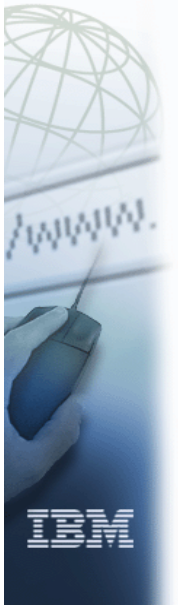
ibm.com

e-business



z/OS UNIX and zFS

Redbooks

International Technical Support Organization

Trademarks

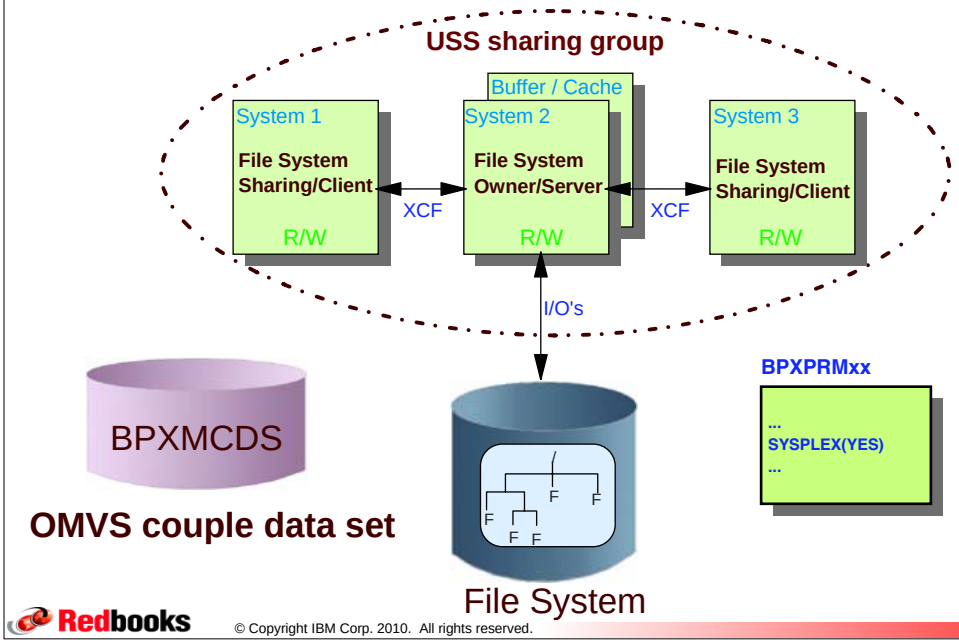
eNetwork	DFSMS/MVS	IMS	RMF
geoManager	DFSMSdfp	IMS/ESA	RS/6000
AD/Cycle	DFSMSdss	IP PrintWay	S/390
ADSTAR	DFSMShsm	IPDS	S/390 Parallel Enterprise Server
AFP	DFSMSrmm	Language Environment	SecureWay
APL2	DFSORT	Multiprise	StorWatch
APPN	Enterprise System 3090	MQSeries	Sysplex Timer
BookManger	Enterprise System 4381	MVS/ESA	System/390
BookMaster	Enterprise System 9000	Network Station	System REXX
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	IBM	QMFr	z/OS
	IBM System z	RACF	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 Manger (Tivoli Systems Inc.)
UNIX (X/Open Company Limited)
Windows (Microsoft Corporation)
Windows NT (Microsoft Corporation)

