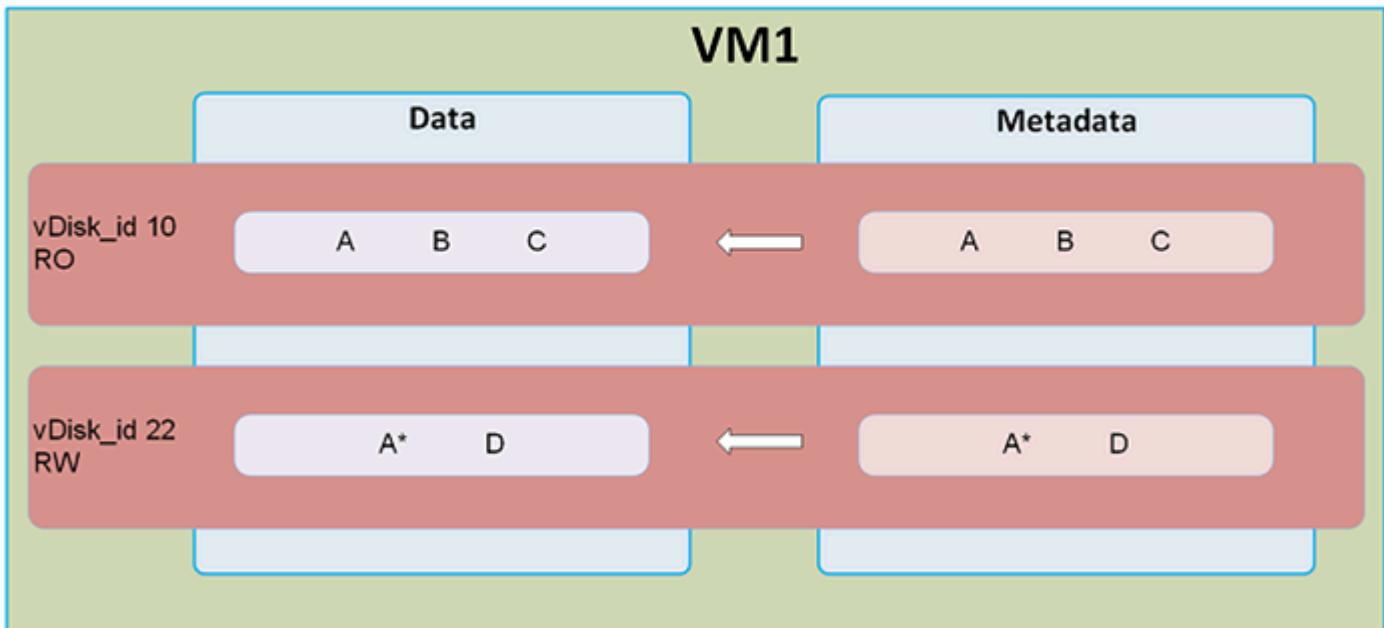


Figure 4: Snapshot: Data Overwrite

Depending on the nature of the overwrite—overwriting the entire block or partially overwriting the block—AOS might have to complete more than just a write operation. If the VM overwrites the entire block, it only needs to write the new data. If the VM only partially overwrites the block, AOS must copy the metadata that isn't overwritten from vDisk_id 10 to vDisk_id 22 so the entire vBlock can be addressed from the child vDisk, vDisk_id 22.

Taking another snapshot creates vDisk_id 33 in Pithos.

