



**APFS**

# **Apple's File System for the Next 30 Years**

**Tim Standing**

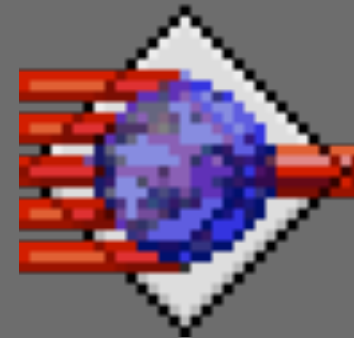
**Vice President Software Engineering - Mac  
OWC, Inc.**

# SoftRAID

## 24 Years of Software RAID for Mac OS



1.0  
1994



2.0  
1996



3.0  
2003



4.0  
2010



5.0  
2014



6.0  
2019

# 1998 = HFS+ and the 1st iMac



# Attempts to Replace HFS+

- 2006: ZFS
- 2007-2011: 2 Rumored, Cancelled Projects
- 2011: Core Storage





# APFS

Apple's file system for the  
next 30 years

# Why Apple created APFS

- 64 bit file system; more files; smaller chunk size on large volumes
- Tuned for SSDs
- Space sharing—dynamic resizing for volumes on the same disk
- Increased protection for volume metadata
- Reserve size and volume quotas
- Copy on Write—snapshots; more efficient storage of different file versions
- Low latency file operations
- More robust encryption
- New source code

# What we want from APFS

- Increased protection from corrupted volumes
- Volume snapshots
- Increased speed with HDDs
- More robust encryption



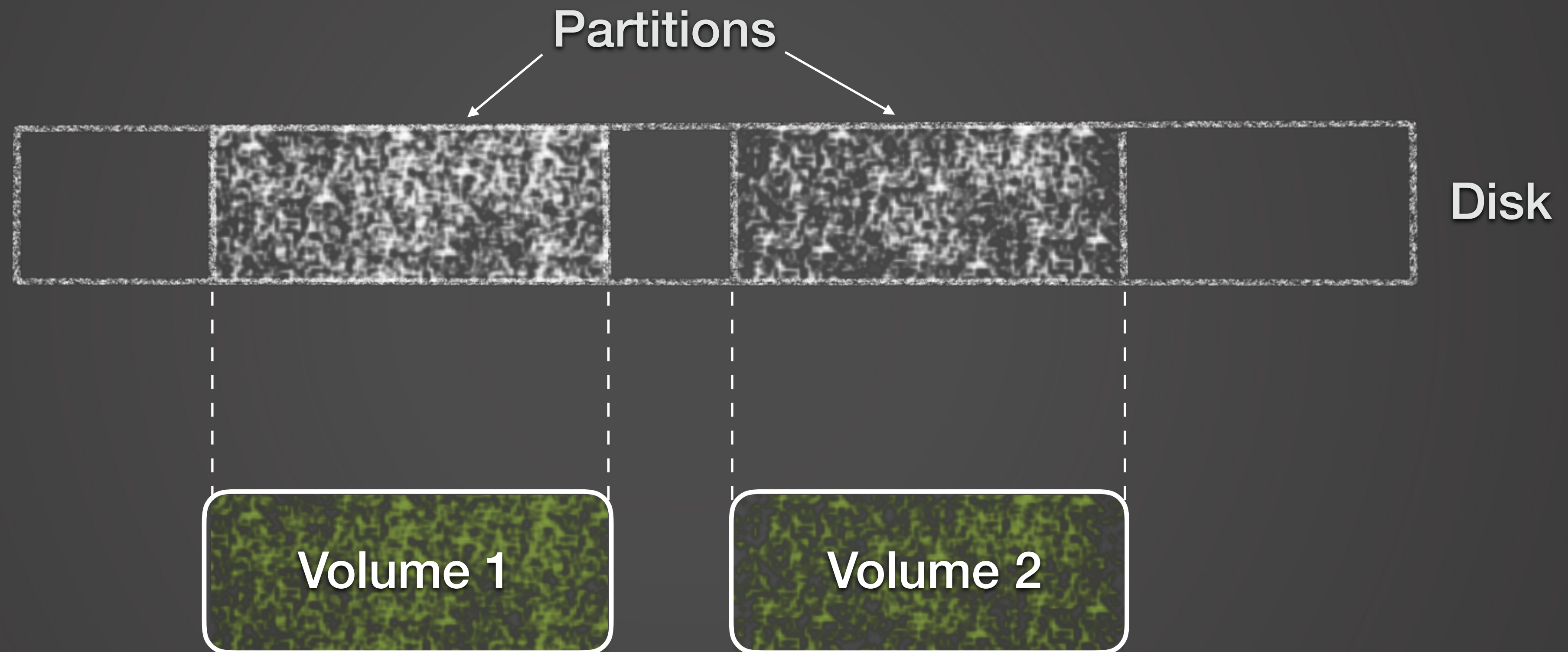
# APFS

- What is volume space sharing
- How does APFS store metadata
- What are the Reserve Size and Quota of a volume
- How does Copy on Write work
- What are APFS snapshots
- APFS encrypted volumes
- How fast is an APFS volume



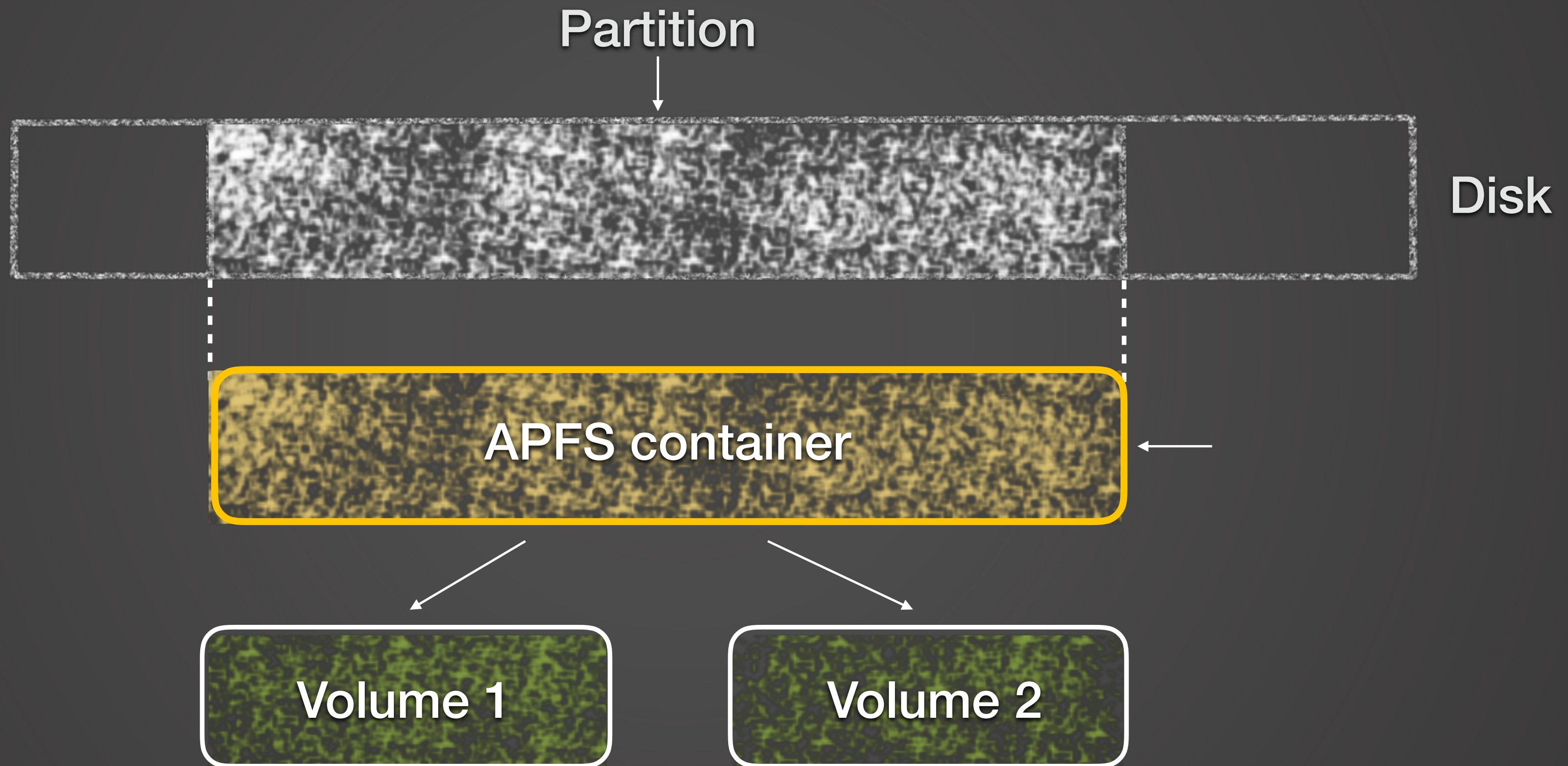
# What is “Volume Space Sharing”?

# 2 HFS+ Volumes on a Disk





# 2 APFS Volumes on a Disk



How does APFS  
store metadata?



# HFS+ Risk of Corruption

Original metadata and file

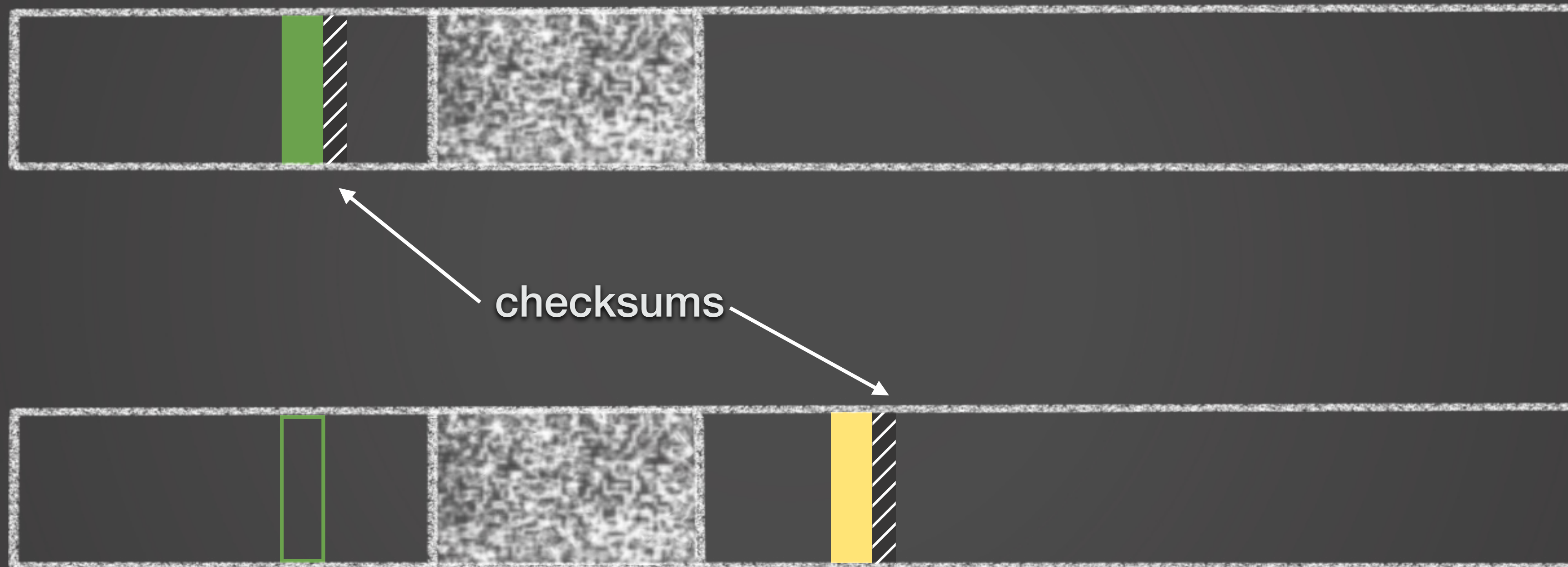


Metadata changes overwritten in place



# APFS: Increased Protection from Corruption

All metadata has a checksum



Metadata updates do not overwrite previous metadata

What are the “Reserve Size” and  
“Quota” of an APFS volume?



