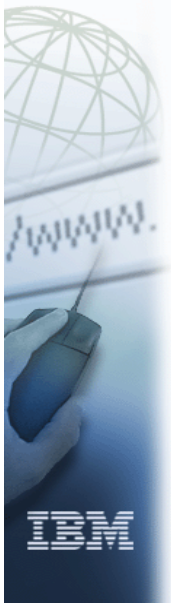


ibm.com



e-business

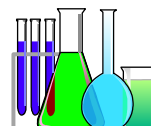


zSeries File System (zFS)



Redbooks

International Technical Support Organization



© Copyright IBM Corp. 2004. All rights reserved.

Trademarks



eNetwork	DFSMS/MVS	IMS	RACF
geoManager	DFSMSdtp	IMS/ESA	RMF
AD/Cycle	DFSMSdss	IP PrintWay	RS/6000
ADSTAR	DFSMShsm	IPDS	S/390
AFP	DFSMSrmm	Language Environment	S/390 Parallel Enterprise Server
APL2	DFSORT	Multiprise	SecureWay
APPN	Enterprise System 3090	MQSeries	StorWatch
BookManger	Enterprise System 4381	MVS/ESA	Sysplex Timer
BookMaster	Enterprise System 9000	Network Station	System/390
C/370	ES/3090	NetSpool	SystemView
CallPath	ES/4381	OfficeVision/MVS	SOM
CICS	ES/9000	Open Class	SOMobjects
CICS/ESA	ESA/390	OpenEdition	SP
CICS/MVS	ESCON	OS/2	VisualAge
CICSplex	First Failure Support Technology	OS/390	VisualGen
COBOL/370	FlowMark	Parallel Sysplex	VisualLift
DataPropagator	FFST	Print Services Facility	VTAM
DisplayWrite	GDDM	PrintWay	WebSphere
DB2	ImagePlus	ProductPac	3090
DB2 Universal Database	Intelligent Miner	PR/SM	3890/XP
DFSMS/MVS	IBM	QMFr	z/OS
			z/OS.e

Domino (Lotus Development Corporation)
DFS (Transarc Corporation)
Java (Sun Microsystems, Inc.)
Lotus (Lotus Development Corporation)

Tivoli (Tivoli Systems Inc.)
Tivoli Management Framework
(Tivoli Systems Inc.)
Tivoli Manager (Tivoli Systems Inc.)

UNIX (X/Open Company Limited)
Windows (Microsoft Corporation)
Windows NT (Microsoft Corporation)

© Copyright IBM Corp. 2004. All rights reserved.

Installation and Documentation



- ❑ Installed first in z/OS V1R2
 - Enabling APAR/PTF - OW50850 / UW82925
 - Coding APAR/PTF - OW51563 / UW83377
- ❑ The support for OS/390 V2R10 and z/OS V1R1 is provided by APAR OW51780
- ❑ z/OS Distributed File Service zSeries Administration, SC24-5989 - (Standard documentation)
- ❑ z/OS Distributed File Service zSeries File System Implementation, SG24-6580-01 - (z/OS V1R6)
 - Publication - 4Q 2004

© Copyright IBM Corp. 2004. All rights reserved.

zFS File Systems



- ❑ zFS is a Unix File System for z/OS
- ❑ A component of the Distributed File Service since 1995
 - Base element - new sub-component, zSeries File System (zFS) introduced in V1R5
- ❑ Using zFS, you can
 - Run applications just like HFS
 - Use zFS in addition to HFS or replace HFS
- ❑ Advantages:
 - Better performance
 - Enhanced administrative functions
 - Less loss of data on system failures

© Copyright IBM Corp. 2004. All rights reserved.

zSeries File System (zFS)



- ❑ zFS is the strategic UNIX Systems Services file system for z/OS
 - The Hierarchical File System (HFS) functionality has been stabilized
 - HFS is expected to continue shipping as part of the operating system and will be supported in accordance with the terms of a customer's applicable support agreement
- ❑ IBM intends to continue enhancing zFS functionality, including RAS and performance capabilities, in future z/OS releases
 - All requirements for UNIX file services are expected to be addressed in the context of zFS only

© Copyright IBM Corp. 2004. All rights reserved.

New Functions in z/OS V1R3



- ❑ zFS file systems support UNIX ACLs
- ❑ MOUNT statements for zFS file systems can now be placed in BPXPRMxx
- ❑ Use the UNIXPRIV class profile, SUPERUSER.FILESYS.PFSCTL, to protect the use of zfsadm commands
- ❑ Use dataspaces for user data cache

© Copyright IBM Corp. 2004. All rights reserved.

z/OS V1R4 zFS Enhancements



- Dynamic configuration
- Dynamic extension
- -grow option
- Duplicate file system names in different aggregates
- System symbols in IOEFSPRM
- Metadata backing cache
- Log file cache
- zFS mounts

© Copyright IBM Corp. 2004. All rights reserved.

z/OS V1R5 zFS Enhancements



- ❑ **Multilevel security (MLS) with zFS**
 - zFS supports security labels for MLS
 - Externals for MLS are contained in RACF, shell, z/OS UNIX APIs, etc.
 - zFS support is in PFS
- ❑ **ServerPac enhancements for zFS**
- ❑ **zFS remount considerations**
- ❑ **Dynamic creation of automounted zFS file systems**
- ❑ **Automount policy handling - HFS and zFS in parallel**
 - Via APAR OA06364 - PTF UA10075

© Copyright IBM Corp. 2004. All rights reserved.

z/OS V1R6 zFS Enhancements



- ❑ Logical file system (LFS) zFS support
- ❑ System management improvements
 - Parmlib search
- ❑ Availability improvements
 - Conditional asserts
- ❑ Performance information from a program
 - Performance monitoring APIs
 - zfsadm query command
- ❑ Automount policy handling - HFS and zFS in parallel

© Copyright IBM Corp. 2004. All rights reserved.

z/OS V1R5 Multilevel Security and zFS



- ❑ Multilevel security (MLS) was added to z/OS V1R5
 - Part of MLS is support for security labels
- ❑ zFS supports security labels for MLS
 - Stores security labels in FSP in metadata
 - Calls RACF for security label processing
- ❑ Externals for MLS are contained in other elements/components
 - (RACF, shell, z/OS Unix APIs, etc.)
- ❑ zFS supports low level functions for MLS

© Copyright IBM Corp. 2004. All rights reserved.

Multilevel Security with z/OS V1R5



- ❑ Support is added allowing security labels to be associated with file system resources and users to provide greater restrictiveness than is possible with POSIX permissions alone
 - ALL systems must be z/OS V1R5
 - Requires use of zFS for ROOT and /dev resources
 - Requires use of zFS for all file systems mounted for read/write access

© Copyright IBM Corp. 2004. All rights reserved.

ServerPac Processing



- ❑ New member, **ZFS**, copied from SIOEPROC to CPAC.PROCLIB in z/OS orders
 - Add to your procedure library concatenation
- ❑ Statement added to BPXPRMFS when needed to enable zFS filesystem support:
 - FILESYSTYPE TYPE(ZFS) ENTRYPOINT(IOEFSCM) ASNAME(**ZFS**)
- ❑ CICS, DB2, and IMS ServerPacs assume that any necessary zFS setup has been done

© Copyright IBM Corp. 2004. All rights reserved.

BPXPRMxx Definitions

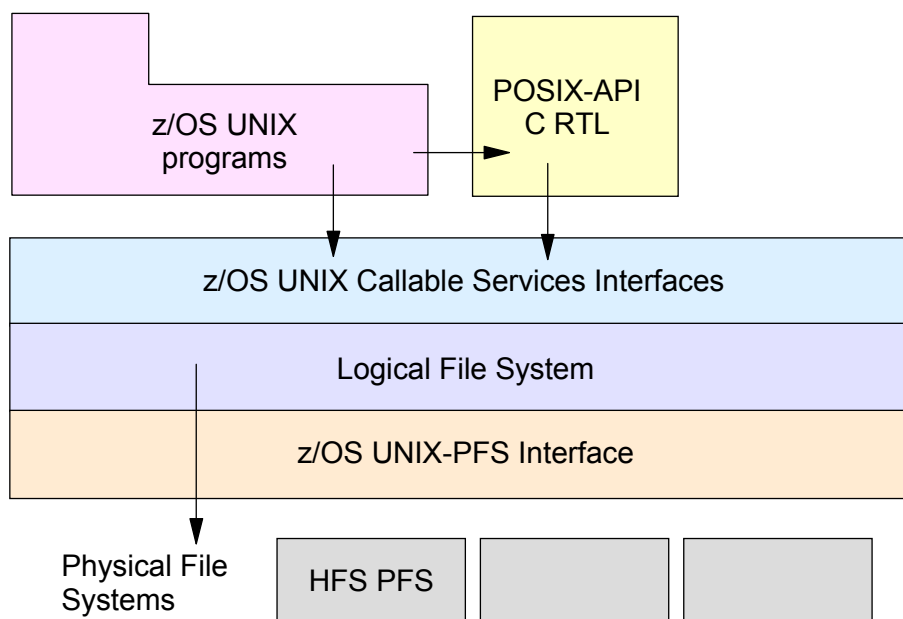


- ❑ zFS defined in z/OS UNIX parmlib member
- ❑ BPXPRMxx FILESYSTYPE statement
 - Defines zFS as a physical file system (PFS)
 - This definition causes z/OS UNIX to start zFS

FILESYSTYPE TYPE(ZFS)
ENTRYPOINT(IOEFSCM)
ASNAME(ZFS)

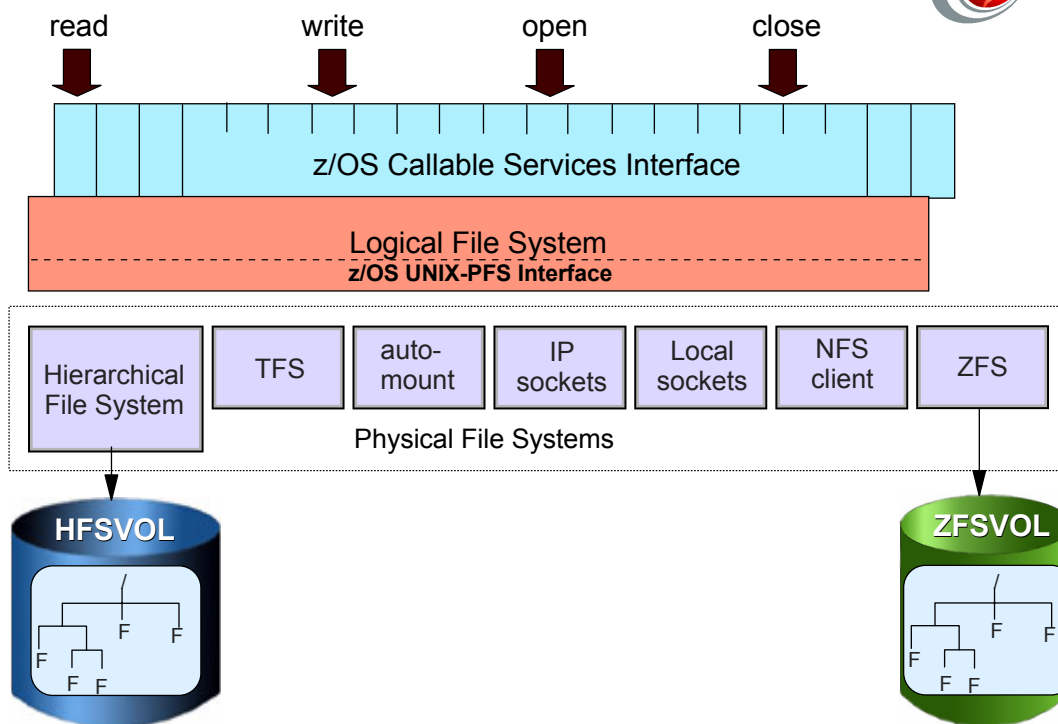
© Copyright IBM Corp. 2004. All rights reserved.