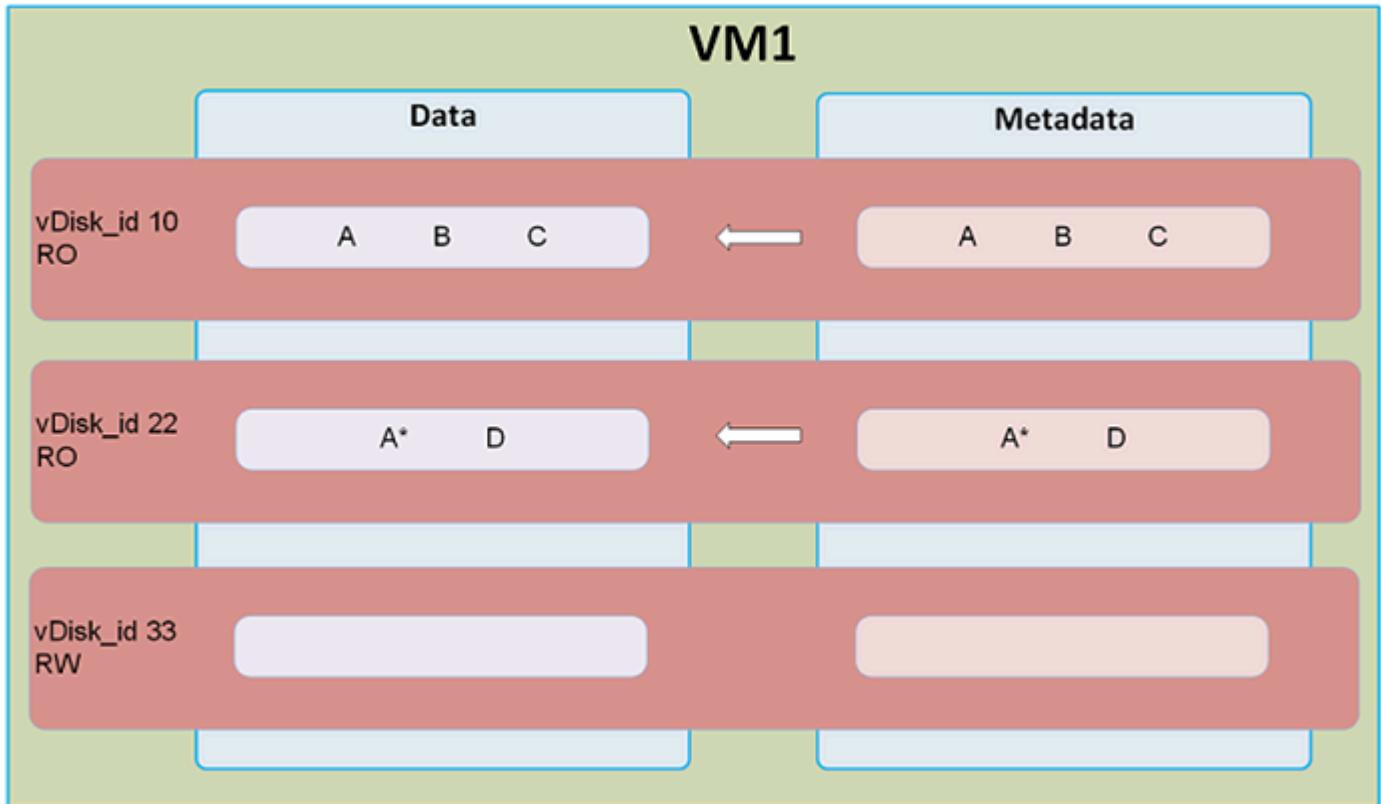


Figure 5: Snapshot: Second Snapshot

When VM1 writes new data, it places this data in vDisk_id 33, which is now allocated on disk.

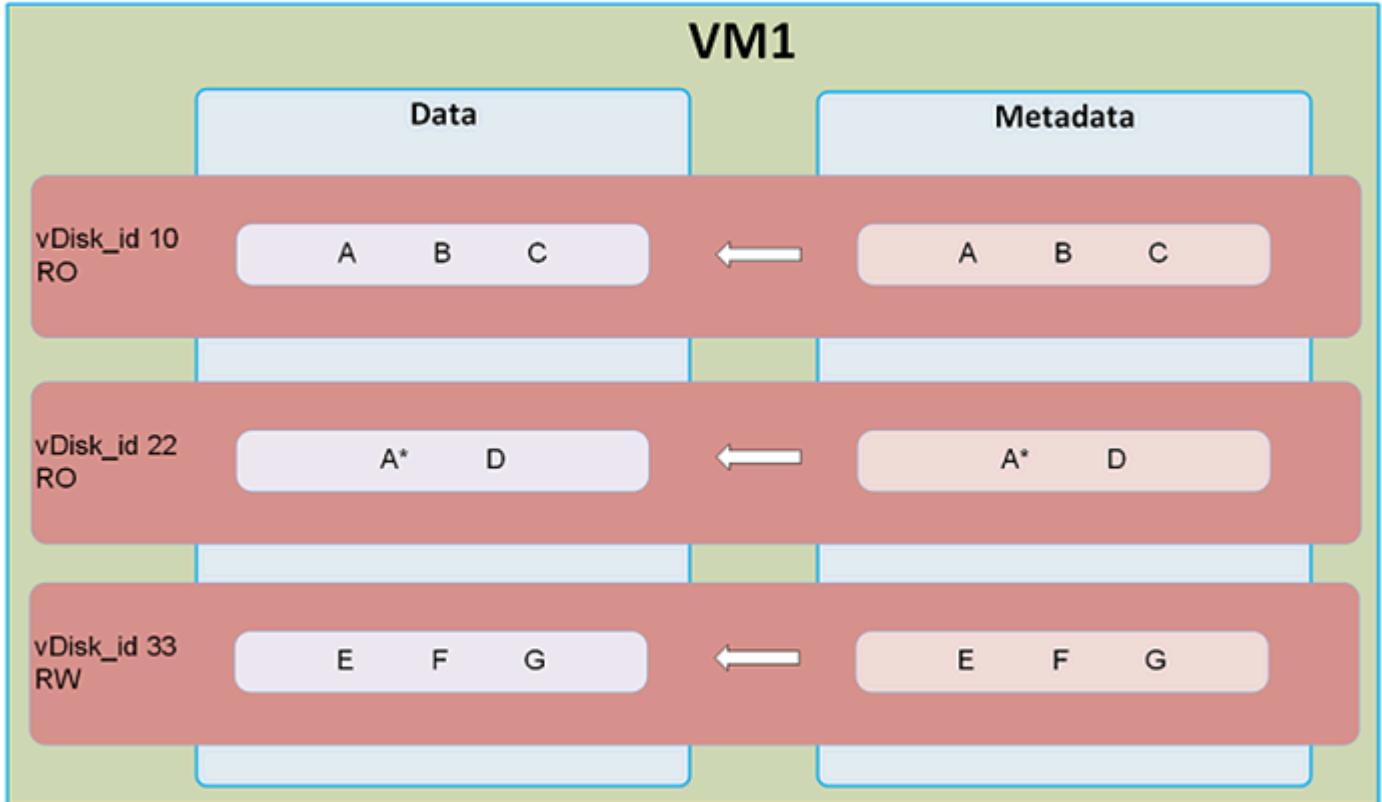


Figure 6: Snapshot: Write to Second Snapshot

Snapshot Maintenance Task

To limit the number of metadata sources that must be read per vDisk to find the data requested by the VM, AOS performs snapshot metadata maintenance. Expanding on the previous figure, the following diagram shows what the vDisk structure looks like after a snapshot maintenance task.