# Reserve Size and Quotas

Reserve size:

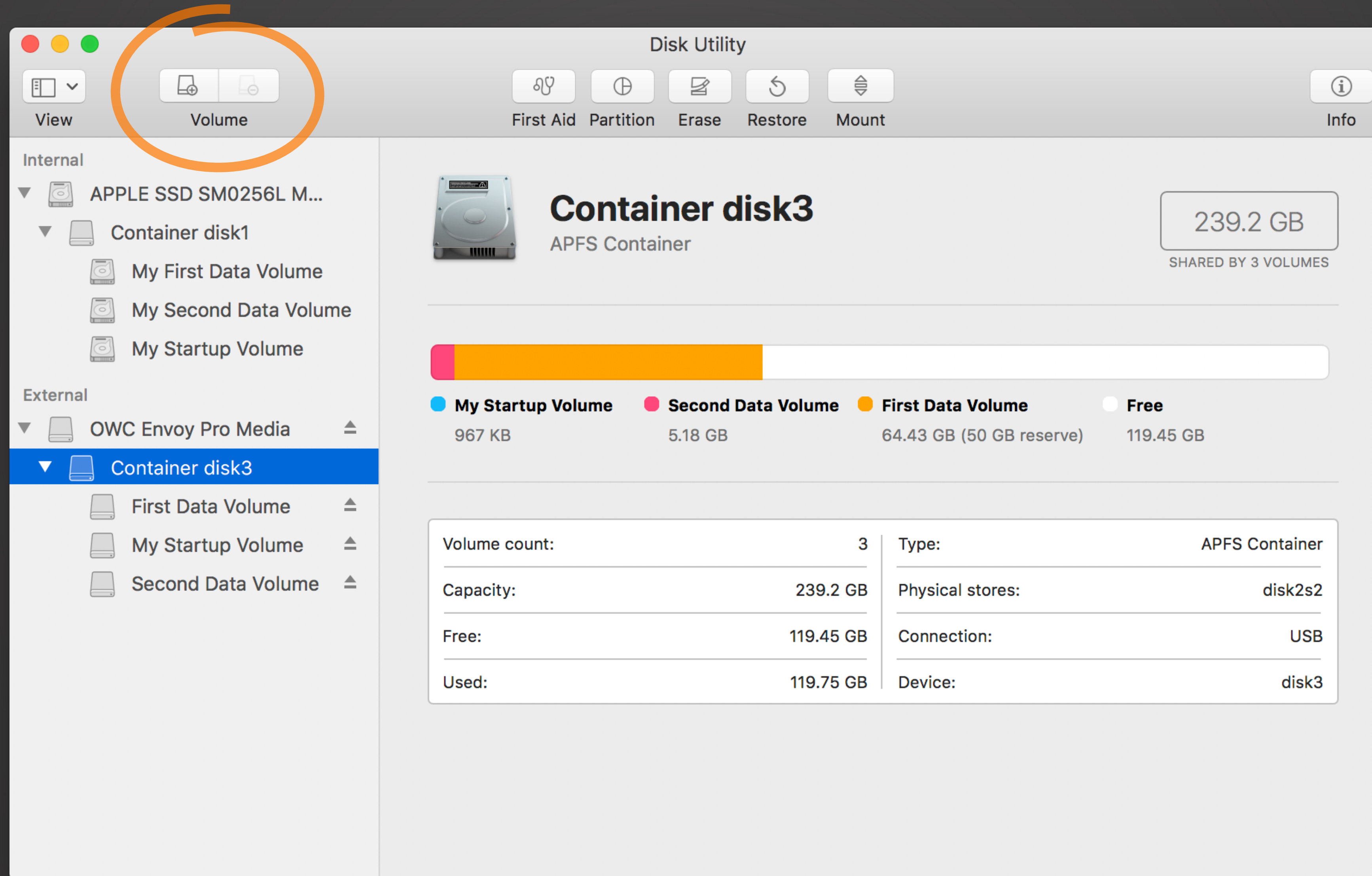
- Amount of space *guaranteed* for use by a volume
- Volume size able to exceed this reserve size

Quota size:

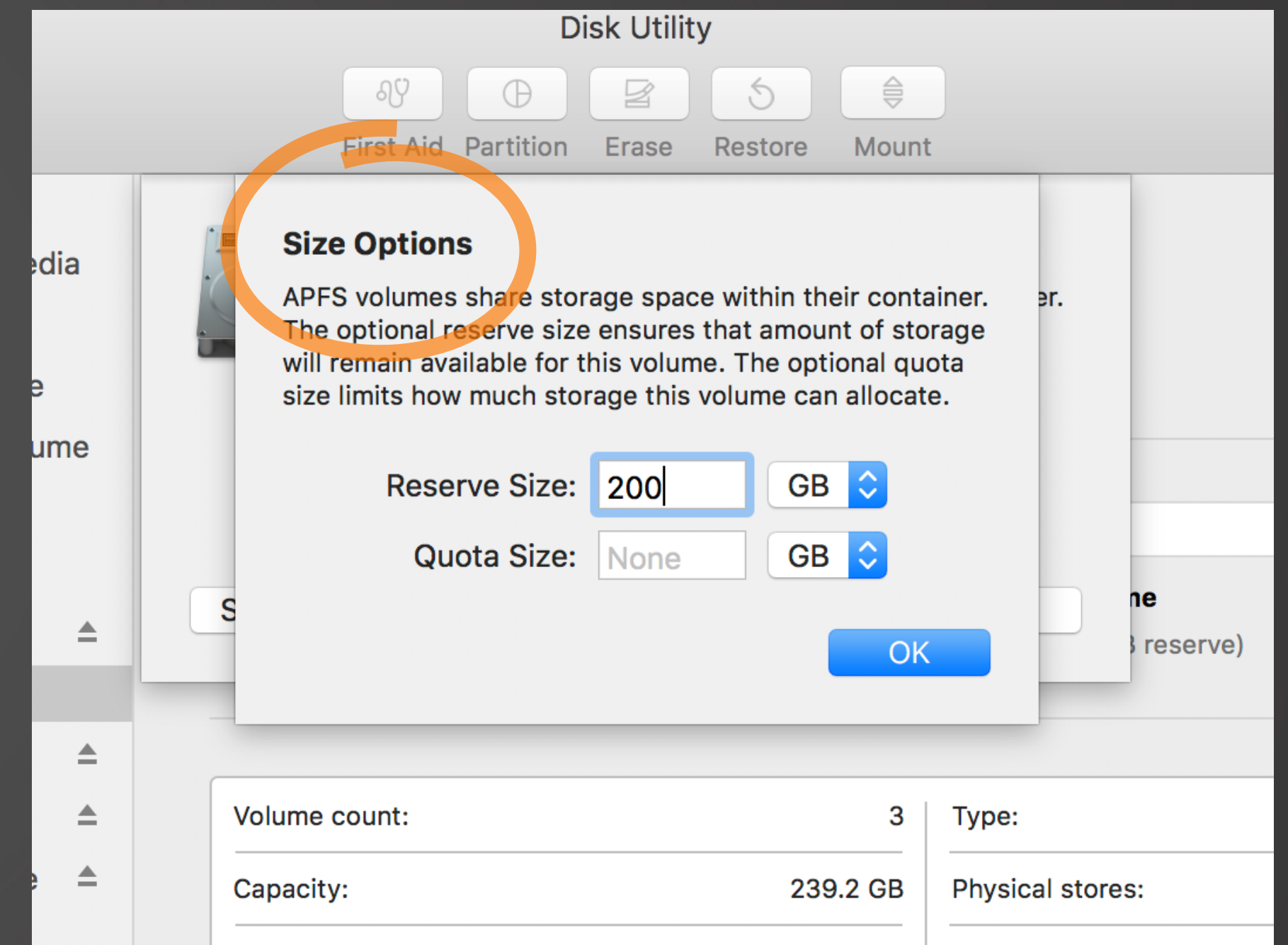
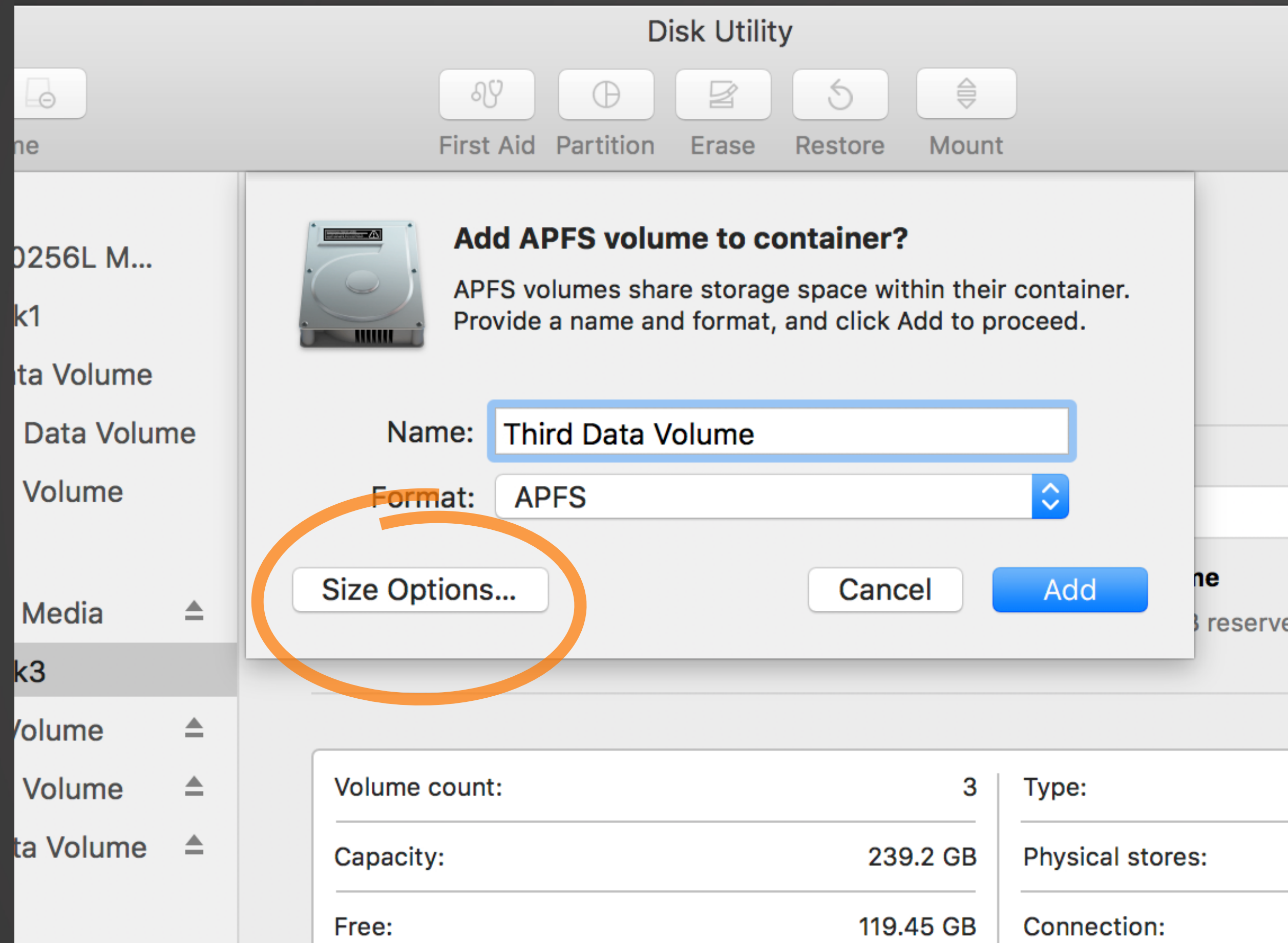
- Limit on how large a volume can be
- No guarantee that the volume can grow that big



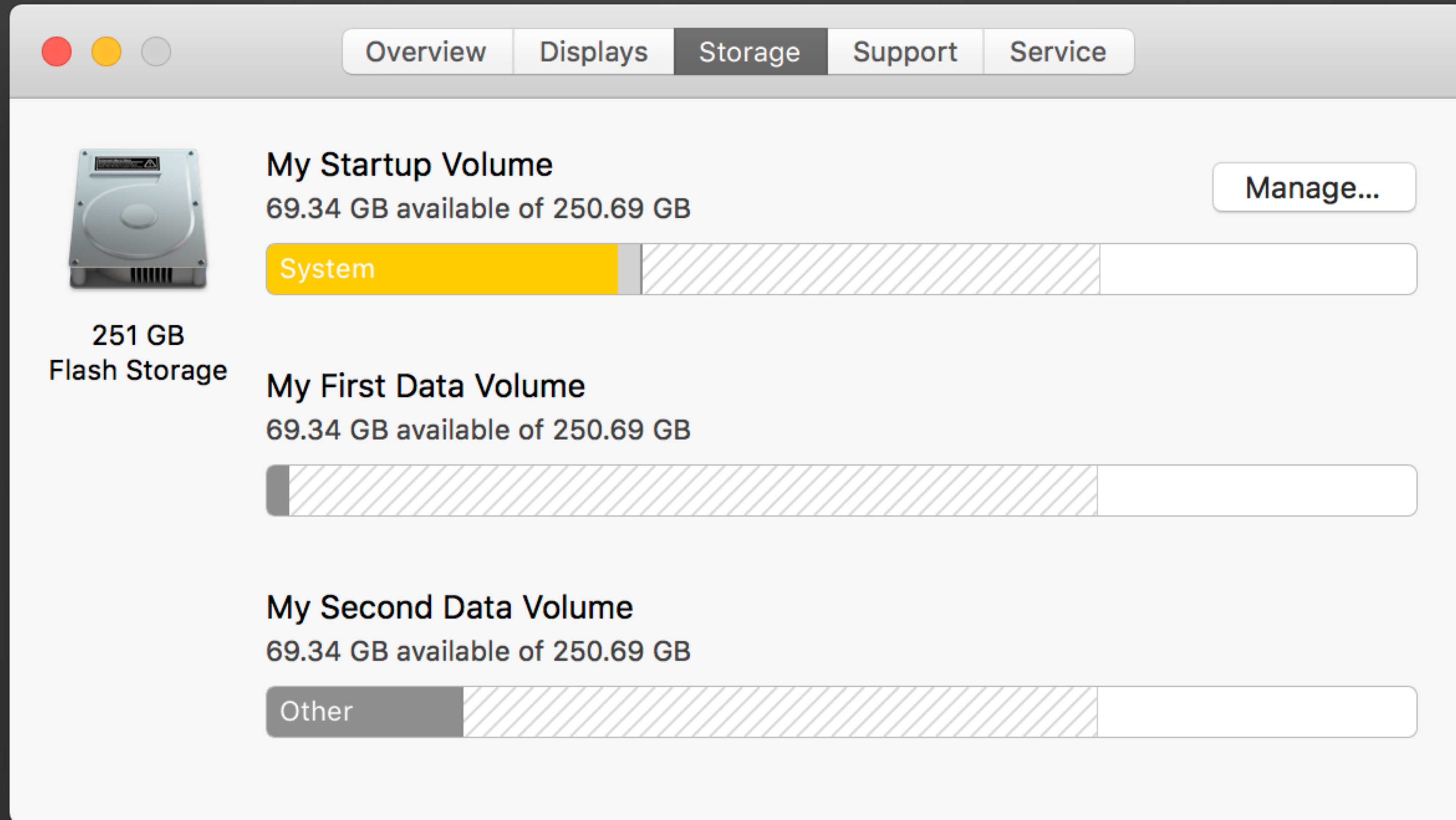
# Creating a New Volume with Reserve Size