UNIX System Services



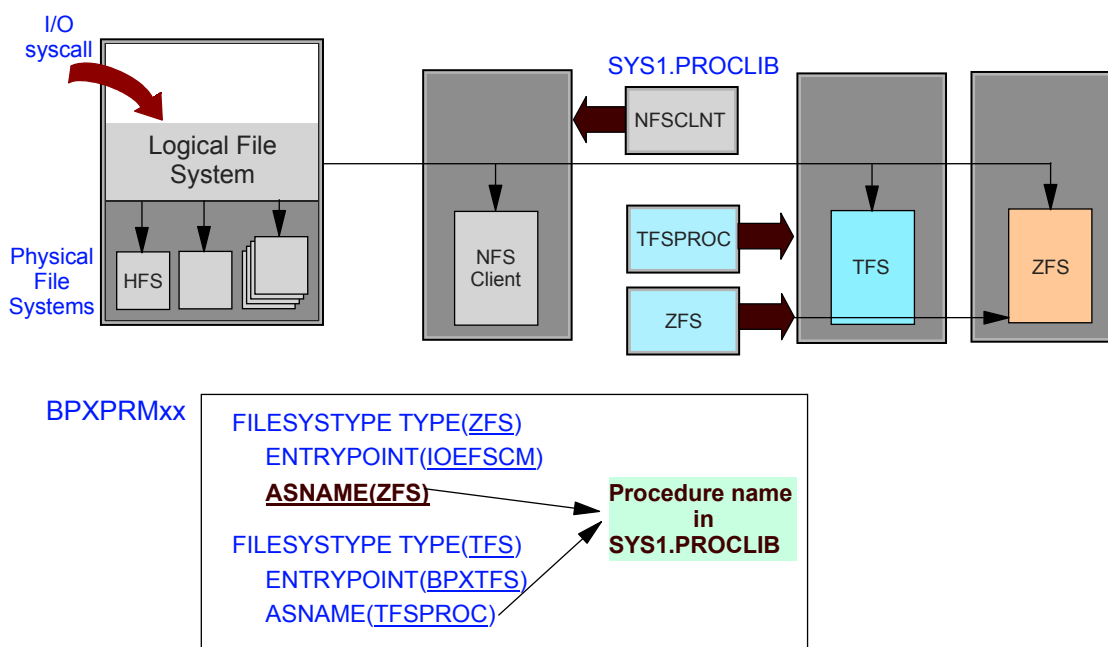
© Copyright IBM Corp. 2004. All rights reserved.

Physical File Systems



© Copyright IBM Corp. 2004. All rights reserved.

zFS Colony Address Space



© Copyright IBM Corp. 2004. All rights reserved.

zFS Procedure



ZFS PROC

```
//ZFS      PROC REGSIZE=0M
//ZFSGO EXEC PGM=BPXVCLNY,REGION=&REGSIZE,TIME=1440
//*STEPLIB DD DISP=SHR,DSN=IOE.SIOELMOD
//IOEZPRM DD DSN=IOE.IOEFSZFS(IOEFSPRM),DISP=SHR
// PEND
```

© Copyright IBM Corp. 2004. All rights reserved.

Dynamic Configuration - z/OS V1R4



- ☐ In previous releases - to change configuration parameters, you had to:
 - Modify the IOEFSPRM file
 - Shutdown and restart the ZFS PFS
- ☐ This causes unmounts and/or moves of zFS file systems (in a sysplex)
 - This can be disruptive to applications and is administratively involved
- ☐ With z/OS V1R4:
 - Use the **zfsadm config** command to change values
 - **zfsadm configquery** to display config values

© Copyright IBM Corp. 2004. All rights reserved.

IOEFSPRM File Options



```
zfsadm config [-admin_threads number]
[-user_cache_size number]
[-meta_cache_size number]
[-log_cache_size number]
[-sync_interval number]
[-vnode_cache_size number]
[-nbs {on|off}]
[-fsfull threshold,increment]
[-aggrfull threshold,increment]
[-trace_dsnPDSE_dataset_name]
[-tran_cache_size number]
[-msg_output_dsn Seq_dataset_name]
[-user_cache_readahead {on|off}]
[-metaback_cache_size number]
[-fsgrow increment,times]
[-aggrgrow {on|off}]
[-allow_dup_fs {on|off}]
[-level]
[-help]
```

New options
in z/OS V1R4

© Copyright IBM Corp. 2004. All rights reserved.

Logical Parmlib Support z/OS V1R6



- ☐ With this new support, a logical parmlib search is used
- ☐ Member names are in the form IOEPRMxx
- ☐ Multiple members can be specified
- ☐ Allows installation to have a common parmlib member that is shared among members of the sysplex but also have an additional member that is unique

The logical parmlib concatenation is a set of up to 10 partitioned data sets defined by PARMLIB statements in the LOADxx member of SYSn.IPLPARM or SYS1.PARMLIB

© Copyright IBM Corp. 2004. All rights reserved.

Specifying Parmlib Members



- ❑ List of member (suffixes) is specified in the FILESYSTYPE statement for ZFS in a BPXPRMxx member

```
FILESYSTYPE TYPE(ZFS) ENTRYPOINT(IOEFSCM)  
ASNAME(ZFS,'SUB=MSTR')  
PARM('PRM=(01,02,03)')
```

```
FILESYSTYPE TYPE(ZFS) ENTRYPOINT(IOEFSCM)  
ASNAME(ZFS,'SUB=MSTR')  
PARM('PRM=(AA,&SYSCClone.)')
```

© Copyright IBM Corp. 2004. All rights reserved.

Searching for IOEZPRM



- ❑ If IOEZPRM DD not specified in ZFS PROC
 - Before V1R6, hard-coded defaults taken for zFS configuration options
 - In V1R6 and later, parmlib members specified in PRM= are searched
 - If no PRM= is specified, IOEPRM00 searched for zFS configuration options and if IOEPRM00 non-existent, hard-coded defaults taken
- ❑ You could wait to remove the IOEZPRM DD until all systems are at z/OS V1R6

© Copyright IBM Corp. 2004. All rights reserved.

zfsadm Commands to Manage File Systems and Aggregates



zfsadm attach	Attach an aggregate
zfsadm apropos	Display first line of help entry
zfsadm detach	Detach an aggregate
zfsadm grow	Grow an aggregate
zfsadm aggrinfo	Obtain information on an attached aggregate
zfsadm clone	Clone a filesystem
zfsadm clonesys	Clone multiple filesystems
zfsadm create	Create a filesystem
zfsadm delete	Delete a filesystem
* zfsadm define	Create a VSAM linear data set aggregate
* zfsadm format	Format an aggregate
zfsadm help	Get help on commands
zfsadm lsaggr	List all currently attached aggregates
zfsadm lsfs	List all file systems on an aggregate or all
zfsadm lsquota	List filesystem information
zfsadm quiesce	Quiesce an aggregate and all file systems
zfsadm rename	Rename a file system
zfsadm setquota	Set the quota for a file system
zfsadm unquiesce	Make the aggregate and all file systems available
+ zfsadm config	Change value of zFS configuration (IOEFSPRM) in memory
+ zfsadm configquery	Query the current value of zFS configuration option
@ zfsadm query	Query or reset the performance counters

* New with z/OS V1R3
+ New with z/OS V1R4
@ New with z/OS V1R6

© Copyright IBM Corp. 2004. All rights reserved.

zFS Aggregates - (Terminology)



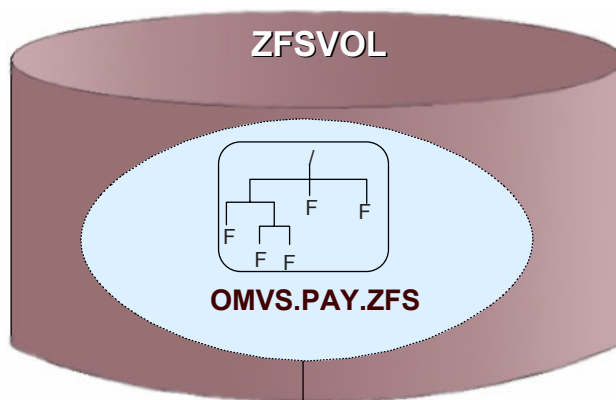
- ☐ An aggregate is a VSAM linear data set (LDS)
- ☐ An aggregate contains one or more zFS file systems
- ☐ Two types of aggregates:
 - HFS compatibility mode - contains 1 zFS file system
 - Multi-file system mode - contains 1 or more zFS file systems
 - Space sharing between file systems in same aggregate
- ☐ File system maximum size as a logical value
- ☐ File system clone (making a read/only copy)
- ☐ zFS File System
 - Can be mounted within the HFS root file system
 - Contains a logical maximum size known as its quota

© Copyright IBM Corp. 2004. All rights reserved.

zFS Compatibility Mode Aggregate



zFS compatibility mode
aggregate

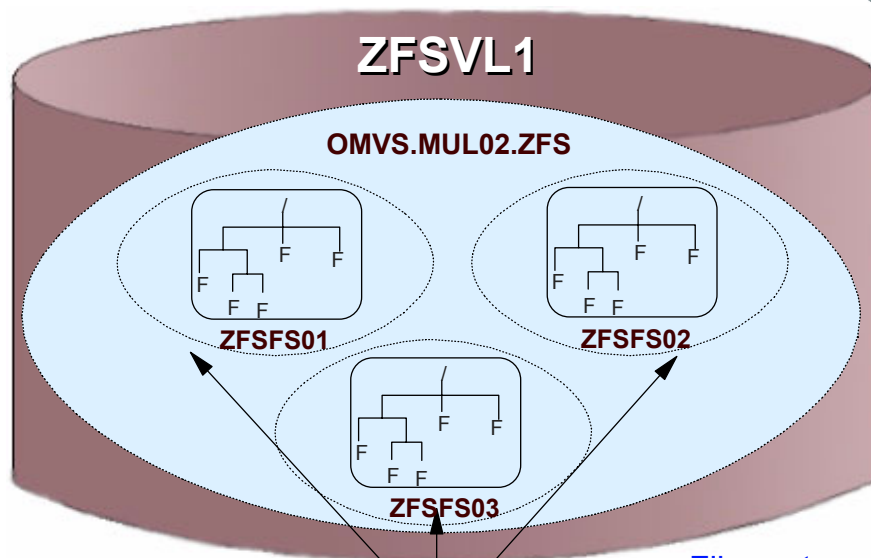


zFS file system

VSAM LDS name=Aggregate name=File system name=OMVS.PAY.ZFS

© Copyright IBM Corp. 2004. All rights reserved.

Multiple File System Aggregate



File system names

ZFSFS01
ZFSFS02
ZFSFS03

zFS file systems

VSAM LDS name=Aggregate name=OMVS.MUL02.ZFS

© Copyright IBM Corp. 2004. All rights reserved.

z/OS V1R3 Define Aggregate



☐ zfsadm command - zfsadm define

```
$> zfsadm define -a OMVS.CMP01.ZFS -volumes totzf1 -cylinders 10 1
IOEZ00248E VSAM linear dataset OMVS.CMP01.ZFS successfully created.
$> su
#> zfsadm format -a OMVS.CMP01.ZFS -compat -owner 316 -p o755
IOEZ00077I HFS-compatibility aggregate OMVS.CMP01.ZFS has been
successfully created
#> /usr/sbin/mount -f OMVS.CMP01.ZFS -t ZFS /u/zfs/cmp01
```

☐ Display quota

```
$> zfsadm lsquota -filesystem OMVS.CMP01.ZFS
```

Filesys Name	Quota	Used	Percent	Used	Aggregate
OMVS.CMP01.ZFS	6335	9	0	2	= 154/6480 (zFS)

© Copyright IBM Corp. 2004. All rights reserved.

Allocate zFS Aggregates



- ☐ An aggregate can contain one or more zFS file systems
- ☐ A zFS file system is equivalent to an HFS file system

```
//RFRZAL JOB (999,POK), 'R F',CLASS=A,MSGCLASS=U,NOTIFY=&SYSUID,
//      REGION=0M
//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//DISK DD DISP=OLD,UNIT=3390,VOL=SER=ZFSVOL
//SYSIN DD *
        DEFINE CLUSTER -
                (NAME (OMVS.MUL01.ZFS) VOL(ZFSVL1,ZFSVL2,ZFSVL3) -
                LINEAR SHAREOPTIONS(2))
```

© Copyright IBM Corp. 2004. All rights reserved.

Allocate a zFS Aggregate - z/OS V1R4



☐ Allocate an aggregate from the ISHELL

➤ Option 4 New ZFS ...

File	Directory	Special_file	Tools	File_systems	Options	Setup	Help
UNIX System Serv				-			
Command ==> _____				1. Mount table...			
Enter a pathname and do one of these:				2. New HFS...			
				3. Mount(0)...			
				4. New ZFS...			
<ul style="list-style-type: none">- Press Enter.- Select an action bar choice.- Specify an action code or command on the command line.							
Return to this panel to work with a different pathname.							
/u/harry						More:	+

EUID=0							

© Copyright IBM Corp. 2004. All rights reserved.