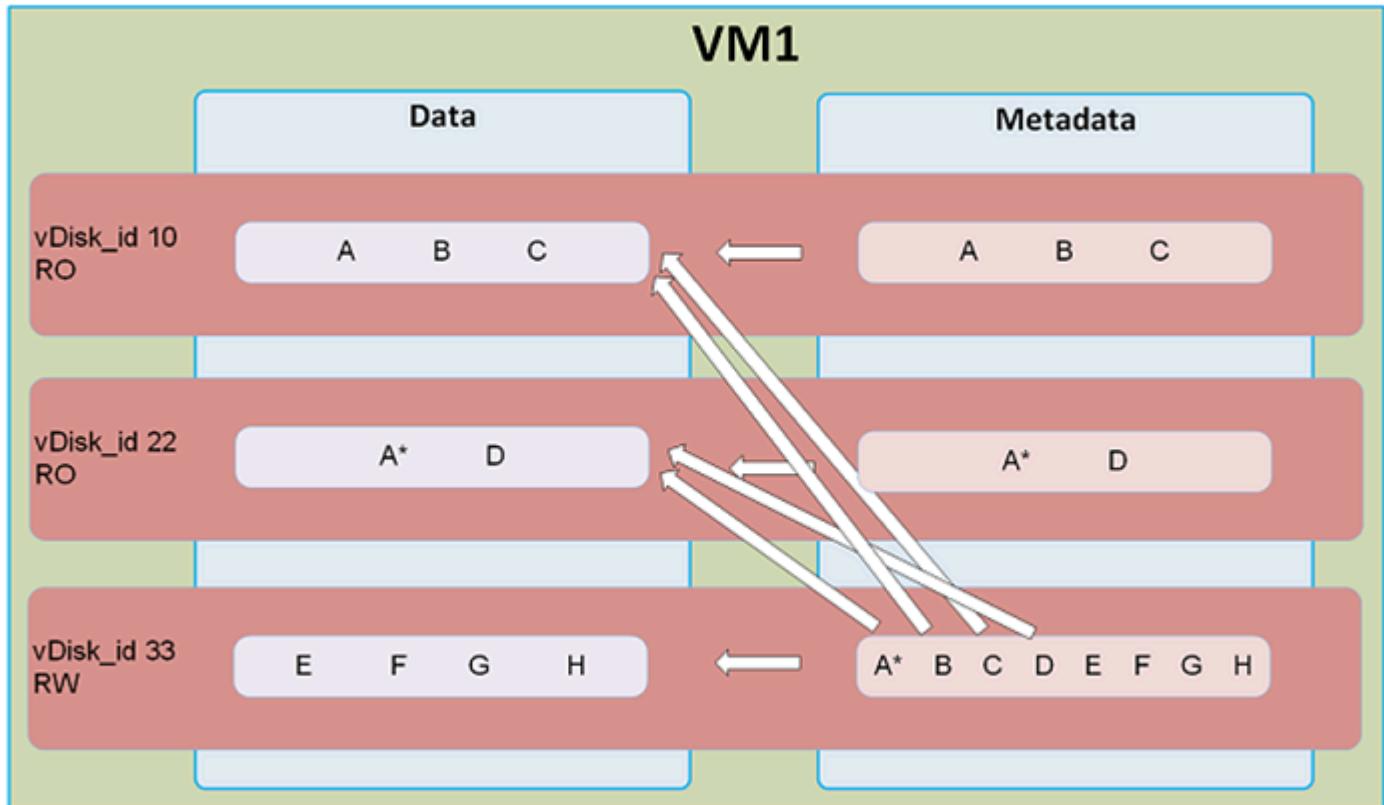


Figure 7: Snapshot: Maintenance Tasks

Clones

This section describes the data and metadata structure for VM clones. In this example, we have two VMs:

1. VM1, which is the original VM we want to clone. VM1 has one vDisk attached. This vDisk is one mount point in the VM1 operating system; for example, / in a Linux-based VM or c:\ in a Windows-based VM.
2. VM2, which results from cloning VM1.

Let's look at how data and metadata evolve during a clone scenario.

At the starting point, VM1 is running and its vDisk_id 10 is in read/write mode (RW in the following figures).

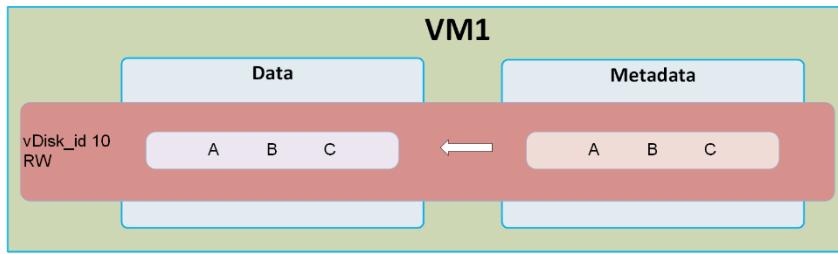


Figure 8: Clone: Starting Point

In this example, VM1 is off. However, you can clone the VM while it's running if necessary.

As shown in the previous figure, VM1 consists of vDisk_id 10. When the system performs a clone operation, it creates VM2 and its vDisk, vDisk_id 28, in Pithos. Because vDisk_id 28 has an internal reference to its parent, vDisk_id 10, it refers to vDisk_id 10 for its information. The black arrow between the VMs illustrates this relationship.

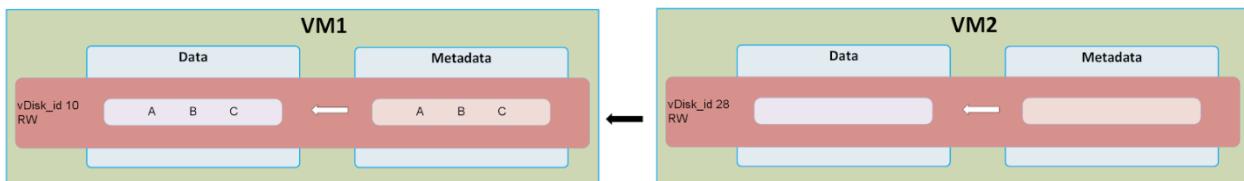


Figure 9: Clone: Initial Clone

VM2 performs its first write operation and places its data, Z, in vDisk_id 28, which is now allocated on disk.