# Creating a New Volume with Reserve Size

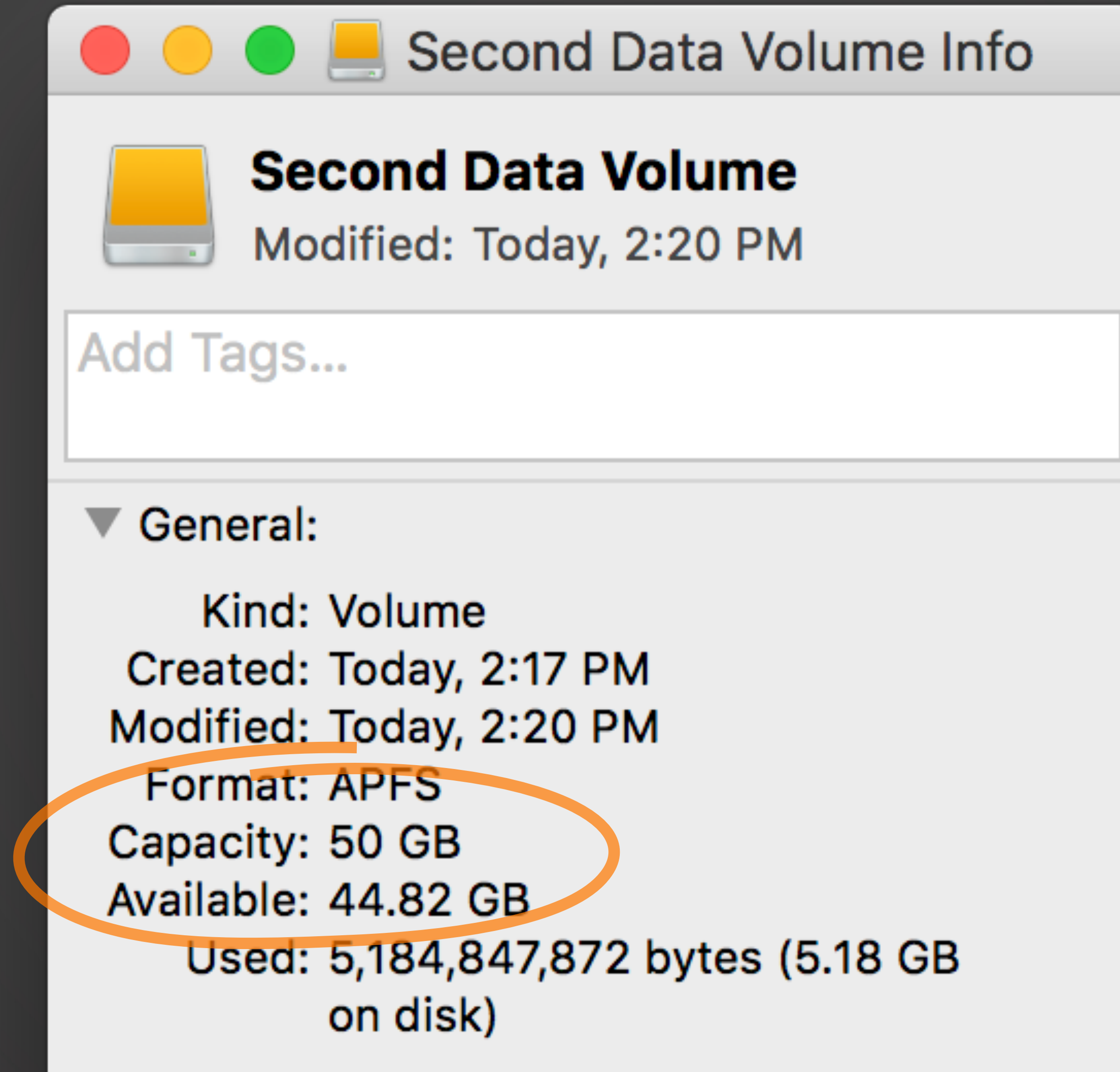


# Which Volume Has a Reserve Size?





# Does This Volume Have a Quota?







```
[Yoda:~ yoda$ diskutil apfs list disk3  
|  
+-- Container disk3 3CF9C723-D299-4A31-8007-F1439E95452E  
=====  
APFS Container Reference:      disk3  
Capacity Ceiling (Size):      499898105856 B (499.9 GB)  
Capacity In Use By Volumes:    200162566144 B (200.2 GB) (40.0% used)  
Capacity Available:            299735539712 B (299.7 GB) (60.0% free)  
|  
+--< Physical Store disk1s2 79B53882-5AA5-47C0-BD07-3B13EFCF70D0  
-----  
| APFS Physical Store Disk:    disk1s2  
| Size:                        499898105856 B (499.9 GB)  
|  
+--> Volume disk3s2 7A13922A-1BA6-42FF-9C11-DD8A93F0375F  
-----  
| APFS Volume Disk (Role):    disk3s2 (No specific role)  
| Name:                       First Data Volume (Case-insensitive)  
| Mount Point:                /Volumes/First Data Volume  
| Capacity Consumed:           200000000000 B (200.0 GB)  
| Capacity Reserve:            200000000000 B (200.0 GB)  
| Capacity Quota:              None  
| Encrypted:                   No  
|  
+--> Volume disk3s1 7E46F79C-183A-449E-872D-6148789AFD11  
-----  
| APFS Volume Disk (Role):    disk3s1 (No specific role)  
| Name:                       Second Data Volume (Case-insensitive)  
| Mount Point:                /Volumes/Second Data Volume  
| Capacity Consumed:           835584 B (835.6 KB)  
| Capacity Reserve:            None  
| Capacity Quota:              100000002048 B (100.0 GB) (0.0% reached)  
| Encrypted:                   No
```

# Does the Volume Have a Reserve Size or Quota?



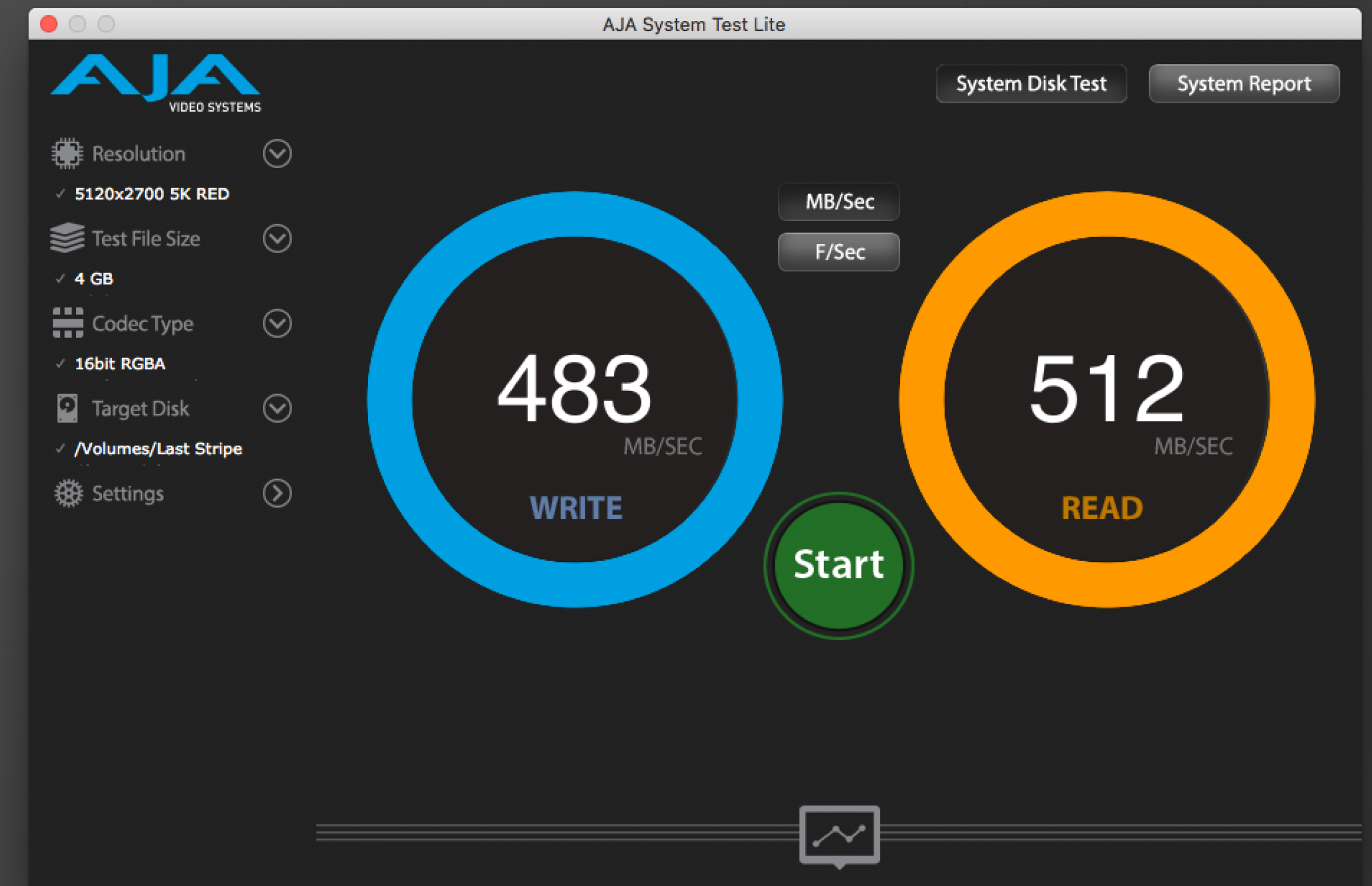
# Speed Differences in HDDs

Huge difference between first & last parts of HDDs  
shown by speeds for SoftRAID 4-disk stripe volumes:

using FIRST 10% of disk



using LAST 10% of disk



# Quota Limitations

- No ability to control over where volumes are in container
- Can't set quotas up to use fastest part of HDDs
- Can't change quota or reserve size after volume is created

How does  
“Copy on Write”  
work?





