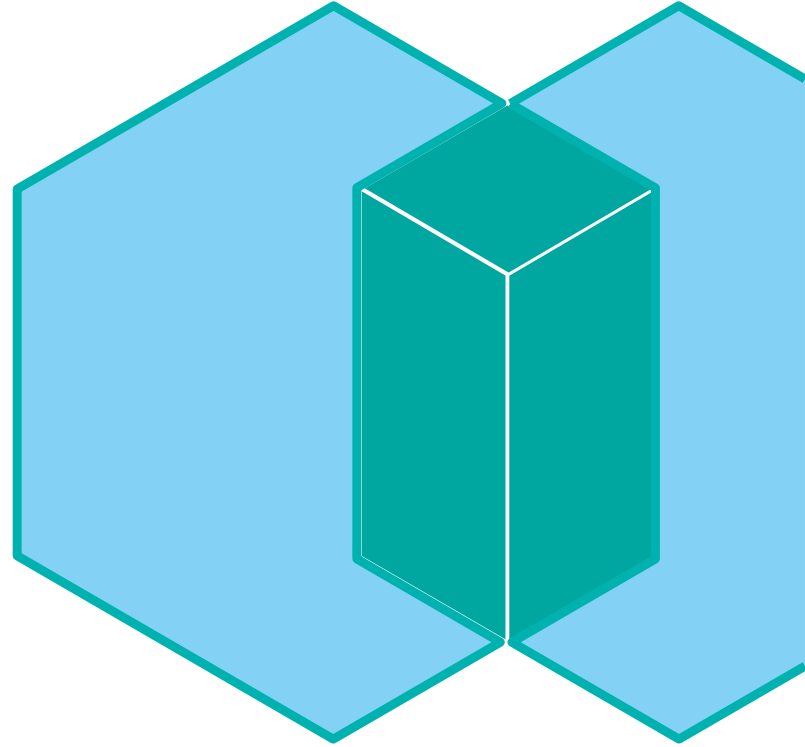# IBM Spectrum Scale

Spectrum Scale 4.2 File compression

Agenda

Spectrum Scale 4.2: Compression Overview

Spectrum Scale 4.2: Compression Current Implementation

Spectrum Scale 4.2: Closer Look to Compression

Spectrum Scale 4.2: Current limitations to Compression implementation

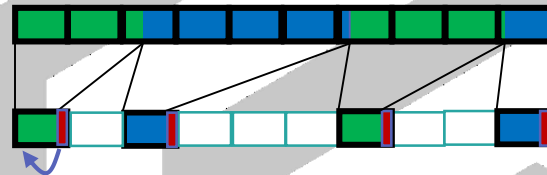Spectrum Scale 4.2: Compression

SPECTRUM SCALE COMPRESSION OVERVIEW

Spectrum Scale 4.2: File Compression

Compression / decompression is controlled per file
- Not live compression
  - Policy-driven background compression/decompression
  - CLI-driven compression/decompression with **mmchattr**
- Platform support: Linux, AIX, Windows
- Compression algorithm/library used: deflate/zlib favors saving space over speed
- Inline read of compressed file
- Inline decompression of compression group upon writes

Improved storage efficiency
- Reclaim space from cold data
- Improvement in storage efficiency depends on the file itself
  - Typically 2:1 to 5:1
  - But could reach 10:1 on trivial files or none on random ones