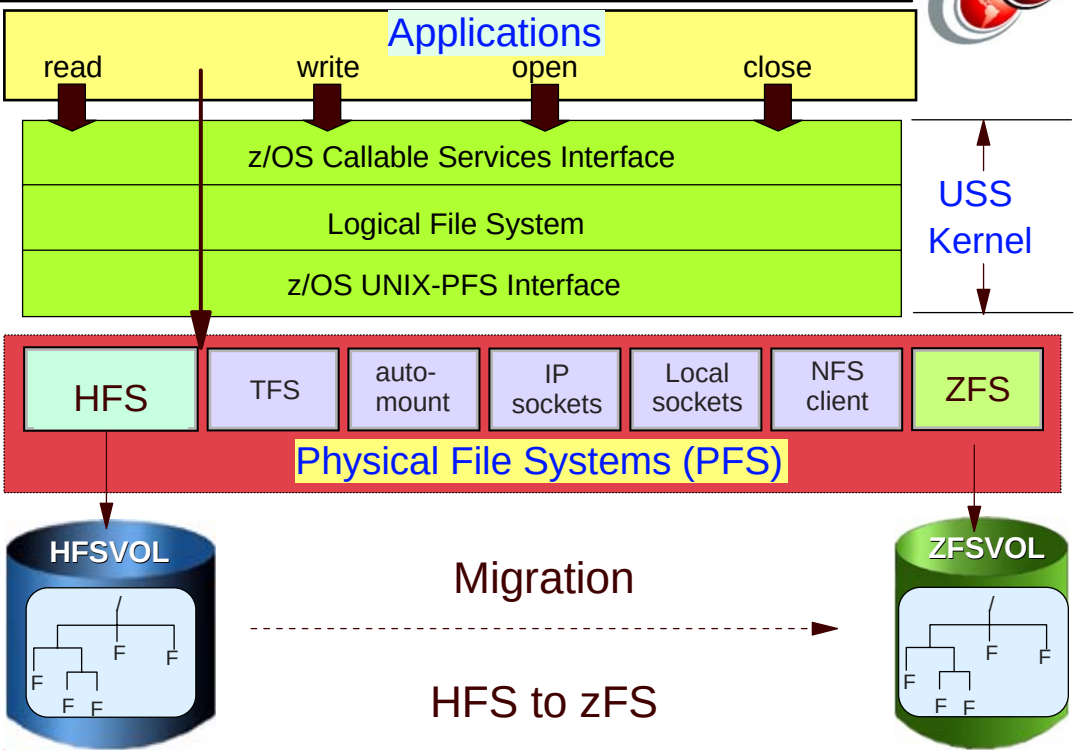
Redbooks

Shared File System Support in a Sysplex

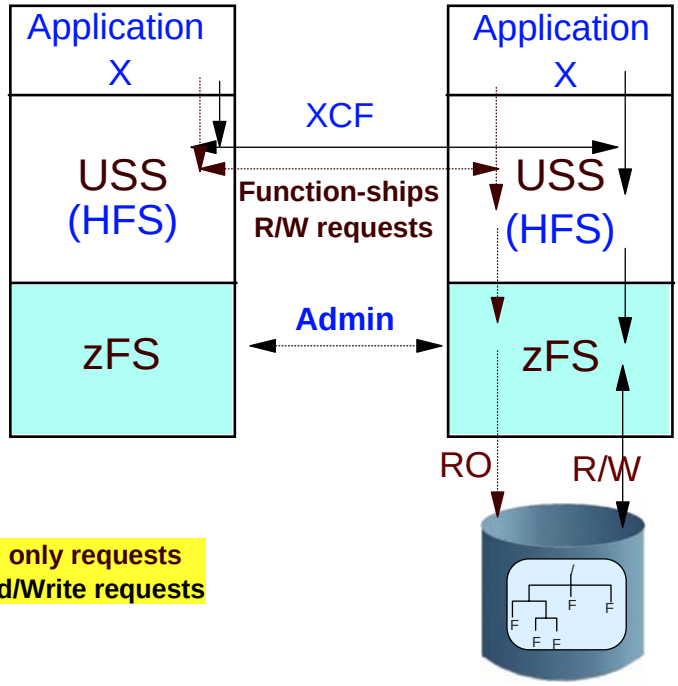
- OS/390 V2R9 - First file sharing - (Only HFS)
- zFS supports shared file systems - (z/OS V1R2)



zFS & HFS Physical File Systems (PFS)



Sysplex File Sharing - Function-Shipping



zFS Sharing Mode Terminology

- Sysplex-unaware
  - A filesystem is sysplex-unaware if the PFS supporting that file system requires it to be accessed through the remote owning system from all other systems in a sysplex (allowing only one connection for update at a time) for a particular mode (read-only or read-write)
  - Access from a system not being the owner (sometimes referred to as being a client) is provided through XCF communication to the owning system (sometimes called “function shipping”)
    - This is controlled by z/OS UNIX and the interface is also known as a PFS named XPFS (“Cross System PFS”)

zFS Filesystems Sysplex-aware

- Beginning with z/OS V1R11, zFS introduced the ability to be able to enable zFS read-write filesystems to be sysplex-aware
  - This means that USS sends all file requests directly down to the local zFS physical file system (PFS)
  - It does not “function ship” the requests
- For z/OS V1R11 - APAR OA29712 updates the support that was shipped with V1R11
  - USS SPE OA29712 is a pre-requisite for the zFS SPE OA29619
- z/OS V1R12 contains OA29712 making V1R11 and V1R12 at equivalent levels for zFS

APAR OA29619

- APAR OA29619 provides a new function that allows users to specify which zFS read-write filesystems are to be made sysplex-aware
- To enable zFS sysplex-aware on a file system basis, specify sysplex=filesys in the IOEFSPRM configuration file
- Leave the sysplex_filesys_sharemode with its default of norwshare
- You can specify it in a shared IOEFSPRM configuration file and each system picks up the specification in a rolling IPL - The sysplex option is ignored by previous releases

New zFS Configuration Options (OA29619)

- Options changed with V1R11 and **OA29619**
  - sysplex = {on | off | **filesys**} - Default is **OFF**
  - **sysplex_filesys_sharemode = norwshare | rwshare**
  - sysplex_admin_level = 0 | 1 | 2 | 3
  - token_cache_size = vnode_cache_size x 2
  - file_threads = 40
  - client_cache_size =128M --------> Cache
  - client_reply_storage = **10M ----> (changed from 40M)**
  - recovery_max_storage = 256M