# COPYING FILES WITH HFS+

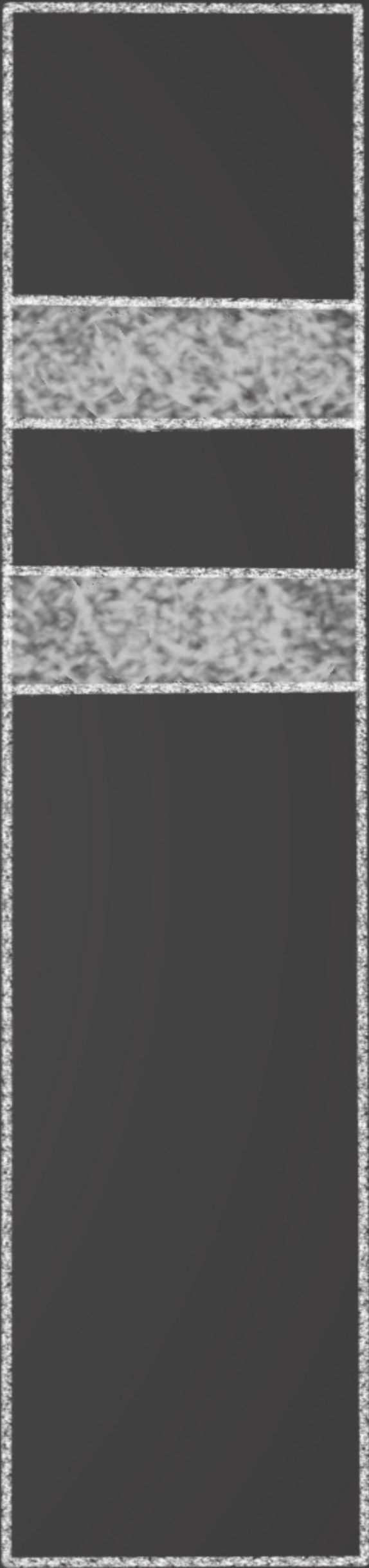
Extents Table		
	Offset	Length
Original File	20401094656	2147483648
	41875931136	8589934592



*"Nina's Birthday.mp4"*

Available space on disk: 90 GB

120 GB





# COPYING FILES WITH HFS+

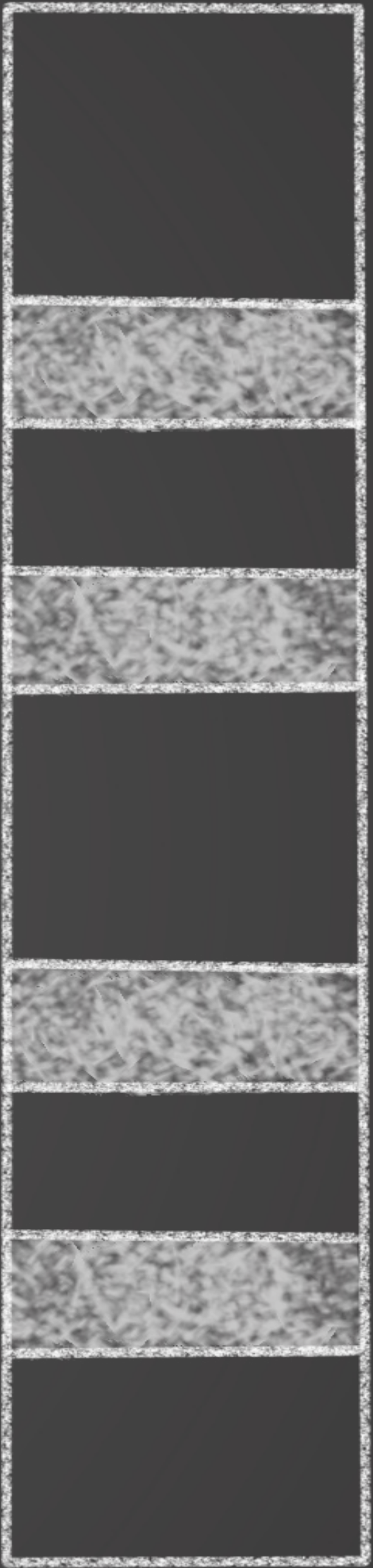
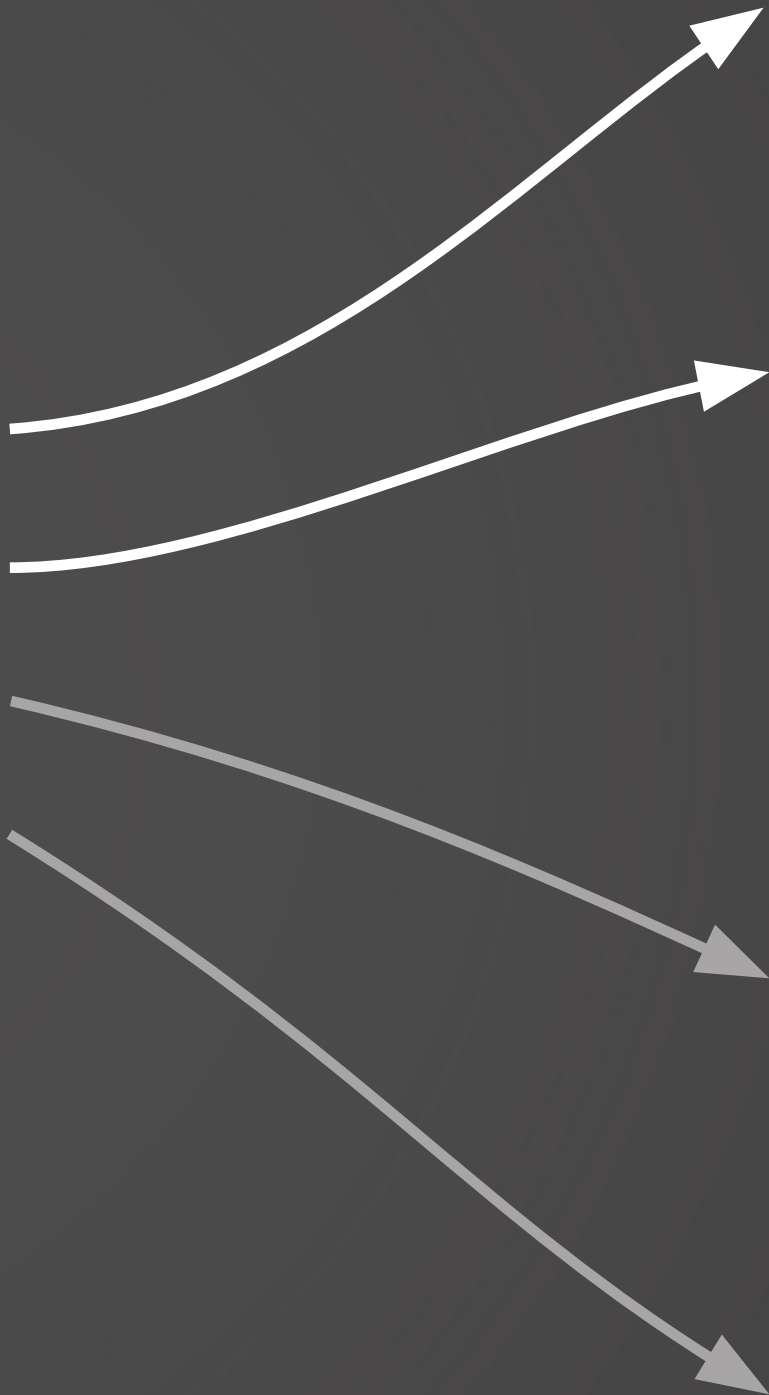
Original file with copy on SSD

Extents Table		
	Offset	Length
Original File	20401094656	2147483648
	41875931136	8589934592
Copy	53687091200	4294967296
	64424509440	6442450944

*"Nina's Birthday.mp4"*

Available space on disk: 80 GB

120 GB







# COPYING FILES WITH HFS+

Editing 4 frames

Extents Table		
	Offset	Length
Original File	20401094656	2147483648
	41875931136	8589934592
Copy	53687091200	4294967296
	64424509440	6442450944

*"Nina's Birthday.mp4"*

Available space on disk: 80 GB

120 GB



Edited frames





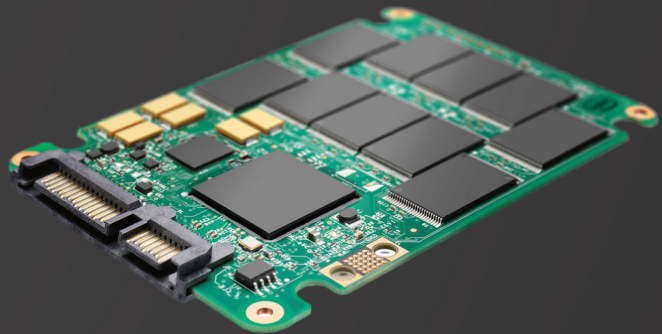
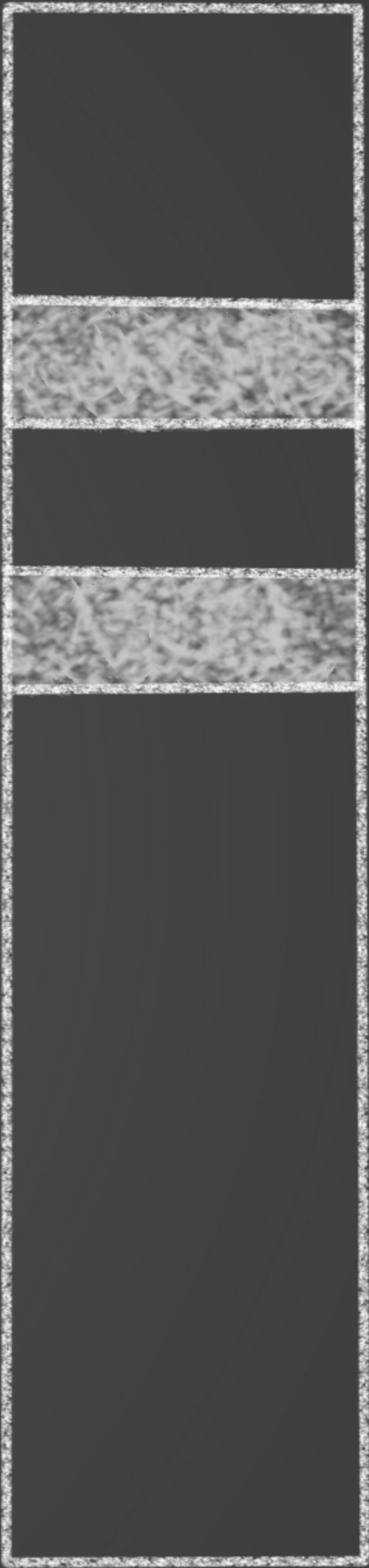
# COPYING FILES WITH APFS

Extents Table		
	Offset	Length
Original File	20401094656	2147483648
	41875931136	8589934592

*"Nina's Birthday.mp4"*

Available space on disk: 90 GB

120 GB



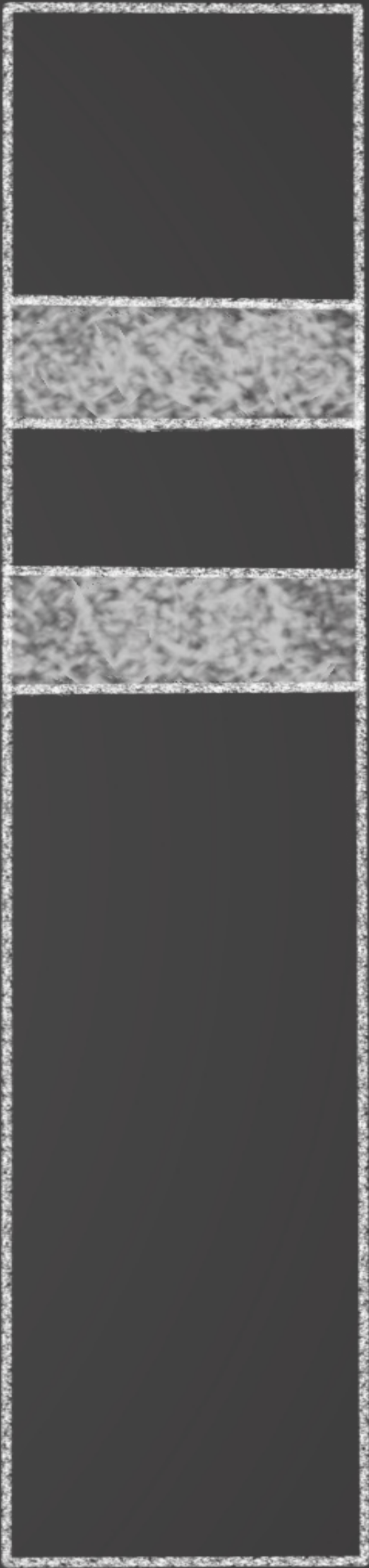
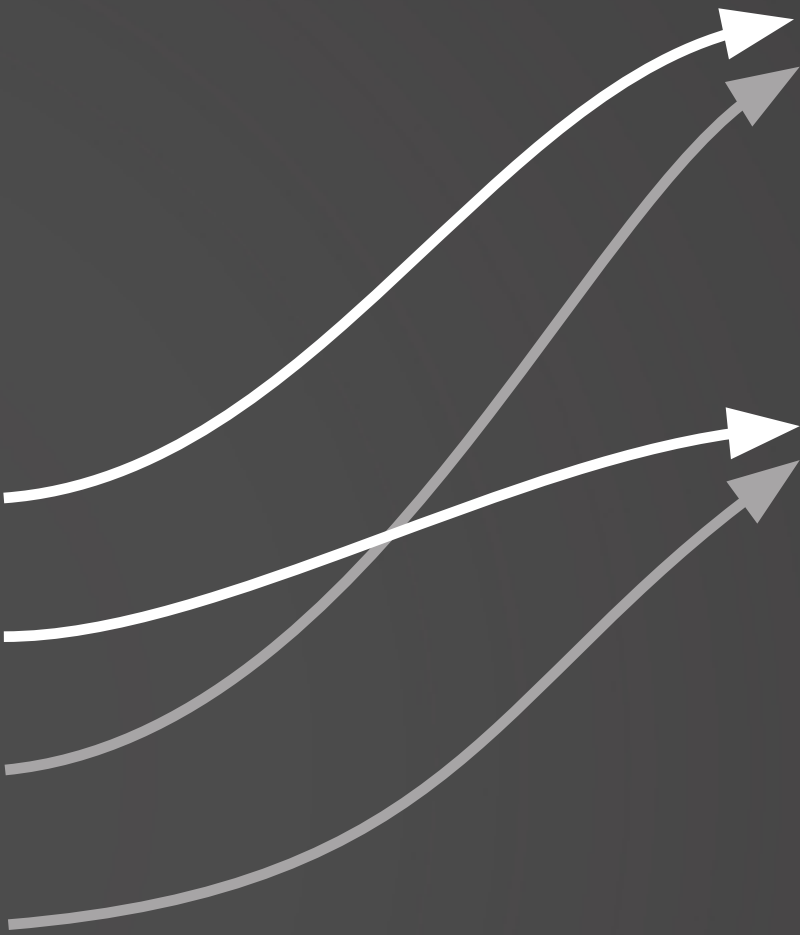




