STORAGE DEVELOPER CONFERENCE

SDC 22 | Fremont, CA
September 12-15, 2022

BY Developers FOR Developers

A SNIA Event

What's New in the MacOS SMB Client

Presented by Brad Suinn

Who am I?

Brad Suinn

- Network File Systems Engineer
- Joined Apple in late 1989
- 1989 – 1993 QA Engineer, 1993+ Development Engineer
- First Project I worked on at Apple
    - Macintosh IIfx
        - Motorola 68030 @ 40 MHz
        - 4 MB RAM expandable to 128 MB
        - 80 or 160 MB Hard Disk Drive
        - System 6.x

Topics

- Sealing algorithm updates
- MultiChannel
- Bracketed throughput algorithm
- File leasing support
- Spotlight/SMB
- Tiered/Online-only file support
- Shadow Copies/SMB Timewarp support
- Questions and maybe answers

Sealing Algorithm Updates

macOS 12 Monterey and later

Sealing Algorithm Updates

- SMB 3.1.1 now supported
  - Preauthententication Integrity check enforced
- Negotiate Context SMB2_ENCRYPTION_CAPABILITIES
  - AES-256-GCM, AES-256-CCM
  - AES-128-GCM, AES-128-CCM

MultiChannel

macOS 11 Big Sur and later

MultiChannel – Channel Discovery

- On the first connection to a MultiChannel capable server, the Client Network Interface Controller (NIC) list is created
  - Client NIC list is updated if NICs are added or removed
- Server NIC list is from server IOCTL reply and updated periodically (10 mins)
- List is created of possible channels to server for each client NIC and excludes current connected NICs
  - Each list entry has a link speed of min(client_NIC_speed, server_NIC_speed)
  - Server RSS interface is limited to 4 clients NICs by default
- List is sorted in order of fastest link speed to slowest with wired preferred over wireless by default

MultiChannel – Trials

- Trials are started from the list starting with the fastest link speed
  - Each trial attempts the TCP connection and SMB Session Binding
- Up to a maximum of 16 trials are started in parallel
- One trial per client NIC to server NIC excluding current connected NICs
- If a trial fails for a client NIC, it will try the next server NIC in the list for that client NIC
- If client or server NIC list changes, then new trials are started

MultiChannel – Trials Example

- Client NICs – 10 Gbs (en0), WiFi (en1), 1 Gbs (en2)
- Server NICs – 10 Gbs (ifIndex10, RSS capable)
- Initial connection is from en0 <-> ifIndex10, thus en0 is excluded from trials
- Sorted list: en2 <-> ifIndex10, en1 <-> ifIndex10
  - Wired is preferred over wireless by default
- Trials in parallel: en2 -> ifIndex10 started, then en1 -> ifIndex10
- List of established channels returned for more processing…

MultiChannel – Active vs Inactive Channels

- Active Channel(s)
  - One or more channels using the fastest link speeds from trials
  - SMB Traffic is round robin’d only over the active channels
- Inactive Channel
  - Next fastest channel is kept as the inactive channel
  - Only one inactive channel is kept
  - If all active channels fail, the inactive channel is promoted to be the active channel and trials are started to find more possible channels
  - After promotion, next fastest channel is designated as inactive channel
- Any remaining channels are disconnected

MultiChannel – Main vs Alternate Channels

- One active channel is designated as Main
  - Just holds some additional internal state tracking
  - If Main channel goes down, then an Alternate channel is promoted to Main
- Other active channels designated as Alternate

