

Deeper dive into File Level Backup

Andrew N. Wilt

DFSMSHsm Development

IBM Tucson

anwilt@us.ibm.com

November 2019

Session DK

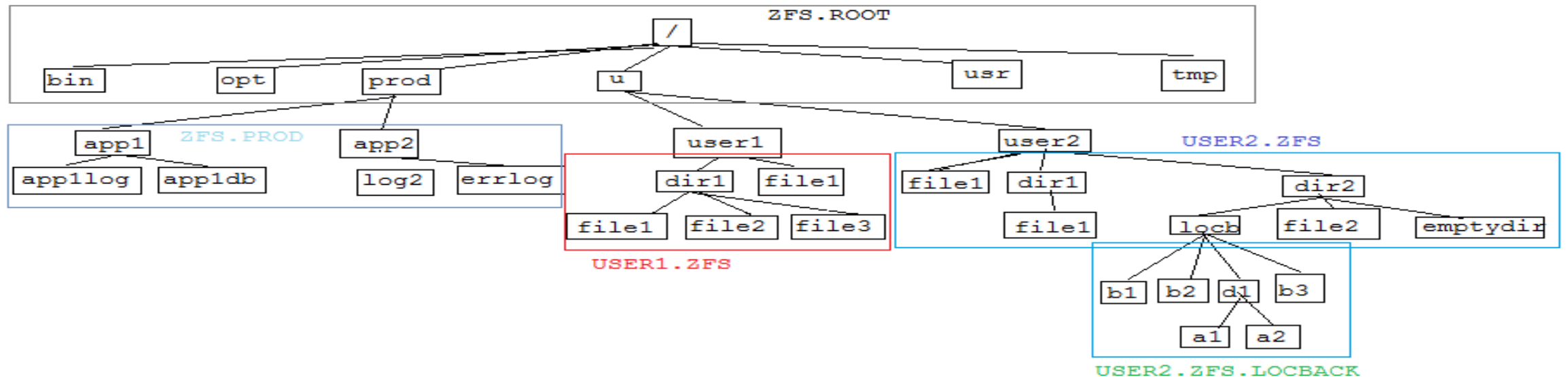


Purpose

- With the Statement of Direction that the Spectrum Protect z/OS UNIX System Services Backup Client being withdrawn from support, DFSMS is enhanced to provide z/OS V2R3 support for Individual UNIX file backup.

What is ZFS and a UNIX file?

- z/OS UNIX data organized in terms of directories, and files.
- Directories and files are contained in Filesystems.
- Filesystems are contained in zFS data sets. (or NFS and others)
- zFS data sets are mounted to empty directories



Previous environment

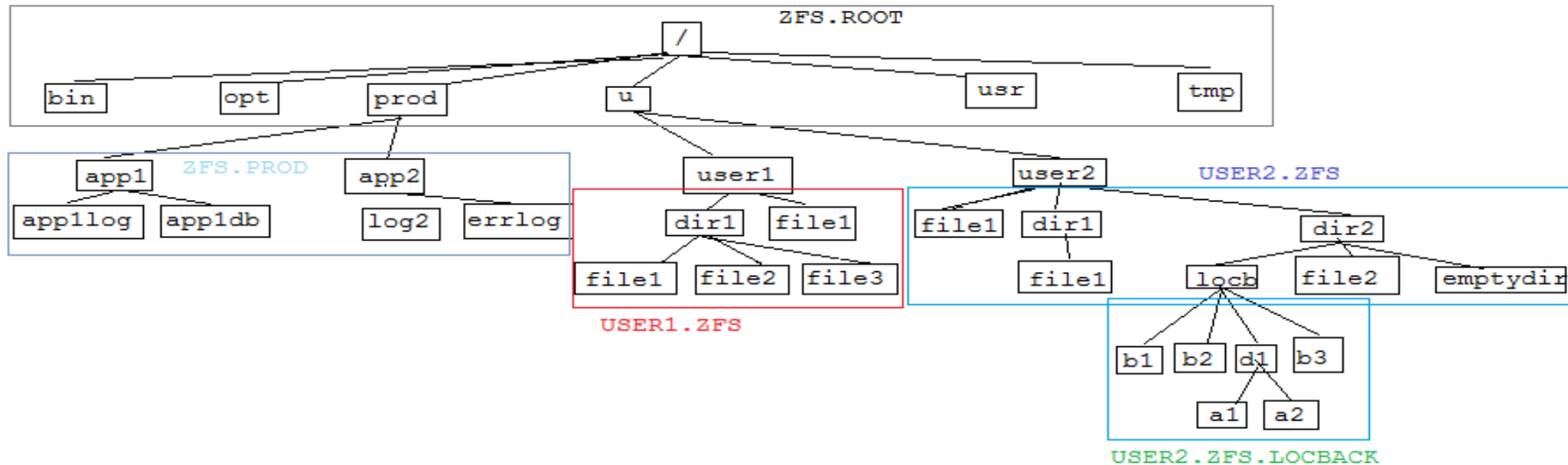
- DFSMSdss and DFSMSHsm only processed zFS data sets.
- Problem – Need to recover only one file? Just recover the zFS data set and mount it to a temporary directory, and copy out your file.