Vision

Which files to compress

When to compress the file data
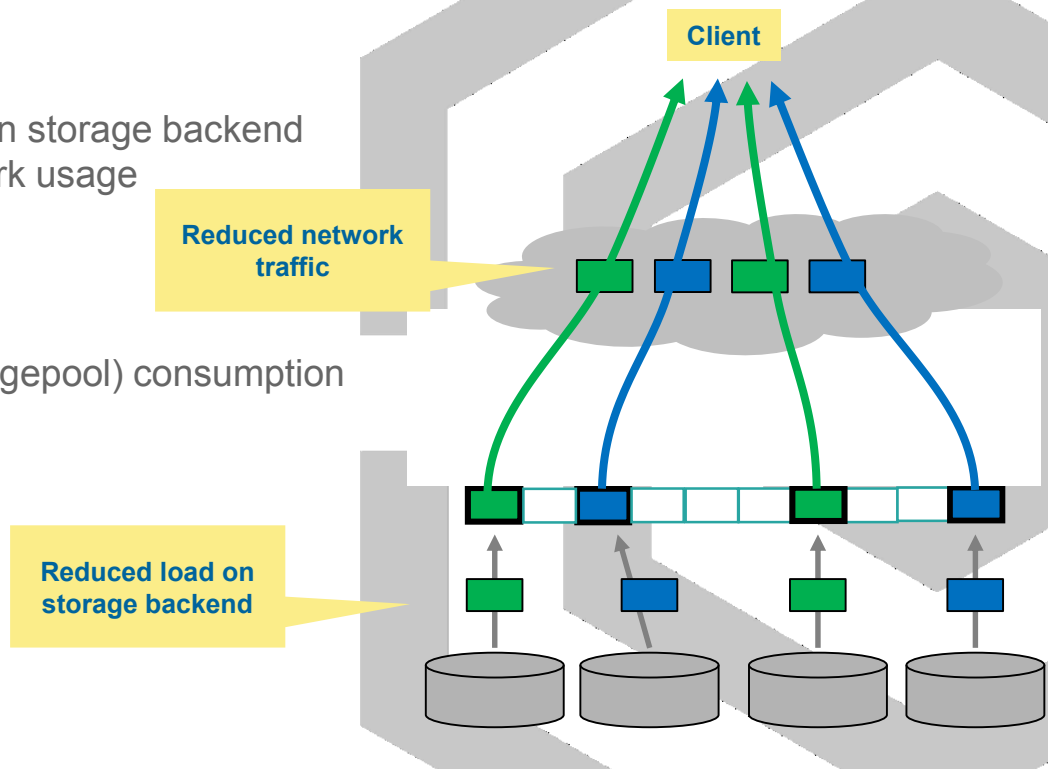
How to compress the file data

Spectrum Scale 4.2: File Compression

Improved i/o bandwidth
- Read/write compressed data reduces load on storage backend
- Read/write compressed data reduces network usage

Improved client side caching
- Caching compressed data reduce cache (pagepool) consumption

Client

Reduced network traffic

Reduced load on storage backend

Spectrum Scale 4.2: File Compression

Please keep in mind …

- With this release compression is primarily for: **Cold Data**

- If storage path is constraint, reading compressed data is slower than reading raw data
  - due to on-the-fly zlib decompression overhead
  - to be addressed by using faster algorithm (such as lz4), hardware acceleration, and other optimizations

- Reduced disk IO can benefit read performance more in a busier system

- **Update to PTF 4.2.0.2 or above** [10 March 2016] Bug fix
  - which could have occurred on previous version when a user write on a compressed file, while mmrestripefs –z is running in background

Spectrum Scale 4.2: Compression

CURRENT IMPLEMENTATION

Compress/Decompress Files

CLI command

- **mmchattr** command:
  - **mmchattr --compression { yes | no } -I { yes | defer } File**
    mmchattr --compression yes -I yes /gpfs/gpfs0/test-64MB-0
    mmchattr --compression yes -I defer /gpfs/gpfs0/test-64MB-0
    mmchattr --compression no -I yes /gpfs/gpfs0/test-64MB-0
    mmchattr --compression no -I defer /gpfs/gpfs0/test-64MB-0

The compression (--compression yes) or decompression (--compression no) is either

- Synchronous with **-I yes** : The file is compressed/decompressed immediately
- Deferred with **–I defer** : the file will be compressed/decompressed with a **mmrestripefs -z** or **mmrestripefile** -z
  - **mmrestripefs gpfs0 –z**
  - **mmrestripefile –z gpfs/gpfs0/test-64MB-0**

- The file is marked **illcompressed** in its flags until the restripe is done

Compress Files synchronously

64 MB

```
[root@vm01 gpfs0]#  ls  -s --block-size=K /gpfs/gpfs0/test-64MB-0
65536K /gpfs/gpfs0/test-64MB-0
[root@vm01 gpfs0]# du -sm /gpfs/gpfs0/test-64MB-0
64      /gpfs/gpfs0/test-64MB-0
[root@vm01 gpfs0]# mmlsattr -L /gpfs/gpfs0/test-64MB-0
file name:              /gpfs/gpfs0/test-64MB-0
metadata replication: 1 max 1
data replication:       1 max 1
immutable:              no
appendOnly:             no
flags:
storage pool name:      system
fileset name:           root
snapshot name:
creation time:          Wed May 11 07:27:27 2016
Misc attributes:        ARCHIVE
Encrypted:              no
[root@vm01 gpfs0]#
```