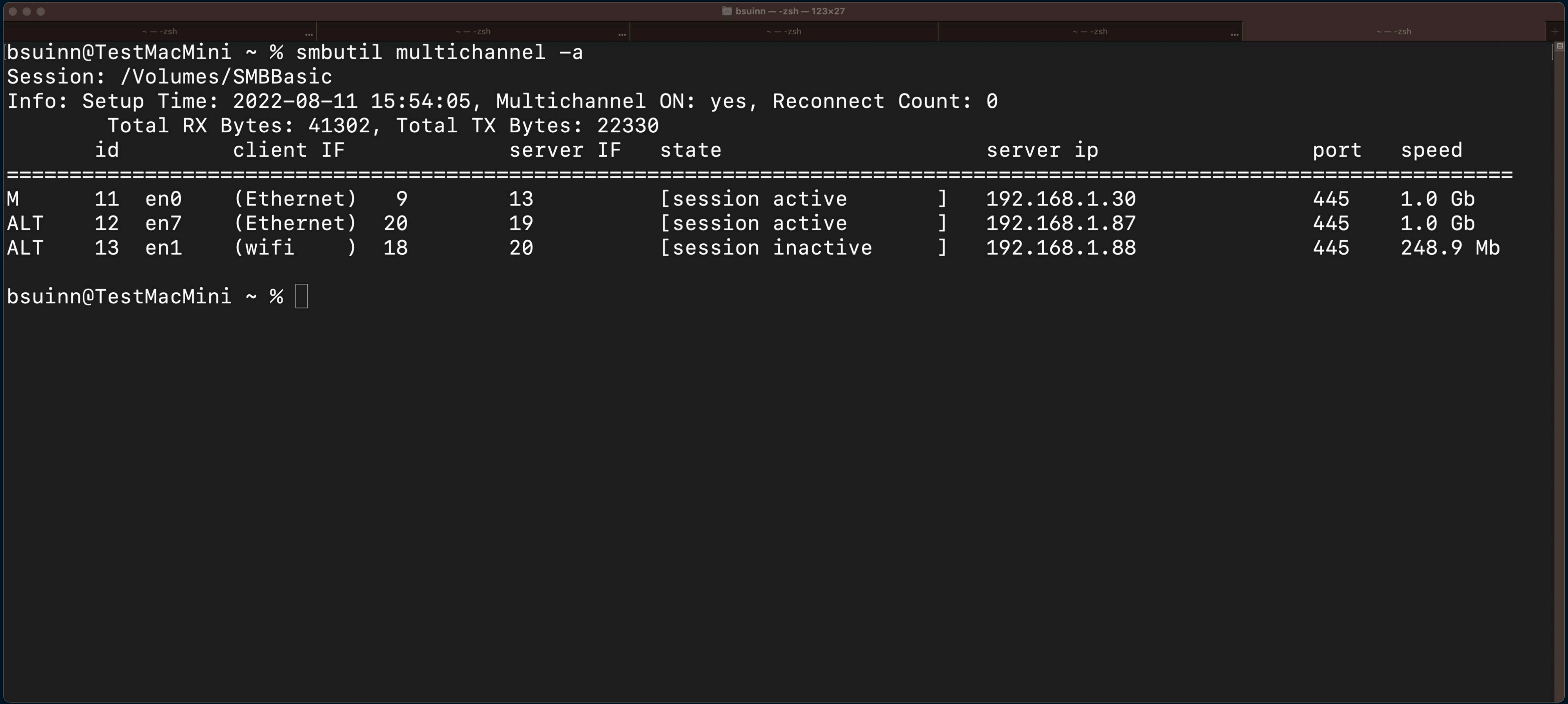
MultiChannel – Fail Over and Reconnect

- When an active channel goes down, its pending traffic fails over to another active channel and is replayed
- When the last active channel goes down, the inactive channel is promoted to active and any pending traffic is replayed
- If all active channels go down and no inactive channels left, then reconnect is attempted

MultiChannel – Debugging

- Use “smbutil multichannel” to view current channel status
- /etc/nsmb.conf, new options added
  - “mc_on” – enable or disable MultiChannel
  - “mc_prefer_wired” – enable to disable preference of wired over wireless links
- /etc/nsmb.conf, set “kloglevel=0x80” to view MultiChannel logging
  - “log stream | grep smb”
- “man nsmb.conf” or “man smbutil” for more information