DFSMS zFS File Level Backup

- **DFSMSHsm**

DFSMShsm File Level Backup – BACKDS / BACKFILE

- **Storage Admin command:**
- BACKDS '/u/user1/file1' TARGET(TAPE)
- BACKDS '/u/user2/*' RECURSE(NOCROSSMOUNTS) CC(PREFERRED) TARGET(DASD) TOTAL
 - /u/user2/file1
 - /u/user2/dir1/file1
 - /u/user2/dir2/file2
 - /u/user2/dir2/emptydir



DFSMShsm File Level Backup – BACKDS / BACKFILE

- **TARGET (DASD | TAPE)** – Override default target DASD or Tape
- **RECURSE (NOCROSSMOUNTS | CROSSMOUNTS)** – Traverse Directory structure. Leave Filesystem?
- **CHANGEDONLY** – Modification date > Backup date – Default
- **TOTAL** – Create backup regardless of dates
- **RETAIN DAYS (nn)** – How long backup should be kept past number of versions (same for Data Sets)
- **CC (STANDARD | PREFERRED | REQUIRED)** – ZFS File Snapshot processing
- **NEWNAME (newdir) DATE (yyyy/mm/dd) TIME (hhmmss)** – Backup files in request as if they came from newdir

DFSMS File Level Backup – DFSMShsm Recover

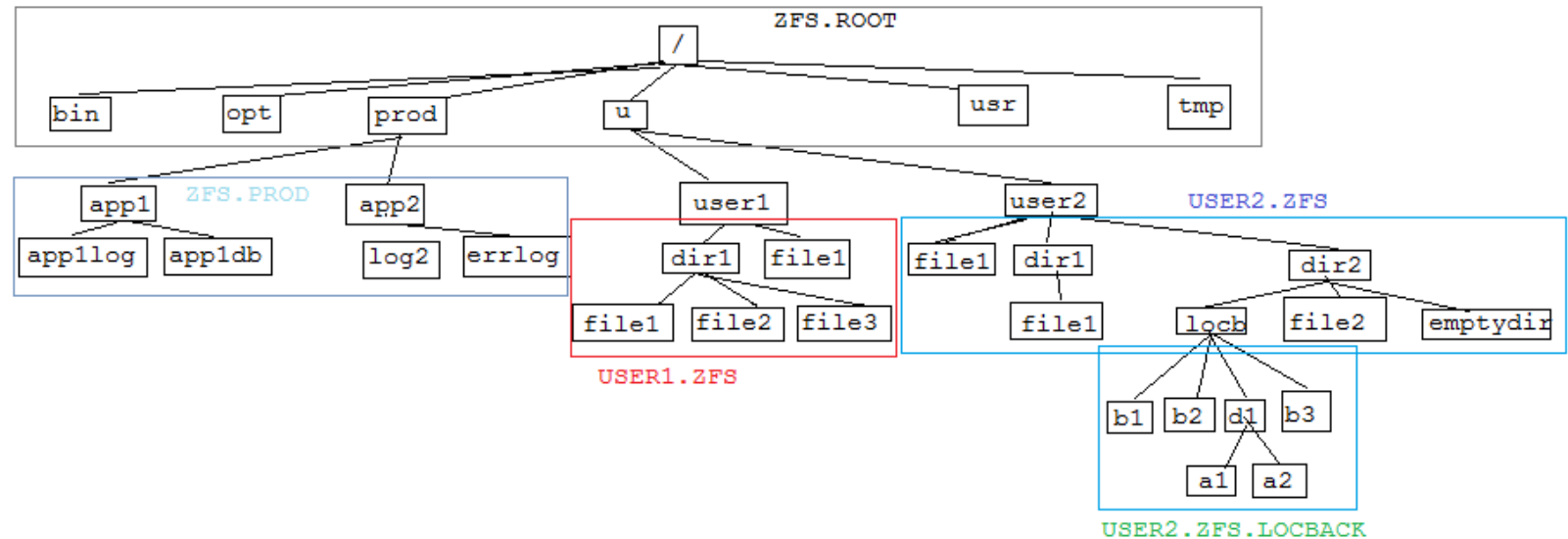
- **Storage Administrator command:**

```
RECOVER 'filename' DATE (yyyy/mm/dd) TIME (hhmmss) GENERATION (gennum)
VERSION (vernum) REPLACE RECURSE (NOCROSSMOUNTS | CROSSMOUNTS)
```

- RECOVER '/u/user1/file1' REPLACE

- RECOVER '/u/user2/dir2/*' REPLACE RECURSE(CROSSMOUNTS)

```
/u/user2/dir2/locb/b1
/u/user2/dir2/locb/b2
/u/user2/dir2/locb/d1/a1
/u/user2/dir2/locb/d1/a2
/u/user2/dir2/locb/b3
/u/user2/dir2/file2
/u/user2/dir2/emptydir/
```