- The file before compression
  - 64 MB

Compress Files synchronously

```
[root@vm01 gpfs0]# mmchattr --compression yes -I yes /gpfs/gpfs0/test-64MB-0
[root@vm01 gpfs0]# mmlsattr -L /gpfs/gpfs0/test-64MB-0
file name:            /gpfs/gpfs0/test-64MB-0
metadata replication: 1 max 1
data replication:     1 max 1
immutable:            no
appendOnly:           no
flags:
storage pool name:    system
fileset name:         root
snapshot name:
creation time:        Wed May 11 07:27:27 2016
Misc attributes:      ARCHIVE COMPRESSED
Encrypted:            no
[root@vm01 gpfs0]#
```

COMPRESSED

- The file after sync compression
  - ARCHIVE COMPRESSED

Compress Files synchronously

< 7MB

```
[root@vm01 gpfs0]#  ls  -s --block-size=K /gpfs/gpfs0/test-64MB-0
6592K /gpfs/gpfs0/test-64MB-0
[root@vm01 gpfs0]# du -sm /gpfs/gpfs0/test-64MB-0
7       /gpfs/gpfs0/test-64MB-0
[root@vm01 gpfs0]#
```

- The file after sync compression
  - Size 6592KB

Compress Files asynchronously, defer option

64 MB

```
[root@vm01 gpfs0]#  ls  -s --block-size=K /gpfs/gpfs0/test-64MB-urand
65536K /gpfs/gpfs0/test-64MB-urand
[root@vm01 gpfs0]# du -sm /gpfs/gpfs0/test-64MB-urand
64      /gpfs/gpfs0/test-64MB-urand
[root@vm01 gpfs0]# mmlsattr -L /gpfs/gpfs0/test-64MB-urand
file name:              /gpfs/gpfs0/test-64MB-urand
metadata replication: 1 max 1
data replication:     1 max 1
immutable:            no
appendOnly:           no
flags:
storage pool name:    system
fileset name:         root
snapshot name:
creation time:        Wed May 11 07:27:43 2016
Misc attributes:      ARCHIVE
Encrypted:            no
[root@vm01 gpfs0]#
```

- The file before compression
  - 64 MB

Compress Files asynchronously, defer option

defer

```
[root@vm01 ~]# mmchattr --compress yes -I defer /gpfs/gpfs0/test-64MB-urand
[root@vm01 ~]# mmlsattr -L /gpfs/gpfs0/test-64MB-urand
file name:              /gpfs/gpfs0/test-64MB-urand
metadata replication: 1 max 1
data replication:       1 max 1
immutable:              no
appendOnly:             no
flags:                  illcompressed
storage pool name:      system
fileset name:           root
snapshot name:
creation time:          Wed May 11 07:27:43 2016
Misc attributes:        ARCHIVE COMPRESSED
Encrypted:              no
```

ILLCOMPRESSED

- The file after async compression
  - ARCHIVE COMPRESSED
  - Flag illcompressed

Compress Files asynchronously, defer option

mmrestripefs -z

```
[root@vm01 ~]# mmrestripefile -z /gpfs/gpfs0/test-64MB-urand
[root@vm01 ~]# mmlsattr -L /gpfs/gpfs0/test-64MB-urand
file name:              /gpfs/gpfs0/test-64MB-urand
metadata replication: 1 max 1
data replication:       1 max 1
immutable:              no
appendOnly:             no
flags:
storage pool name:      system
fileset name:           root
snapshot name:
creation time:          Wed May 11 07:27:43 2016
Misc attributes:        ARCHIVE COMPRESSED
Encrypted:              no
```

- The file is restriped
- It is now COMPRESSED

Compress Files asynchronously, defer option

64 MB

```
01 ~]# ls -l  --block-size=K /gpfs/gpfs0/test-64MB-urand
  r--. 1 root root 65536K May 11 17:45 /gpfs/gpfs0/test-64MB-urand
[root@vm01 ~]# du -sm /gpfs/gpfs0/test-64MB-urand
64      /gpfs/gpfs0/test-64MB-urand
[root@vm01 ~]#
```