# COPYING FILES WITH APFS

Original file with copy on SSD

Extents Table		
	Offset	Length
Original File	20401094656	2147483648
	41875931136	8589934592
Copy	20401094656	2147483648
	41875931136	8589934592



*"Nina's Birthday.mp4"*

Available space on disk: 90 GB

120 GB

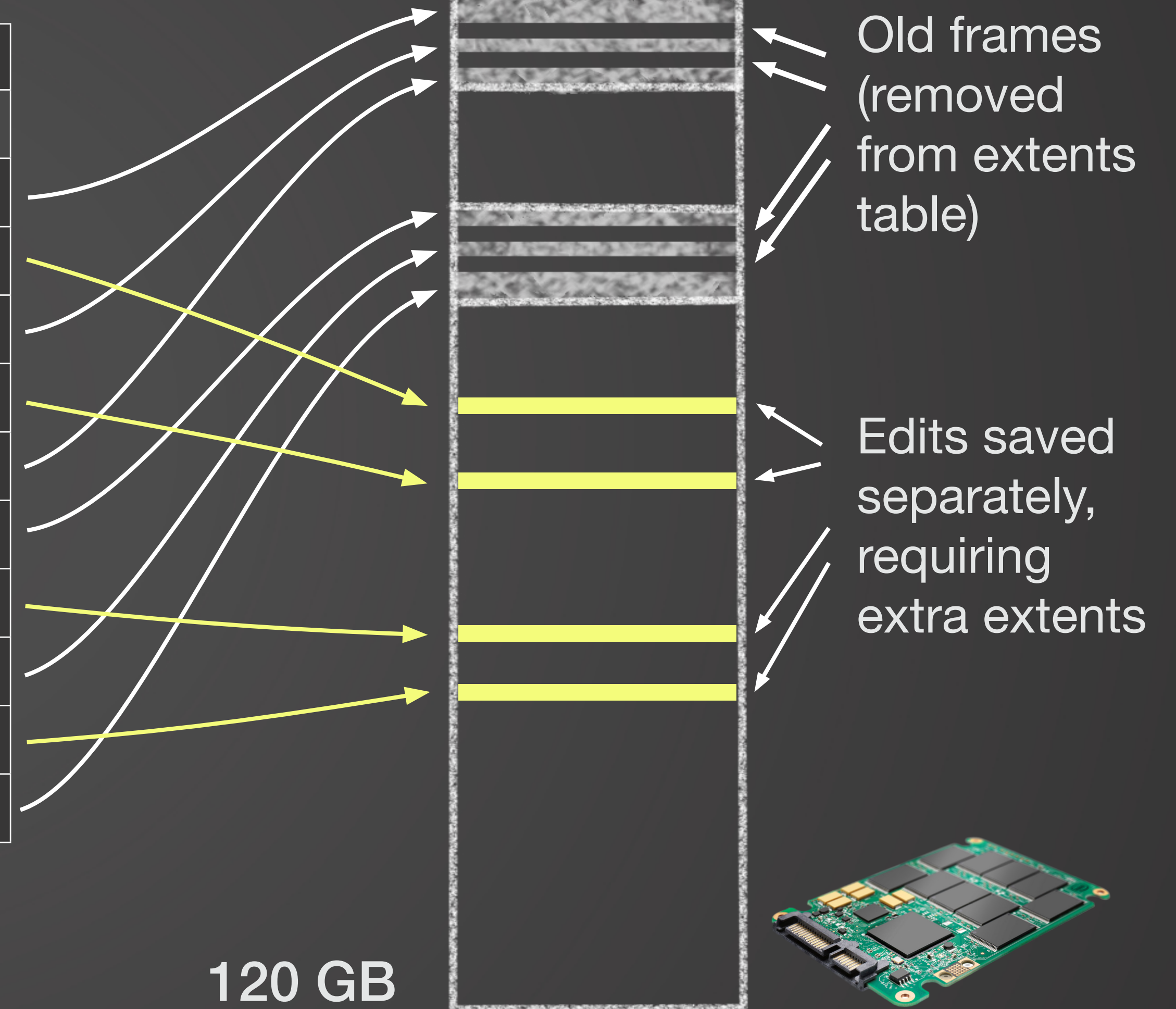




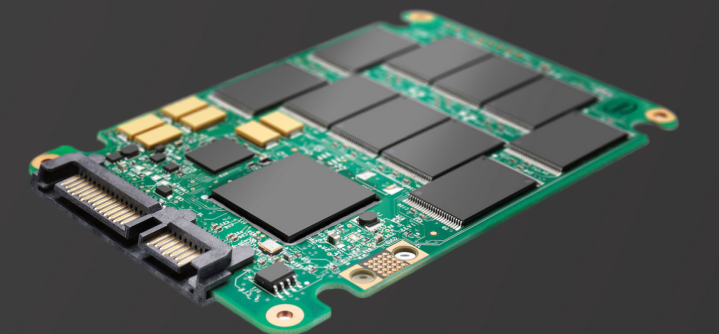
# COPYING FILES WITH APFS

## Original file after editing 4 frames

Extents Table		
	Offset	Length
shared with copy	20401094656	104857600
changed frame	75161927680	1048576
shared with copy	20507000832	106954752
changed frame	80530636800	2097152
shared with copy	20616052736	1932525568
shared with copy	41875931136	1073741824
changed frame	86973087744	106954752
shared with copy	43056627712	2147483648
changed frame	94489280512	4294967296
shared with copy	4294967296	966787072

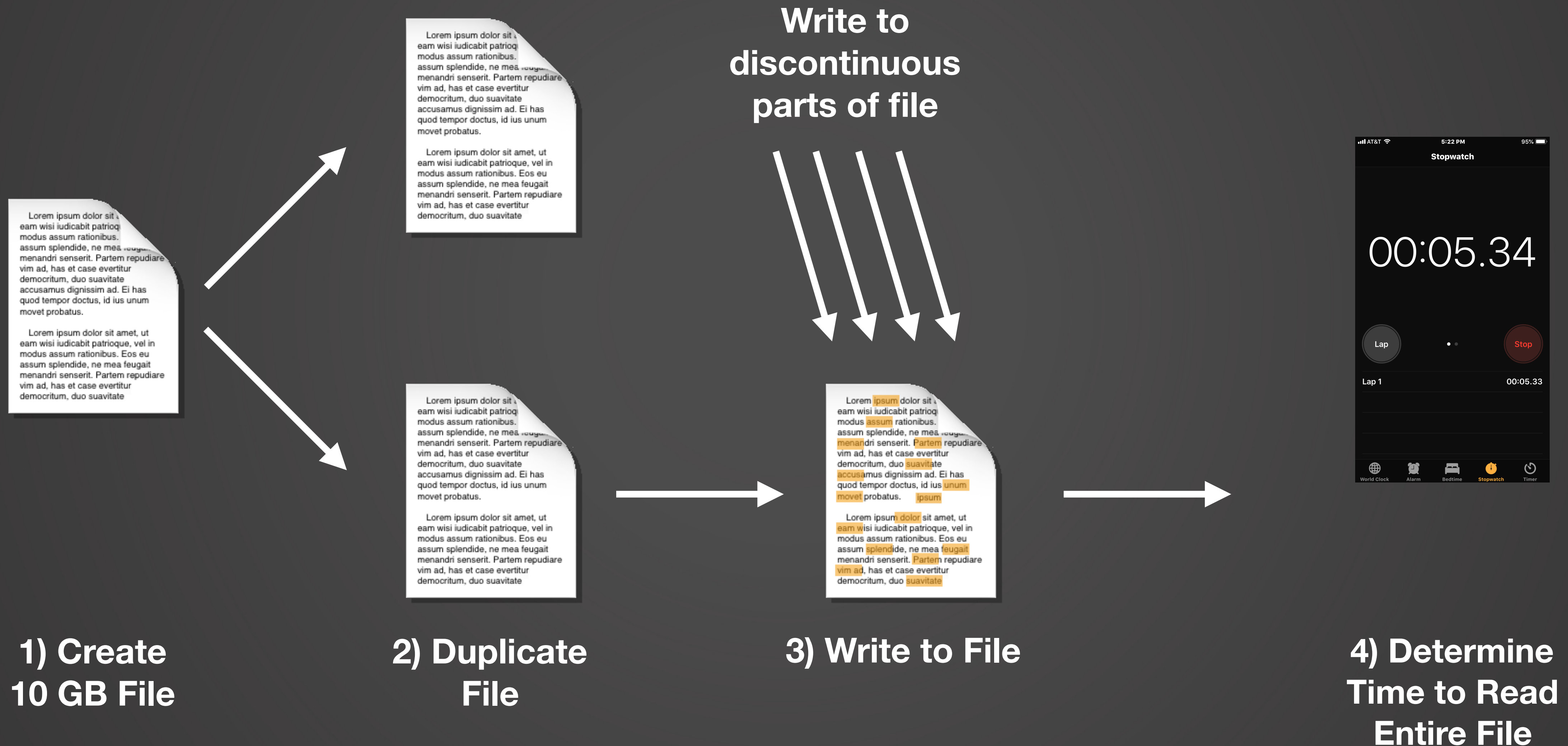


"Nina's Birthday.mp4"