DFSMShsm File Level Backup - LIST

- Storage Administrator:

```
LIST FILENAME ('/u/ibmuser/cicsdata/daily.runlog.txt') BCDS TERM
```

- Terminal Output:

```
FILE=/u/ibmuser/cicsdata/daily.runlog.txt
```

```
BACK FREQ = ***, MAX ACTIVE BACKUP VERSIONS = ***  
FROMFS=ZFS.UDIR
```

```
BDSN=DFHSM.BACK.TRMRQ04.CICSDATA.DAILY.A4169 BACKVOL=BACK01  
BACKDATE=14/06/18 BACKTIME=14:14:58 GEN=000 VER=003 UNS/RET= NO  
EXTENDED ACL=YES RETDAYS=***** TYPE=FILE
```

```
BDSN=DFHSM.BACK.TKRSI14.CICSDATA.DAILY.A4169 BACKVOL=BACK01  
BACKDATE=14/06/18 BACKTIME=14:13:52 GEN=001 VER=002 UNS/RET= NO  
EXTENDED ACL=YES RETDAYS=***** TYPE=FILE
```

DFSMShsm File Level Backup - LIST

- LIST FILELEVEL('/u/user1/') BCDS OUTDATASET(ADMIN.FILELIST.OUTPUT)
- **TSO Command:**
- HLIST FILELEVEL('/u/ibmuser/cicsdata/') BCDS ODS(ADMIN.CICSDATA.OUTPUT)
- **UNIX shell command:**
- hlist -X /u/ibmuser/cicsdata/

DFSMShsm File Level Backup

- **DFSMShsm Functions Updated:**
- AUDIT FILES() BCDS
- BDELETE (HBDELETE, hbdelete)
- FIXCDS
- QUERY
- CANCEL
- ALTERPRI
- ALTERDS

DFSMSHsm File Level Backup - Security

- Storage Administrator can:
 - Create backups of all files (BACKDS)
 - Recover any file (RECOVER)
- Anyone with Search permission to a file can:
 - Create a backup (HBACKDS, hbackup) if they have read access, or is owner
 - Recover a file (HRECOV, hrecover) if they have write permission to file and parent directory, or is owner
- File Access Control Lists (FACL) are checked, and saved in backup.

DFSMShsm Recursion and wildcards

- Directory processing and wildcards
- BACKDS/HBACKDS '/u/dir1/' RECURSE(NCM)
 - Backup all files in /u/dir1 and process sub-directories, but don't cross into a different filesystem
- BACKDS/HBACKDS '/u/dir1/ab*.txt' RECURSE(NCM)
 - Backup files in /u/dir1 that match the pattern, ab*.txt. Process sub-directories for files that match the pattern, but don't cross into a different filesystem.
- hbackup "/u/dir1/ab*.txt" vs. hbackup /u/dir1/ab*.txt
 - The UNIX shell will expand wildcards that match names if "" is not used.

zFS File Snapshot - Caveat

- DFSMShsm ConcurrentCopy (or DFSMSdss CLONE) UNIX file processing must be performed on Owing System for zFS mounted with NORSHARE
- If the zFS is mounted with RWSHARE, then the processing can be done on any system.

Storage Administrator – DFSMShsm Tasks

- Backup Target
 - Command Backups (All UNIX file backups) target ML1 volumes (if not Direct to Tape) – Moved to Daily Backup volumes during Automatic Backup processing
- Tasking
 - SETSYS DSBACKUP(DASD(TASKS(14) TAPE(TASKS(4)))
- SETSYS BACKUP VERSIONS(12) FREQUENCY(0)
 - VERSIONS used for Files in non-SMS ZFS data sets
 - FREQUENCY ignored
- Backup Copy Data Set Name
 - *prefix.BACK.Tcccchh.user1.user2.Xydd*

Storage Administrator – DFSMShsm tasks

- Bigger BCDS needed

Each file = MCB + n*UFN

```
ALTER DFHSM.BCDS NEWNAME (DFHSM.BCDS.OLD)
  IF MAXCC = 0 THEN DO
    ALTER DFHSM.BCDS.DATA NEWNAME (DFHSM.BCDS.OLD.DATA)
    ALTER DFHSM.BCDS.INDEX NEWNAME (DFHSM.BCDS.OLD.INDEX)
  END

REPRO INDATASET (DFHSM.BCDS.OLD) OUTDATASET (DFHSM.BCDS)
```


DFSMShsm – Facility Class checking

- STGADMIN.ARC.ENDUSER.HBACKDS.RCRS.CM
- STGADMIN.ARC.ENDUSER.HBACKDS.RCRS.NCM
- STGADMIN.ARC.ENDUSER.HRECOVER.RCRS.CM
- STGADMIN.ARC.ENDUSER.HRECOVER.RCRS.NCM
- STGADMIN.ARC.BACKDS.DELETE
- STGADMIN.ARC.ENDUSER.HBACKDS.DELETE

Function Statistics Records

- SETSYS SMF(254)
 - Type 255 is FSR
- '40'x bit at offset 12A is UNIX file
- VOLSER is always %UNIX%
- Shortened UNIX filename in FSRDSN field.
 - `/test/longdirtest2/1...test9/arclfile.plx170`
- Full Filename at end of Record, after Tape volume entries.
- How many UNIX file backups have been run this week?