Allocate an Aggregatez/OS V1R4



☐ New panel to allocate the zFS aggregate

File	Directory	Special_file	Tools	File_systems	Options	Setup	Help
Create a zFS File System							
Enter the fields as required then press Enter.							
C	File system name OMVS.TEST.ZFS						
E	Owning User ROGERS (Number or user name)						
	Owning Group SYS1 (Number or group name)						
	Permissions 700 (3 digits, each 0-7)						
R	Primary cylinders 10						
	Secondary cylinders 5						
	Storage class						
	Management class						
	Data class						
E	Volume names						
F1=Help F3=Exit F6=Keyshelp F12=Cancel							

© Copyright IBM Corp. 2004. All rights reserved.

zFS Aggregates on Disk



- ❑ A zFS aggregate is an array of 8K blocks
- ❑ Three special objects in all zFS aggregates that take up space that cannot be used for user files:
 - Log file - Used to record metadata changes and is by default 1% of the disk size
 - Bitmap - Records which blocks are free on disk, and is as big as needed and depends on size of the aggregate
 - Aggregate File System List - Describes the file systems contained in the aggregate
 - Compatibility mode aggregates - only 1 8KB block
 - Multi-file system aggregates - Depends on number of file systems

© Copyright IBM Corp. 2004. All rights reserved.

zFS Disk Space



New with z/OS V1R5

- ❑ `zfsadm aggrinfo` shows aggregate disk space usage
 - Based on the number of 8K blocks
 - Subtracts the space reserved for the 3 objects

```
ROGERS @ SC65:/u/rogers>zfsadm aggrinfo
IOEZ00368I A total of 9 aggregates are attached.
ZFSFR.ROOT.ZFS (R/W COMP): 292908 K free out of total 1440000
ZFSFR.ZFSG.ZFS (R/W COMP): 6374 K free out of total 7200
ZFSFR.ZFSF.ZFS (R/W COMP): 6374 K free out of total 7200
ZFSFR.ZFSE.ZFS (R/W COMP): 6374 K free out of total 7200
ZFSFR.ZFSD.ZFS (R/W COMP): 6374 K free out of total 7200
ZFSFR.ZFSC.ZFS (R/W COMP): 57760 K free out of total 576000
ZFSFR.ZFSB.ZFS (R/W COMP): 57760 K free out of total 576000
ZFSFR.ZFSA.ZFS (R/W COMP): 6374 K free out of total 7200
ZFSFR.ROOT1.ZFS (R/W COMP): 292908 K free out of total 1440000
```

© Copyright IBM Corp. 2004. All rights reserved.

zFS Threshold Monitoring Space Usage



New with z/OS V1R5

❑ zFS threshold monitoring function aggrfull reports:

- Space usage based on total aggregate disk size
- Has the space for the three special objects - total disk space and amount used on disk in its messages
- The aggrfull message shows units in 8K blocks

```
ROGERS @ SC65:/u/rogers>zfsadm config -aggrfull "(80,5)"
```

```
IOEZ00300I Successfully set -aggrfull to (80,5)
```

```
ROGERS @ SC65:/u/rogers>zfsadm configquery -aggrfull
```

```
IOEZ00317I The value for configuration option -aggrfull is  
(80,5).
```

Number of 8K blocks in the total aggregate

Number of 8K blocks used in the aggregate

```
IOEZ00078W zFS aggregate Name exceeds %d%% full (%u/%u)  
(WARNING)
```

```
IOEZ00079I zFS aggregate Name is now below %d%% full (%u/%u)
```

© Copyright IBM Corp. 2004. All rights reserved.

zFS Aggregate Space



```
ROGERS @ SC65:/u/rogers>zfsadm aggrinfo ZFSFR.ZFSG.ZFS
```

```
ZFSFR.ZFSG.ZFS (R/W COMP): 6374 K free out of total 7200
```

```
ROGERS @ SC65:/u/rogers>zfsadm lsquota ZFSFR.ZFSG.ZFS
```

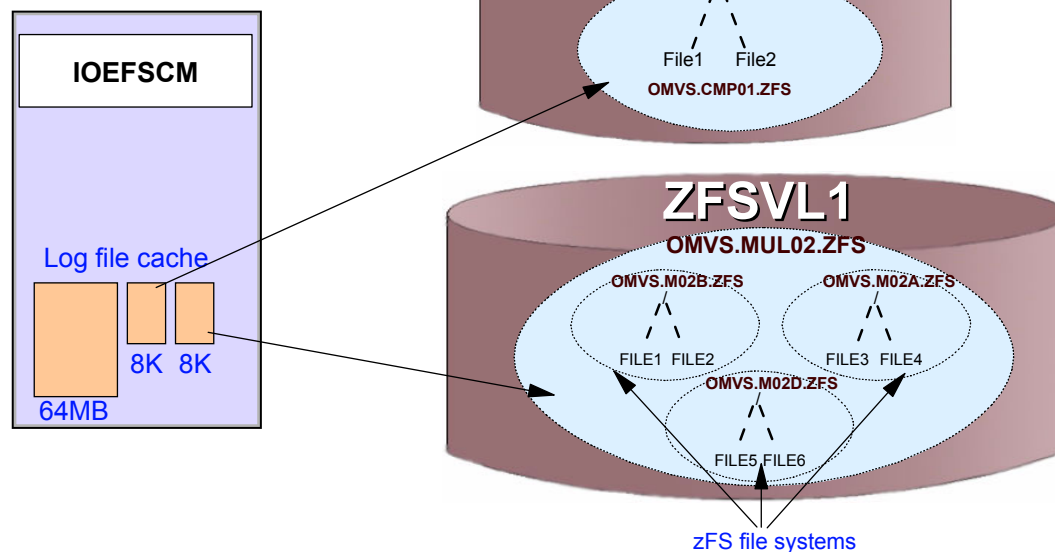
Filesys Name	Quota	Used	Percent Used	Aggregate
ZFSFR.ZFSG.ZFS	7047	673	9	11 = 826/7200 (zFS)

© Copyright IBM Corp. 2004. All rights reserved.

Log File Cache (V1R3 and lower)



zFS address space



© Copyright IBM Corp. 2004. All rights reserved.

Log File Cache

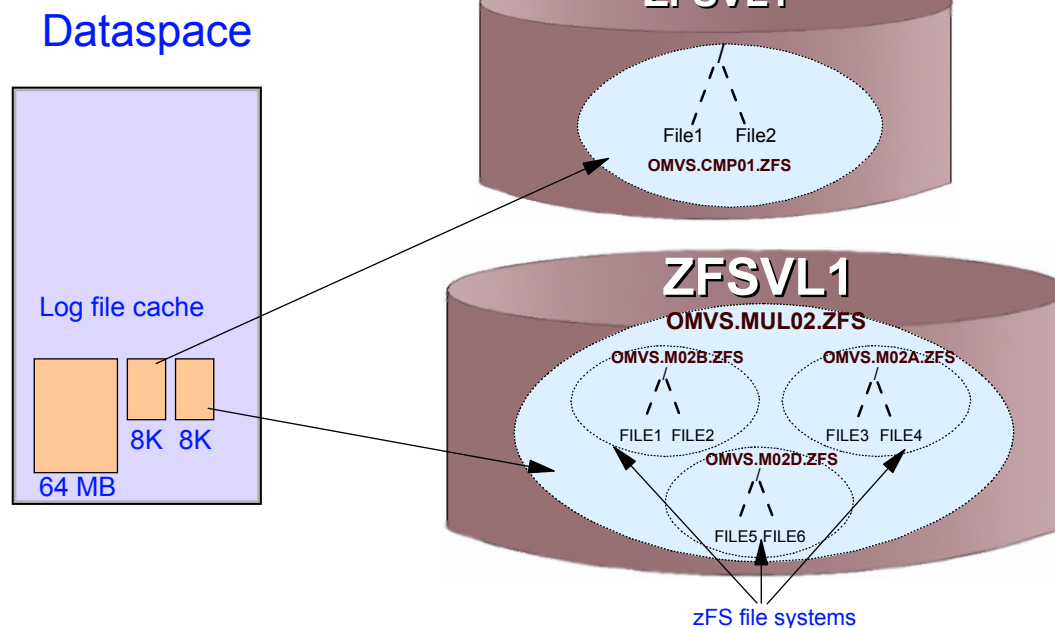


- ☐ With **V1R4**, the log file cache is in a dataspace
 - Default - (64 MB)
- ☐ Frees up space for caches in ZFS address space

Cache	Dataspace?	How many?
User data	Yes	32
Log file	Yes	1
Metaback	Yes	1
Meta	No	
Vnode	No	
Transaction	No	

© Copyright IBM Corp. 2004. All rights reserved.

Log File Cache V1R4



© Copyright IBM Corp. 2004. All rights reserved.

Aggregates and Multiple File Systems



- ❑ zFS aggregates are all capable of containing multiple file systems, even compatibility mode aggregates
- ❑ Compatibility mode aggregates can have backup file systems in them that take space if the clone operation is used
- ❑ Each file system has a quota represented in 1KB fragments
- ❑ The quota of a file system is a logical number and can be smaller or larger than the size of the disk (if the size of the disk were expressed in 1KB fragments)

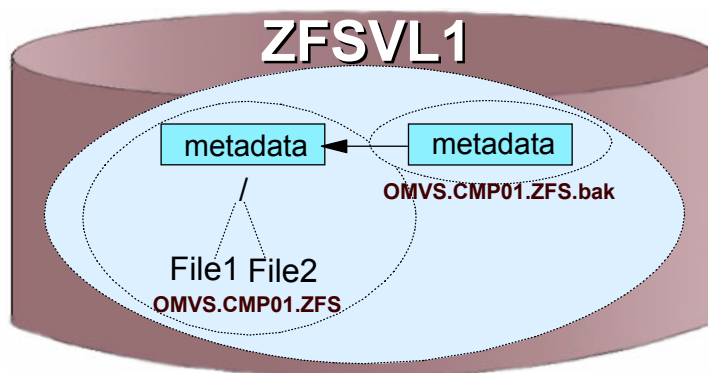
© Copyright IBM Corp. 2004. All rights reserved.

Backup File System - zFS Clone



`zfsadm clone -filesystem OMVS.CMP01.ZFS`

IOEZ00225I File system OMVS.CMP01.ZFS successfully cloned.

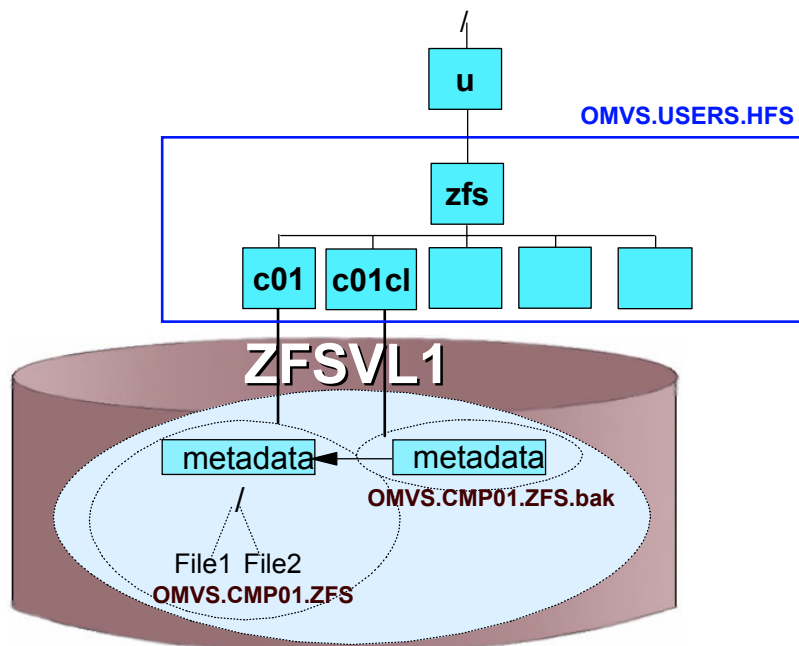


© Copyright IBM Corp. 2004. All rights reserved.

zFS File System Clone



```
MOUNT FILESYSTEM(''OMVS.CMP01.ZFS.bak'')  
MOUNTPOINT('/u/zfs/c01cl') TYPE(READ) NOAUTOMOVE
```



© Copyright IBM Corp. 2004. All rights reserved.

Using the Clone



- ❑ After a clone operation creates the backup file system, if the read-write file system user data is updated, zFS does the following:
 - Makes sure that new physical blocks are allocated to hold the updates
 - Maintains the backup file system's data pointers to the original data
 - Keeps the backup file system as an exact copy at the point-in-time the clone was made when updates to the read-write file system are made

© Copyright IBM Corp. 2004. All rights reserved.