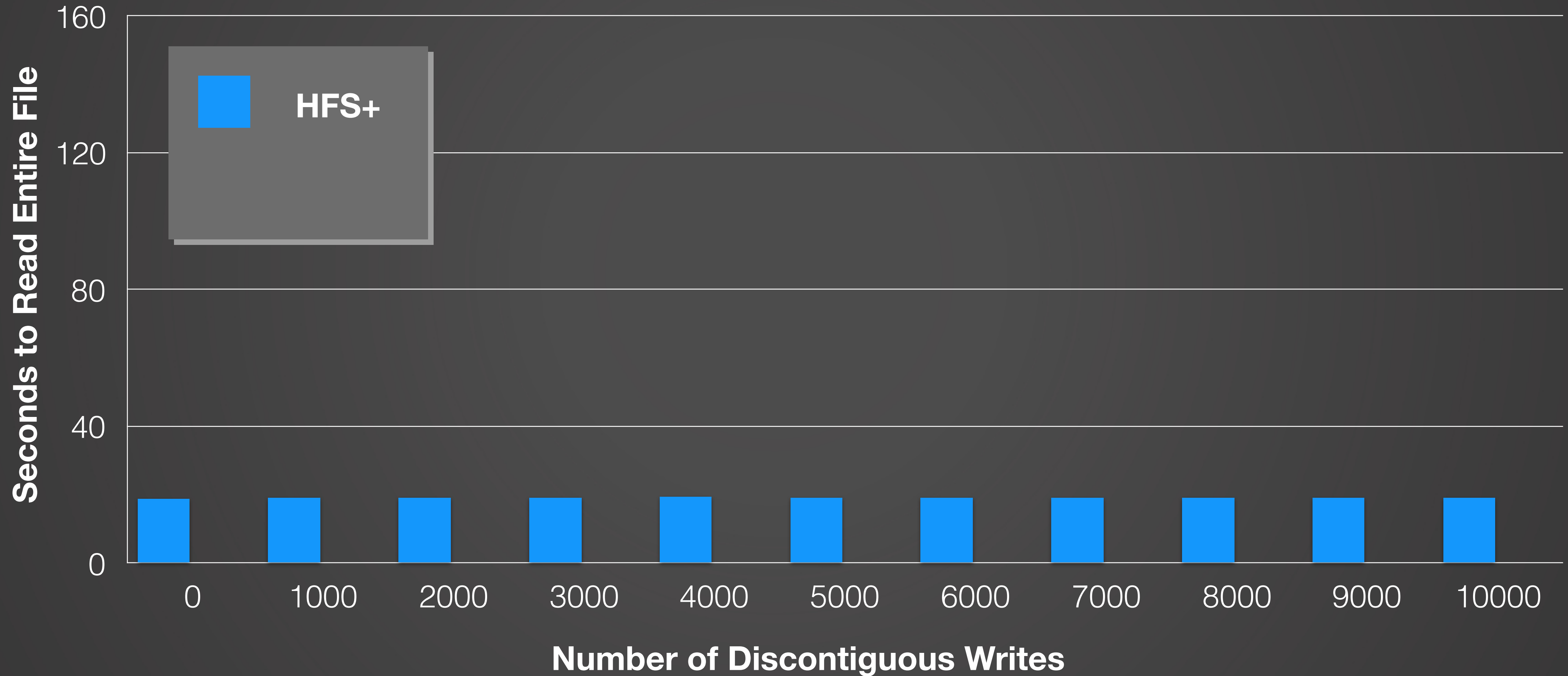


# How to test the effect of copy on write with HDDs?

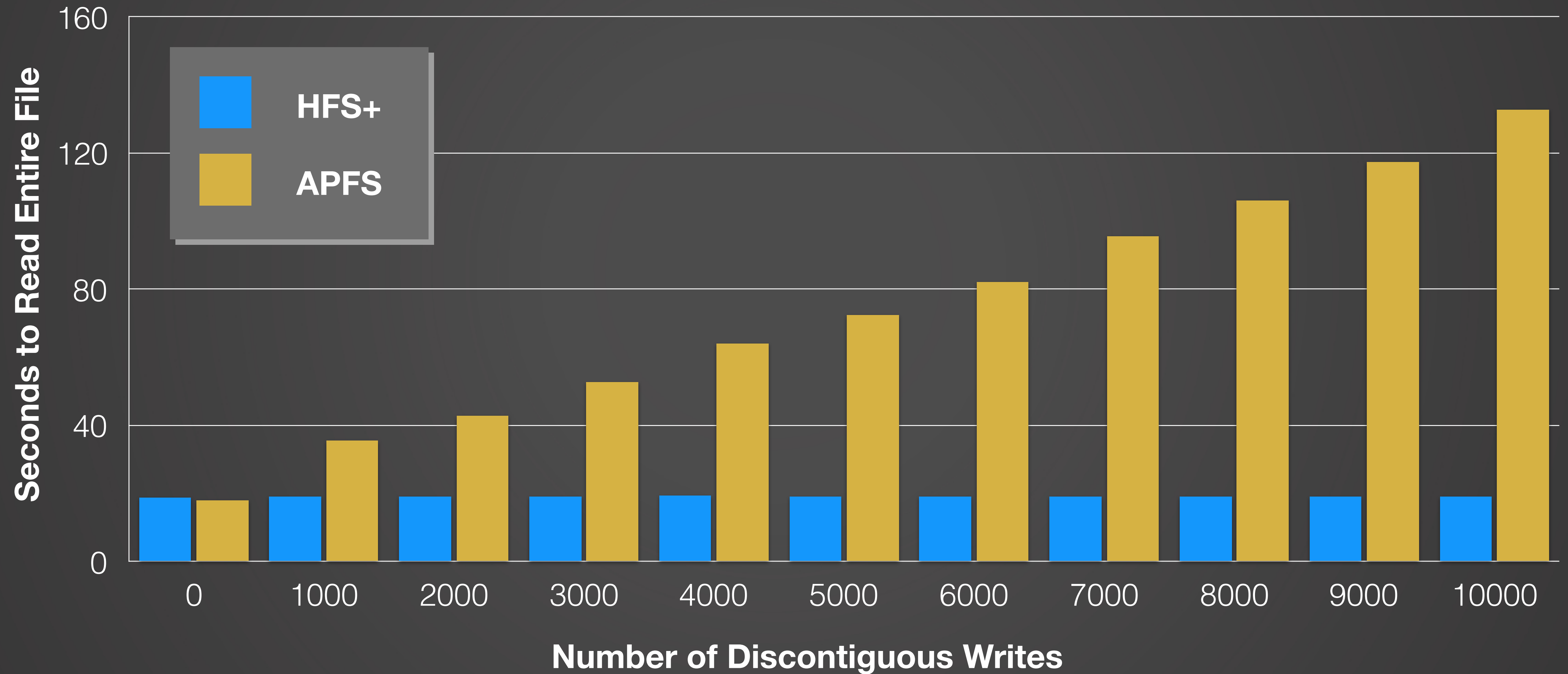




# Time to read 10 GB file from HFS+ volume

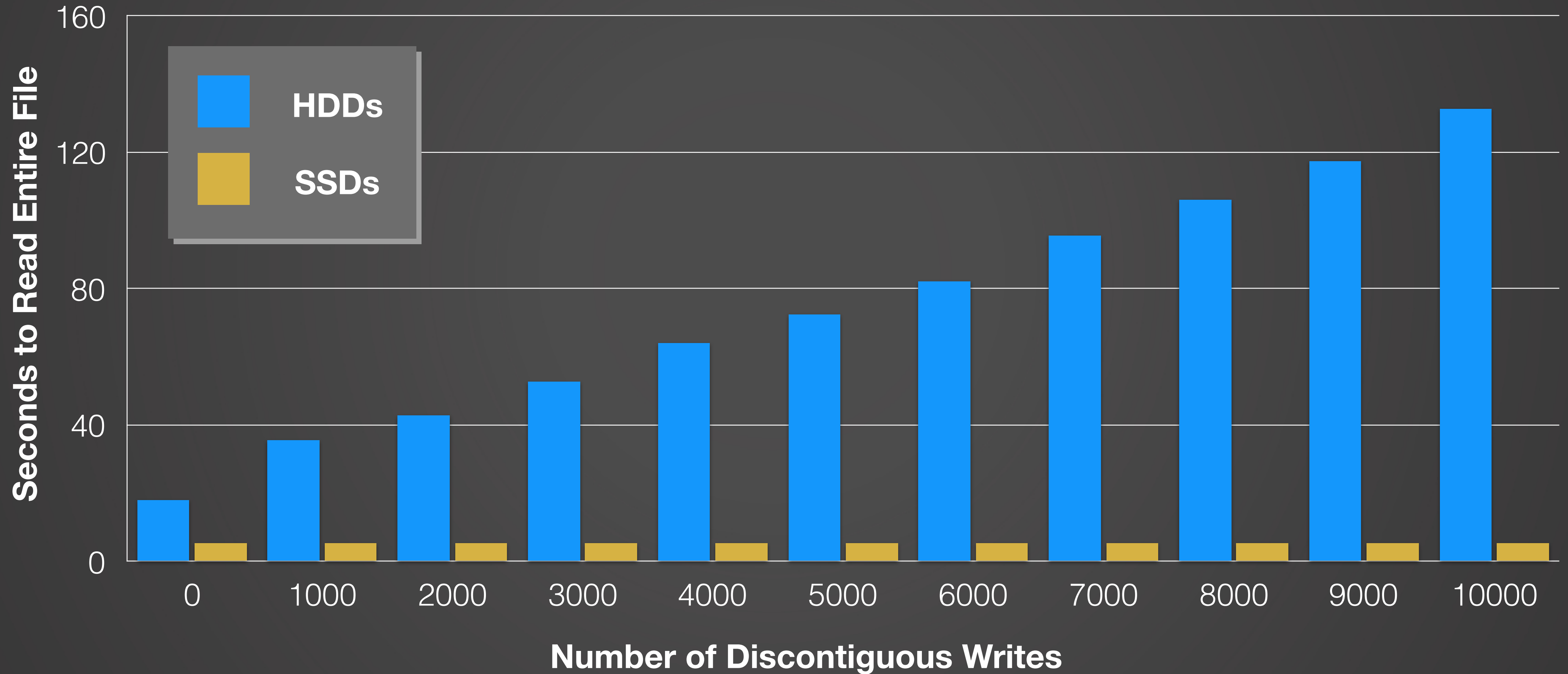


# Time to read 10 GB file from HFS+ vs. APFS





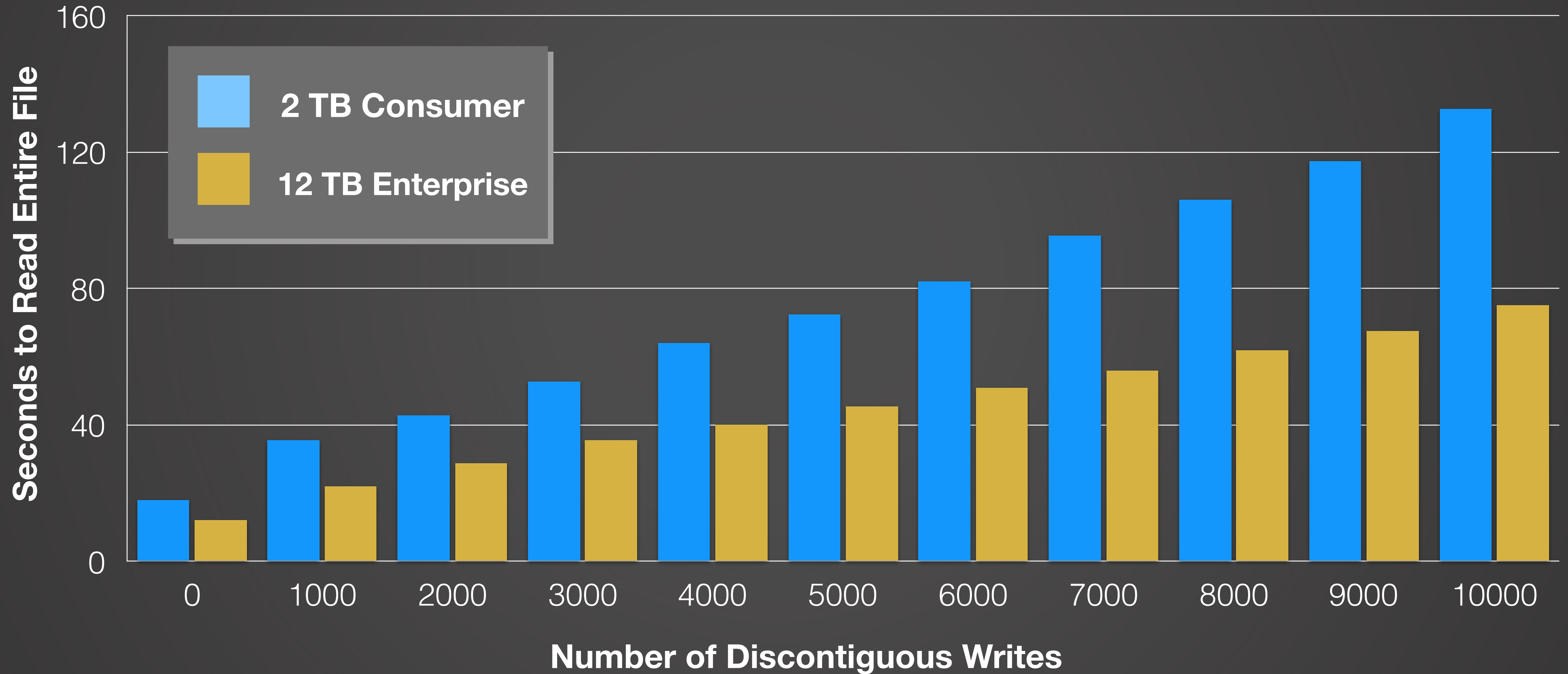
# Time to read 10 GB file from HDDs and SSDs (APFS Volumes)





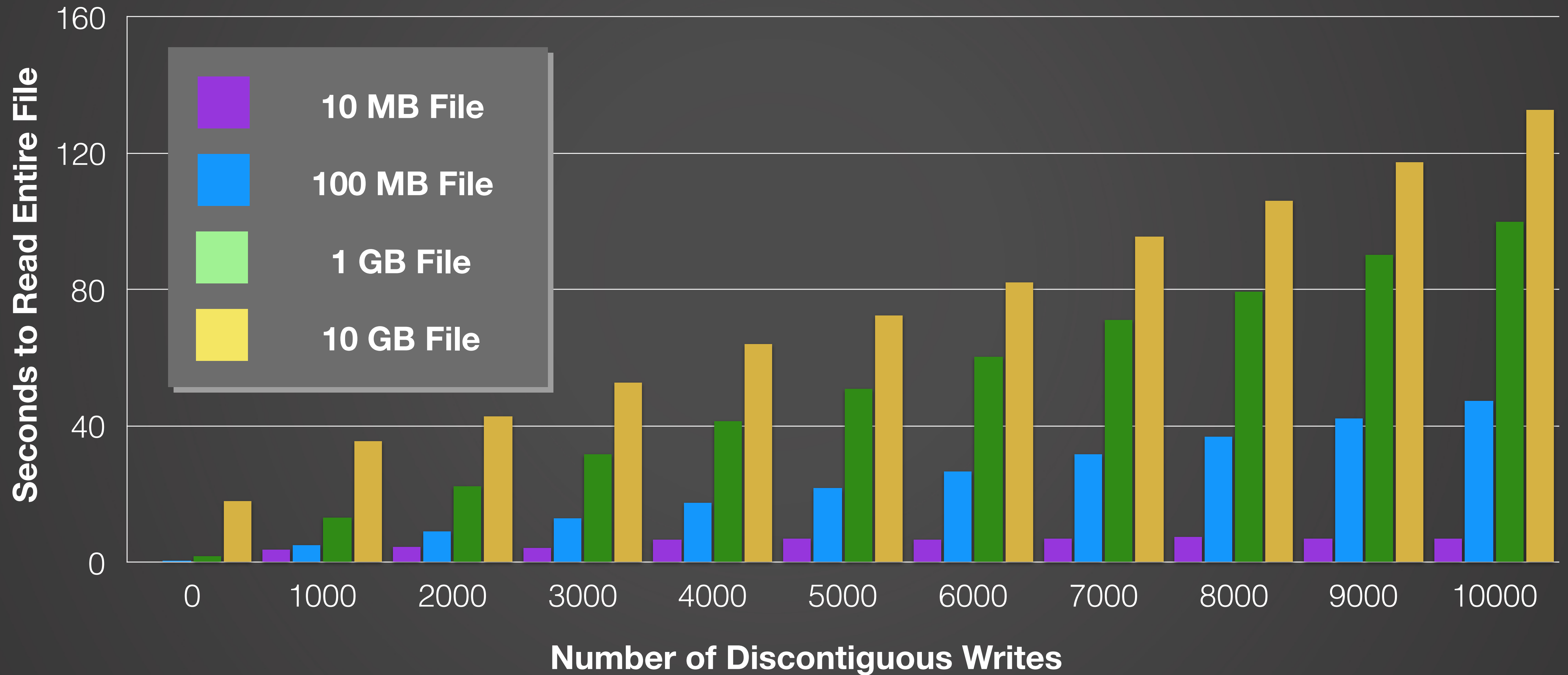


# Time to read 10 GB file from different HDDs (APFS Volumes)





# Time to read different size files from HDDs (APFS Volumes)



# Automatic defragmentation of APFS Volumes

## Status of automatic defragmentation

```
standing — -bash — 81x5
[MacBook-Pro:~ standing$ diskutil apfs defragment /Volumes/StripeVolume status ]
APFS Volume defragmentation is currently enabled
```