Reporting

- DCOLLECT
 - UNIX file backup records skipped during DCOLLECT processing

- REPORT

```
REPORT VOLUMES (%UNIX%) FUNCTION (BACKUP)
FROMDATE (2019/01/01) SUMMARY ODS (OUT.DS)
```

NUMBER	-----READ-----	-----WRITTEN-----	-----REQUESTS-----	AVERAGE	-----AVERAGE	TIME-----								
HSM	FUNCTION	DATASETS	TRK/BLK	K-BYTES	TRK/BLK	K-BYTES	SYSTEM	USER	FAILED	AGE	QUEUED	WAIT	PROCESS	TOTAL
BACKUP														
DAILY	BACKUP	0003715	00008072	000334611	00000000	000000000	000000	03906	00191	00000	0191	00028	00018	00237
DELETE	BACKUPS	0001224	00000000	000000000	00000000	000000000	001224	00000	00000	00129	0000	00000	00000	00000

Backup Expiration

- Management Class – Number of backup versions (Data Set Exists)
- SETSYS VERSIONS(nn) – files from non-SMS zFS
- At backup time – roll off extra versions
- EXPIREBV command
 - See if file deleted. If so, mark date found to be deleted
 - Expire versions that met RETAINDDAYS(nn)
 - Expire extra versions - Number of backup versions (Data Set Deleted) and NONSMSVERSIONS(CATALOGEDDATA(days)) – for deleted files

DFSMS zFS File Level Backup

- **Initial Limitations:**

- **File Types Supported:**

- Regular Files (sparse files when using ZFS File Snapshot only)
- Directory Files
- FIFO Files
- Symbolic Links
 - Reference is backed up, DSS will not attempt to resolve the reference
 - Will not follow symbolic links

- **File Types Not Supported:**

- Socket Files
- Special Character Files
- CLOUD as target not supported.

Spectrum Protect Users

- Look in dsmsched.log for directories and files that are scheduled
- `dsmc query backup /. * -subdir=yes`

```
dsmc backup /dir1/*
hbackup -o "/dir1/*"
```

```
dsmc archive -deletefiles /prod/SAP/logs/*
BACKDS `/prod/SAP/logs/*' DELETE TOTAL
```

- **Monthly Archives:**

```
BACKDS `/prod/' RETAIN_DAYS(365) RECURSE(NCM)
```

What to back up?

- **SYS1.PARMLIB(BPXPRMxx)**

```

MOUNT FILESYSTEM('MVSZFS.ETC.ZFS')
        TYPE(ZFS)
        MODE(RDWR)
        MOUNTPOINT('/etc')
  
```

- **AUTOMOUNT**

/etc/u.map

Example:

```

name          *
type          ZFS
filesystem    AUTOMNT.<uc_name>.ZFS
mode          rdwr
duration      nolimit
delay         10
  
```

Spectrum Protect Backup Import

- `dsmc restore /prod/webapp1/ /tmp/2017-07-08/ -
pitdate=7/8/2017 -subdir=yes -
preservepath=Complete`
 - `BACKDS '/tmp/2017-07-08/prod/webapp1/*' RECURSE
NEWNAME ('/prod/webapp1/') DATE(2017/07/08)`
- or
- `hbackup -oX -D 2017/07/08 -N / /tmp/2017-07-08/`

Scheduling via Batch TSO

- Define PDS IBMUSER.BACKDS.LONGNAME with:

```

Organization . . . : PO
Record format . . . : VB
Record length . . . : 1048
Block size . . . . : 1052
  
```

- Member BACKDSLID:

```

00010048WTO ***** STEP04 *****
00020057WTO HW BACKDS '/test/longdirtest/longdirectorytest..' RECURSE
00030061HSEND BACKDS-
00040068 '/test/longdirtest/longdirectorytest1/longdirectorytest2/ ...
  
```

- JCL:

```

//STEPTSO EXEC PGM=IKJEFT01 EXECUTE TSO COMMAND(S)
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD DSN=IBMUSER.BACKDS.LONGNAME (BACKDSLID) , DISP=SHR
  
```

Scheduling via Batch UNIX (BPXBATC

- **JCL:**

```
//STEP1      EXEC  PGM=BPXBATCH,REGION=8M
//SYSPRINT  DD    SYSOUT=*
//STDOUT    DD    SYSOUT=*
//STDERR    DD    SYSOUT=*
//STDENV    DD    DUMMY
//STDPARM   DD    *
SH /bin/sh -c '
```

```
set -x;
cd /prod/SAP/apps/ ;
hbackup -Xo "./*" ;
hlist -X /prod/SAP/apps/ ;
'
/*
```