Underlined options are with OA29619

zFS Sysplex State - (sysplex_state)

- The sysplex-state of the sysplex
  - (0) indicates that zFS is not in a shared file system environment (normal for V1R6 and prior releases)
  - (1) indicates that zFS is in a shared file system environment (normal for V1R7 and above in a shared file system environment) - This also indicates that zFS is non-sysplex aware for RW filesystems; that is either sysplex=off is specified or is the default
  - (2) indicates that zFS is running in a sysplex-aware environment with sysplex=on
  - (3) indicates that zFS is running in a sysplex-aware environment with sysplex=filesys (With APAR 29619)

```
ROGERS @ SC74:/u/rogers>zfsadm configquery -sysplex_state
IOEZ00317I The value for configuration option -sysplex_state is 3.
```

Admin Levels in a Mixed Sysplex

- zFS in V1R9 or V1R10 with sysplex_admin_level=1
  - Uses and follows the XCF Admin protocol among systems with level 0 and 1 and does not initialze if a system with XCF Admin level 3 is active in the sysplex
- zFS in V1R9 or V1R10 with sysplex_admin_level=2
  - Uses and follows the new XCF Admin protocol among systems with levels 2 or 3 and does not initialze if a system with XCF Admin level 0 is active in the sysplex
- zFS in R11 and R12 running with sysplex_admin_level=3
  - By default uses and follows the new XCF Admin protocol among systems with levels 2 or 3 and does not initialze if a system with XCF Admin level 0 or 1 is active in the sysplex

zFS Admin Levels with z/OS V1R12

- **0** - Admin level of zFS in z/OS V1R9 or R10 without APAR OA25026
- **1** - Default admin level interface used in R9 and 10 when APAR OA25026 is applied, also called the conditioning level
- **2** - Used for zFS R11 that allows zFS V1R9 and V1R10 to run together with zFS V1R11 and z/OS V1R12 (toleration mode)
- **3** - This is the zFS Admin level of V1R11 - Any specification of parameter sysplex_admin_level in V1R11 is ignored as zFS in V1R11 and higher always runs at level **3**

Sysplex=filesys

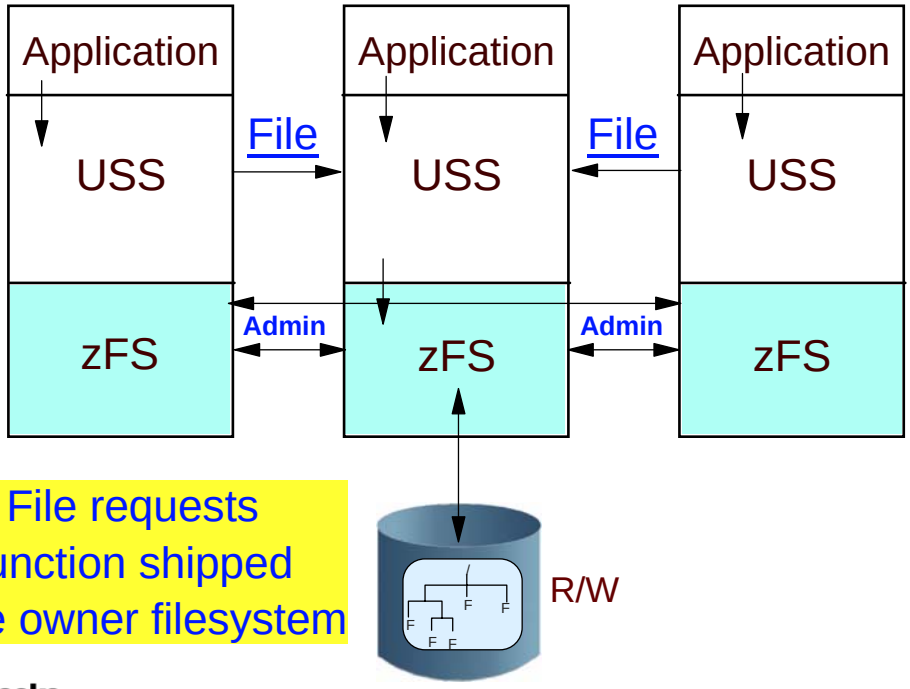
- **sysplex=filesys** - zFS can be configured sysplex=filesys to allow some zFS read-write mounted file systems owned on that system to be sysplex-aware file systems and some to be non-sysplex aware file systems
  - When you run sysplex=filesys, the zFS PFS runs sysplex-aware, but each zFS file system is mounted non-sysplex aware (by default)
  - zFS is enabled to allow zFS read-write file systems to be sysplex-aware but it must be explicitly requested on a file system basis