- Automatic defragmentation built into APFS
- Enabled via diskutil command in Terminal

## Enabling automatic defragmentation

```
standing — -bash — 81x5
[MacBook-Pro:~ standing$ diskutil apfs defragment /Volumes/StripeVolume enable ]
APFS Volume defragmentation has been successfully enabled
```

- Disabled by default

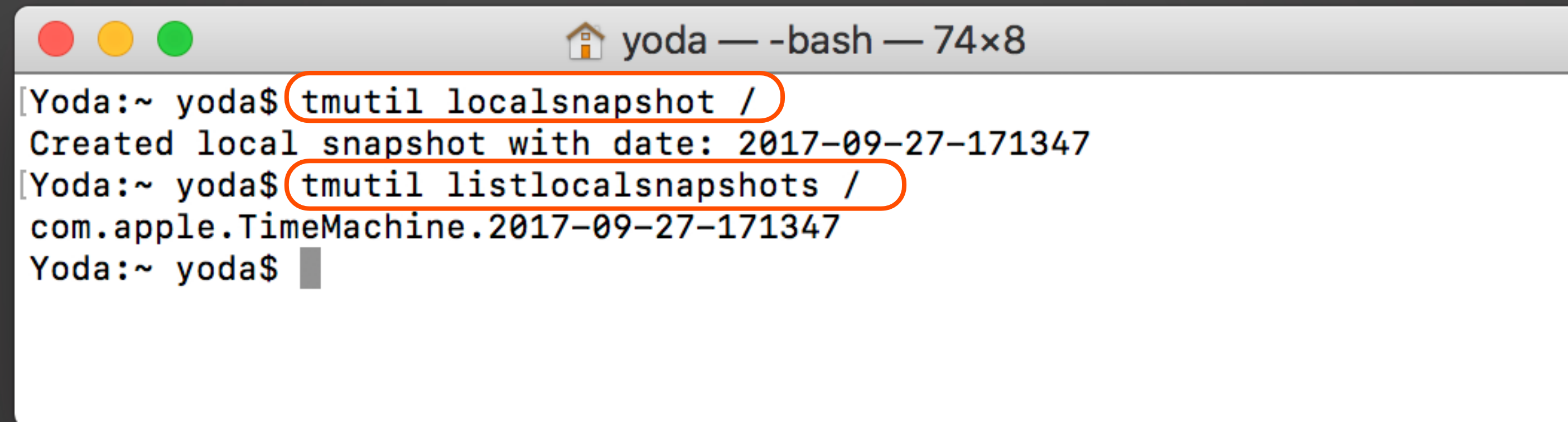


# What are APFS snapshots?

# APFS Snapshots

- Instantaneous picture of all the files on an APFS volume
- Stored in the same container as the APFS volume
- Created in 1 - 2 seconds
- Take up very little space as they use the copy on write technology
- Boot into Recovery Mode to revert a volume to a previous snapshot

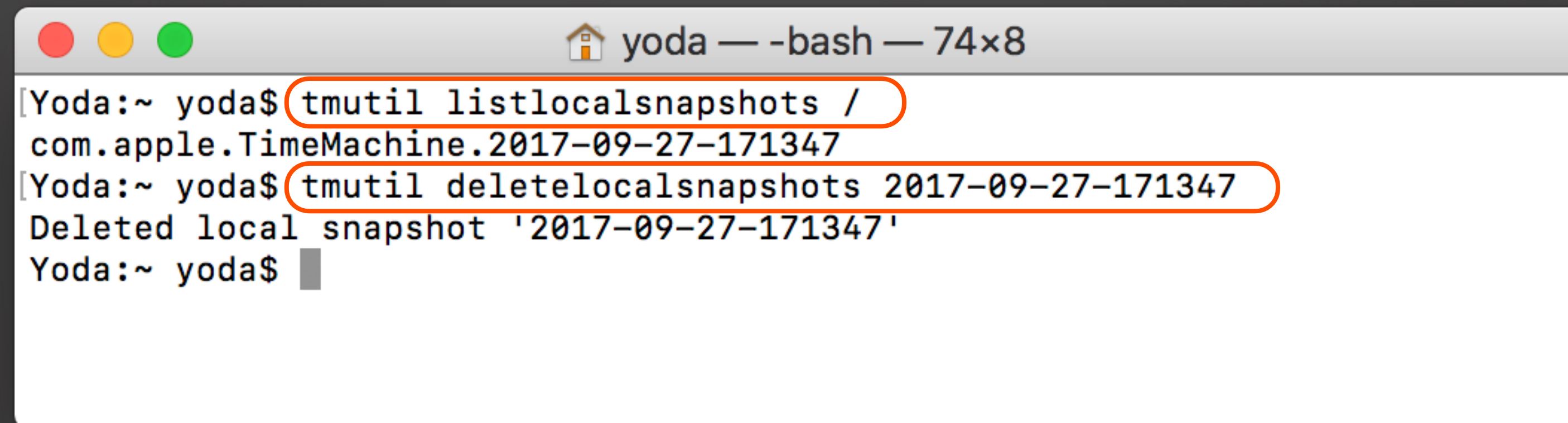
# Creating an APFS Snapshot

A screenshot of a macOS terminal window. The title bar shows a home icon, the name "yoda", and the shell "-bash" with a window size of "74x8". The terminal text shows the execution of two commands: "tmutil localsnapshot /" and "tmutil listlocalsnapshots /". The first command's output is "Created local snapshot with date: 2017-09-27-171347". The second command's output is "com.apple.TimeMachine.2017-09-27-171347". Both command lines are circled in red. The prompt "Yoda:~ yoda\$" is visible at the start of each line, and a cursor is shown at the end of the final line.

```
Yoda:~ yoda$ tmutil localsnapshot /  
Created local snapshot with date: 2017-09-27-171347  
Yoda:~ yoda$ tmutil listlocalsnapshots /  
com.apple.TimeMachine.2017-09-27-171347  
Yoda:~ yoda$
```