MultiChannel Example

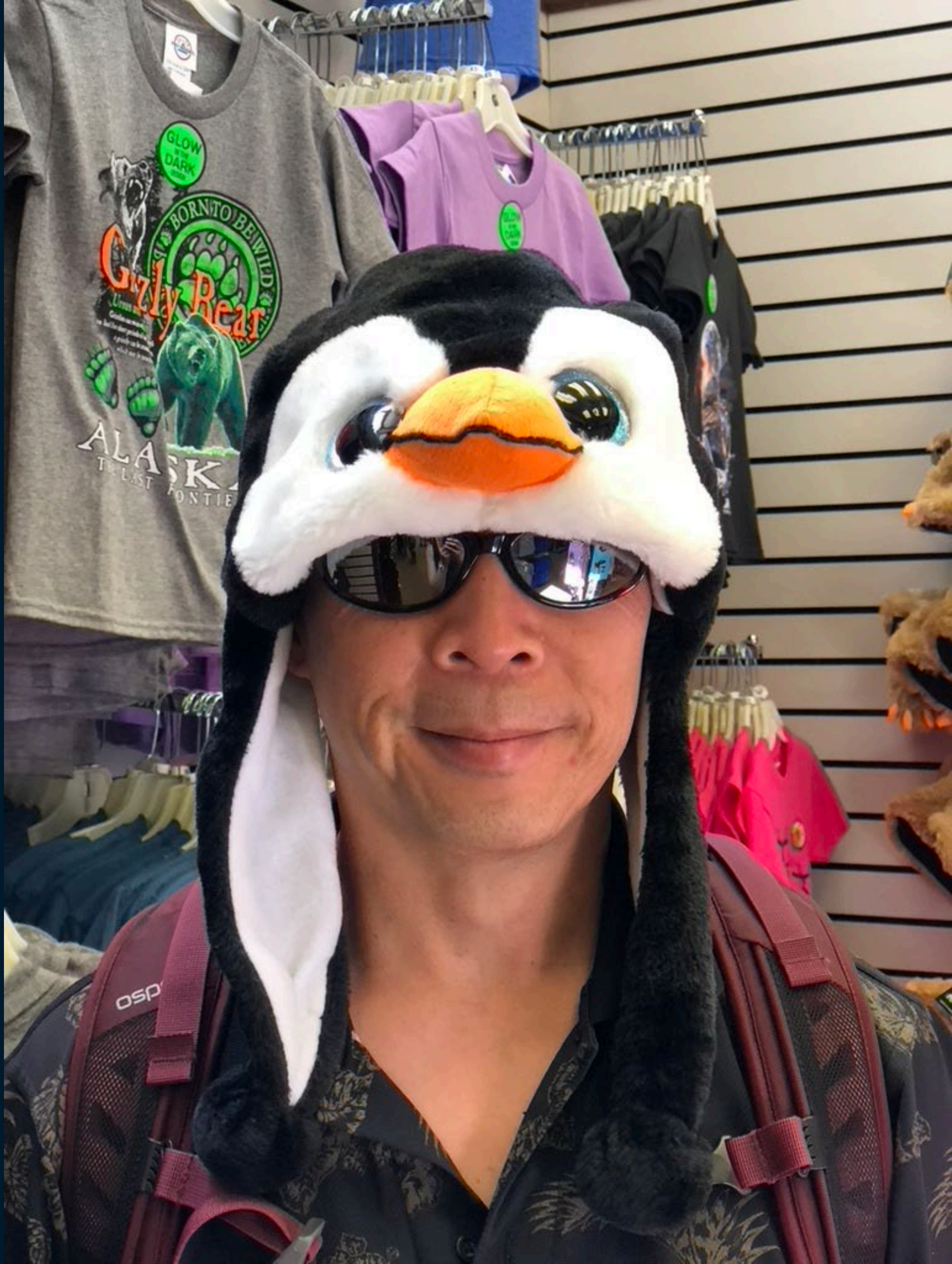


```
bsuinn@TestMacMini ~ % smbutil multichannel -a
Session: /Volumes/SMBBasic
Info: Setup Time: 2022-08-11 15:54:05, Multichannel ON: yes, Reconnect Count: 0
        Total RX Bytes: 41302, Total TX Bytes: 22330
       id         client IF              server IF   state                     server ip                 port   speed
========================================================================================================================
M      11  en0    (Ethernet)   9         13          [session active       ]   192.168.1.30              445    1.0 Gb
ALT    12  en7    (Ethernet)  20         19          [session active       ]   192.168.1.87              445    1.0 Gb
ALT    13  en1    (wifi    )  18         20          [session inactive     ]   192.168.1.88              445    248.9 Mb

bsuinn@TestMacMini ~ % 
```

