z/OS 3.2 IBM Education Assistant

Solution Name: Enhance zFS Shrink Capabilities

Solution Element(s): zFS

July 2025

Agenda

- Trademarks
- Objectives
- Overview
- Usage & Invocation
- Interactions & Dependencies
- Upgrade & Coexistence Considerations
- Installation & Configuration
- Summary
- Appendix

Trademarks

- See url http://www.ibm.com/legal/copytrade.shtml for a list of trademarks.
- Additional Trademarks:
  - None

Objectives

- Describe new usability enhancements to zFS shrink support
- Describe use and benefits of new and modified options when running the zFS shrink command

Overview

- Who (Audience)
  - IBM zFS administrators
- What (Solution)
  - Usability enhancements to zFS shrink
- Wow (Benefit / Value, Need Addressed)
  - As a zFS Administrator, I wish to shrink my zFS file system(s), so that I can save disk space using a more user-friendly command which does not require me to perform any up-front mathematical calculations or manually mount or re-mount file systems.

Usage & Invocation (1)

- zFS Filesystems can be shrunk by invoking the zfsadm shrink command:
  - zfsadm shrink -aggregate name {-size Kbytes [-noai] | -release percent [-noai] | -cancel }
    [-remountro] -help -level -trace file_name
- Parameters
  - -aggregate name (modified)
    - The name of the aggregate to shrink. The name can include the wildcard character * at the beginning, the end, or at both the beginning and end. If the wildcard is used, each mounted aggregate whose name matches the name pattern will have a shrink command invoked against it.
  - -size Kbytes
    - The new size of the aggregate in 1K units. It is rounded up to an 8K boundary if necessary. This new size must be larger than the minimum size of an aggregate, and must be less than the current size of the aggregate minus 8K. If the amount of free space requested to be released is found to be larger than what is free in the aggregate, zFS will automatically increase the Kbytes value by at least 1 MB. This will result in less free space being released than was requested. This may occur due to file system activity or fragments converted to blocks during the shrink operation.
  - -noai (modified)
    - This indicates that zFS is not to automatically increase the Kbytes value specified on the –size option or calculated via the –release option if there is not enough free space to satisfy the request.

Usage & Invocation (2)

- -release percent (new)
  - The percentage of free space that is to be released during the shrink operation. The percent value can be from 1 – 100. zFS will internally generate a new size for the aggregate based on the current size of the aggregate and how many 8K blocks are currently free at the time the command is issued. If the amount of free space requested to be released is found to be larger than what is free in the aggregate, zFS will automatically increase the size value by at least 1 MB. This will result in less free space being released than was requested. This can occur if there is activity in the file system during the shrink operation, or if 8K blocks need to be allocated to store objects that were being kept in fragments.
- -cancel
  - Cancels an in-progress shrink operation if there is one occurring for the aggregate.

Usage & Invocation (3)

- -remountro (new)
  - The use of the -remountro option indicates to zFS that if the aggregate is mounted read-only, it can be mounted read/write in order for the shrink operation to complete. After the shrink operation completes, the aggregate is remounted read-only. In some cases the remount to read-only might fail (for example, the user session issuing the shrink command was canceled) and the aggregate might remain read/write. If you do not want it to remain read/write, you will need to manually remount the aggregate read-only. Be careful when using this option because the aggregate might be mounted on another system that is not part of your single system or shared file system environment. While the file system is mounted read/write, applications can change the file system.

Usage & Invocation (4)

- -level
  - Prints the level of the zfsadm shrink command. This option is useful when you are diagnosing a problem. Except for -help, all other valid options that are specified with -level are ignored.
- -help
  - Prints the online help for this command. All other valid options that are specified with this option are ignored
- -trace file_name
  - Specifies the name of the file that will have the trace records written into it. The trace file can be a z/OS UNIX file, an existing MVS sequential data set, or a member of either an existing partitioned data set (PDS) or partitioned data set extended (PDSE). Use this option only at the direction of IBM Support.

Interactions & Dependencies (1)

- Software Dependencies
  - None
- Hardware Dependencies
  - None

Interactions & Dependencies (2)

- Exploiters (API)
  - See the Shrink Aggregate section of the zFS application programming interface information chapter in the z/OS 3.2 File System Administration Guide if there is a need to shrink file systems via an API.
    - The structure SH_REQ now supports the following new sh_flag values:
      - SHR_RELEASE_PCT 0x04 – API caller is providing an integer release pct from 1-100 instead of a length in KBytes
      - SHR_ATTACH 0x08 - API caller is requesting an attach if the aggregate is not mounted
      - SHR_REMOUNT 0x10 – API caller is indicating the aggregate has been temporarily remounted read-write
    - The sh_length field must contain the percentage of free space to be released if SHR_RELEASE_PCT is specified via sh_flag. If SHR_RELEASE_PCT is not specified, the sh_length field must contain the new size of the aggregate in 1K units.
    - Wildcards will not be allowed for the aggregate name when using the API directly. Characters within the aggregate name will be validated by zFS.
    - If these new features of the SH_REQ structure are used with a ZFS kernel that does not have the new Shrink RFE support, then an error will occur.
    - When using the SHR_REMOUNT flag via the API the zFS kernel assumes the calling program is responsible for the remount to read/write prior to the shrink call and remount to read-only post shrink.

Upgrade & Coexistence Considerations

- To exploit this solution, all systems in the Plex must be at the new z/OS level: Yes
- List any toleration/coexistence APARs/PTFs.
  - None.
- List anything that doesn't work the same anymore.
  - File systems mounted as read-only are now eligible for shrink if new -remountro option is used
  - Single unmounted file systems are now eligible for shrink via an implicit attach
  - Multiple mounted file systems can be shrunk in one request if the leading and/or trailing wildcards are used for the aggregate name
  - A specific size in KBytes no longer needs to be specified on shrink if the -release option is used
- Upgrade involves only those actions required to make the new system behave as the old one did.
  - None.
- Coexistence applies to lower level systems which coexist (share resources) with latest z/OS systems.
  - None.

Installation & Configuration

- Enhanced Shrink Capabilities will be backfitted to R2.5 and R3.1 via APAR OA67431
- Shrink support is provided for both v4 and v5 filesystems
- For v4 filesystems, IBM recommends conversion to v5 which is enabled by default. To disable automatic conversion to v5 when performing shrink activities see the z/OS 3.2 File System Administration guide (for example, see the optional parameter -o 'NOCONVERTTOV5' on mount request)

Summary

- An overview of the new usability enhancements to zFS shrink support was provided
- The use and benefits of new / modified options when running the zFS shrink command was covered in detail
- The z/OS 3.2 File System Administration guide can be consulted for additional details on the new options supported by the zfsadm shrink command and how to invoke the new shrink features via an API

Appendix

- Publication references
  - z/OS 3.2 File System Administration
  - z/OS 3.2 File System Messages and Codes