Useful Commands: zfsadm fsinfo -path

```

ezu129.tuc.stglabs.ibm.com - PuTTY
# zfsadm fsinfo -path ./
File System Name: ZFS.UDIR

*** owner information ***
Owner:                SYSTEM1          Converttov5:          OFF,n/a
Size:                 362880K           Free 8K Blocks:      35370
Free 1K Fragments:    7              Log File Size:        3632K
Bitmap Size:          56K              Anode Table Size:     288K
File System Objects:  1103             Version:              1.5
Overflow Pages:       0              Overflow HighWater:   0
Thrashing Objects:    0              Thrashing Resolution: 0
Token Revocations:    0              Revocation Wait Time: 0.000
Devno:                21              Space Monitoring:     0,0
Quiescing System:     n/a              Quiescing Job Name:   n/a
Quiescor ASID:        n/a              File System Grow:     ON,0
Status:               RW,NS,NE,NC
Audit Fid:            C4F9E2F7 E2F0FFF0 0000
Backups:              0              Backup File Space:    0K

File System Creation Time: Nov 30 06:35:41 2017
Time of Ownership:       Aug  7 04:08:52 2019
Statistics Reset Time:   Aug  7 04:08:52 2019
Quiesce Time:           n/a
Last Grow Time:         n/a

Connected Clients:      n/a

Legend: RW=Read-write, NS=Mounted NORWSHARE, NE=Not encrypted
        NC=Not compressed
# █
  
```

Useful Commands: zfsadm fileinfo

```

ezu129.tuc.stglabs.ibm.com - PuTTY
n=EF176324.
# zfsadm fileinfo output.txt
  path: /test/output.txt
  ***  global data  ***
  fid          116,1          anode          41799,5304
  length       1283          format         BLOCKED
  1K blocks    8             permissions    644
  uid,gid      0,500         access acl     0,0
  dir model acl na           file model acl na
  user audit   F,F,F        auditor audit  N,N,N
  set sticky,uid,gid 0,0,0    seclabel      none
  object type  FILE          object linkcount 1
  object genvalue 0          dir version   na
  dir name count na         dir data version na
  dir tree status na        dir conversion na
  file format bits 0x0,0,0    file charset id 0x0
  file cver      none        charspec major,minor na
  direct blocks 0x000054F7
  indirect blocks none
  mtime         Jan 16 05:34:07 2018    atime          Sep 10 07:01:45 2018
  ctime         Aug 7 04:31:41 2019      create time    Jan 16 05:20:37 2018
  reftime       Aug 7 04:31:41 2019
  not encrypted
  not compressed
# █

```

DFSMShsm zFS File Level Backup

- Consideration:
 - Initial backup of a file can take longer due to CDS I/O overhead
 - Backing up a large number of files will considerably grow the BCDS.
- Tasking level:
 - Tasking level controlled by current backup max tasks.
SETSYS DSBACKUP(DASD(TASKS(12) TAPE(TASKS(8)))

DFSMShsm – Follow-On Enhancements

1. Storage Administrator Mode for UNIX programs – OA57454
2. DELETE parameter for Backup – OA57454
3. EXCLUDE parameter for Backup and Recover – OA57868
4. Separate HSMPlex for UNIX files
5. RECOVER to NEWDIR - OA58612
6. Automatic Backup (Identify new directories/Mounted Filesystems)
7. Management Class per Directory
8. RENAME on RECOVER

DFSMS zFS File Level Backup

- **DFSMSdss**

DFSMScss zFS File Level Backup

- “As a Storage Administrator, I want to be able to write batch JCL to use the DFSMScss DUMP and RESTORE command to request a backup of a zOS UNIX file using absolute path names”

DUMP -

```
PATH(INCL('dir1/file1' 'file1')) WORKINGDIRECTORY('/u/user1/') -  
OUTPUTDD(DUMPDD) CLONE(PREFERRED) TOLERATE(WRITERS)-  
ADMINISTRATOR
```

RESTORE -

```
PATH(INCL('dir1/file1' 'file1')) WORKINGDIRECTORY('/u/user2/') -  
INPUTDD(DUMPDD) ADMINISTRATOR
```


DFSMdss zFS File Level Backup

DUMP PATH(INCL('dir1/file1' 'file1'))

WORKINGDIRECTORY('/u/user1/')

**OUTDD(DUMPDD) CLONE(PREFERRED) TOLERATE(WRITERS)
ADMINISTRATOR**

- **PATH(INCLUDE(...))** - Files specified will be backed up
 - Includes backing up file attributes for directories along the path
 - Needed to recreate directories upon Restore
- **WORKINGDIRECTORY** tells DSS where it will begin processing
- **INCLUDE** paths will be concatenated to the **WORKINGDIRECTORY**
 - Working Directory path attributes will **not** be backed up