- The file is compressed
- Its size is same due to is complexity

Compress/Decompress Files with management policy

Policy-Driven, management policy

- Similar to data replication factor change through **MIGRATE** policy rule
- **MIGRATE** supports **COMPRESS('yes')** and **COMPRESS ('no')**
- **Example**
  - RULE 'COMPRESSION' MIGRATE COMPRESS('yes') WHERE NAME LIKE 'blue%'
  - This rule compress all files starting by blue

- mmapplypolicy [-I {yes | defer}] –P <policy file name>

- Per-file Compression Control and Status bits in inode MISC_ATTRIBUTES
  - **c** the file is **selected to be compressed**
  - **K** some data blocks are **ill-compressed**

File attributes with compression

COMPRESSED	illcompressed	Meaning
Not displayed	Not displayed	The file is **not compressed**. Any pending decompression is completed
Not displayed	Displayed	The file is marked for **decompression** which is **deferred**
Displayed	Not displayed	The file is **compressed**. Any pending compression is completed
Displayed	Displayed	The file is marked for **compression**. Compression is **deferred**, or part of the file is decompressed due to a file update or memory mapping

Spectrum Scale 4.2: Compression

CLOSER LOOK TO COMPRESSION

A Closer Look At a Compressed File

```
[root@vm03 ~]# ls -i /gpfs/gpfs0/test-64MB-0
14848 /gpfs/gpfs0/test-64MB-0
[root@vm03 ~]# tsdbfs gpfs0 listda 14848
Inode 14848 snap 0       2:569856  status=USERFILE indirection=2
Inode Disk Pointers:
    0:  2:418048    1:  3:216448    2: (null)        ...
  323: (null)
Indirect block starting in sector 2:418048:
  magic=0x112DF307 generation=0x173E...  blockNum=0 inodeNum=14848
  indirection level=1
  checksum=0x9D...035B
  Data pointers:
    0:  3:314752    1:  1:416768    2:  2:1028864    3:  3:722816
    4:  1:......8   5:  2:722816    6:  3:212736     7:  1:1028864
    8:  2:926848    9:  3:824832   10:  1:212736    11:  2:212736
   12:  3:926848   13:  1:314752   14:  2:416768    15:  3:926720
   16:  1:926848   17:  2:314752   18:  3:722688    19:  1:926720
   20:  2:722688   21:  3:416640   22:  1:824704    23:  2:824704
   24:  3:212608   25:  1:314624   26:  2:416640    27:  3:1028736
   28:  1:1028736  29:  2:926720   30:  3:314624    31:  1:722688
   32:  2:314624   33:  3:824704   34:  1:416640    35:  2:1028736
   36:  3:314240   37:  1:212608   38:  2:212608    39:  3:416256
   40:  1:212224   41:  2:1028352  42:  3:824320    43:  1:314240
   44:  2:926336   45:  3:926336   46:  1:926336    47:  2:314240
```

Before the compression of the file

- **tsdbfs gpfs0 listda i** :display disk address list for inode i in the file system gpfs0

- The inode of the file has an address on disk 2 sector 569856

- This sector contains indirect blocks addresses

- The indirect blocks have addresses of the data on disks at sector n: 3:314752
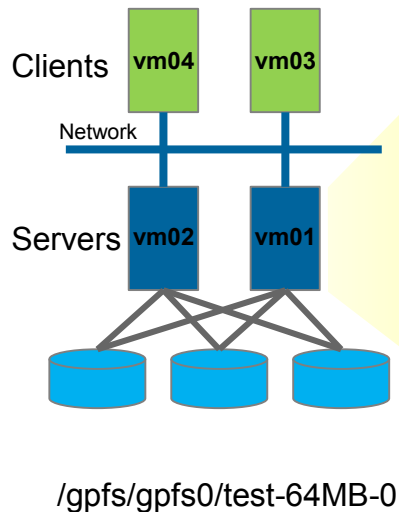
A Closer Look At a Compressed File

```
[root@vm03 ~]# tsdbfs gpfs0 listda 14848
Inode 14848 snap 0 address 2:569856 status=USERFILE indirection=2
Inode Disk Pointers:
    0:  2:418048    1:  3:216448    2:  (null)        ...
  323:  (null)
Indirect block starting in sector 2:418048:
  magic=0x112DF307 generation=0x173EDE5A blockNum=0 inodeNum=14848
  indirection level=1
  checksum=0x44217D6D
  Data pointers:
    0: C 3:1032064    1:  (znull)        ...        9:  (znull)
   10: C 1:828032    11:  (znull)        ...       19:  (znull)
   20: C 2:826112    21:  (znull)        ...       29:  (znull)
   30: C 3:828032    31:  (znull)        ...       39:  (znull)
   40: C 1:1032064   41:  (znull)        ...       49:  (znull)
   50: C 2:728064    51:  (znull)        ...       59:  (znull)
   60: C 3:726144    61:  (znull)        ...       69:  (znull)
   70: C 1:726144    71:  (znull)        ...       79:  (znull)
   80: C 2:216064    81:  (znull)        ...       89:  (znull)
   90: C 3:828160    91:  (znull)        ...       99:  (znull)
  100: C 1:930176   101:  (znull)        ...      109:  (znull)
  110: C 2:420096   111:  (znull)        ...      119:  (znull)
  120: C 3:420096   121:  (znull)        ...      129:  (znull)
  130: C 1:1032192  131:  (znull)        ...      139:  (znull)
```