Bracketed Throughput Algorithm

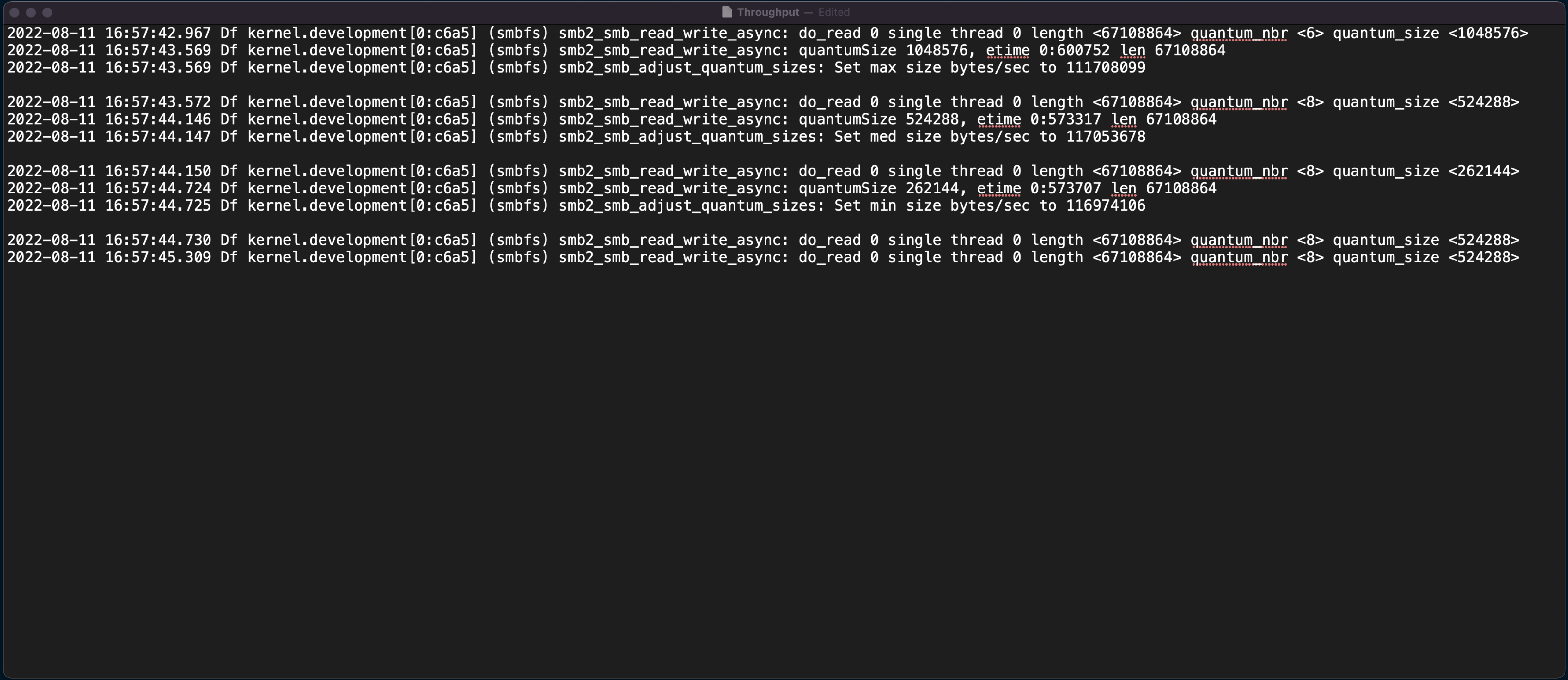
Optimizing Read/Write Throughput



Bracketed Throughput Algorithm

- Three sets tried
  - Min - 8 IO requests of 128 KB (Reads) or 256 KB (Writes)
  - Med - 8 IO requests of 512 KB
  - Max - 6 IO requests of 1.25 MB (Reads) or 1 MB (Writes)
- Calculate Bytes Per Second throughput for Min, Med, Max
- Use the set that has the fastest throughput
- Selected set expires after 60 seconds and next IO will recheck all sets
- Min/Med/Max values are selected from past empirical testing results
- /etc/nsmb.conf, set “kloglevel=0x04” to view current values and throughput results

Bracketed Throughput Example



```
2022-08-11 16:57:42.967 Df kernel.development[0:c6a5] (smbfs) smb2_smb_read_write_async: do_read 0 single thread 0 length <67108864> quantum_nbr <6> quantum_size <1048576>
2022-08-11 16:57:43.569 Df kernel.development[0:c6a5] (smbfs) smb2_smb_read_write_async: quantumSize 1048576, etime 0:600752 len 67108864
2022-08-11 16:57:43.569 Df kernel.development[0:c6a5] (smbfs) smb2_smb_adjust_quantum_sizes: Set max size bytes/sec to 111708099

2022-08-11 16:57:43.572 Df kernel.development[0:c6a5] (smbfs) smb2_smb_read_write_async: do_read 0 single thread 0 length <67108864> quantum_nbr <8> quantum_size <524288>
2022-08-11 16:57:44.146 Df kernel.development[0:c6a5] (smbfs) smb2_smb_read_write_async: quantumSize 524288, etime 0:573317 len 67108864
2022-08-11 16:57:44.147 Df kernel.development[0:c6a5] (smbfs) smb2_smb_adjust_quantum_sizes: Set med size bytes/sec to 117053678

2022-08-11 16:57:44.150 Df kernel.development[0:c6a5] (smbfs) smb2_smb_read_write_async: do_read 0 single thread 0 length <67108864> quantum_nbr <8> quantum_size <262144>