Update the Clone - (clonesys)



- ❑ `zfsadm clonesys -aggregate OMVS.CMP01.ZFS`
 - Update previous clone
- ❑ `zfsadm clonesys -aggregate OMVS.MUL02.ZFS`
 - Clone a set of file systems

```
IOEZ00219I Clonesys starting for aggregate OMVS.MUL02.ZFS, prefix * all
filesystems *
IOEZ00225I File system OMVS.M01.ZFS successfully cloned.
IOEZ00225I File system OMVS.M02.ZFS successfully cloned.
IOEZ00225I File system OMVS.m03.ZFS successfully cloned.
IOEZ00225I File system OMVS.M04.ZFS successfully cloned.
IOEZ00216I Clone ending for aggregate OMVS.MUL02.ZFS (Total: 4, Failed: 0,
Time.340)
ROGERS @ SC43:/>zfsadm lsfs
IOEZ00129I Total of 8 file systems found for aggregate OMVS.MUL02.ZFS
OMVS.m03.ZFS RW (Mounted R/W)      8 K alloc      9 K quota On-line
OMVS.m03.ZFS.bak BK (Not Mounted)   9 K alloc      9 K quota On-line
OMVS.M01.ZFS RW (Mounted R/W)     24 K alloc     77 K quota On-line
OMVS.M01.ZFS.bak BK (Not Mounted)  77 K alloc     77 K quota Online
.....
```

© Copyright IBM Corp. 2004. All rights reserved.

Format the zFS Aggregate



- ❑ Stand alone utility to format:
 - IOEAGFMT format utility
 - zFS multiple file aggregates
 - Compatibility mode aggregates
- ❑ ZFSADM command to format
 - zfsadm format -aggregate name [-initialempty blocks] [-size blocks] [-logsize blocks] [-overwrite] [-compact] [-owner {uid | name}] [-group {group_id | name}] [-perms decimal | octal | hex_number] [-level] [-help]
- ❑ Does not need zFS colony address space to be active
- ❑ Does not use the IOEFSPRM configuration file

© Copyright IBM Corp. 2004. All rights reserved.

Format Aggregates



A compatibility mode aggregate is formatted with this JCL:

```
//RFRAGF JOB (999,POK),'R F',CLASS=A,MSGCLASS=U,NOTIFY=&SYSUID,
// REGION=0M
//STEP1 EXEC PGM=IOEAGFMT,
//          PARM=(' -aggregate omvs.cmp01.zfs -compat ')
//SYSPRINT DD SYSOUT=*
//STDOUT DD SYSOUT=*
//STDERR DD SYSOUT=*
//CEEDUMP DD SYSOUT=*
```

A multi-file system aggregate is formatted with this JCL:

```
//RFRAGF JOB (999,POK),'R F',CLASS=A,MSGCLASS=U,NOTIFY=&SYSUID,
// REGION=0M
//STEP1 EXEC PGM=IOEAGFMT,
//          PARM=(' -aggregate omvs.mul01.zfs ')
//SYSPRINT DD SYSOUT=*
//STDOUT DD SYSOUT=*
//STDERR DD SYSOUT=*
//CEEDUMP DD SYSOUT=*
```

© Copyright IBM Corp. 2004. All rights reserved.

IOEAGFMT Successful Messages



```
IOEZ00004I Loading dataset 'omvs.cmp01.zfs'.
IOEZ00005I Dataset 'omvs.cmp01.zfs' loaded successfully.
*** Using default initialempty value of 1.
*** Using default number of (8192-byte) blocks: 17999
*** Defaulting to 179 log blocks(maximum of 19 concurrent transactions).
Done. /dev/lfs1/omvs.cmp01.zfs is now an zFS aggregate.
IOEZ00071I Attaching aggregate OMVS.CMP01.ZFS to create hfs-compatible file
system
IOEZ00074I Creating file system of size 140487K, owner id 0, group id 2,
permissions xLED
IOEZ00048I Detaching aggregate OMVS.CMP01.ZFS
IOEZ00077I HFS-compatibility aggregate OMVS.CMP01.ZFS has been successfully
created
```

```
IOEZ00004I Loading dataset 'omvs.mul01.zfs'.
IOEZ00005I Dataset 'omvs.mul01.zfs' loaded successfully.
*** Using default initialempty value of 1.
*** Using default number of (8192-byte) blocks: 17999
*** Defaulting to 179 log blocks(maximum of 19 concurrent transactions).
Done. /dev/lfs1/omvs.mul01.zfs is now an zFS aggregate.
```

© Copyright IBM Corp. 2004. All rights reserved.

Attaching a Multi-file Mode Aggregate



- ☐ Compatibility mode aggregates do not require attach
- ☐ Multi-file mode aggregates require an attach
- ☐ 3 ways to attach
 - Attach at zFS colony address space startup by having an entry in the IOEFSPRM file
 - define_aggr R/W attach cluster(OMVS.MUL01.ZFS)
 - Using the IOEZADM program in a submitted job
 - PARM=('attach -aggregate OMVS.MUL01.ZFS')
 - Using the zfsadm attach command
 - zfsadm attach -aggregate omvs.mul01.zfs -aggrfull 90,5 -nbs

© Copyright IBM Corp. 2004. All rights reserved.

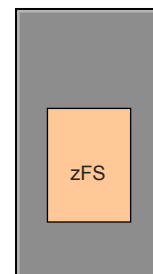
Defining Mult-File zFS File Systems



- ☐ Use IOEZADM program in JCL
- ☐ Use zfsadm command from OMVS shell

zfsadm create -filesystem OMVS.M01A.ZFS -size 5000 -aggregate OMVS.MUL01.ZFS

```
//ZFZADM JOB , 'ZFS Create Filesys', NOTIFY=ROGERS,
//          CLASS=A, MSGCLASS=X, MSGLEVEL=(1,1), TIME=1440
//*
//ZFZADM EXEC PGM=IOEZADM, REGION=0M,
//          PARM=('create -filesystem OMVS.M01A.ZFS
//              -aggregate omvs.mul01.zfs -size 5000')
//STEPLIB DD DISP=SHR, DSN=IOE.SIOELMOD
//SYSPRINT DD SYSOUT=T
//STDOUT DD SYSOUT=T
//STDERR DD SYSOUT=T
//SYSUDUMP DD SYSOUT=T
//CEEDUMP DD SYSOUT=T
//*
```



Colony Address Space

IOEZ00099I File system OMVS.M01A.ZFS created successfully

© Copyright IBM Corp. 2004. All rights reserved.

ISHELL Support for zFS (z/OS V1R5)



ISHELL provides ability to manage zFS aggregates

```
File Directory Special_file Tools File_systems Options Setup Help
Command ==> UNIX System Serv
Enter a pathname and do one of these:
    - Press Enter.
    - Select an action bar choice.
    - Specify an action code or command on the command line.

Return to this panel to work with a different pathname.

/u
More: +

EUID=0
```

- 5 1. Mount table...
2. New HFS...
3. Mount(0)...
4. New ZFS...
5. ZFS aggregates...

© Copyright IBM Corp. 2004. All rights reserved.

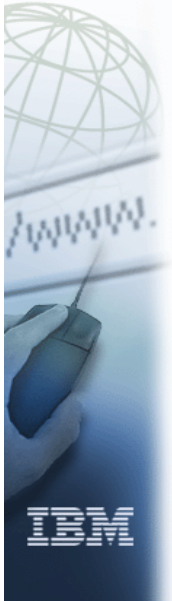
© Copyright IBM Corp. 2004. All rights reserved

© Copyright IBM Corp. 2004. All rights reserved

ibm.com



e-business



Growing Aggregates and File Systems

Understanding zFS Format of Large Aggregates



Redbooks

International Technical Support Organization



© Copyright IBM Corp. 2004. All rights reserved.

Grow an Aggregate



☐ Grow the size of an aggregate

- Size 0 - Indicates to use secondary extent allocation
- Specify size in KB

```
ROGERS @ SC43: />zfsadm aggrinfo omvs.mul02.zfs  
OMVS.MUL02.ZFS (R/W MULT): 50806 K free out of total 51272 (2000 reserved)
```

```
#> zfsadm grow -aggregate OMVS.MUL02.ZFS -size 0  
IOEZ00173I Aggregate OMVS.MUL02.ZFS successfully grown  
OMVS.MUL02.ZFS (R/W MULT): 61598 K free out of total 62072 (2000 reserved)
```

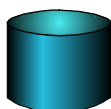
© Copyright IBM Corp. 2004. All rights reserved.

New -grow Option - z/OS V1R4

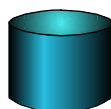


- ❑ When an aggregate is initially formatted using IOEAGFMT or zfsadm format command
 - Size, -size specified must be able to be allocated in the primary allocation and one extension
- ❑ If you wanted to format a three volume aggregate with IOEAGFMT - not possible
 - Requires a primary and at least two extensions
- ❑ Need to use **zfsadm format** command to specify:
 - `zfsadm format -size 900720 -grow 300240`

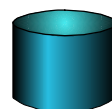
300240 8K blocks



300240 8K blocks



300240 8K blocks

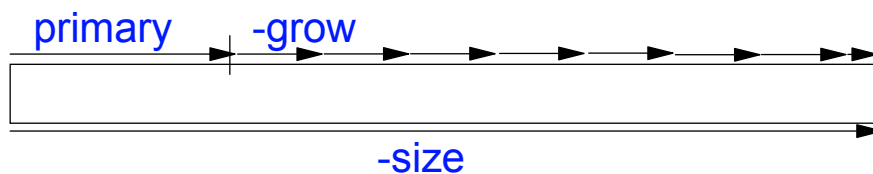


© Copyright IBM Corp. 2004. All rights reserved.

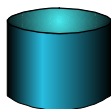
New -grow Option - z/OS V1R4



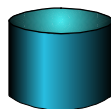
- ❑ Now, IOEAGFMT and **zfsadm format** provide the -grow option
 - Specifies the increment that will be used for extension when -size is larger than the primary allocation
 - Extends by -grow amount until -size is satisfied



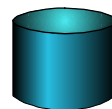
300240 8K blocks



300240 8K blocks



300240 8K blocks



© Copyright IBM Corp. 2004. All rights reserved.

Dynamic Aggregate Extension - V1R4



- ❑ Pre **z/OS V1R4**, aggregates/quotas must be grown via:
 - **zfsadm grow** command
 - File system quota must increase by **zfsadm setquota**
- ❑ With **z/OS V1R4**, aggregates and file system quotas can be dynamically increased
 - VSAM LDS must have a secondary allocation + space

© Copyright IBM Corp. 2004. All rights reserved.

Dynamic Aggregate Extension - V1R4



- ❑ Ways to specify aggregate extension
 - **zfsadm config** command - aggrgrow on | off
 - New option in IOEFSPRM file - **aggrgrow=on|off**
 - mount command - parm('aggrgrow')
 - mount filesystem('omvs.test.zfs') mountpoint('/tmp/test') type(zfs) mode(rdwr) parm('aggrgrow')
- ❑ Multi-file mode aggregate dynamic extension
 - Attach the aggregate using IOEFSPRM file
 - define_aggr R/W attach aggrgrow cluster(OMVS.TEST.ZFS) **aggrgrow | noaggrgrow**
 - **zfsadm attach** command - -aggrgrow or -noaggrgrow
`zfsadm attach -aggregate OMVS.TEST.ZFS -aggrgrow`

© Copyright IBM Corp. 2004. All rights reserved.

Dynamic Aggregate Extension - V1R4



❑ When the aggregate fills and an option is specified

- Aggregate is extended using a secondary allocation
- Secondary allocation is formatted
- Becomes available to application

```
IOEZ00312I Dynamic growth of aggregate OMVS.TEST.ZFS in progress, (by
user JANE) .
IOEZ00329I Attempting to extend OMVS.TEST.ZFS by a secondary extent.
IOEZ00324I Formatting to 8K block number 360 for secondary extents of
OMVS.TEST.ZFS
IOEZ00309I Aggregate OMVS.TEST.ZFS successfully dynamically grown (by
user JANE) .
```

© Copyright IBM Corp. 2004. All rights reserved.

Dynamic File System Quota Increase



❑ File systems have a quota (maximum size)

- Logical number used when additional blocks added
- Can be smaller, equal, or larger than space in aggregate
- When quota is reached - file system is full

❑ Ways to dynamically increase file system quota - V1R4

- IOEFSPRM file - **fsgrow=(increment,times)**
 - **increment** - in k-bytes up to 2147483647
 - **times** - Number of times to extend quota
- **mount** command - **parm('fsgrow=(increment,times)')**
- **zfsadm config** command - **-fsgrow 500,4**
 - **fsgrow(500,4)** means grow the quota by 500K bytes up to 4 times

© Copyright IBM Corp. 2004. All rights reserved.

Dynamic File System Quota Increase ...