Do not make any zFS read-write file systems sysplex-aware until you have all systems in the shared file system environment at z/OS V1R11 with sysplex=filesys active. To make a zFS read-write file system sysplex-aware when running sysplex=filesys on all systems, you must unmount the file system, specify RWSHARE as a mount PARM and then mount the file system

zFS Filesystems Mounted Sysplex-unaware

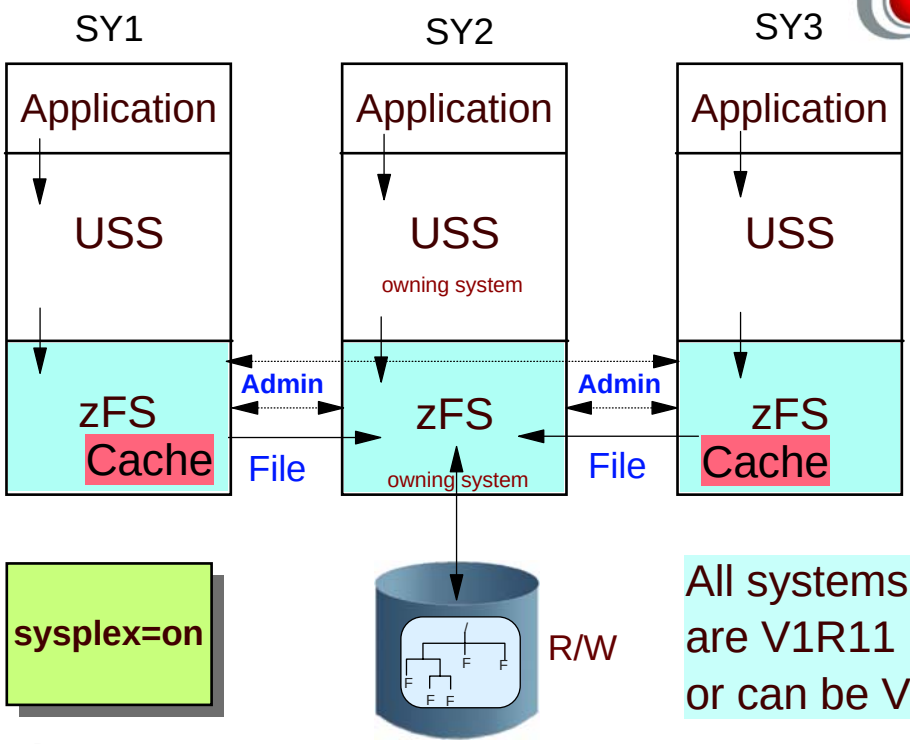
- If R/W mounted zFS file systems in V1R10 or older or in V1R11 and V1R12 when zFS has been started with sysplex=off
- One system is owning system, other systems are “clients”
- A filesystem is locally mounted on owning system, but also externally mounted/available on all systems in the sysplex
- The PFS supporting that filesystem requires it to be accessed through the remote owning system from all other systems in a sysplex (allowing only one connection for update at a time) for a particular mode (RO or R/W)
- The system that connects to the file system is called the file system owner - Other system’s access is provided through XCF communication with the filesystem owner

Defining zFS Sysplex-unaware



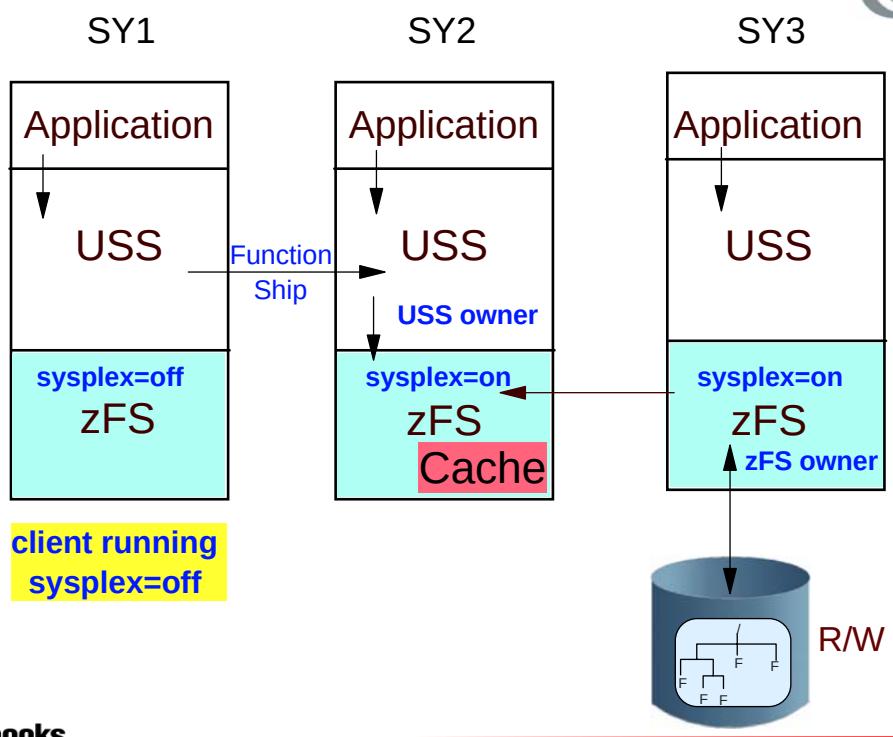
File - File requests are function shipped to the owner filesystem