2022-08-11 16:57:44.724 Df kernel.development[0:c6a5] (smbfs) smb2_smb_read_write_async: quantumSize 262144, etime 0:573707 len 67108864
2022-08-11 16:57:44.725 Df kernel.development[0:c6a5] (smbfs) smb2_smb_adjust_quantum_sizes: Set min size bytes/sec to 116974106

2022-08-11 16:57:44.730 Df kernel.development[0:c6a5] (smbfs) smb2_smb_read_write_async: do_read 0 single thread 0 length <67108864> quantum_nbr <8> quantum_size <524288>
2022-08-11 16:57:45.309 Df kernel.development[0:c6a5] (smbfs) smb2_smb_read_write_async: do_read 0 single thread 0 length <67108864> quantum_nbr <8> quantum_size <524288>
```

File Leasing Support

macOS 12 Monterey (and Earlier) File Leasing Behavior

- Only files that were locally opened with O_EXLOCK (share none) or O_SHLOCK (share read/share delete) would request a lease and durable handle
- Durable Handle V1 and Lease V1 format are used unless it is a Time Machine/SMB mount
  - Note: Time Machine is the macOS Backup mechanism
  - Time Machine/SMB mount will check for V2 support
- After a lease break, client will not try to upgrade the lease
- Local data caching controlled separately from current lease state
- Not that many files are opened with O_EXLOCK or O_SHLOCK

macOS 13 Ventura File Leasing Behavior

- All files now will request a lease and durable handle
- If an open file has a broken lease, it will periodically attempt to upgrade back to the original granted lease using a compound Create/Close
- Local data caching controlled by current lease state
- Reconnect will have more files to do durable handle reconnects on
- Assumes Durable Handle V2 and Lease V2 is supported if SMB 3.x dialect
  - [MS-SMB] 3.2.4.3.5