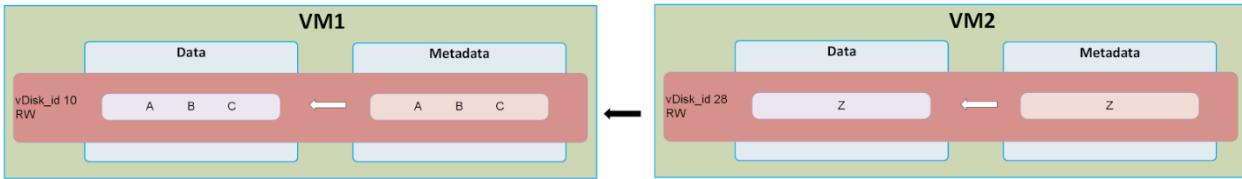


Figure 10: Clone: First Write

The system starts VM1. VM1 writes data D, with the following results:

- vDisk_id 10 becomes anonymous read-only (RO in the following figures), and the system performs a snapshot-on-write operation.
- VM1 places its data in a new vDisk, vDisk_id 22, in Pithos and allocates it on disk.

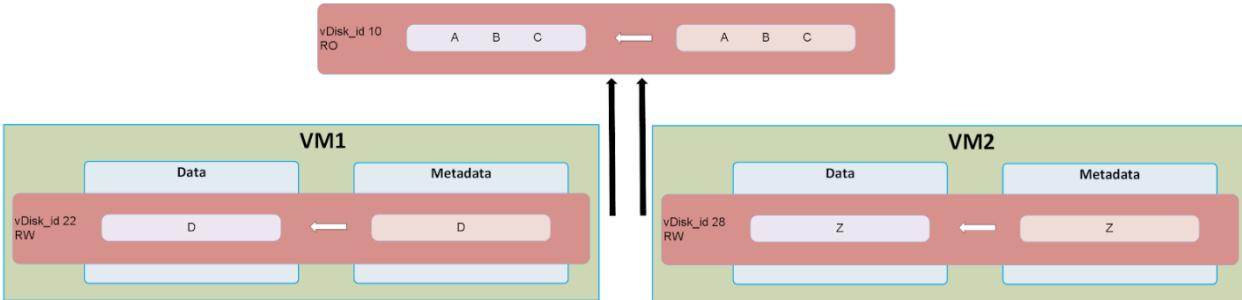


Figure 11: Clone: Source Writing Data

Both VMs overwrite data:

- VM1 overwrites data A from its read-only vDisk, so VM1's data A and D are now in VM1 vDisk_id 22.
- VM2 overwrites data A from VM1, so VM2's data A and Z are now in VM2 vDisk_id 28.

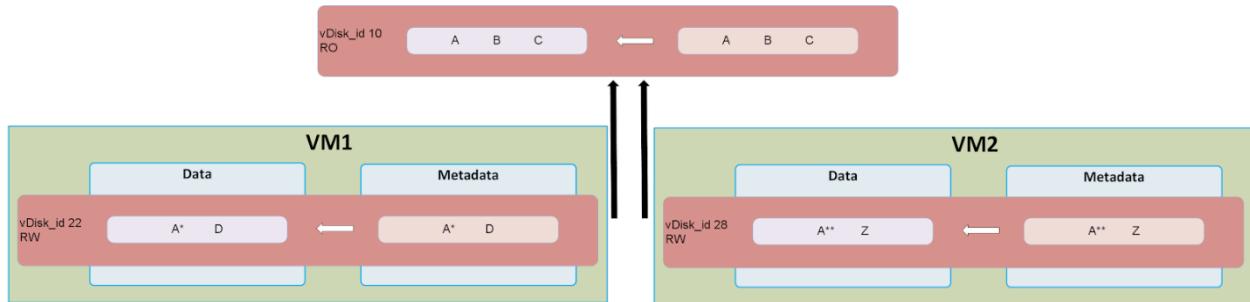


Figure 12: Clone: Overwrite

AOS applies the same metadata maintenance logic described in the snapshot scenario for each VM in the cloning scenario.

Clones and Snapshots

This section expands on the final figure in the Clones section and illustrates what happens when you take a snapshot of VM1 and VM2.

When you take a snapshot of VM1, vDisk_id 22 for VM1 enters read-only mode (RO in the following figures) and the system creates a new vDisk, vDisk_id 33, in Pithos.

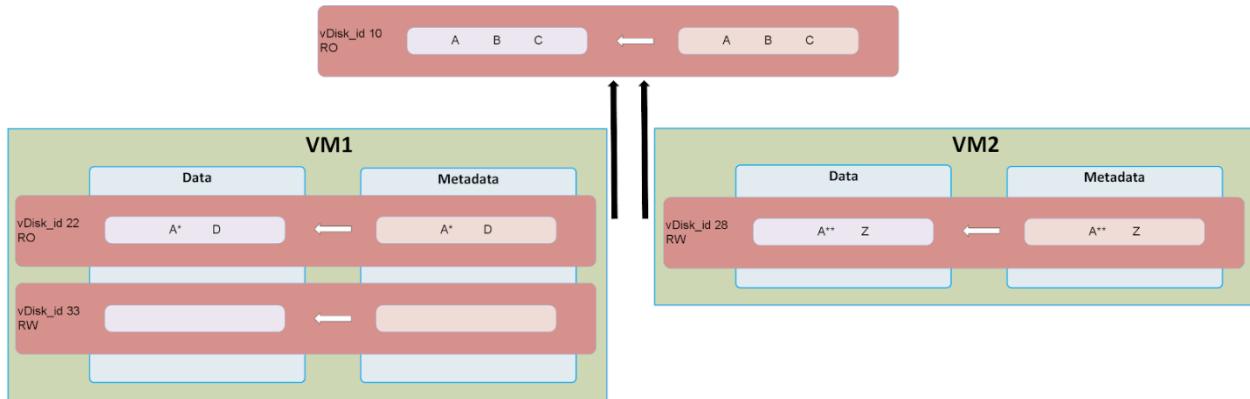


Figure 13: Clones and Snapshots: Taking a Second Snapshot of VM1

When VM1 writes new data, it places this data in vDisk_id 33, which is now allocated on disk.