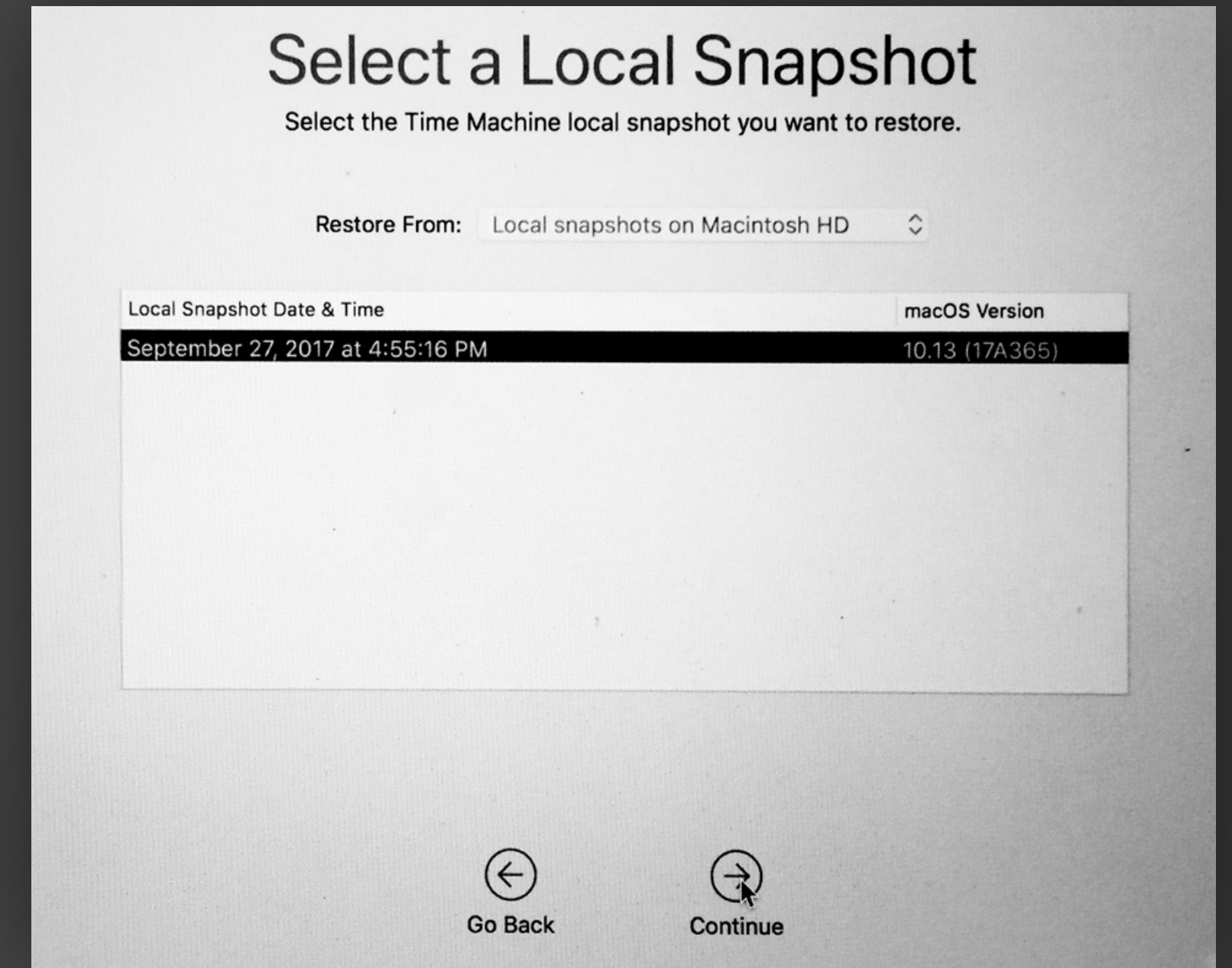
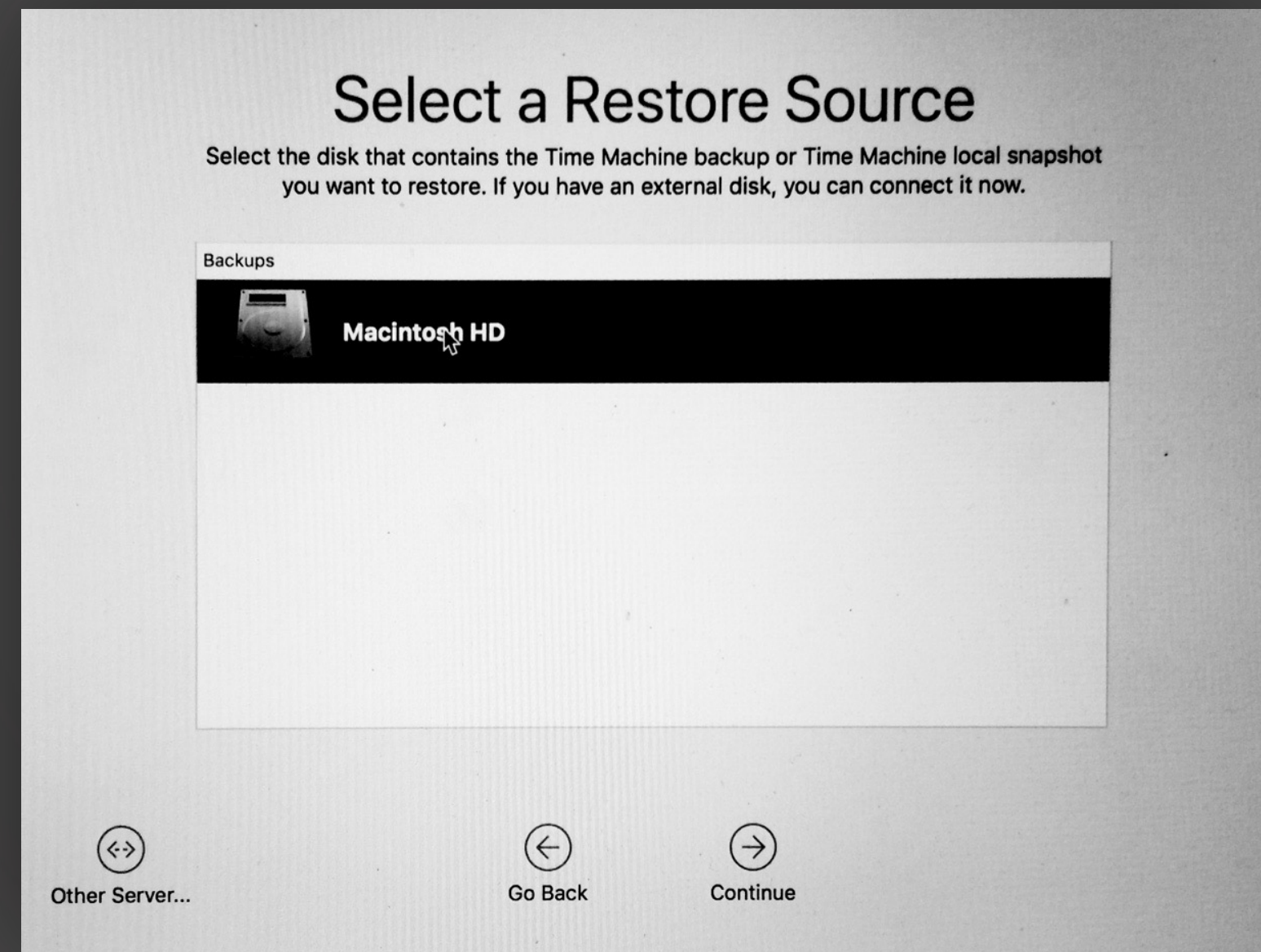
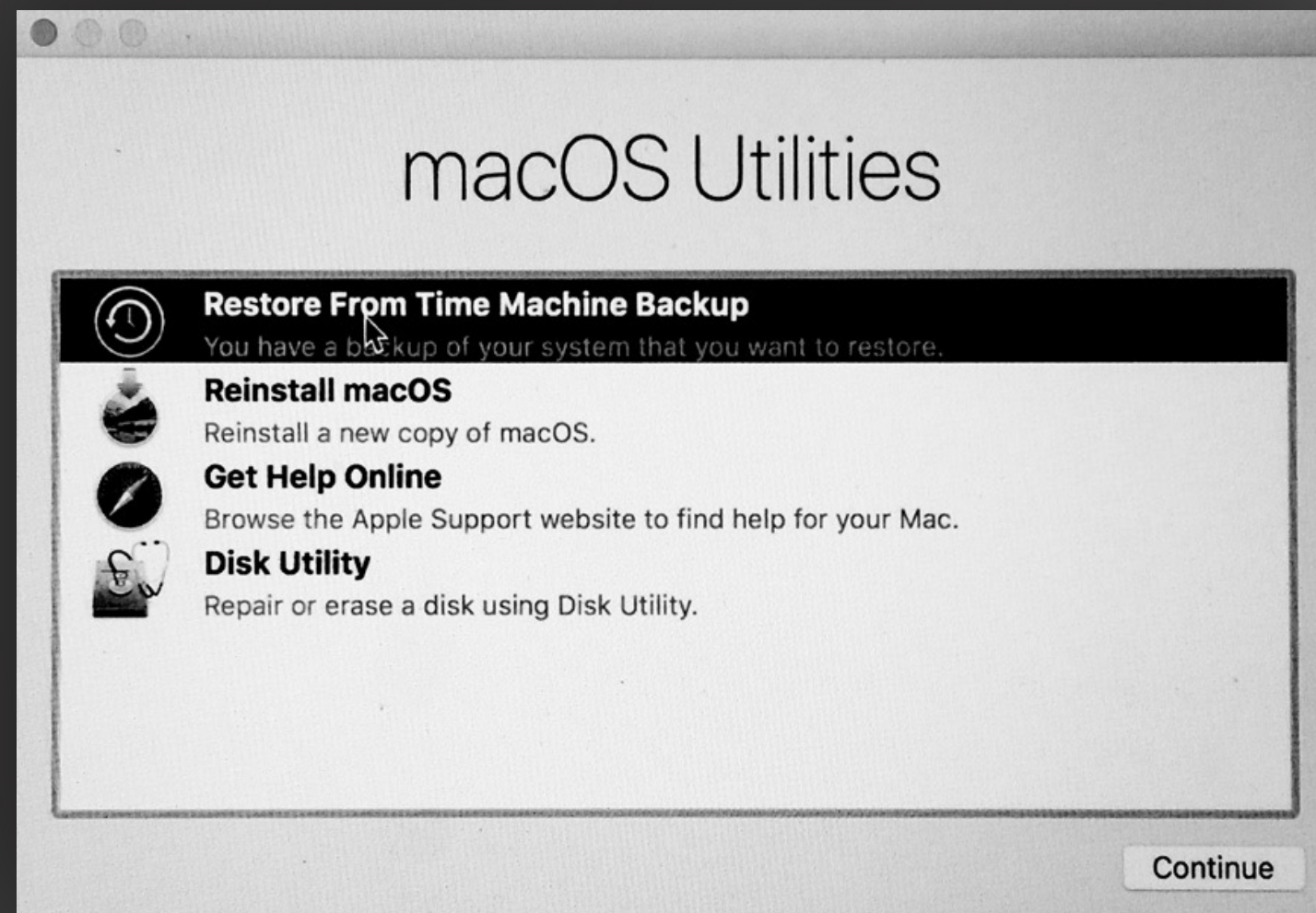
# Deleting an APFS Snapshot

A screenshot of a macOS terminal window. The title bar shows a home icon, the name "yoda", and the command "-bash" with a window size of "74x8". The terminal text shows two commands being executed. The first command, "tmutil listlocalsnapshots /", is highlighted with a red oval and returns the output "com.apple.TimeMachine.2017-09-27-171347". The second command, "tmutil deletelocalsnapshots 2017-09-27-171347", is also highlighted with a red oval and returns the output "Deleted local snapshot '2017-09-27-171347'". The prompt "Yoda:~ yoda\$" is visible at the end of each line.

```
Yoda:~ yoda$ tmutil listlocalsnapshots /  
com.apple.TimeMachine.2017-09-27-171347  
Yoda:~ yoda$ tmutil deletelocalsnapshots 2017-09-27-171347  
Deleted local snapshot '2017-09-27-171347'  
Yoda:~ yoda$
```



# Restoring from an APFS Snapshot





# APFS Encrypted Volumes

- Encryption is integrated into the file system—more secure
- Supports encrypted startup volumes
- Convert to encrypted volume while in use





```
[Yoda:~ yoda$ diskutil apfs list disk8  
|  
+-- Container disk8 A32DC745-250D-4281-9AFF-1CE8960966CA  
=====
```

APFS Container Reference:	disk8
Capacity Ceiling (Size):	239847653376 B (239.8 GB)
Capacity In Use By Volumes:	137658368 B (137.7 MB) (0.1% used)
Capacity Available:	239709995008 B (239.7 GB) (99.9% free)

```
|  
+--< Physical Store disk2s2 5F34E397-FBD9-423D-BE89-8C2282014E0B  
|-----  
| APFS Physical Store Disk: disk2s2  
| Size: 239847653376 B (239.8 GB)  
|  
+--> Volume disk8s1 B283C6B1-8AE4-4D3C-B07B-CA4ABDB25C58  
-----
```

APFS Volume Disk (Role):	disk8s1 (No specific role)
Name:	My Encrypted Data (Case-insensitive)
Mount Point:	/Volumes/My Encrypted Data
Capacity Consumed:	897024 B (897.0 KB)
Encrypted:	No

```
Yoda:~ yoda$
```

# Encrypting the Startup Volume

1. List all the APFS containers and volumes

```
[Yoda:~ yoda$ sudo diskutil apfs encryptvolume disk8s1 -user disk  
[Password:  
[Passphrase for the new "Disk" user (B283C6B1-8AE4-4D3C-B07B-CA4ABDB25C58):  
[Repeat passphrase:  
Starting background encryption of the new "Disk" user on disk8s1  
The new "Disk" user will be the only one who has initial access to disk8s1  
The new APFS crypto user UUID will be B283C6B1-8AE4-4D3C-B07B-CA4ABDB25C58  
Background encryption is ongoing; see "diskutil apfs list" to see progress  
Yoda:~ yoda$
```

2. Start the encryption  
(Relatively slow -  
6 minutes /GB)

**How fast is an  
APFS volume?**



# Speed of APFS Volumes

Two OWC ThunderBlades striped together as RAID 0



ThunderBlade





# APFS vs HFS+ Speed

Two OWC ThunderBlades striped together as RAID 0

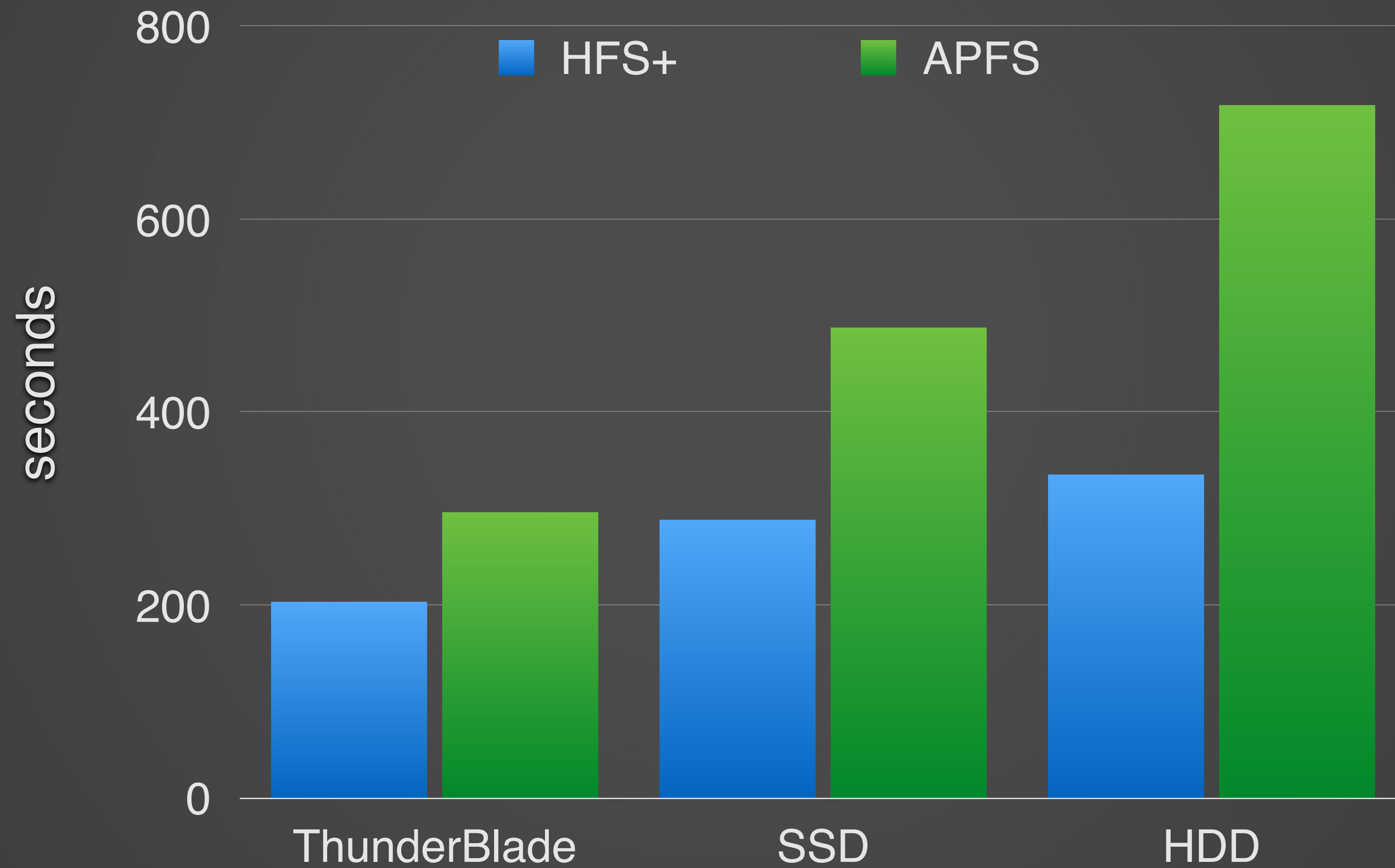
with APFS



with HFS+



# Finder Copy Speed



# Recommendations

- Don't use APFS on HDDs—EVER
- Don't encrypt or decrypt a volume with more than 200 GB of contents (200GB = 24 hours to complete)
- Only use APFS for volumes which are actively being backed up
- Use snapshots as a precaution before every software install or system update
- Expect non-Time Machine backups and file copies to take twice as long as HFS+





# Q & A