❑ Display quota

```
$> zfsadm lsquota -filesystem OMVS.TEST.ZFS
Filesys Name      Quota    Used    Percent Used  Aggregate
OMVS.TEST.ZFS     1159      9        0      11 = 146/1296 (zFS)
```

❑ Compatibility mode aggregates

- Only the **aggrgrow** option can extend the size
 - **fsgrow** option is ignored

Before dynamic extension:

Filesys Name	Quota	Used	Percent Used	Aggregate
OMVS.TEST.ZFS	1159	9	0	11 = 146/1296 (zFS)

After dynamic extension:

Filesys Name	Quota	Used	Percent Used	Aggregate
OMVS.TEST.ZFS	1807	1577	87	88 = 1714/1944 (zFS)

© Copyright IBM Corp. 2004. All rights reserved.

ibm.com



e-business



Mounting zFS File Systems



Redbooks

International Technical Support Organization

© Copyright IBM Corp. 2004. All rights reserved.

Mounting zFS File Systems



- ❑ zFS aggregates (Compatibility and Multi-file)
 - Place in /etc/rc
 - /usr/sbin/mount -t ZFS -f OMVS.CMP01.ZFS /u/zfs/c01
 - Mount command
 - Automount facility
 - AUTOMOVE option in a shared hfs (sysplex) ****
 - Place mount statement in BPXPRMxx member
- ❑ **** Multiple file aggregates
 - Does not support AUTOMOVE

Restriction: No BPXPRMxx mount statements in z/OS V1R2

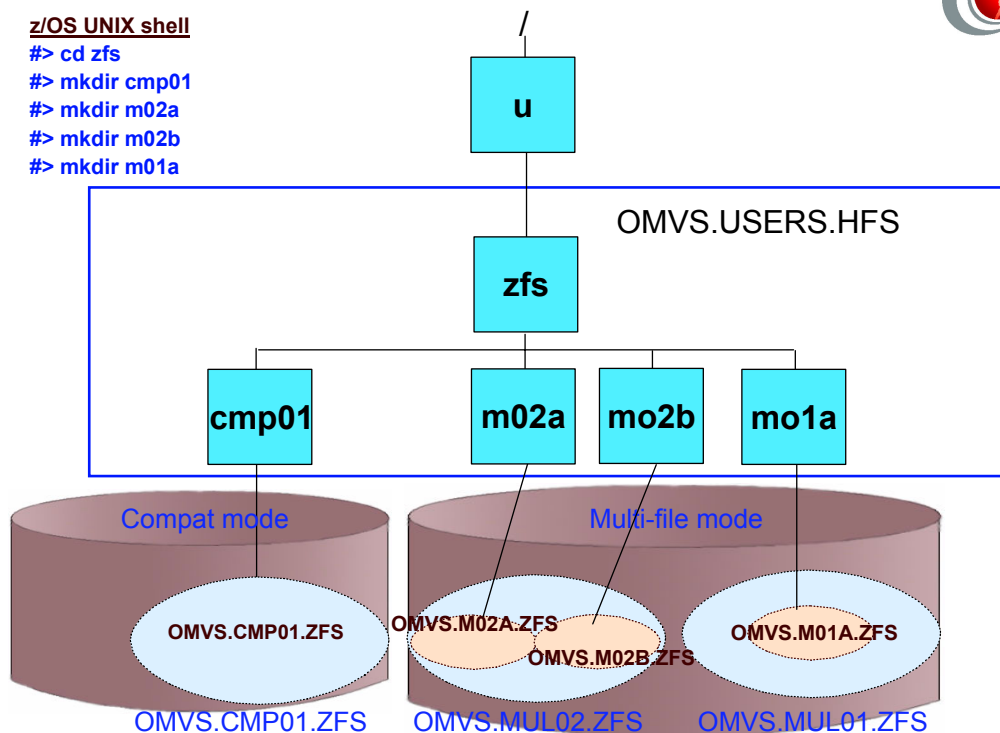
© Copyright IBM Corp. 2004. All rights reserved.

zFS File Systems - Direct Mount



z/OS UNIX shell

```
#> cd zfs
#> mkdir cmp01
#> mkdir m02a
#> mkdir m02b
#> mkdir m01a
```



© Copyright IBM Corp. 2004. All rights reserved.

Mount Command from TSO/E examples



Mount a multi-file mode file system

```
MOUNT FILESYSTEM('OMVS.M01A.ZFS') TYPE(ZFS)  
MODE(RDWR) MOUNTPOINT('/u/zfs/m01a')
```

Mount a compatibility mode file system

```
MOUNT FILESYSTEM('OMVS.CMP01.ZFS') TYPE(ZFS)  
MODE(RDWR) MOUNTPOINT('/u/zfs/cmp01')
```

- ❑ Mounts could be done via
 - /etc/rc
 - BPXPRMxx member

© Copyright IBM Corp. 2004. All rights reserved.

Automount Policy for zFS



1.

BPXPRMxx		
FILESYSTYPE	TYPE(AUTOMNT)	ENTRYPOINT(BPXTAMD)

2.

/etc/auto.master

/u	/etc/u.map
/z	/etc/z.map

<asis_name>

3.

/etc/z.map

### ZFS automount map file for mount point /z ###	
name	*
type	ZFS
filesystem	OMVS.<uc_name>.ZFS
mode	rdwr
duration	nolimit
delay	10

© Copyright IBM Corp. 2004. All rights reserved.

Automount Policy for HFS



1.

BPXPRMxx
FILESYSTYPE TYPE(AUTOMNT) ENTRYPOINT(BPXTAMD)

2. /etc/auto.master

/u	/etc/u.map
----	------------

3. /etc/u.map

HFS automount map file for mount point /u
name *
type HFS
filesystem <uc_name>.HFS
mode rdwr
duration nolimit
delay 10

© Copyright IBM Corp. 2004. All rights reserved.

Automount of zFS File Systems

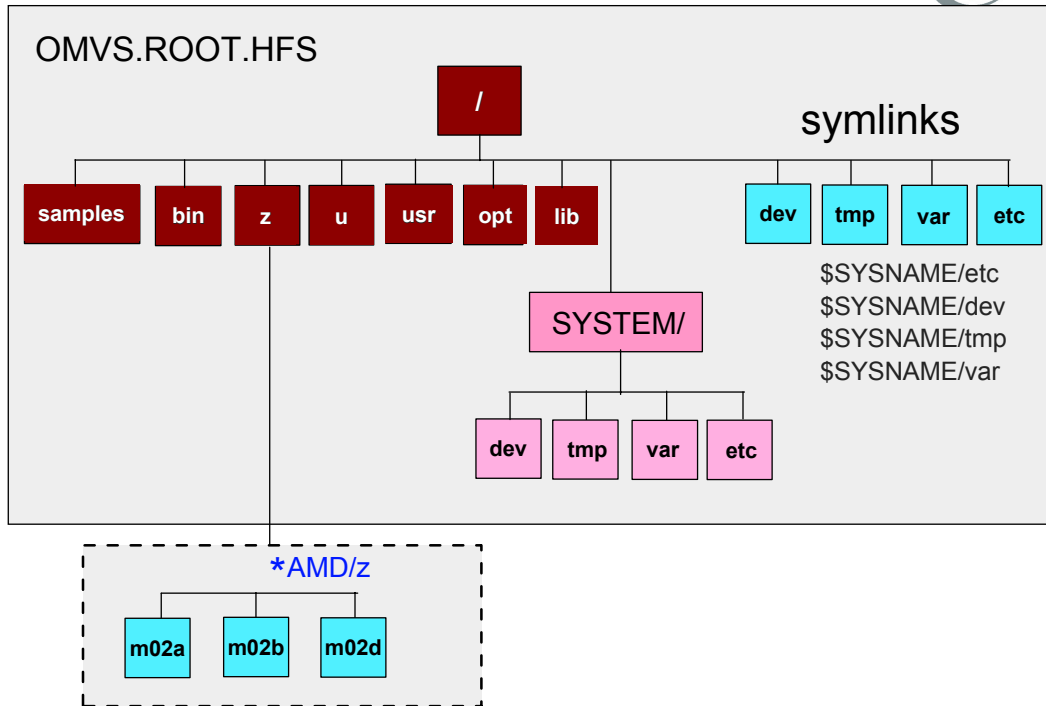


☐ Users or programs reference file system

```
ROGERS @ SC43:>cd /z/m02a
ROGERS @ SC43:>/z/m02a>ls -al
total 336
drwxr-xr-x  9 HERING  SYS1      736 Apr  2 01:43 .
drwxr-xr-x  9 HERING  SYS1      736 Apr  2 01:43 ..
-rwxr-xr-x  1 HERING  SYS1    1869 Mar 14 16:27 .profile
-rwxr-xr-x  1 HERING  SYS1     689 Mar 14 16:29 .setup
-rw-----  1 HERING  SYS1   3033 Apr  1 15:29 .sh_history
drwxr-xr-x  2 HERING  SYS1     256 Mar 15 16:08 bin
drwx-----  4 RC43    SYS1   1088 Apr  4 00:35 test
-r--r--r--  1 RC43    SYS1     147 Mar 17 14:41 test.extattr
drwx-----  2 RC43    SYS1     256 Mar 31 15:29 test1
lrwxrwxrwx  1 RC43    SYS1        15 Apr  4 00:35 test1.sl ->
/u/hering/test
1
drwxr-xr-x  2 HERING  SYS1     256 Mar 14 14:58 test2
drwxr-xr-x  2 HERING  SYS1     256 Mar 19 00:43 test3
```

© Copyright IBM Corp. 2004. All rights reserved.

zFS File Systems Mounted (automount)



© Copyright IBM Corp. 2004. All rights reserved.

copytree or pax Command



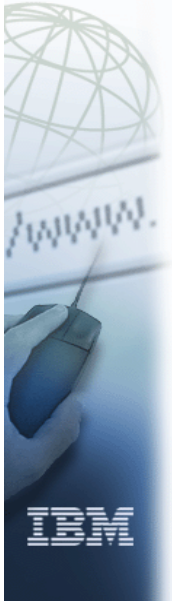
- ❑ Use to copy HFS file system to zFS file system
 - Can do with or without an intermediate archive file - pax
- ❑ HFS file system mounted at /etc/hfs1
 - Copy this into an empty zFS file mounted at /tmp/zfs/zfs1
 - cd /etc/hfs1
 - pax -rwv /tmp/zfs/zfs1
- ❑ copytree available from /samples
 - copytree /etc/hfs1 /tmp/zfs/zfs1

© Copyright IBM Corp. 2004. All rights reserved.

ibm.com



e-business



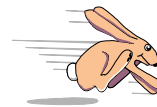
zFS Root, Recovery, Sysplex Sharing,

Working with zFS File Systems



Redbooks

International Technical Support Organization



© Copyright IBM Corp. 2004. All rights reserved.

Using a zFS Root File System



☐ Copy HFS root to zFS root

- Took about 10 minutes - about 2000 MB of data

```
#> zfsadm define -aggregate OMVS.ROOTCOMP.ZFS -storageclass SCCOMP
-megabytes 2000 100
IOEZ00248E VSAM linear dataset OMVS.ROOTCOMP.ZFS successfully created.
#> zfsadm format -aggregate OMVS.ROOTCOMP.ZFS -compat
IOEZ00077I HFS-compatibility aggregate OMVS.ROOTCOMP.ZFS has been
successfully created
#> mkdir -m 700 /u/zfs/clone
#> /usr/sbin/mount -f OMVS.ROOTCOMP.ZFS -t ZFS /u/zfs/clone
```

© Copyright IBM Corp. 2004. All rights reserved.

Make zFS the Root



- ❑ Messages issued for the IPL with a zFS root file system and another IPL for HFS file system as root
 - Showed no essential differences in time intervals from the start of OMVS for the following events:
 - zFS is active
 - The root file system is mounted
 - ETC file system is mounted
 - BPXOINIT has been started
 - OMVS initialization is completed

```
ROOT      FILESYSTEM('OMVS.ROOTCOMP.ZFS')
          TYPE(ZFS)                /* TYPE OF FILE SYSTEM          */
          MODE(RDWR)               /* (OPTIONAL) CAN BE READ OR RDWR. */
                                   /* DEFAULT = RDWR                */
```

© Copyright IBM Corp. 2004. All rights reserved.

DFSMSdss Dump and Restore



- ❑ DFSMS new function APARs for z/OS V1R4
 - OW57046 - Catalog flag for zFS aggregate
 - OW57015 - DFSMS catalog support
 - OW57141 - DSS support
 - OW57017 - Listcat support
 - OA02713 - HSM support
- ❑ Current method for backup without APARs - restore
 - 1. Quiesce the aggregate (this drains any activity and suspends any new requests)
 - 2. Backup the aggregate (and all the file systems)
 - 3. Unquiesce the aggregate (allowing zFS activity to continue)

- APARS allow DFSMS to automatically quiesce the aggregate
- Concurrent Copy can be used

© Copyright IBM Corp. 2004. All rights reserved.