DFSMSdss zFS File Level Backup

- ADMINISTRATOR indicates user is a DFSMSdss-authorized storage administrator if the user has READ access to its RACF facility profile.
 - File security checks will succeed regardless of the file's permissions/ACL
- TOLERATE(WRITERS) indicates it is OK for DSS to backup the file even though it may be open for write intent by other applications
- OUTDD describes the output data set that will hold the backup contents
 - TAPE or DASD
- RESET – Set the last backup date to the current date/time

DFSMS zFS File Level Backup

- **CLONE (NONE, PREFERRED, REQUIRED)**
 - Indicates backups will occur from a ZFS File Snapshot of the regular file
 - DSS will release the base file and Read from the ZFS File Snapshot for the Backup
 - Enables applications to quickly have access to their files
- **NOTIFYCLONE** – Notify the result of ZFS File Snapshot processing attempt

DFSMS zFS File Level Backup

```
DUMP PATH(INCL('dir1/file1' 'file1'))  

WORKINGDIRECTORY('/u/user1/')  

OUTDD(DUMPDD) CLONE(PREFERRED) TOLERATE(WRITERS)  

ADMINISTRATOR
```

```
ADR650I (001)-UDFLT(01) ALL PATHS ARE RELATIVE TO WORKING DIRECTORY
```

```
    /u/user1/
```

```
ADR651I (001)-DTUNX(01) PATH FILTERING IS COMPLETE. 3 OF 3 FILES WERE SELECTED
```

```
ADR454I (001)-UPRTT(01) THE FOLLOWING FILES WERE SUCCESSFULLY PROCESSED
```

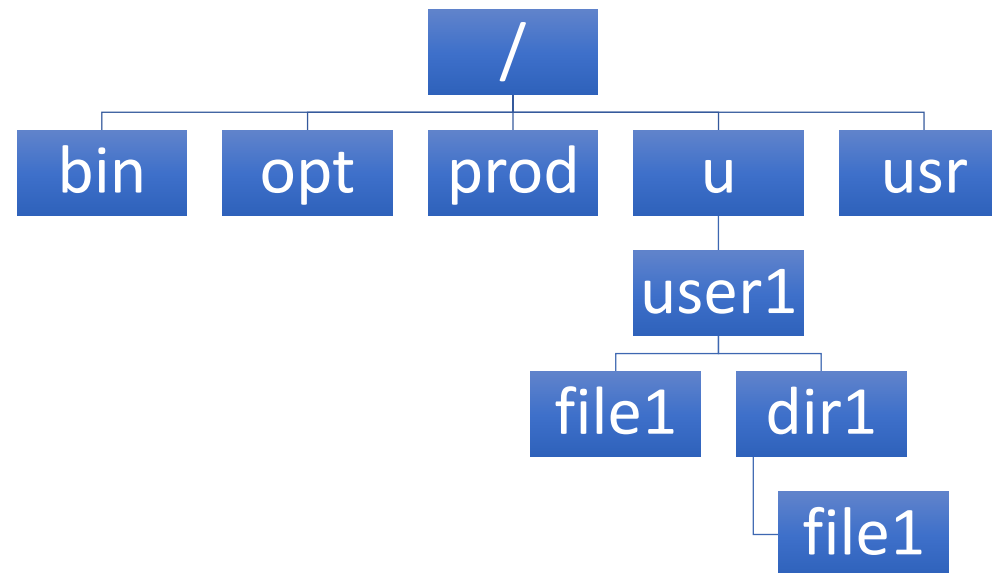
```
  - ./file1
```

```
  d ./dir1
```

```
  - ./dir1/file1
```

DFSMS zFS File Level Backup – Part 1

DUMP PATH(INCL('dir1/file1' 'file1'))
WORKINGDIRECTORY('/u/user1/')
OUTDD(DUMPDD) CLONE(PREFERRED) TOLERATE(WRITERS)
ADMINISTRATOR



DFSMS zFS File Level Backup

**RESTORE PATH(INCL('dir1/file1' 'file1'))
 WORKINGDIRECTORY('/u/user2/')
 INDD(*DUMPDD*) ADMINISTRATOR REPLACEUNCONDITIONAL**

- **PATH(INCLUDE(...))** - Files to be restored
 - Files restored to the specified working Directory
- **WORKINGDIRECTORY** tells DSS path to Restore to
 - Same as Dump Source – Files created with dump attributes
 - Different - Files created as 'new' files
 - ADMINISTRATOR - Always creates files with dump attributes
- **ADMINISTRATOR** indicates user is a DFSMSdss-authorized storage administrator if the user has READ access to its RACF facility profile.
 - File security checks will succeed regardless of the file's permissions/ACL
 - Both Source (Dump contents) and Target (Pre-existing files)
- **REPLACEU** – overwrite regular files only