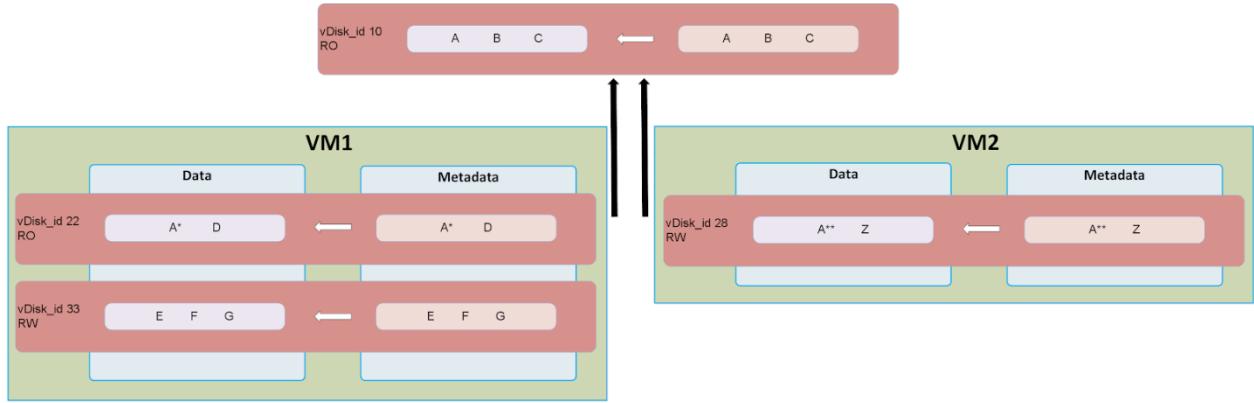


Figure 14: Clones and Snapshots: First Write to Second VM1 Snapshot

When you take a snapshot of VM2, vDisk_id 28 for VM2 enters RO mode and the system creates a new vDisk, vDisk_id 38, in Pithos.

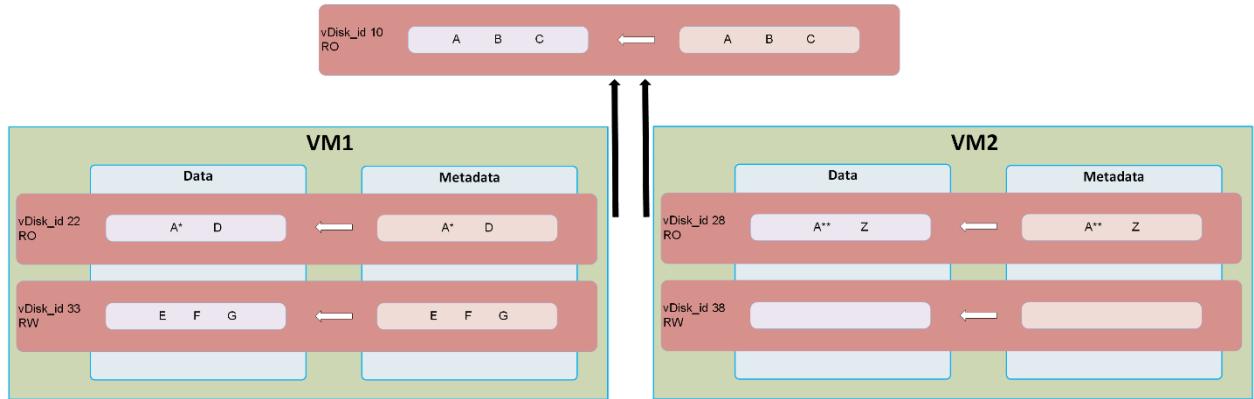


Figure 15: Clones and Snapshots: Taking a Second Snapshot of VM2

Both VMs overwrite data B and create new data K. The system allocates VM2's vDisk_id 38 on disk.

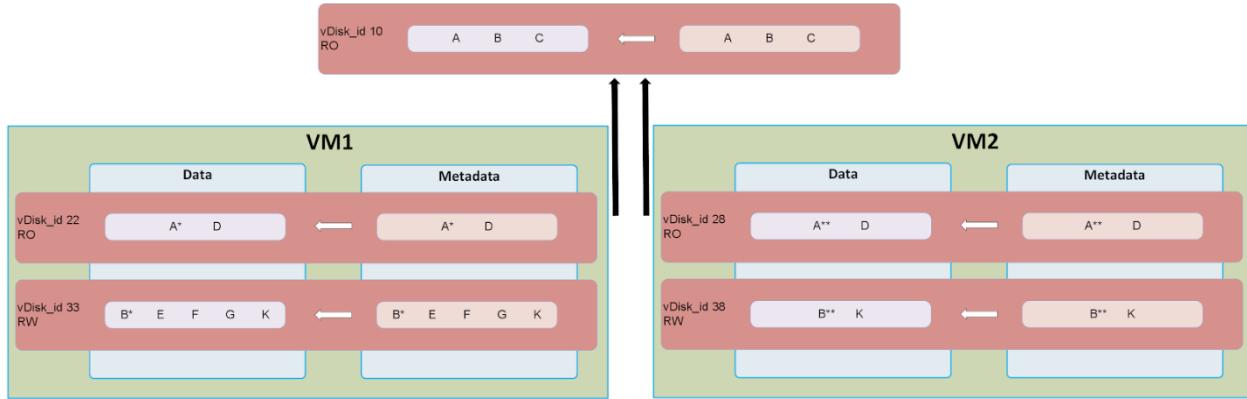


Figure 16: Clones and Snapshots: New Writes and Overwrites

Cloning Multiple VMs from a Template

This section describes the differences between cloning VMs from a snapshot and cloning VMs from a template VM.

Typically, a template VM is what we call a VM where you have installed the OS, management software, and, in some cases, the application binary, which we can use to create other VMs.

Cloning VMs from a Template

Creating clones from a template VM results in the vDisk structure described in the next two subsections. Each clone operation occurs in two steps. A template VM is typically off during the cloning process (though it can be running if required), then restarted if required when the cloning operation is complete.