Defining zFS as Sysplex-aware



All systems are V1R11 or can be V1R12

zFS Mixed Environment

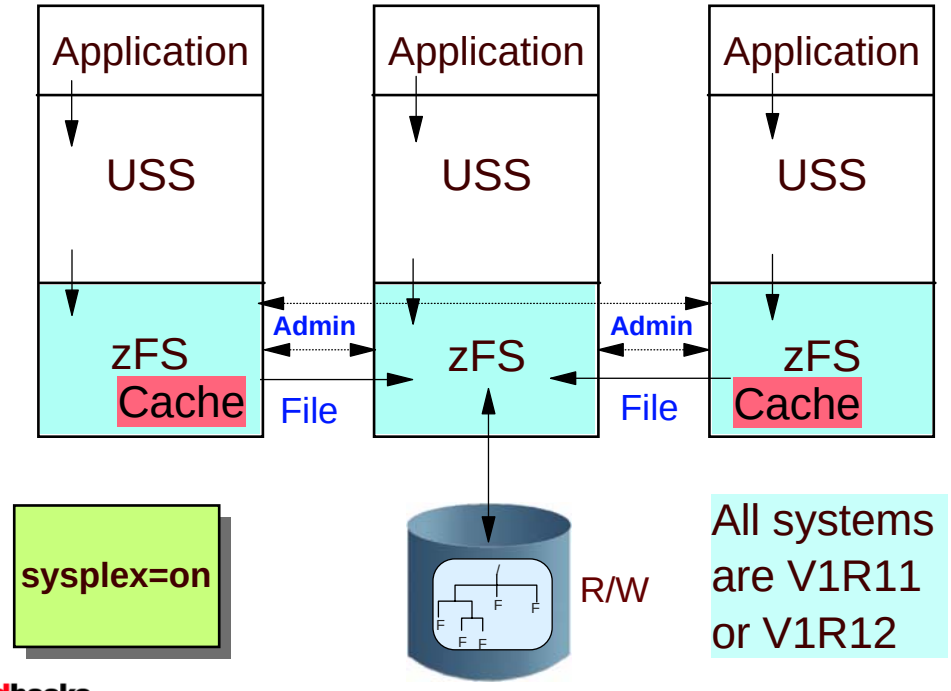


Sysplex-aware File System with V1R11

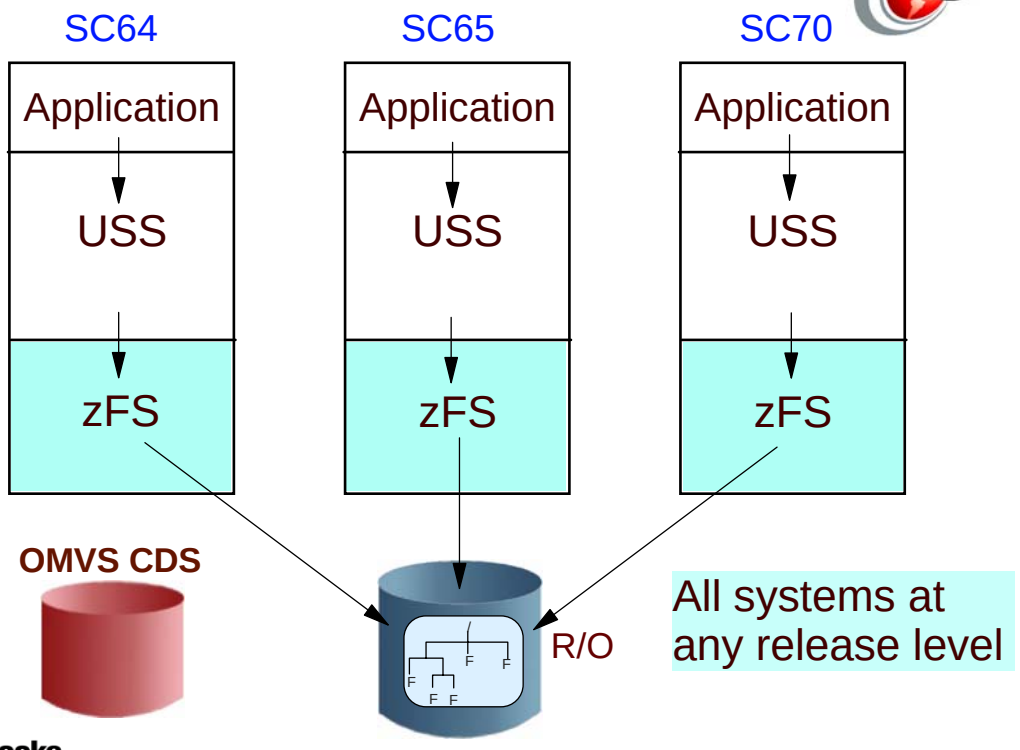
- zFS is sysplex-aware for file systems mounted read-write (new with zFS V1R11)
  - This means that USS sends all file requests directly down to the local zFS physical file system (PFS)
  - It does not “function ship” the requests
  - The local zFS either sends the request to the owning zFS system or it satisfies the request from the local zFS cache
  - In many cases, this improves the pathlength over the function shipping model