DFSMS zFS File Level Backup

```
RESTORE PATH(INCL('dir1/file1' 'file1'))  

WORKINGDIRECTORY('/u/user2/')  

INDD(DUMPDD) ADMINISTRATOR REPLACEUNCONDITIONAL
```

```
ADR650I (001)-URFLT(01) ALL PATHS ARE RELATIVE TO WORKING DIRECTORY  

      /u/user2/
```

```
ADR454I (001)-UPRTT(01) THE FOLLOWING FILES WERE SUCCESSFULLY PROCESSED  

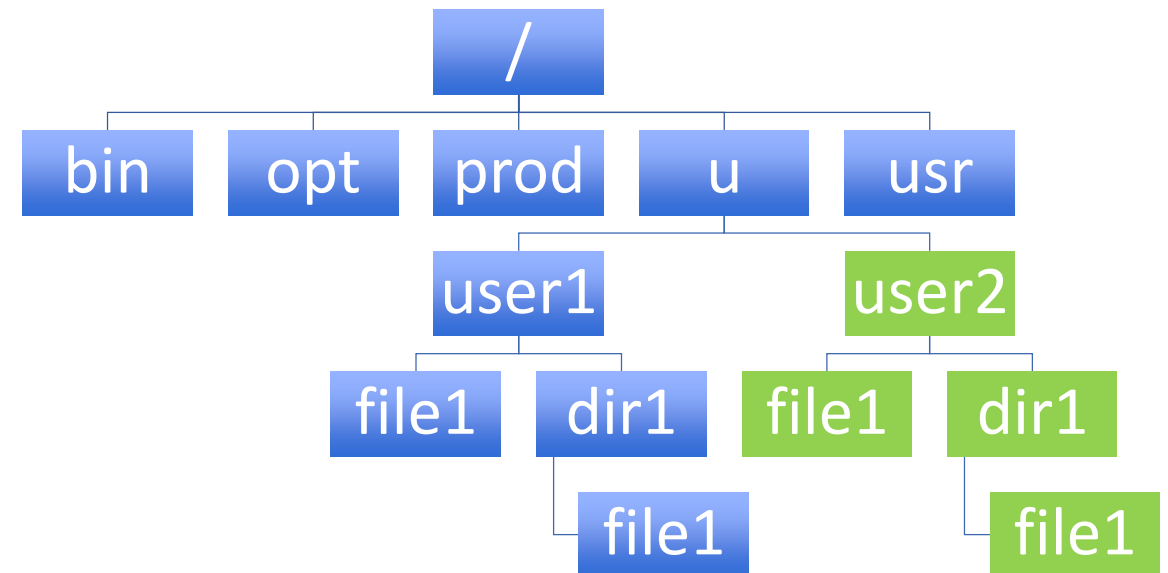
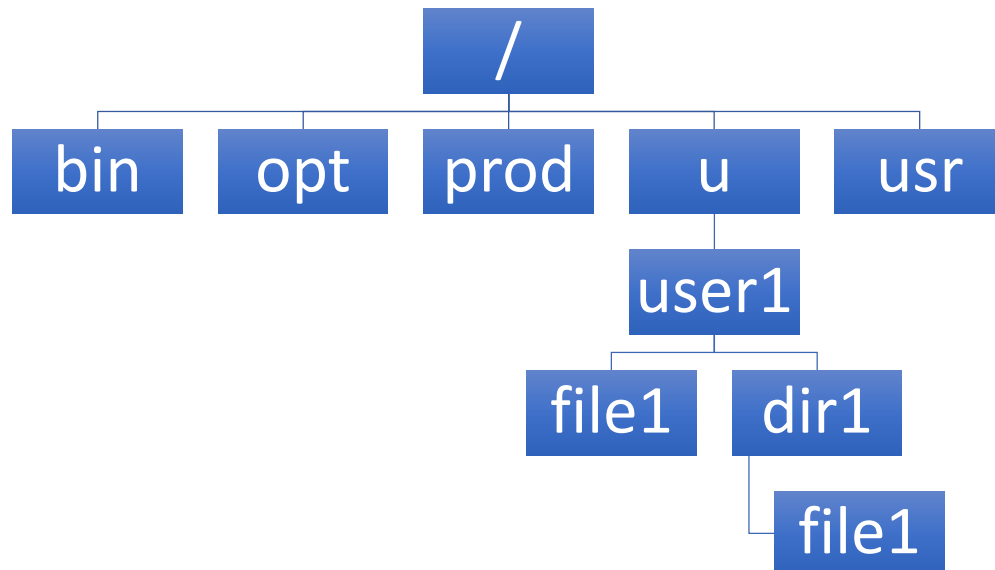
      - ./file1  

      d ./dir1  

      - ./dir1/file1
```

DFSMS zFS File Level Backup

**RESTORE PATH(INCL('dir1/file1' 'file1'))
WORKINGDIRECTORY('/u/user2')
INDD(DUMPDD) ADMINISTRATOR**



DFSMSSDSS Serialization

FUNCTION	Request READ intent	Request WRITE intent	Deny Shared Readers	Deny shared Writers
DUMP	YES	NO	YES*	YES*
RESTORE	NO	YES	YES	YES