After the compression of the file

- **tsdbfs gpfs0 listda i** :display disk address list for inode i in the file system gpfs0

- The inode of the file has not changed

- Indirect blocks addresses have not changed

- The data blocks have change
  - Grouped by 10 in **compression group**
  - Compressed data blocks have compressed flag: **C**
  - Deallocated block have **znull**

Accessing compressed files


```
[root@vm01 gpfs0]# mmchattr --compression yes -I yes /gpfs/gpfs0/test-64MB-0
[root@vm01 gpfs0]# mmlsattr -L /gpfs/gpfs0/test-64MB-0
file name:              /gpfs/gpfs0/test-64MB-0
metadata replication: 1 max 1
data replication:       1 max 1
immutable:              no
appendOnly:             no
flags:
storage pool name:      system
fileset name:           root
snapshot name:
creation time:          Wed May 11 07:27:27 2016
Misc attributes:        ARCHIVE COMPRESSED
Encrypted:              no
[root@vm01 gpfs0]#  ls  -s --block-size=K /gpfs/gpfs0/test-64MB-0
6592K /gpfs/gpfs0/test-64MB-0
[root@vm01 gpfs0]# du -sm /gpfs/gpfs0/test-64MB-0
7       /gpfs/gpfs0/test-64MB-0
[root@vm01 gpfs0]#
```