Clone Scenario 1

This scenario discusses what happens when a clone operation creates multiple VMs from a template VM, the cloned VMs write data, and the template VM starts and writes data.

When the system creates the VMs Clone1, Clone2, and Clone3, it enters one vDisk for each VM (vDisk_id 28, 38, and 48) in Pithos. vDisk_id 28, 38, and 48 are allocated on disk when the first write to them occurs. Because vDisk_id 28, 38, and 48 each have an internal reference to their parent, vDisk_id 10, they

refer to vDisk_id 10 for their information. The black arrow between the VMs illustrates this relationship.

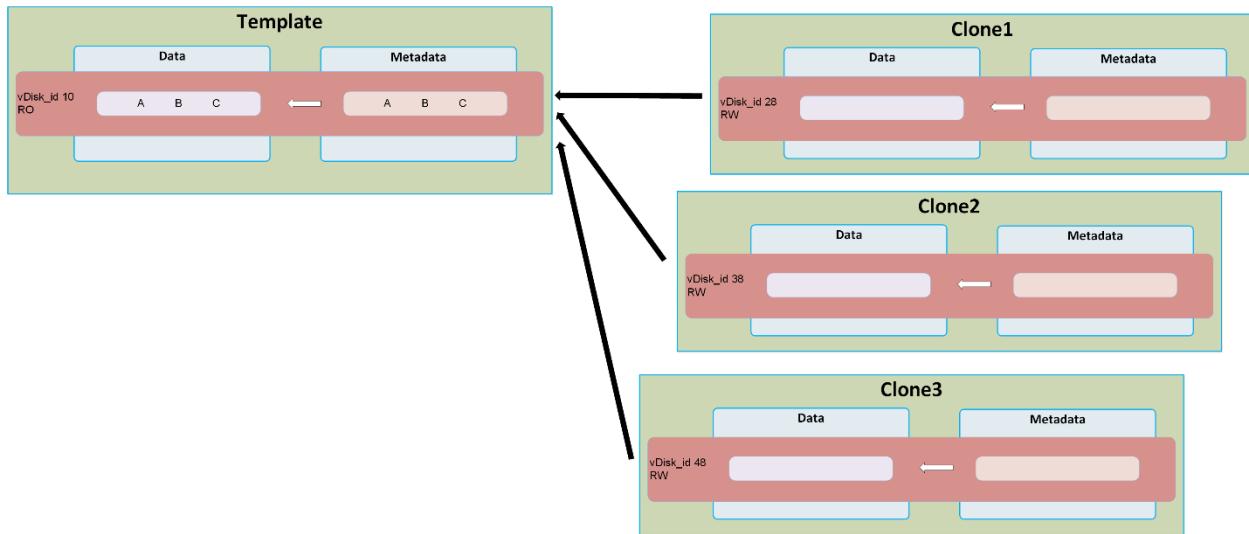


Figure 17: Cloning from a Template: First Clones

The system starts the template VM, Clone1, and Clone2, and all three VMs write data.

- vDisk_id 10 becomes anonymous read-only (RO in the following figures), and the system performs a snapshot-on-write operation.
- The template VM places its data, D, in a new vDisk, vDisk_id 22, in Pithos and allocates it on disk.
- Clone1 VM places its data, Y, in vDisk_id 28, which is allocated on disk.
- Clone2 VM places its data, Z, in vDisk_id 38, which is allocated on disk.

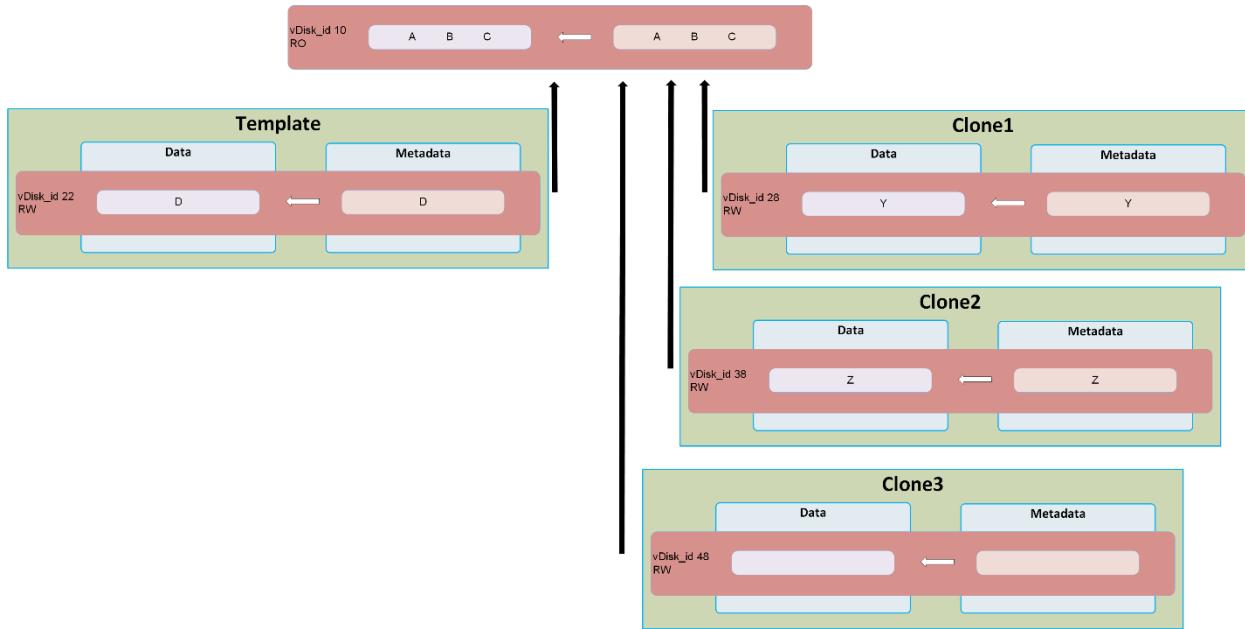


Figure 18: Cloning from a Template: New Writes to Template, Clone1, and Clone2

Clone Scenario 2

This scenario discusses what happens when a clone operation creates VMs from a template VM and the template VM and the clones start and write data.