DFSMSSdss File Permissions after Restore

Attribute	Attribute Determined by		
	ADMINISTRATOR keyword specified	No ADMINISTRATOR keyword	
		Same Location	Different Location
Permissions (<code>Attrmode</code>)	Source value	Source value	DSS defaults
Owning UID (<code>AttrUid</code>)	Source value	Source value	Effective UID
Owning GID (<code>AttrGid</code>)	Source value	Source value	Effective GID
Sticky bit (<code>AttrNoDelFiles</code>)	Source value	Source value	Not set
Shared library (<code>AttrShareLibMask</code>)	Source value	Not set	Not set
APF authorized program (<code>AttrApfAuthMask</code>)	Source value	Not set	Not set
Program controlled (<code>AttrProgCtlMask</code>)	Source value	Not set	Not set

DFSMSdss File Permissions after Restore (cont)

Attribute	Attribute Determined by		
	ADMINISTRATOR keyword specified	No ADMINISTRATOR keyword	
		Same Location	Different Location
Auditor audit (AttrAuditorAudit)	Source value	Source value	Not set
Auditor user (AttrUserAudit)	Source value	Source value	Not set
Last access time (AttrAtime64)	Source value	Source value	Current time
Last modification time (AttrMtime64)	Source value*	Source value*	Current time
Last file stat change time (AttrCtime64)	Source value	Source value	Current time
Last reference time (ATTRREFTIME64)	Source value	Source value	Not set
Security label (ATTRSECLABEL)	Source value	Source value	System default

DFSMSSdss Dump Contents

- How to list what files are in a Dump Data Set

```
//FILELIST EXEC PGM=ADRDSSU, PARM='TYPRUN=NORUN'
//SYSPRINT DD SYSOUT=*
//DUMPIN DD DISP=SHR, DSN=DUMP.FILES.UNKNOWN
//SYSIN DD *
  RESTORE PATH(INCL('*')) WORKINGDIRECTORY('/')
  INDD(DUMPIN)
/*
```

DFSMS zFS File Level Backup

Thank you

- OA52703 – Initial DFSMSHsm support
- OA52836 – DFSMSdss support
- OA54218 – UNIX System Services support
- OA56145 – ZFS support
- OA55165 – RACF support

- FIXCAT: UNIXFILEBACKUP/K

DFSMS Publication updates:

- <http://publibz.boulder.ibm.com/zoslib/pdf/OA52703.pdf>

Additional Info

- DFSMShsm
- AUTOMOUNT directories are supported
- Migrated Automount ZFS will be recalled and mounted when referenced
- R/O Root and filesystems supported

```
# hbackup include/  
ARC1000I /usr/include/rapi_int.h BACKDS PROCESSING ENDED  
ARC1000I /usr/include/irrspim.h BACKDS PROCESSING ENDED  
ARC1000I /usr/include/slapi-plugin.h BACKDS PROCESSING ENDED  
ARC1000I /usr/include/ldapssl.h BACKDS PROCESSING ENDED  
ARC1000I /usr/include/ldap.h BACKDS PROCESSING ENDED  
ARC1000I /usr/include/snmpntfy.h BACKDS PROCESSING ENDED  
ARC1000I /usr/include/snmpmgr.h BACKDS PROCESSING ENDED  
ARC1000I /usr/include/rapi.h BACKDS PROCESSING ENDED  
ARC1000I /usr/include/lber.h BACKDS PROCESSING ENDED  
ARC1000I /usr/include/xutility.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/xtree.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/xstring.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/xloctime.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/xlocmes.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/xlocinfo.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/vector.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/valarray.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/string.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/sstream.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/ostream.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/memory.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/list.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/istream.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/fstream.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/deque.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/complex.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/bitset.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/algorithm.t BACKDS PROCESSING ENDED  
ARC1000I /usr/include/massv.h BACKDS PROCESSING ENDED  
ARC1000I /usr/include/mass_simd.h BACKDS PROCESSING ENDED  
ARC1000I /usr/include/mass.h BACKDS PROCESSING ENDED  
ARC1000I /usr/include/omp.h BACKDS PROCESSING ENDED  
ARC1000I /usr/include/unexpect.h BACKDS PROCESSING ENDED  
ARC1000I /usr/include/typeinfo.h BACKDS PROCESSING ENDED  
ARC1000I /usr/include/terminat.h BACKDS PROCESSING ENDED  
ARC1000I /usr/include/new.h BACKDS PROCESSING ENDED
```

Please submit your session feedback!

- Do it online at <http://conferences.gse.org.uk/2019/feedback/DK>
- This session is DK

1. What is your conference registration number?

🔦 This is the three digit number on the bottom of your delegate badge

2. Was the length of this presentation correct?

🔦 1 to 4 = "Too Short" 5 = "OK" 6-9 = "Too Long"

1 2 3 4 5 6 7 8 9

3. Did this presentation meet your requirements?

🔦 1 to 4 = "No" 5 = "OK" 6-9 = "Yes"

1 2 3 4 5 6 7 8 9

4. Was the session content what you expected?

🔦 1 to 4 = "No" 5 = "OK" 6-9 = "Yes"

1 2 3 4 5 6 7 8 9