Clients — vm04 vm03

Network

Servers — vm02 vm01

/gpfs/gpfs0/test-64MB-0

The compressed file is 6.5 MB

Accessing compressed files: reads

6.5 MB
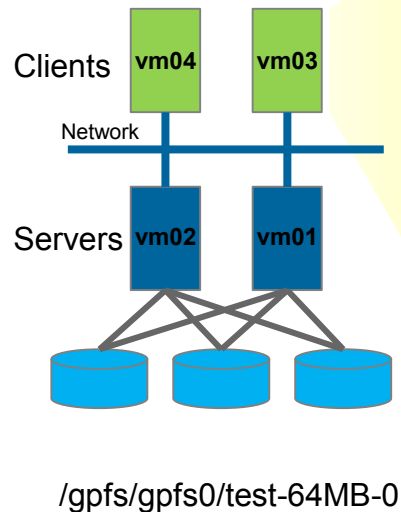
```
[root@vm03 ~]# ls -s --block-size=K /gpfs/gpfs0/test-64MB-0
6592K /gpfs/gpfs0/test-64MB-0
[root@vm03 ~]# ifconfig enp0s8
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.103  netmask 255.255.255.0  broadcast 192.168.56.255
        inet6 fe80::a00:27ff:fe36:df3a  prefixlen 64  s
        ether 08:00:27:36:df:3a  txqueuelen 1000   (Ethe
        RX packets 89526  bytes 6351123   (6.0 MiB)
        RX errors 0  dropped 0  overruns 0  frame
        TX packets 89954  bytes 38666214 (36.8 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[root@vm03 ~]# dd if=/gpfs/gpfs0/test-64MB-0 of=/dev/nu
131072+0 records in
131072+0 records out
67108864 bytes (67 MB) copied, 0.982337 s, 68.3 MB/s
[root@vm03 ~]# ifconfig enp0s8
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.103  netmask 255.255.255.0  broad
        inet6 fe80::a00:27ff:fe36:df3a  prefixlen 64  s
        ether 08:00:27:36:df:3a  txqueuelen 1000   (Eth
        RX packets 94656  bytes 1346311   (12.8 MiB)
        RX errors 0  dropped 0  overruns 0  frame
        TX packets 90547  bytes 38825508 (37.0 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Before the read the RX is 6 MB

Read the file from the client over the network

After the read the RX is 12.8 MB : size of compressed file on disk

Clients vm04 vm03

Network

Servers vm02 vm01

/gpfs/gpfs0/test-64MB-0

Accessing compressed files: reads

Reading compressed files

- The file is decompressed on the fly where the read is issued
- The file remains compressed on disk
- Reduced disk IOs , only the compressed blocks are retrieved from the disks
- Reduced network transfers, only the compressed blocks are send on the network
- The read performance is increased when
  - The cost of decompression is balanced by the gain of blocks transfer
  - So it depends on the network type and storage performance

Example on VMs	Compressed	Decompressed
Read from NSD server (disk local)	81.2 MB/s	125 MB/s
Read from client	60.9 MB/s	55.0 MB/s

Decompression slows down on the NSD server

Compressed read faster on the client

Accessing compressed files: writes
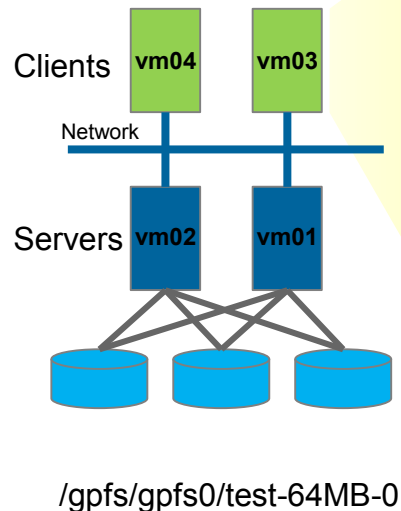