Do not make any zFS read-write file systems sysplex-aware until you have all systems in the shared filesystem environment at z/OS V1R11 with sysplex=filesys active

Defining zFS as Sysplex-aware



Sysplex-aware File Systems (read-only)



Sysplex-aware for zFS File Systems

- zFS PFS now allows a file system to be locally accessed on all systems in a sysplex for a particular mode (R/W)
  - Then the zFS PFS is sysplex-aware for that mode
- New option ----------> sysplex=on
  - Define in the IOEZPRM file - or
  - Define in the IOEPRMxx parmlib member
- If SYSPLEX(YES) is specified in the BPXPRMxx parmlib member, this sysplex configuration option controls whether zFS runs sysplex-aware or not

zFS Cache Management

- When the zFS PFS is sysplex-aware
  - z/OS UNIX now sends requests to the local zFS and zFS takes on the responsibility to forward the request, if necessary
  - For many requests, it is not necessary because zFS caches data locally and can satisfy the request from its cache
  - Cache consistency is maintained through a token management mechanism and may provide improved performance in a shared file system environment

zFS Cache Management

- When a request is received by the local zFS PFS, zFS determines if the request can be satisfied from its cache
- If it is a read or lookup or readdir and the data is contained in the cache, the request can be satisfied without communication with the zFS owning system
- Write requests are written forwarded to the owner, unless we know the space is already allocated on disk
- Caching improves performance

z/OS V1R11 and sysplex=on Summary

- zFS is sysplex-aware
- USS sends requests directly to local zFS PFS, regardless of USS owner
- zFS decides whether it needs to forward the request to the zFS owning system
- zFS caching and tokens allow it to sometimes avoid XCF communications
- zFS moves zFS aggregates on system failure and to minimize XCF processing
- Only compatibility mode zFS aggregates are supported

Maintenance Levels Needed

- Before you can run zFS sysplex-aware on a file system basis, you must have z/OS V1R11 zFS
  - With APAR OA29619 and z/OS V1R11 UNIX APAR OA29712 installed and active on all of your LPARs
- V1R11 systems: - In addition, conditioning APAR OA29786 must be installed and active on all your LPARs
- z/OS V1R9 and V1R10: Finally, if you use the SMB server, APAR OA31112 must also be installed.

zFS Sysplex-aware on a Filesystem Basis

- Specifying sysplex=filesys and sysplex_filesys_sharemode=norwshare
  - Each system selects this specification
- When running with sysplex=filesys, a new mount parameter can be used to specify whether a file system is mounted sysplex-aware:
  - RWSHARE | NORWSHARE
- MOUNT command

```
MOUNT FILESYSTEM('OMVS.PRV.COMPAT.AGGR001')
TYPE(ZFS)
MOUNTPOINT('/etc/mountpt') PARM('RWSHARE') MODE(RO)
```

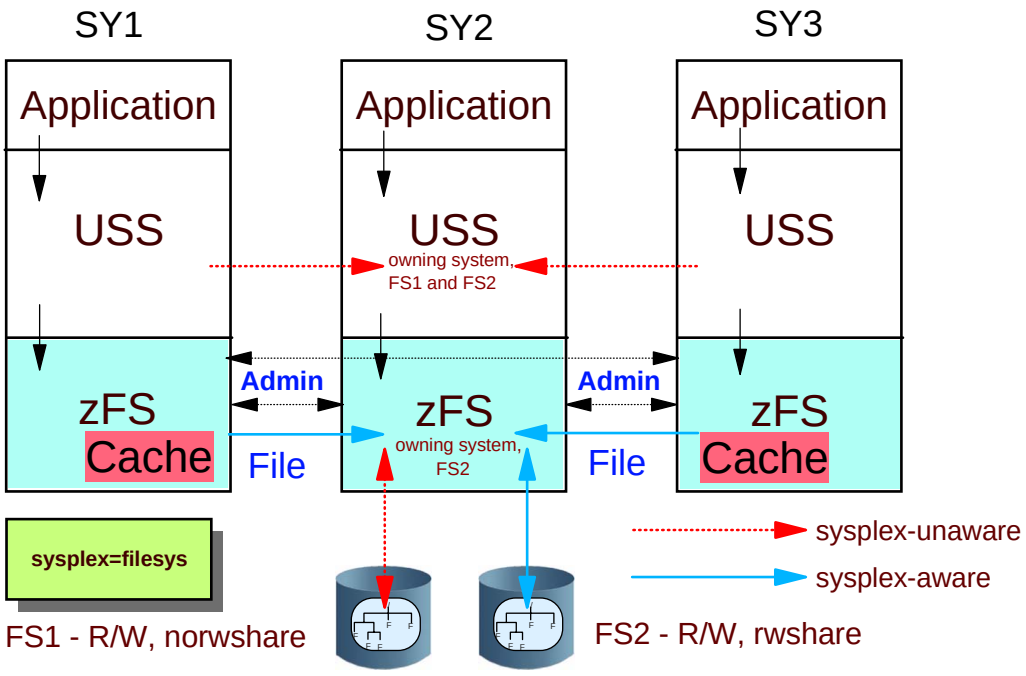
The new sysplex=filesys is documented in a refresh of manual z/OS Distributed File Service zFS Administration, SC24-5989

