z/OS 2.4 IBM Education Assistance (IEA)

Solution (Epic) Name: zFS High Availability Mount Option

Element(s)/Component(s): zFS

Agenda

- Trademarks
- Session Objectives
- Overview
- Usage & Invocation
- Interactions & Dependencies
- Migration & Coexistence Considerations
- Installation
- Session Summary
- Appendix

Trademarks

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

Session Objectives

- Describe benefits of High Availability (HA) mount option for zFS
- Describe what environment HA is useful for.
- Show how to enable and use HA
- Describe migration and coexistence considerations

Overview

- Users can indicate a zFS sysplex-aware R/W mounted file system should use the high-availability zFS sysplex protocols and interfaces which:
  - Guarantees that when a system experiences an outage in a sysplex, applications on other sysplex members do not receive errors.
    - Current z/OS Unix sysplex sharing (NORWSHARE) will present errors to applications that have open files when the owner goes down.
    - The current zFS sysplex sharing (RWSHARE) will also present errors to applications that have open files when the owner goes down.
    - And in both of the existing function cases, prior client operations might have been rolled back by the new owner causing loss of directory or file updates which confuse applications.
  - Guarantees that when a system experiences an outage in a sysplex, no prior directory or file update is lost.
  - No errors and no loss of data.
    - (Though resolving a dead system does take extra time to resolve orphaned calls to the dead owner).
- Some Target Users: Websphere Application Server (WAS) file systems and others.

Usage & Invocation - I

- New MOUNT parameter (HA | NOHA):
  - Explicitly indicates if high-availability should be used for a file system or not.
    - If not specified, the default comes from the IOEFSPRM options file.
- New IOEFSPRM parameter (HA = [ON | OFF])
  - The default high-availability setting for file systems mounts
    - Default is OFF, no high-availability. Not all file systems require HA support, example: automove-unmount.
    - Is used if the user does not explicitly state HA or NOHA for their mount parameter.
  - zfsadm config –ha [ON | OFF] can be used to change the default dynamically.
- Can dynamically enable or disable the HA function of a file system.
  - zfsadm chaggr [–ha | -noha] *file_system_name*
    - This results in a same-mode re-mount of the file system using the new desired option.

Usage & Invocation - II

- zfsadm fsinfo will show if a file system is HA:

```
ZFSAGGR.BIGZFS.ZFSTEST.V5ENC          DCEIMGHQ RW,RS,GD,EN,CO,HA
ZFSAGGR.BIGZFS.ZFSTEST1.EXTATTR       DCEIMGHQ RW,RS,GD,NE,NC,HA

Legend: RW=Read-write,GD=AGGRGROW disabled,RS=Mounted RWSHARE,EN=Encrypted
     NC=Not compressed,HA=High availability,NE=Not encrypted,CO=Compressed
```

- Can select a listing of file systems that are HA:
  - ***zfsadm fsinfo –select HA***
- FSINFO is an API available to application programs, updates for HA include:
  - ***fr_selection*** **–** *New input bit (0x20000000) – allows application to select only HA file systems when listing file systems.*
  - ***fo_flags*** **–** *New output bit (0x20000) – indicates corresponding file system is HA mounted.*

Interactions & Dependencies

- HW Dependencies: None
- SW Dependencies: None
- Performance Impact:
  - Results in log file syncs for each directory update made by sysplex non-owners
    - Owners do not need to sync the log file for each directory update, but they would still be affected by clients forcing the owner to sync the log file.
  - Results in log file sync for any asynchronous write-behind for files performed by non-owners.
    - Clients still cache directory and file contents.
    - Clients still perform asynchronous read-ahead and write-behind for files with direct disk access.
  - Workloads heavy in directory updates from non-owners are impacted, heavy file activity is only slightly impacted.
    - Heavy directory update workloads are not common.
- If a file system is being written by non-owners, then it should be automove eligible:
  - To ensure one of the client systems can assume ownership if original owner has an outage.

Migration & Coexistence Considerations

- 2.3 HA SPE being provided to allow z/OS 2.3 systems to support HA function.
- Systems need to be at the z/OS 2.4 level or 2.3 + HA SPE to use HA:
  - If there are HA mounted file systems in the sysplex, zFS will not allow down-level members without the HA support to join the sysplex.
    - → zFS will not allow a chaggr to enable HA if down level systems exist in the sysplex
    - → An attempt to mount a file system as HA, when there are down level members will result in a non-HA file system.
- z/OS Unix Pre-req APAR: OA57157 – required for zfsadm chaggr.

Migration Path

1. Ensure that all sysplex members have the required service levels and that its unlikely you will need to go back to the prior levels.
2. (Optional) If its desired all RW file systems to be HA:
   - Set IOEFSPRM option HA=ON
   - Issue zfsadm config –ha on for each sysplex member
3. To enable HA for existing mounted file systems desired to be HA:
   - zfsadm chaggr –ha for each RW mounted file system.

- → If you need to put on prior service or release levels due to unforeseen circumstances, reverse the process:
  - Set IOEFSPRM HA=OFF, zfsadm config –ha off for each member, if you enabled it by default.
  - zfsadm fsinfo –select HA will list HA file systems.
  - zfsadm chaggr –noha for each file system that has the HA option.

Installation

- No changes to installation.

Session Summary

- Described HA and its benefits
- Described who might benefit from HA
- Provided usage considerations, including performance
- Presented how to enable or disable the function
- Discussed migration and coexistence considerations

Appendix

- z/OS File System Administration – Updated for HA function, including a section dedicated to using HA.
- z/OS File System Messages and Codes – Updated for HA function.