```
[root@vm03 ~]# dd if=/dev/urandom of=/gpfs/gpfs0/test-64MB-0 bs=64K count=1 conv=notrunc
1+0 records in
1+0 records out
65536 bytes (66 kB) copied, 0.00535237 s, 12.2 MB/s
[root@vm03 ~]# ls -i /gpfs/gpfs0/test-64MB-0
47360 /gpfs/gpfs0/test-64MB-0
[root@vm03 ~]# tsdbfs gpfs0 listda 47360
Inode 47360 snap 0 address 3:652928 status=USERFILE indirection=2
Inode Disk Pointers:
    0:  3:411904    1:  1:430608    2:  (null)       ...
  323:  (null)
Indirect block starting in sector 3:411904:
  magic=0x112DF307 generation=0x7AC95DD3 blockNum=0 inode
  indirection level=1
  checksum=0xA8E36593
  Data pointers:
    0:  3:1030144    1:  1:418048    2:  2:928128    3:  3:928128
    4:  1:826112    5:  2:214016    6:  3:316032    7:  1:214016
    8:  2:724096    9:  3:418048   10: C 1:215296   11: (znull)
          ...      19: (znull)     20: C 2:929408   21: (znull)
          ...      29: (znull)     30: C 3:929408   31: (znull)
          ...      39: (znull)     40: C 1:317312   41: (znull)
          ...      49: (znull)     50: C 2:827392   51: (znull)
          ...      59: (znull)     60: C 3:215296   61: (znull)
          ...      69: (znull)     70: C 1:929408   71: (znull)
```

Modify 64KB in first data block

The compression group containing the modified block is decompressed

Clients: vm04 vm03

Network

Servers: vm02 vm01

/gpfs/gpfs0/test-64MB-0

Accessing compressed files: writes

Clients

vm04 **vm03**

Network

Servers

vm02 **vm01**
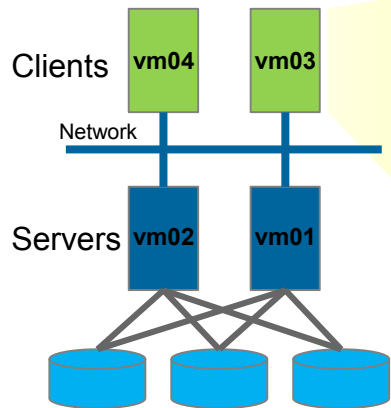
```
[root@vm03 ~]# mmlsattr -L /gpfs/gpfs0/test-64MB-0
file name:              /gpfs/gpfs0/test-64MB-0
metadata replication: 1 max 1
data replication:      1 max 1
immutable:             no
appendOnly:            no
flags:                 illcompressed
storage pool name:     system
fileset name:          root
snapshot name:
creation time:         Wed May 11 23:09:39 2016
Misc attributes:       ARCHIVE COMPRESSED
Encrypted:             no
[root@vm03 ~]#
```

File is illcompressed

/gpfs/gpfs0/test-64MB-0

Accessing compressed files

Writing in compressed files

- The parts of the files touched are decompressed
- The decompression is **synchronous** on the disks
- The granularity is 10 data blocks decompressed
  - 1 data block touched, implies the 9 other of the decompression group are decompressed
- The file become **illcompressed**
- It needs to be restriped with **mmrestripefs -z** or **mmrestripefile -z**

Spectrum Scale 4.2: Compression

CURRENT LIMITATIONS

Current Limitations

File compression for cold data
- Zlib data compression library is used. It favors **space over speed**
- If the calculated potential saving space is below 10%, the compression group (10 data blocks) is not compressed

The following operations are not supported:
- Compressing files in snapshots
- Compressing a clone or clone parent, or directories
- Compressing files in an AFM cache site or in an AFM-based asynchronous Disaster Recovery (DR) fileset
- Compressing small files (files that consume fewer than two subblocks, compressing small files into an inode)
- Compressing files in an FPO environment or horizontal storage pools
- Cloning a compressed file

On Windows:
- **mmapplypolicy** command is not available. **mmchattr** command is available to compress/decompress.
- Compression of files in Windows hyper allocation mode is not supported.
- The following Windows APIs are not supported:
  - FSCTL_SET_COMPRESSION to enable/disable compression on a file
  - FSCTL_GET_COMPRESSION to retrieve compression status of a file
- In Windows Explorer, in the Advanced Attributes window, the compression feature is not supported.

DIO, Mapped Files, mmrestorefs, etc.

Compress Direct IO file is not supported

- Direct IO is turned into buffered IO on compressed files