zFS Commands

```
/* From the OMVS shell ..............
$> zfsadm configquery -sysplex_state
IOEZ00317I The value for configuration option -sysplex_state is 3.
$> zfsadm configquery -syslevel
IOEZ00644I The value for configuration option -syslevel is:
zFS kernel: z/OS    zSeries File System
Version 01.11.00 Service Level OA29619 - HZFS3B0.
Created on Wed Jan 13 09:39:20 EST 2010.
sysplex(filesys,norwshare) interface(3)

/* MVS command.......................
f zfs,query,level
IOEZ00639I zFS kernel: z/OS    zSeries File System
 Version 01.11.00 Service Level OA29619 - HZFS3B0.
 Created on Wed Jan 13 09:39:20 EST 2010.
 sysplex(filesys,norwshare) interface(3)
 IOEZ00025I zFS kernel: MODIFY command - QUERY,LEVEL completed
successfully.
```


zFS Sysplex-aware on a Filesystem Basis



zFS Sysplex-aware on a Filesystem Basis

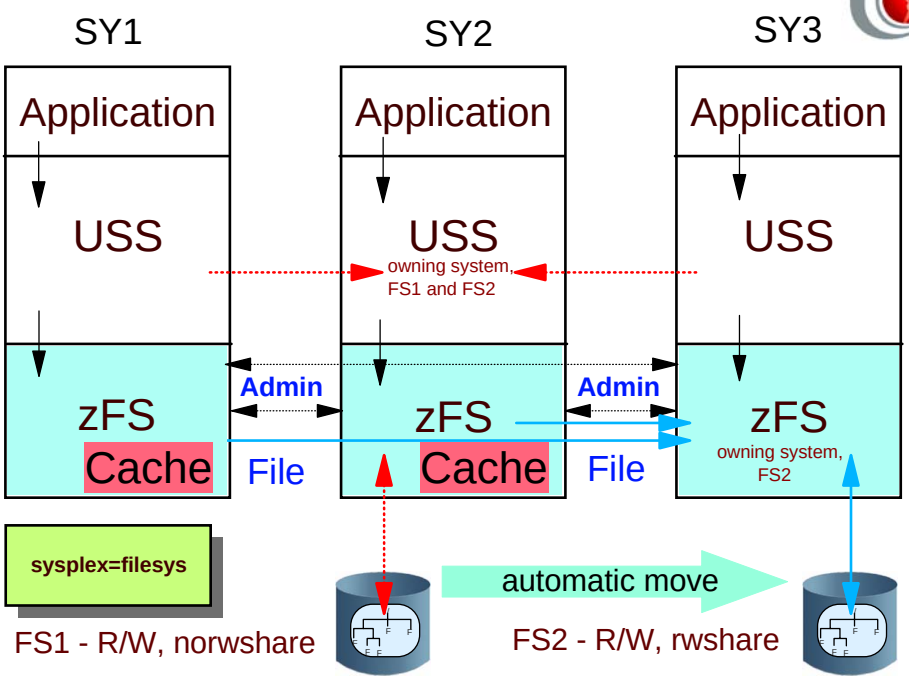
- When you run zFS sysplex-aware on a file system basis on all your members, the zFS PFS initializes as sysplex-aware
  - It can individually determine which file system is sysplex-aware and which is not based on the mount parameters rwshare and norwshare
- MOUNT commands for filesystems

```
For file system FS2
MOUNT FILESYSTEM('OMVS.PRV2.COMPAT.AGGR001') TYPE(ZFS)
MODE(RDWR) MOUNTPOINT('/usr/mountpt2') PARM('RWSHARE')
For file system FS1
MOUNT FILESYSTEM('OMVS.PRV1.COMPAT.AGGR001') TYPE(ZFS)
MODE(RDWR) MOUNTPOINT('/usr/mountpt1') PARM('NORWSHARE')
```


Automatic Movement of Filesystems

- Beginning with z/OS V1R11, as a part of supporting read-write mounted file systems that are accessed as sysplex-aware
  - zFS automatically moves zFS ownership of a zFS filesystem to the system that has more read-write activity coming from it compared to the total amount from all systems
  - zFS can move its ownership of zFS read-write sysplex-aware file systems dynamically based on system usage