Back up with DFSMSdss Logical Dump



☐ Back up the zFS aggregate (and all the file systems)

```
//ZFSBKUP1 JOB (OS390), 'PROGRAMMER', CLASS=A,
//          MSGCLASS=X, MSGLEVEL=(1,1)
//*-----
//* THIS JOB QUIESCES A ZFS AGGREGATE, DUMPS IT, THEN UNQUIESCES IT.
//*-----
//DUMP      EXEC PGM=ADDRSSU, REGION=4096K
//SYSPRINT DD  SYSOUT=*
//SYSABEND DD  SYSOUT=*
//OUT       DD  DSN=hlq.AGGR004.BACKUP,
//          DISP=(NEW,CATLG,DELETE), SPACE=(CYL,(5,1),RLSE)
//SYSIN      DD  *
DUMP DATASET(INCLUDE(hlq.ZFS.AGGR004)) -
CONCURRENT -
OUTDD(OUT)
/*
//
```

☐ DFSMSdss automatically quiesces the zFS aggregate before dumping the data set and unquiesces it when it is done

© Copyright IBM Corp. 2004. All rights reserved.

Restore Aggregate with DFSMSdss



- ☐ Restore aggregate with DFSMSdss logical restore
- ☐ Into a new aggregate OMVS.PRIV.AGGR005.LDS0005 if the original aggregate hlq.ZFS.AGGR004 still exists

```
//ZFSREST1 JOB (OS390), 'PROGRAMMER', CLASS=A,
//          MSGCLASS=X, MSGLEVEL=(1,1)
//*-----
//* THIS JOB RESTORES A ZFS AGGREGATE.
//*-----
//ZFSREST EXEC PGM=ADDRSSU, REGION=0M
//SYSPRINT DD  SYSOUT=*
//SYSABEND DD  SYSOUT=*
//INDS      DD  DISP=SHR, DSN=SUIMGUR.ZFS.DUMP1
//SYSIN      DD  *
RESTORE DATASET(INCLUDE(**)) -
CATALOG -
RENAMEU( -
(hlq.ZFS.AGGR004, -
OMVS.PRIV.AGGR005.LDS0005) -
) -
WRITECHECK -
INDD(INDS)
```

© Copyright IBM Corp. 2004. All rights reserved.

zFS Recovery



- ❑ zFS is a logging file system
 - It logs metadata updates
 - On a system crash, the log is replayed to bring the file system to a consistent state
- ❑ I/O requests are started immediately (asynchronously) so on a system crash, most data is already on disk
- ❑ Salvager utility - provides aggregate recovery
 - **-recoveryonly** - Recover the specified aggregate by replay of log of metadata changes
 - **-verifyonly** - Verify the aggregate structure to determine if it contains any inconsistencies and report
 - **-salvageonly** - Salvage the aggregate and attempt to repair any inconsistencies it finds

© Copyright IBM Corp. 2004. All rights reserved.

zFS Recovery Processing



- ❑ Recovery is the replaying of the log on the aggregate
- ❑ The log records all changes made to metadata as a result of operations such as:
 - File creation and deletion
- ❑ Problems that can occur:
 - Basic structure of the aggregate bad
 - Log mechanism is damaged
 - Storage medium of the aggregate is suspect
- ❑ Use the `ioeagslv` utility to verify or repair the structure of the aggregate

© Copyright IBM Corp. 2004. All rights reserved.

Salvager Repairs



- ❑ Not all aggregates can be salvaged as follows:
 - Extensive damage to the structure of the metadata
 - Damage to the physical disk
 - Aggregate inconsistencies
 - Damage to user data on an aggregate
 - Problems that modified the contents of a file but did not damage the structure of an aggregate or change the metadata of the aggregate

© Copyright IBM Corp. 2004. All rights reserved.

zFS Asserts - Current Support



- ❑ If a zFS routine is called and a pointer to some information is passed to the routine, the routine might check that the pointer is non-zero - these are called asserts
 - Currently, zFS has asserts (2C3 ABEND) that occur when internal logic problem is detected
 - EOM ABEND conditions are not recovered
- ❑ In these two cases, zFS Abends and goes down causing zFS file systems to be moved (or unmounted) and applications to fail

© Copyright IBM Corp. 2004. All rights reserved.

zFS Conditional Asserts - z/OS v1R6



- ❑ zFS has modified the asserts to conditional asserts
- ❑ This means that zFS will Abend but will not go down
- ❑ The scope of the failure is limited to an aggregate where possible (we may disable the aggregate)
 - For some asserts, an aggregate may be marked disabled for writing
 - This means that the aggregate will need to be detached and reattached before it can be written to
 - Other aggregates can be used

IOEZ00337E zFS kernel: non-terminating exception 2C3 occurred, reason
EA8001B7 abend psw 70C1000 8B4CCD46
IOEZ00422E Aggregate PLEX.JMS.AGGR001.LDS0001 disabled for writing

© Copyright IBM Corp. 2004. All rights reserved.

zFS Conditional Asserts - z/OS v1R6



- ❑ Sometimes, a disabled aggregate will not be able to be detached
- ❑ You can still run with other aggregates
- ❑ A disabled aggregate that could not be detached will be unavailable until stop and restart of zFS

IOEZ00433E Internal error, aggregate PLEX.JMS.AGGR001.LDS0001
cannot be detached from zFS

© Copyright IBM Corp. 2004. All rights reserved.

zFS Conditional Asserts - z/OS v1R6



- ❑ In order to help with hang conditions involving zFS
 - F ZFS,HANGBREAK - operator command
 - Causes waiting users to receive error
 - Dumps/failures may occur
 - Aggregates may get disabled
 - Users may get hung again
- ❑ zFS will also provide recovery code for some EOM failures
- ❑ This means that the operation in progress will either complete or backout and zFS will continue to run
- ❑ For EOM cases not handled, zFS will still Abend and go down

© Copyright IBM Corp. 2004. All rights reserved.

ibm.com



Shared HFS Support



Redbooks

International Technical Support Organization

© Copyright IBM Corp. 2004. All rights reserved.

zFS Supports the shared HFS sysplex



- ❑ Users can access zFS data that is owned by another system in the sysplex
- ❑ zFS file systems are automoved on system failure
- ❑ Only systems running zFS see zFS file systems
- ❑ zFS compatibility mode file systems can be automoved and automounted
 - Only to a system where zFS is running
- ❑ File systems in multi-file system aggregates are not fully supported in a shared HFS environment
- ❑ The IOEFSPRM file cannot be shared across systems in a sysplex when the file contains:
 - A multi-file system aggregate specification, or
 - A msg_output_dsn specification, or
 - A trace_dsn specification

© Copyright IBM Corp. 2004. All rights reserved.

System Symbols in IOEFSPRM File



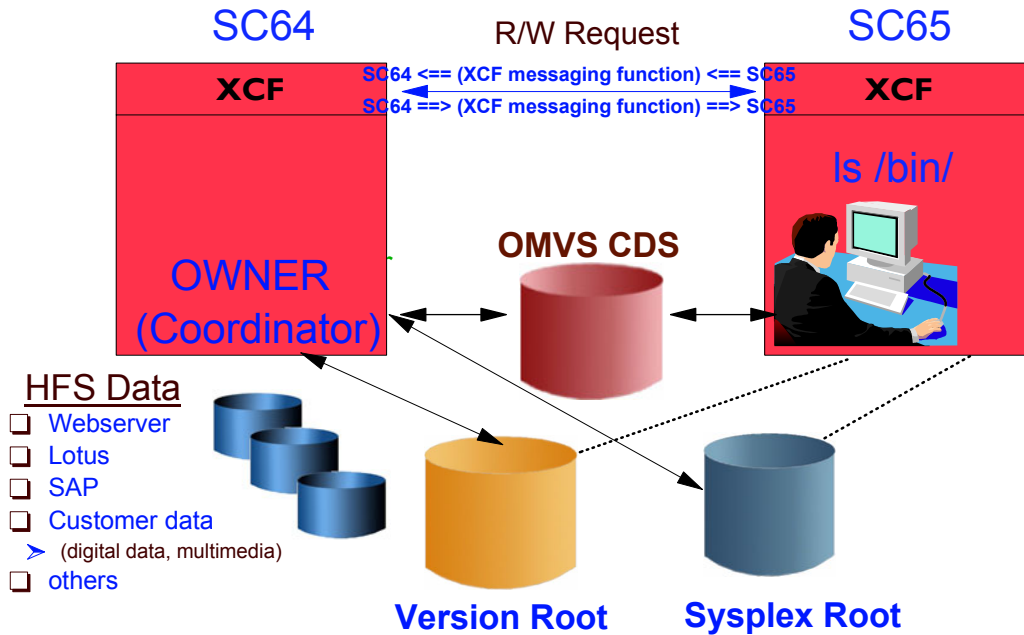
- ❑ With V1R4, system symbols can be specified for data set names in IOEFSPRM
 - Makes possible to share IOEFSPRM file in sysplex
- ❑ Examples:

```
msg_output_dsn=HLQ.&SYSNAME..ZFS.MSGOUT
trace_dsn=HLQ.&SYSNAME..ZFS.TRACEOUT

define_aggr cluster(HLQ.&SYSNAME..AGGR001)
```

© Copyright IBM Corp. 2004. All rights reserved.

Accessing Shared Sysplex File Systems



© Copyright IBM Corp. 2004. All rights reserved.

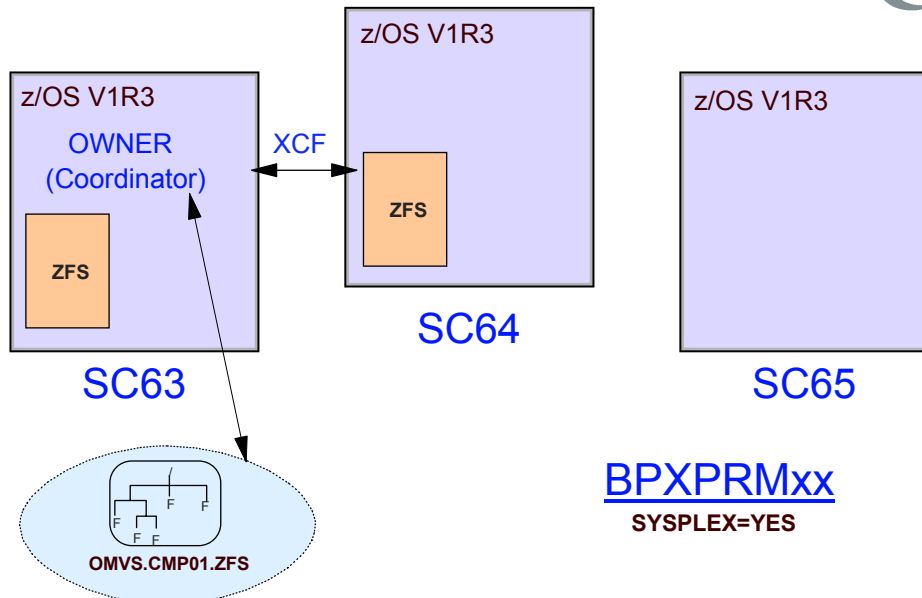
Sysplex Rules



- ☐ Only systems running zFS see zFS files
- ☐ Compatibility mode aggregates
 - Support AUTOMOVE and automount
- ☐ Multi-file mode aggregates
 - Mount NOAUTOMOVE
- ☐ All systems at V1R2+ - All or some running zFS
 - System running zFS goes down
 - File systems mounted NOAUTOMOVE are lost

© Copyright IBM Corp. 2004. All rights reserved.

zFS Sysplex Sharing



For full sysplex support, zFS must be running on all systems in the sysplex and all zFS file systems must be compatibility mode file systems (that is, they cannot be file systems in multi-file system aggregates).

© Copyright IBM Corp. 2004. All rights reserved.

ibm.com



e-business

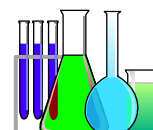


z/OS V1R4 AUTOMOVE System List



Redbooks

International Technical Support Organization



© Copyright IBM Corp. 2004. All rights reserved.

Automove System List



- ❑ In a shared sysplex environment, AUTOMOVE(YES) on MOUNT moves the ownership of the filesystem to some other system in the sysplex if the current server system for that filesystem is brought down
- ❑ The system that becomes the new server is random
- ❑ z/OS V1R4 provides the capability to specify which system or systems in a sysplex will takeover as server for a filesystem
 - A system list is added to the AUTOMOVE parameter for MOUNT
 - The list begins with either include or exclude, (abbreviated i or e) followed by a list of system names

© Copyright IBM Corp. 2004. All rights reserved.

Automove Parameter for Mounts



- ❑ BPXPRMXX parmlib member MOUNT statement or TSO/E MOUNT command
- ❑ Shell mount command
- ❑ ISHELL panels for mounts
- ❑ C program, assembler program, or REXX program
 - AUTOMOVE(indicator, name1, name2,.....)
 - **i** - Provides a prioritized list of systems where the file system may be moved - If no system can takeover as the new owner, the file system is unmounted
 - **e** - This system list provides a list of systems to where the file system may not be moved

© Copyright IBM Corp. 2004. All rights reserved.