Spotlight/SMB Support

Current and upcoming support

macOS 12 Monterey (and Earlier) Spotlight/SMB Behavior

- macOS SMB server - Spotlight is used which is a proprietary protocol
- Non macOS SMB Server - Crawling mode is used
- After a share is mounted, the named pipe of mdsscv is attempted to be opened for Spotlight/SMB
- If mdssvc is not found, then Crawling mode is used where the client Spotlight “crawls” through the share over the network and builds a local database
  - No content search is available
  - Database is destroyed when the share is unmounted

macOS 13 Ventura Spotlight/SMB Behavior

- WSP is now supported for Spotlight/SMB
- After a share is mounted, the named pipe of mdsscv is attempted to be opened for Spotlight/SMB
- If mdssvc is not found, then WSP is attempted
  - [MS-WSP]
- If WSP is found, then Spotlight queries are translated into WSP queries and WSP results are translated to Spotlight results
- If Spotlight/SMB and WSP are not found, then use Crawling mode

Spotlight/WSP

- Standard attributes like Filename, Size, Dates
- Advanced attributes like
  - Title, Copyright, Author, Publishers, Vendors
  - DPI Resolution, Pixel Height, Pixel Width, ISO, Aperture
  - Album Name, Recording Year, Composer, Track Number
- File content searching supported
- Logical (AND/OR/NOT), comparisons (==, <, >), RegEx

Tiered/Online-only File Support

- macOS 11 Big Sur and later

Tiered/Online-only File Support - Declarations

- Using reparse tag of IO_REPARSE_TAG_STORAGE_SYNC (0x8000001e)
- File Attribute Bits
  - L (0x00000400) FILE_ATTRIBUTE_REPARSE_POINT
  - M (0x00400000) FILE_ATTRIBUTE_RECALL_ON_DATA_ACCESS
    - Definitive answer bit. Open will not recall file, IO will recall file
    - Not user settable and used by newer Microsoft servers (2019+)
  - O (0x00001000) FILE_ATTRIBUTE_OFFLINE
    - Not always set and used by older Microsoft servers (<= 2016)
  - P (0x00000200) FILE_ATTRIBUTE_SPARSE_FILE
    - User settable so not as reliable. Used by older Microsoft servers (<= 2016)