When the system creates the **Clone1** VM, it enters **Clone1** VM's **vDisk_id 28** in Pithos. **vDisk_id 28** is allocated on disk when the first write to it occurs. Because **vDisk_id 28** has an internal reference to its parent, **vDisk_id 10**, it refers to **vDisk_id 10** for its information. The black arrow between the VMs illustrates this relationship.

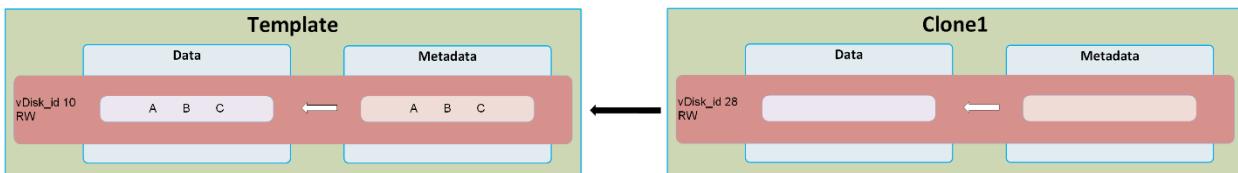


Figure 19: Cloning from a Template: First Clone

The system starts the template VM and both VMs write data:

- **vDisk_id 10** becomes anonymous read-only (RO in the following figures), and the system performs a snapshot-on-write operation.

- The template VM places its data, D, in a new vDisk, vDisk_id 22, which is created in Pithos and allocated on disk.
- Clone1 VM places its data, Z, in vDisk_id 28, which is allocated on disk.

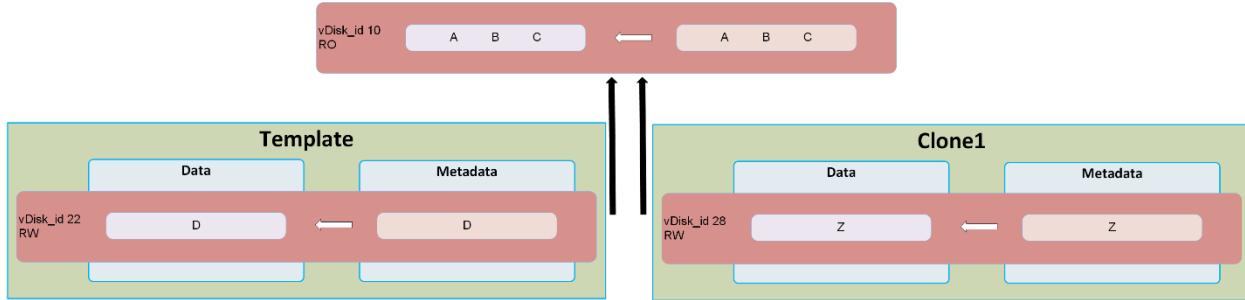


Figure 20: Cloning from a Template: New Writes to Template and Clone1

When the system creates the Clone2 VM, it enters Clone2 VM's vDisk_id 38 in Pithos. Because vDisk_id 38 has an internal reference to its parent, vDisk_id 22, it refers to vDisk_id 22 for its information. The black arrow between the VMs illustrates this relationship.

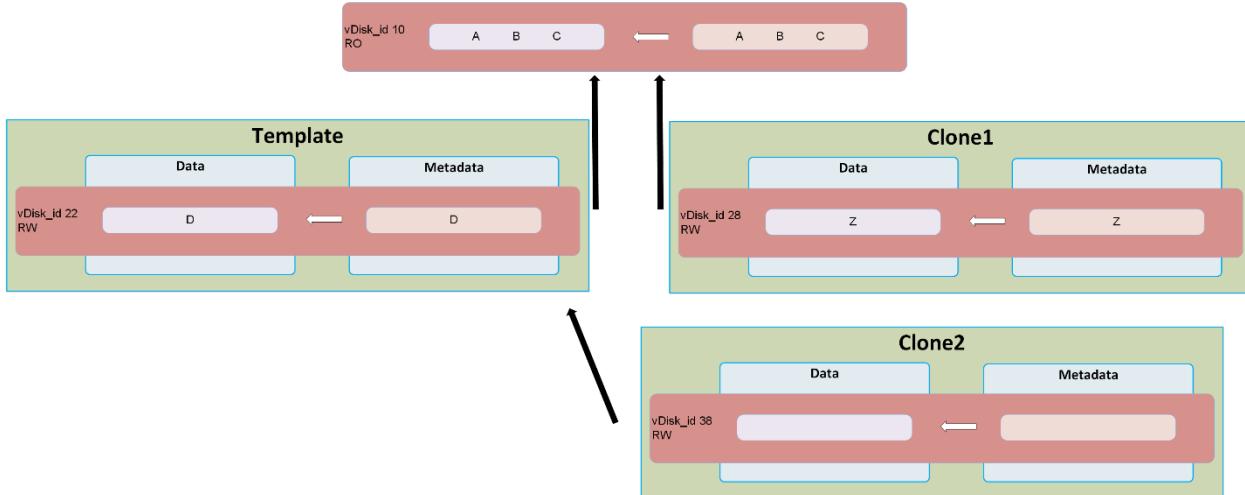


Figure 21: Cloning from a Template: Second Clone

Both the template VM and Clone2 VM write data:

- vDisk_id 22 becomes anonymous read-only (RO in the following figures), and the system performs a snapshot-on-write operation.

- The template VM places its data in a new vDisk, vDisk_id 40, which the system creates in Pithos and allocates on disk.
- Clone2 places its data, Y, in vDisk_id 38.