Automatic Movement of Filesystems



zFS and Partial Release

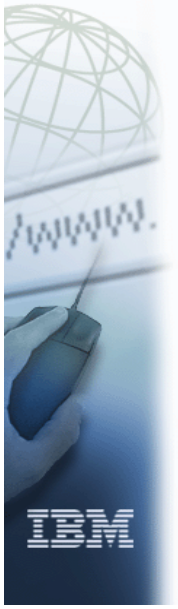
- Partial release is used to release unused space from the end of an extended format data set
  - Specify in a management class or JCL RLSE subparameter
- Data sets whose ending high used RBA is in track managed space, all space after the high used RBA is released on a CA boundary up to the high allocated RBA
  - If the high used RBA is not on a CA boundary, the high used amount is rounded to the next CA boundary
- Data sets whose ending high used RBA, in cylinder managed space, all space after the high used RBA is released on an MCU boundary up to high allocated RBA
  - If the high used RBA is not on an MCU boundary, the high used amount is rounded to the next MCU boundary

Partial Release Restrctions

- Partial release processing is supported only for extended format data sets
- Only the data component of the VSAM cluster is eligible for partial release
- Alternate indexes opened for path or upgrade processing are not eligible for partial release - The data component of an alternate index when opened as cluster could be eligible for partial release
- Partial release processing is not supported for temporary close
- Partial release processing is not supported for data sets defined with guaranteed space

ibm.com

e-business



z/OS UNIX Enhancements

Redbooks

International Technical Support Organization

Health Checker Exploitation of BPX.SUPERUSER

- Before z/OS V1R12, superuser authority was required via a user profile with an OMVS UID being set to 0
- With z/OS V1R12, you can associate the Health Checker address space with the following definitions:
  - READ access to the BPX.SUPERUSER resource in the FACILITY class and a non-zero UID
  - This is available only in z/OS V1R12 and future releases

```
ADDUSER hcsuperid OMVS(UID(non-zero) HOME('/') PROGRAM('/bin/sh')) NOPASSWORD
PERMIT BPX.SUPERUSER CLASS(FACILITY) ID(hcsuperid) ACCESS(READ)
RDEFINE STARTED HZSPROC.* STDATA(USER(hcsuperid)
```


Health Checker USS_HFS_DETECTED

- Beginning with z/OS V1R5, HFS was no longer considered the strategic filesystem, and zFS became the strategic filesystem
  - This new check verifies all file systems mounted and issues an exception message if any HFS file systems are found
- This check runs any time an HFS file system is successfully mounted
  - Unless overridden by the RUN_ON_MOUNT=NO parameter
- The check will also run any time a F BPXOINIT,FILESYS=REINIT command is issued

Health Checker USS_HFS_DETECTED

- This check is a quick way to identify which HFS filesystems are mounted on the system and a reminder to migrate to zFS
  - This check is run with setting “Low Severity” once a day
  - The check is scheduled to run every 24 hours
  - The check has two parameters
- **RUN_ON_MOUNT** - This indicates wether the check should run after the successful mount of an HFS file system - You can specify **YES** or **NO**
- **HFS_LIST** - You can specify a list of HFS file systems that you wish to ignore for this check - A filesystem specified for this parameter will not cause the check to issue an exception message

Check with RUN_ON_MOUNT=YES

```
#> /usr/sbin/mount -f hering.testmnt.hfs test
#> df -v test | head -l3 | tail -1
HFS, Read/Write, Device:197, ACLS=Y
```

- Response in SYSLOG to the mount

```
HZS0001I CHECK(IBMUSS,USS_HFS_DETECTED): 235
BPXH068E One or more HFS file systems mounted.
```

- Part of the output in the SDSF health check display

```
BPXH069I The following HFS file systems were found:
-------------------------------------------------
HERING.TESTMNT.HFS
OMVS.SC74.VAR
OMVS.SC74.ETC
OMVS.DB2V8.UK05586.HFS
```

Directory Caching Display Tool

- This z/OS UNIX tool displays whether z/OS UNIX directory caching is active
  - It is important for helping to resolve performance issues when z/OS UNIX directory caching is lost due to file system movement
  - This normally only happens when the shared file system environment is mixed
- You can get the tool from:
  - ftp://ftp.software.ibm.com/s390/zos/tools/wjsip/wjsipndc.txt

    Note: Be sure to convert it from ISO8859-1 to IBM-1047 or IBM-037 when placing the file to your z/OS system

File System Monitoring Tool (FSMON)

- This wjsfsmon utility is mentioned in APAR OA29619 and can help you to determine which systems are accessing your zFS read-write file systems and whether your zFS read-write file systems are accessed from multiple systems
- See the z/OS UNIX Tools and Toys web site - to download the tool and its documentation:
  - http://www.ibm.com/servers/eserver/zseries/zos/unix/tools
- You can also directly go to the UNIX Tools site:
  - http://www-03.ibm.com/systems/z/os/zos/features/unix/bpxa1ty2.html
- Retrieve it from:
  - ftp://ftp.software.ibm.com/s390/zos/tools/wjsfsmon/wjsfsmon.txt