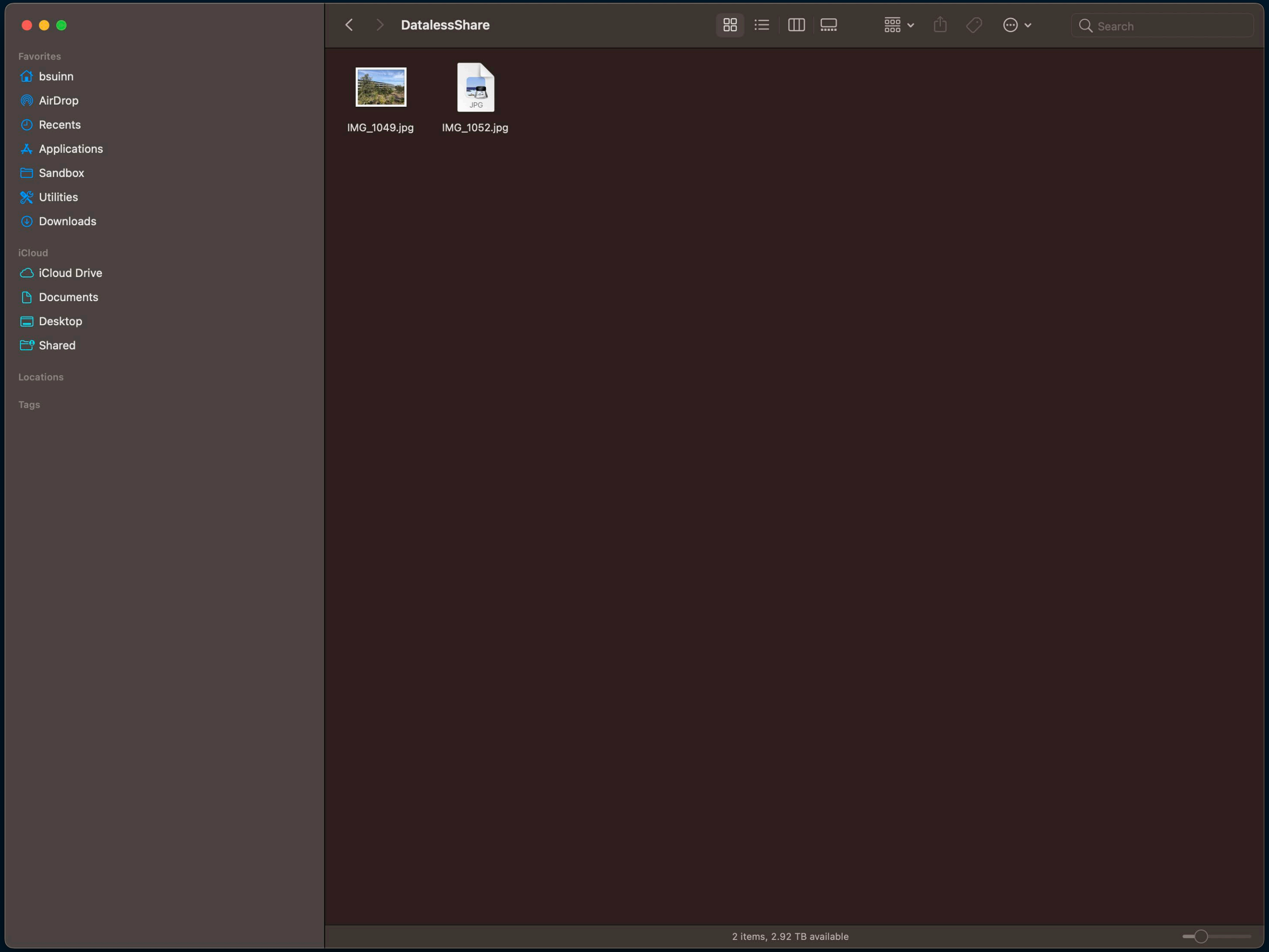
Tiered/Online-only File Support – SMB Client Behavior

- File must be a reparse point and have a tag of IO_REPARSE_TAG_STORAGE_SYNC
- If M bit is set or (P or O) bits are set then it is a currently tiered file
- Once a file is fully recalled, L, M, O, P bits are cleared by the server
- Metadata is from actual file except physical size which is the amount of current recalled data
- Querying for or reading named streams will not recall the file