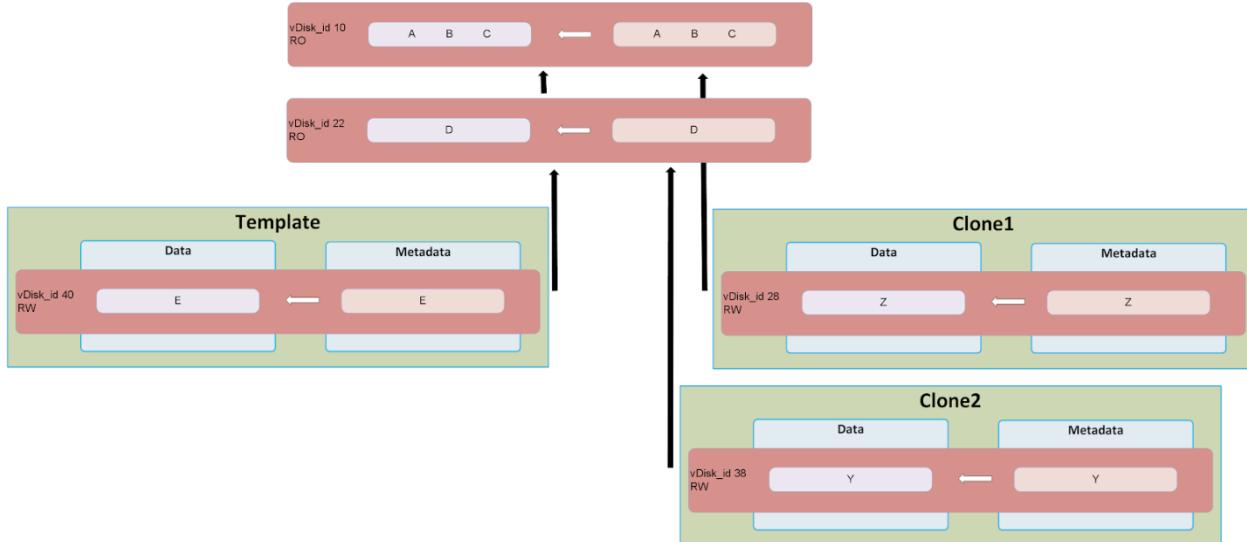


Figure 22: Cloning from a Template: New Writes to Template and Clone2

Cloning VMs from a Snapshot

Taking a snapshot of the template VM and cloning from that snapshot results in a vDisk structure that looks like the following figure.

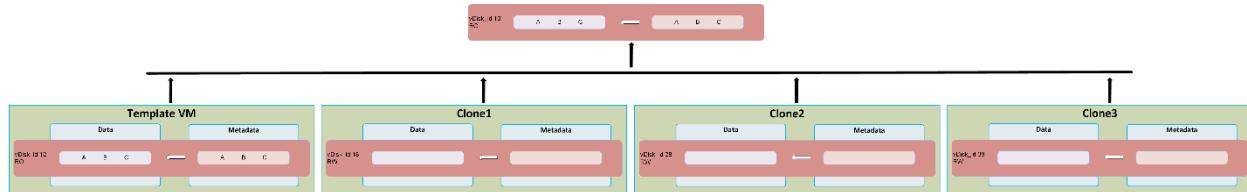


Figure 23: Cloning VMs from a Snapshot

The advantage of this method is that all clones and the template VM connect directly to the source information in vDisk_id 10. This direct connection can lead to lower disk I/O latency for read operations.

When the template VM and the cloned VMs write data, they write to their own vDisks (which initially only exist in Pithos) that the system allocated on disk during the first write operation.

3. Nutanix Recommendations

Snapshot and clone operations are fast and efficient because Nutanix minimizes the number of operations, including creating new constructs on disk, that AOS performs. The following points summarize our guidance on using Nutanix snapshots and clones:

- For optimal performance, clone VMs from a template VM snapshot or template VM that isn't running.
- Clone VMs from a template VM snapshot when you need multiple versions of the template VM.
- For easy management, clone VMs from a template VM.
- Clones created from snapshots don't include any updates made to the template VM after the creation of the snapshot used for cloning.
- For information about supported recovery point objectives (RPOs), refer to the [Prism Element Data Protection Guide](#).

About Nutanix

Nutanix is a global leader in cloud software and a pioneer in hyperconverged infrastructure solutions, making clouds invisible and freeing customers to focus on their business outcomes. Organizations around the world use Nutanix software to leverage a single platform to manage any app at any location for their hybrid multicloud environments. Learn more at www.nutanix.com or follow us on Twitter [@nutanix](https://twitter.com/nutanix).

List of Figures

Figure 1: Snapshot: Starting Point.....	7
Figure 2: Snapshot: Initial Snapshot.....	8
Figure 3: Snapshot: First Write.....	9
Figure 4: Snapshot: Data Overwrite.....	10
Figure 5: Snapshot: Second Snapshot.....	11
Figure 6: Snapshot: Write to Second Snapshot.....	12
Figure 7: Snapshot: Maintenance Tasks.....	13
Figure 8: Clone: Starting Point.....	14
Figure 9: Clone: Initial Clone.....	14
Figure 10: Clone: First Write.....	15
Figure 11: Clone: Source Writing Data.....	15
Figure 12: Clone: Overwrite.....	16
Figure 13: Clones and Snapshots: Taking a Second Snapshot of VM1.....	16
Figure 14: Clones and Snapshots: First Write to Second VM1 Snapshot.....	17
Figure 15: Clones and Snapshots: Taking a Second Snapshot of VM2.....	17
Figure 16: Clones and Snapshots: New Writes and Overwrites.....	18
Figure 17: Cloning from a Template: First Clones.....	19
Figure 18: Cloning from a Template: New Writes to Template, Clone1, and Clone2.....	20
Figure 19: Cloning from a Template: First Clone.....	20
Figure 20: Cloning from a Template: New Writes to Template and Clone1.....	21
Figure 21: Cloning from a Template: Second Clone.....	21
Figure 22: Cloning from a Template: New Writes to Template and Clone2.....	22
Figure 23: Cloning VMs from a Snapshot.....	22