Select the attribute to change panel



Main ISHELL Panel -> File_systems pulldown menu
-> Option 1 - Mount Table -> Modify(M)

Work with Mounted File Systems

Select the attribute to change

Select the attribute to change:

1. Change mount mode to R/O
2. Change Owning system from SC65
3. Change automove attribute...

New owning system _____

F1=Help F3=Exit F6=Keyshelp F12=Cancel

OMVS.SC63.XML.HFS Available
OMVS.SC64.USRLCAL Available
OMVS.SYSLOGD.HFS Available
OMVS.TWS810.TWCTP.HFS Available
WTSCPLX2.DOMINO.HFS Available
WTSCPLX2.JAVA118.V001122.HFS Available
WTSCPLX2.JAVA213.V010424.HFS Available
WTSCPLX2.JAVA213.V010714.HFS Available
WTSCPLX2.JAVA213.V010926.HFS Available
WTSCPLX2.MQSERIES.MA88.HFS Available
WTSCPLX2.SC63.SYSTEM.HFS Available
WTSCPLX2.SC64.SYSTEM.HFS Available
WTSCPLX2.SC65.SYSTEM.HFS Available
WTSCPLX2.SYSPLEX.ROOT Available

© Copyright IBM Corp. 2004. All rights reserved.

Set automove attribute



Select 4 or 5

Work with Mounted File Systems

Select the attribute to change

Set automove attribute

Select the automove attribute:

1. Yes
2. No
3. Unmount
4. Include systems
5. Exclude systems

System names for Include or Exclude

Code system names

F1=Help F3=Exit F6=Keyshelp F12=Cancel

WTSCPLX2.JAVA213.V010926.HFS Available
WTSCPLX2.MQSERIES.MA88.HFS Available
WTSCPLX2.SC63.SYSTEM.HFS Available
WTSCPLX2.SC64.SYSTEM.HFS Available
WTSCPLX2.SC65.SYSTEM.HFS Available
WTSCPLX2.SYSPLEX.ROOT Available

Change an Automove System List



☐ setomvs command

- setomvs filesys,filesystem='omvs.cmp01.zfs',automove=e(sc64)

☐ chmount shell command

- chmount -a i,SC64,SC65 /tmp/test

☐ Display automove system list

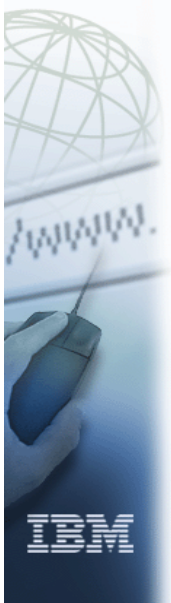
- d omvs,file

```
d omvs,file
```

```
HFS          126 ACTIVE          RDWR
NAME=OMVS.CMP01.ZFS
PATH=/SC63/tmp/test1
OWNER=SC63 AUTOMOVE=I CLIENT=N
INCLUDE SYSTEM LIST:  SC64      SC65
```

© Copyright IBM Corp. 2004. All rights reserved.

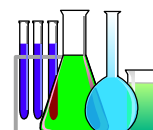
ibm.com



z/OS V1R6 LFS support of sysplex zFS – PFS termination in Shared HFS



Redbooks
International Technical Support Organization



© Copyright IBM Corp. 2004. All rights reserved.

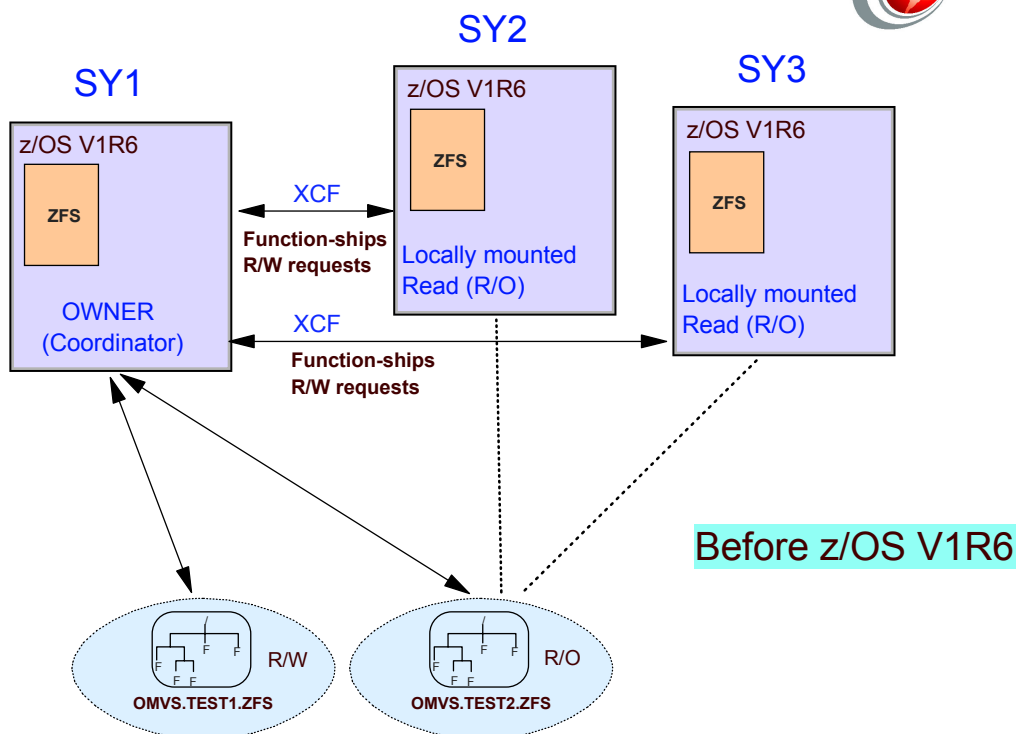
LFS Sysplex Support



- ❑ LFS termination of a PFS, such as zFS
 - Improves availability of file systems on the system where a PFS is terminating
- ❑ **Before:** PFS termination is that file systems for the terminating PFS, and subtrees of those file systems, get moved to another system (if locally owned), and then get locally unmounted and become unavailable on the system where the PFS is terminating - If they could not be moved, then they become globally unmounted.
- ❑ **z/OS V1R6:** If ownership of these file systems can be moved to another system in the sysplex, and then allow for function-shipping requests on the system where a PFS is terminating and avoid the local unmounts -This provides improved availability of file systems

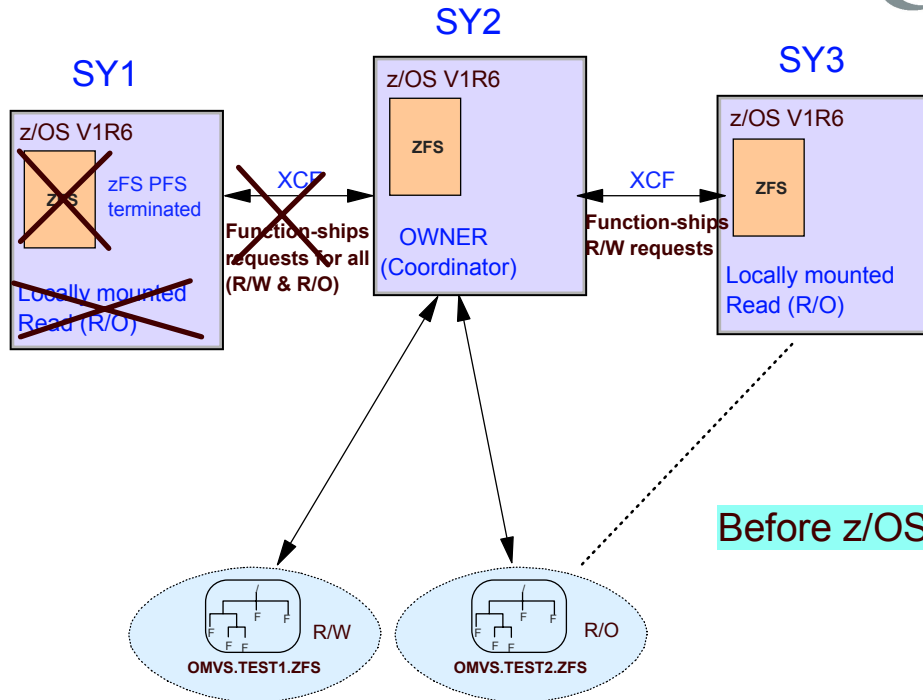
© Copyright IBM Corp. 2004. All rights reserved.

Three systems in a USS sharing environment accessing two file systems



© Copyright IBM Corp. 2004. All rights reserved.

Same environment after changing the ownership to another system



© Copyright IBM Corp. 2004. All rights reserved.

z/OS V1R6 Design



- ❑ Ownership of file systems can be moved to another system in the sysplex, and then allow for function-shipping requests on the system where a PFS is terminating and avoid the local unmounts
 - Improves availability of file systems
- ❑ If the file system is sysplex-aware (locally mounted), but not owned by the system where the PFS is terminating, then the file system will be converted to function-shipping to the owner (no move occurs).

Sysplex-aware - Capable of mounting locally on the systems - For example, R/O on all zFS file systems

Sysplex-unaware - Not capable of mounting locally on all systems - Function ships the request to owner - For example, R/W zFS file systems

© Copyright IBM Corp. 2004. All rights reserved.

z/OS V1R6 Design

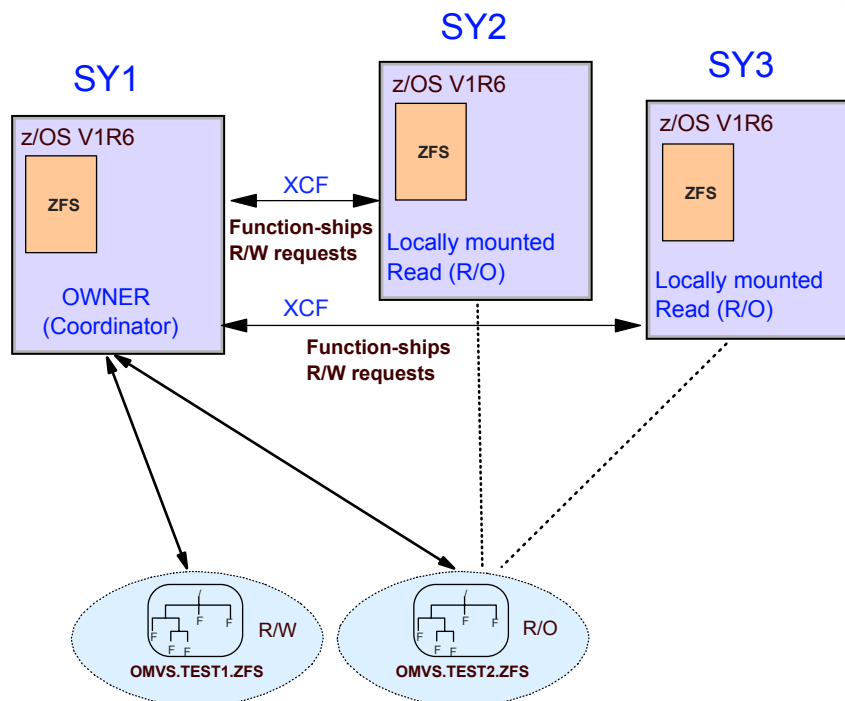


- If the file system is sysplex-aware (locally mounted), but not owned by the system where the PFS is terminating, then the file system is converted to function-shipping to the owner (no move occurs)
- If the reply to re-start the PFS is "I" (Do not restart the PFS), then the file systems are locally unmounted as before
- If the reply to re-start the PFS is "R", then any sysplex-aware file systems converts back from function-shipping to local mount - sysplex-unaware file systems remain function-shipping to the current owner

nn BPXF032D FILESYSTYPE ZFS TERMINATED. REPLY 'R' WHEN READY TO RESTART

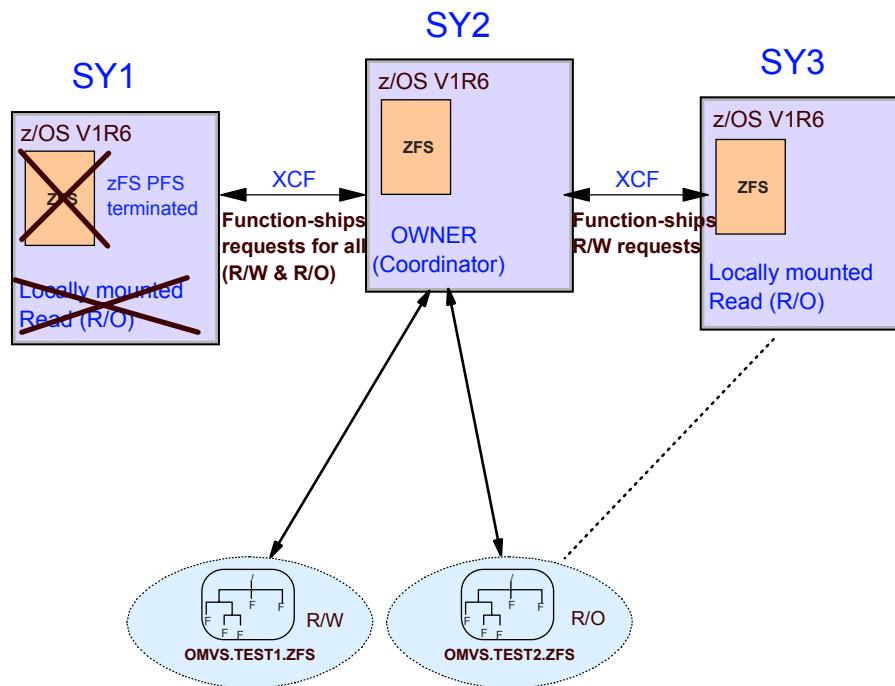
© Copyright IBM Corp. 2004. All rights reserved.