- To help enforce direct IO, Spectrum Scale does not compress a file that is already opened for Direct IO. An error is returned the compression command

Memory Mapped File

- If the file is already compressed, upon page-in request, Spectrum Scale automatically decompresses the paged-in region and set the illcompressed flag of the file.
- Spectrum Scale does not compress a file already memory mapped. An error is returned
- HSM, backup, ADR, AFM: Not compression-aware
  - HSM migrate/recall and backup/restore decompressed data and cause files to become illcompressed
  - Use mmrestripefs or mmrestripefile command to recompress the recalled/restored files

mmrestorefs: Restored data from snapshot is decompressed

- Do not compress files system (using policy, mmchattr, or mmrestripefs) during mmrestorefs to avoid a data loss risk (this will be fixed in PTF1).

HSM, backup, ADR, AFM: Not compression-aware

- HSM migrate/recall and backup/restore decompressed data and cause files to become illcompressed
- mmrestripefs or mmrestripefile command to recompress the recalled/restored files.

Thank you.

IBM
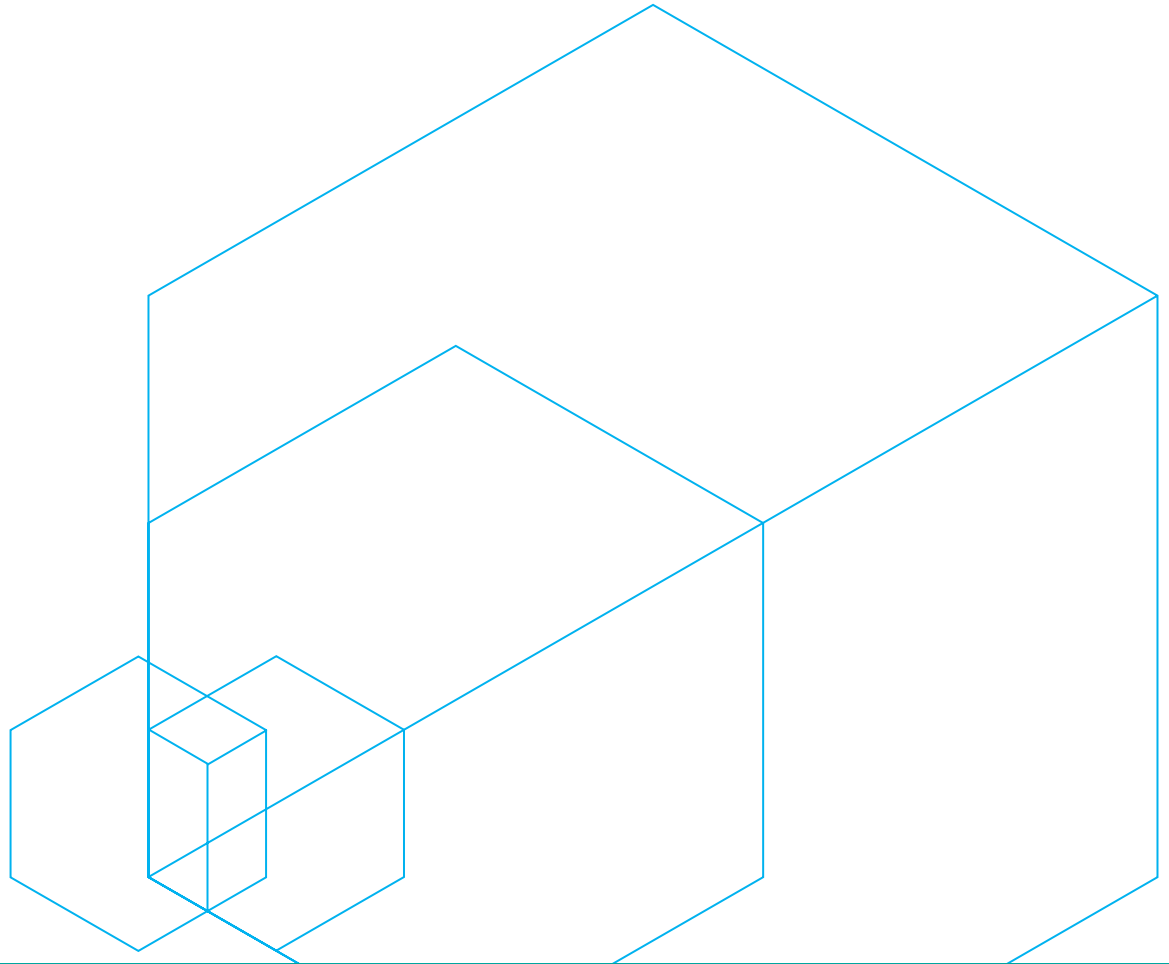
ibm.com/systems

Information and trademarks

Special notices

This document was developed for IBM offerings in the United States as of the date of publication. IBM may not make these offerings available in other countries, and the information is subject to change without notice. Consult your local IBM business contact for information on the IBM offerings available in your area.

Information in this document concerning non-IBM products was obtained from the suppliers of these products or other public sources. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. Send license inquires, in writing, to IBM Director of Licensing, IBM Corporation, New Castle Drive, Armonk, NY 10504-1785 USA.

All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

The information contained in this document has not been submitted to any formal IBM test and is provided "AS IS" with no warranties or guarantees either expressed or implied.

All examples cited or described in this document are presented as illustrations of the manner in which some IBM products can be used and the results that may be achieved. Actual environmental costs and performance characteristics will vary depending on individual client configurations and conditions.

IBM Global Financing offerings are provided through IBM Credit Corporation in the United States and other IBM subsidiaries and divisions worldwide to qualified commercial and government clients. Rates are based on a client's credit rating, financing terms, offering type, equipment type and options, and may vary by country. Other restrictions may apply. Rates and offerings are subject to change, extension or withdrawal without notice.

IBM is not responsible for printing errors in this document that result in pricing or information inaccuracies.

All prices shown are IBM's United States suggested list prices and are subject to change without notice; reseller prices may vary.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

Any performance data contained in this document was determined in a controlled environment. Actual results may vary significantly and are dependent on many factors including system hardware configuration and software design and configuration. Some measurements quoted in this document may have been made on development-level systems. There is no guarantee these measurements will be the same on generally-available systems. Some measurements quoted in this document may have been estimated through extrapolation. Users of this document should verify the applicable data for their specific environment.