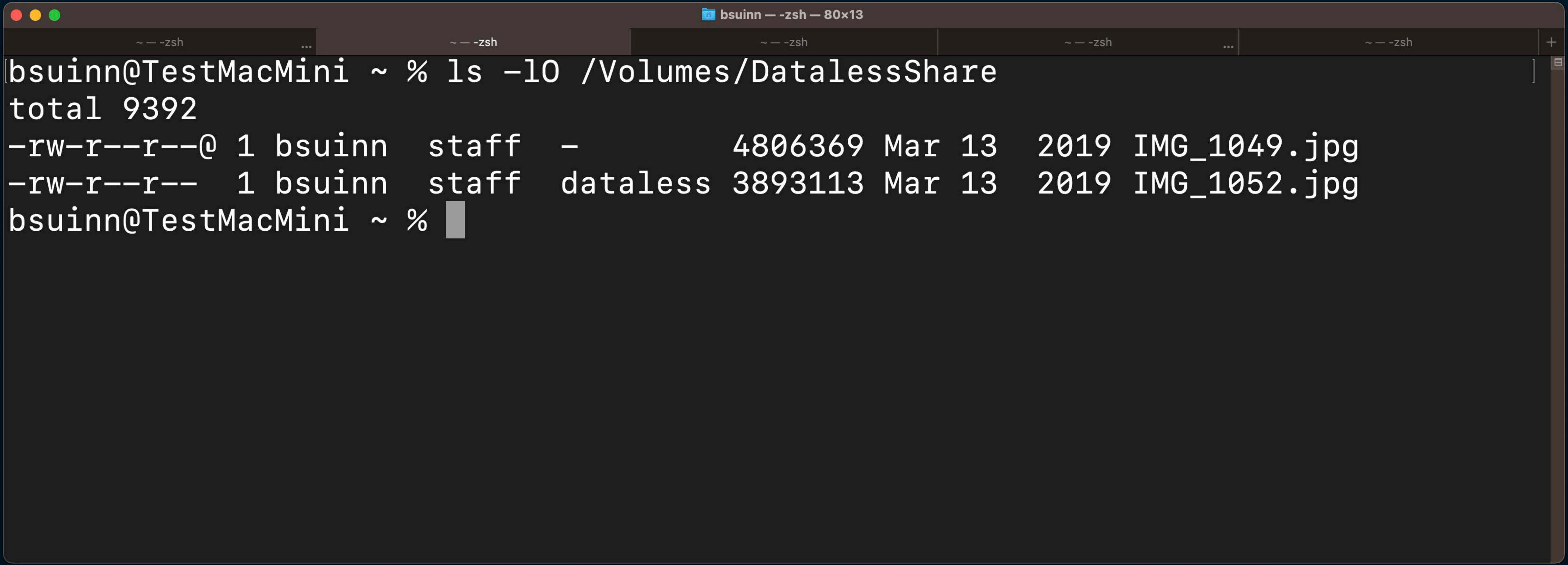
Tiered/Online-only File Support – macOS Client Behavior

- If M bit is set
  - Opening file is allowed. Only IO operations will recall the file
- If P or O bit is set
  - Assume opening file or IO will recall the file
- Process is checked to see if it has the entitlement to recall the file
  - If no entitlement, the operation is returned with an error
    - Example, Finder thumbnail generation

Tiered/Online-only Finder Example



Tiered/Online-only Terminal Example



```
bsuinn@TestMacMini ~ % ls -lO /Volumes/DatalessShare
total 9392
-rw-r--r--@ 1 bsuinn  staff  -        4806369 Mar 13  2019 IMG_1049.jpg
-rw-r--r--  1 bsuinn  staff  dataless 3893113 Mar 13  2019 IMG_1052.jpg
bsuinn@TestMacMini ~ % 
```

Shadow Copies / SMB Timewarp Support

macOS 11 Big Sur and later

Shadow Copies/SMB Timewarp Definitions

- Shadow Copy
  - A snapshot of a volume that duplicates all of the data on a volume at one instant in time
- SMB Timewarp
  - Create Context SMB2_CREATE_TIMEWARP_TOKEN
  - Allows client to request the server open a version of a file or directory at a previous point in time

Shadow Copies/SMB Timewarp Support

- List out available snapshots on a mounted share in gmt_token format
  - “smbutil snapshot [-a] [-m mount_path]”
- Mount a share with a snapshot
  - “mount_smbfs -t <@gmt_token> //<SMB URL> <mount_point>”
  - “mount -t smbfs -o snapshot=<@gmt_token> <SMB URL> <mount_point>”
  - Mount is Read Only
  - Finder will show share name as “ShareName@gmt_token”
- All Create requests have the Timewarp Create Context

Questions?

Please take a moment to rate this session.

Your feedback is important to us.