Three systems in a USS sharing environment accessing two file systems



© Copyright IBM Corp. 2004. All rights reserved.

Same sharing environment after changing the ownership to another system



© Copyright IBM Corp. 2004. All rights reserved.

AUTOMOVE Behavior



- ❑ MOUNT allows AUTOMOVE(YES) or AUTOMOVE(UNMOUNT)
- ❑ If AUTOMOVE(NO) or if an automove syslist is specified, it changes to AUTOMOVE(YES) and a new message BPXF234I is issued
- ❑ Remount will not change the AUTOMOVE setting -So a remount from R/W to R/O when the AUTOMOVE is NO, will not change it to AUTOMOVE(YES), even it is now sysplex-aware
- ❑ PFS termination ignores AUTOMOVE(NO) or AUTOMOVE(UNMOUNT) if sysplex-aware, and will go ahead and try to move ownership and then perform a local to function-ship conversion

© Copyright IBM Corp. 2004. All rights reserved.

AUTOMOVE Behavior



- ❑ Move to any systems in the sysplex (SYSNAME=*) will ignore a automove syslist if sysplex-aware and will consider all systems as move candidates - It has always ignored AUTOMOVE(NO) and AUTOMOVE(UNMOUNT) if sysplex-aware
- ❑ Dead system recovery and takeover has always ignored AUTOMOVE(NO) and AUTOMOVE(UNMOUNT) for sysplex-aware, and has still attempted to have all systems try takeover - Honors the automove syslist regardless of sysplex-awareness - Now ignores automove syslist as well if sysplex-aware, and will allow all systems to try takeover
- ❑ For sysplex-aware, if no system could take it over, AUTOMOVE(UNMOUNT) will unmount the file system and its subtree, but for AUTOMOVE(NO) or syslist it will become unowned

© Copyright IBM Corp. 2004. All rights reserved.

z/OS V1R6 AUTOMOVE Handling Change



- ❑ A move of a file system that is either AUTOMOVE(NO) or has Automove Syslist to a new z/OS V1R6 owner
 - Changes to AUTOMOVE(YES), and issue BPXF234I
 - This is true for manual move and file system Dead System Recovery and unowned file system takeover processing

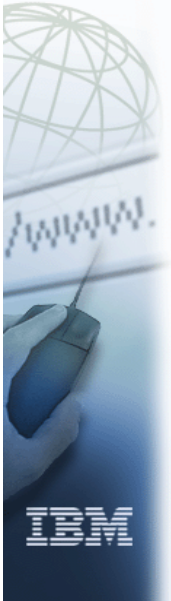
BPXF234I "FILE SYSTEM OMVS.TEST1.ZFS WAS MOUNTED WITH AUTOMOVE(YES)"

© Copyright IBM Corp. 2004. All rights reserved.

ibm.com



e-business



Distributed File Service zSeries File System (zFS)



Redbooks

International Technical Support Organization



© Copyright IBM Corp. 2004. All rights reserved.

Duplicate File System Names



- ☐ File system names must be unique among all attached aggregates on a system
- ☐ It is possible to create the same file system name on two different aggregates by not having them attached at the same time
- ☐ When the second aggregate is attached, a error message occurs and the duplicate file system is unavailable

© Copyright IBM Corp. 2004. All rights reserved.

Duplicate File System Names



- ❑ With V1R4, a new config option allows duplicate file system names (in different aggregates)
 - **allow_duplicate_filesystems=on**
 - **filesystems=off** is the default
 - IOEZ00097E File system ZFSA already exists.
- ❑ Commands that specify zFS file systems can specify aggregate name to qualify it
- ❑ If file system name is ambiguous, (if duplicate and aggregate not specified) the command fails
 - IOEZ00314E The file system name ZFSA is not unique. Its aggregate name must also be specified.

© Copyright IBM Corp. 2004. All rights reserved.

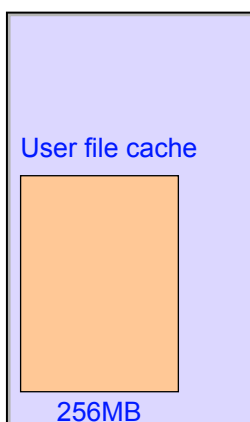
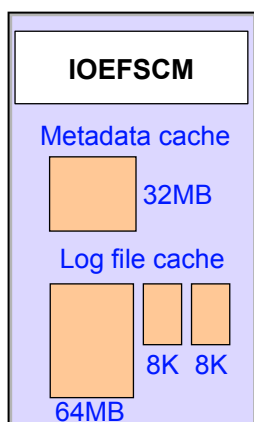
Pre-V1R4 Cache



Default sizes shown

zFS address space

Dataspace



- ❑ Metadata cache
 - All directory contents
 - File status information
 - File system structures
 - Also caching of data for files smaller than 7K

© Copyright IBM Corp. 2004. All rights reserved.

Metadata Backing Cache - z/OS V1R4



- ☐ With V1R4, a new backing cache contains an extension to the meta cache and resides in a dataspace - Specify in IOEFSPRM file
 - metaback_cache_size=64M, fixed
 - Values allowed: 1 MB to 2048 MB
- ☐ Used as a "paging" area for metadata
- ☐ Allows a larger meta cache for workloads that need large amounts of metadata
- ☐ Only needed if meta cache is constrained

© Copyright IBM Corp. 2004. All rights reserved.

ibm.com



HFS to zFS Migration ITSO Redbook Tool



Redbooks
International Technical Support Organization



© Copyright IBM Corp. 2004. All rights reserved.

ITSO zFS Redbook



- ❑ z/OS Distributed File Service zSeries File System Implementation, SG24-6580-01 - (z/OS V1R6)
 - Publication - 4Q 2004
- ❑ Redbook migration tool published in book
 - Download from Redbook website
- ❑ Current Redbook SG24-6580-00 has:
 - JCL for using COPYTREE and PAX for migrations

© Copyright IBM Corp. 2004. All rights reserved.

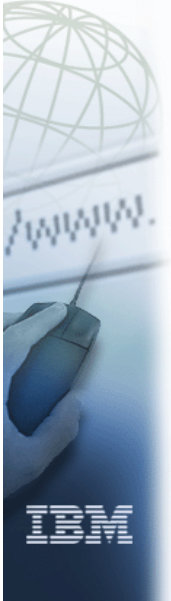
HFS to zFS Migration



- ❑ Input: List of source data sets (HFS data sets)
 - Tool (REXX) creates a list of HFS data sets with
 - Suggested settings for the target zFS aggregates
 - You may change settings (storage group, size, etc.)
- ❑ Tool provides a final input data set for a job that performs the migration process
- ❑ Tool has option to determine whether you want to replace the old HFS with the new zFS if HFS is currently mounted

© Copyright IBM Corp. 2004. All rights reserved.

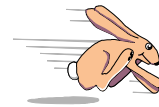
ibm.com



zFS vs HFS Performance



Redbooks
International Technical Support Organization



© Copyright IBM Corp. 2004. All rights reserved.

Performance Monitoring APIs - z/OS V1R6



- ❑ zFS provides six new pfscctl APIs to retrieve performance counters
 - Locks
 - Storage
 - User data cache
 - iocounts
 - iobyaggr
 - iobydasd
- ❑ zFS provides a new zfsadm command (query) to query/reset the performance counters

zfsadm query [-locking] [-storage] [-usercache] [-iocounts]
[-iobyaggregate] [-iobydasd] [-reset]

© Copyright IBM Corp. 2004. All rights reserved.

HFS vs zFS Comparison



- ☐ Choose either the zFS or the HFS file system
- ☐ Define the number of processes which run in parallel to read or write 1 MB blocks of data.
- ☐ Specify the number of I/Os that each process performs
- ☐ Select the percentage of reads among all I/Os. R70 means that 70% of the I/Os are reads and 30% are writes
- ☐ Provide a seed value to create predictable random numbers for offsets into the large file when doing an I/O access

© Copyright IBM Corp. 2004. All rights reserved.

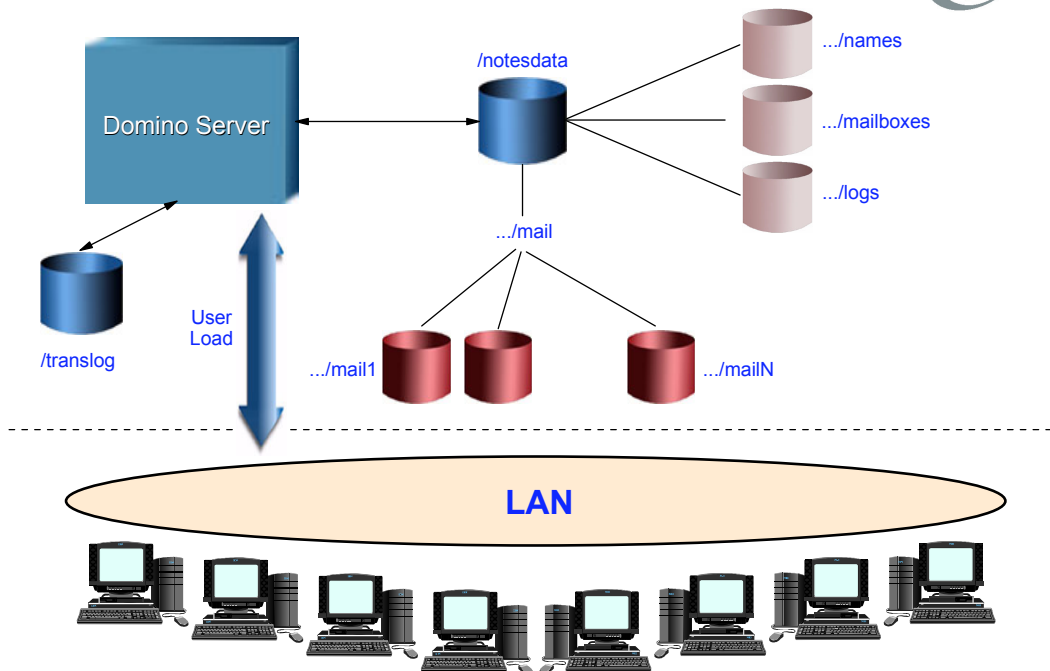
HFS vs zFS Comparison Results



	Processes	AVG secs	MIN secs	MAX secs
HFS	10	512.72	507.39	516.31
ZFS1	10	261.64	249.47	268.75
ZFS2	10	230.73	212.76	240.99
HFS	20	992.99	972.35	1002.94
ZFS1	20	478.90	455.40	496.29
ZFS2	20	470.53	438.43	487.82
HFS	40	1964.11	1869.39	1998.09
ZFS1	40	944.53	894.27	979.17
ZFS2	40	961.50	869.07	994.06

© Copyright IBM Corp. 2004. All rights reserved.

Domino File System Structure



© Copyright IBM Corp. 2004. All rights reserved.

Database used in the Study



- IBM US Directory
 - Compacted to On Disk Structure 41 format
 - 1.24 GB is size
 - 191,000 person documents
 - 1850+ groups
 - 1900+ connection documents

© Copyright IBM Corp. 2004. All rights reserved.

Conclusions



- ❑ Domino servers perform better with zFS in all operations
 - Client experiences higher throughput and lower elapsed times for:
 - Notes client access
 - Access via web browser
- ❑ zFS caching appears to be the key for performance
- ❑ zFS drives only about 1/3 of Read I/O to DASD

© Copyright IBM Corp. 2004. All rights reserved.