```
*****************************************************
***** ZFS : step-by-step by Peerapong.K@Sun.COM *****
*****************************************************

(1) Show physical disk or file disk type for ZFS

#ls -la /zfsdisk/*
-rw------T   1 root     root     104857600 May 17 14:39 /zfsdisk/disk1
-rw------T   1 root     root     104857600 May 17 14:39 /zfsdisk/disk2
-rw------T   1 root     root     104857600 May 17 14:42 /zfsdisk/disk3
-rw------T   1 root     root     104857600 May 17 14:23 /zfsdisk/disk4
-rw------T   1 root     root     157286400 May 17 14:37 /zfsdisk/disk5
-rw------T   1 root     root     157286400 May 17 14:42 /zfsdisk/disk6

NOTIECE : disk1-4 have 100MB in size, disk5-6 have 150MB.

(2) Initial creation of pool (mypool) -- zpool create
# zpool create mypool mirror /zfsdisk/disk1 /zfsdisk/disk2
# zpool list
NAME                    SIZE    USED   AVAIL    CAP  HEALTH     ALTROOT
mypool                 95.5M   52.5K   95.4M     0%  ONLINE     -

# zpool status
  pool: mypool
 state: ONLINE
 scrub: none requested
config:

        NAME                STATE     READ WRITE CKSUM
        mypool              ONLINE       0     0     0
          mirror            ONLINE       0     0     0
            /zfsdisk/disk1  ONLINE       0     0     0
            /zfsdisk/disk2  ONLINE       0     0     0

errors: No known data errors

(3) Initial creation of file systems (myfs) -- zfs create
# zfs create mypool/myfs
# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                 104K  63.4M  24.5K  /mypool
mypool/myfs           24.5K  63.4M  24.5K  /mypool/myfs

(4) Adding extra disks -- zpool add
# zpool add mypool mirror /zfsdisk/disk3 /zfsdisk/disk4

# zpool status
  pool: mypool
 state: ONLINE
 scrub: none requested
config:

        NAME                STATE     READ WRITE CKSUM
        mypool              ONLINE       0     0     0
          mirror            ONLINE       0     0     0
```

```
            /zfsdisk/disk1  ONLINE       0     0     0
            /zfsdisk/disk2  ONLINE       0     0     0
          mirror            ONLINE       0     0     0
            /zfsdisk/disk3  ONLINE       0     0     0
            /zfsdisk/disk4  ONLINE       0     0     0

errors: No known data errors

#zpool list
NAME                    SIZE    USED   AVAIL    CAP  HEALTH     ALTROOT
mypool                  191M    214K    191M     0%  ONLINE     -

# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                 106K   159M  25.5K  /mypool
mypool/myfs           24.5K   159M  24.5K  /mypool/myfs

*** Avail size from the 'zpool list' and 'zfs list' commands may vary slightly
*** as the 'zfs list' command accounts for a small amount of space reserved for
*** the file system level operations that is NOT visible from the 'zpool list'
*** command

(5) Volumes
# zfs create -V 50m mypool/myvol
# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                50.1M   109M  25.5K  /mypool
mypool/myfs           24.5K   109M  24.5K  /mypool/myfs
mypool/myvol          22.5K   159M  22.5K  -

*** no mount point listed for myvol, Volumes are NOT mountable
*** (need to make it UFS to be able to mount)
*** ZFS has guaranteed that there will be 50MB available -
*** the other file systems available space has been reduced by
*** 50MB and the pool has 50MB used

# newfs /dev/zvol/rdsk/mypool/myvol
newfs: construct a new file system /dev/zvol/rdsk/mypool/myvol: (y/n)? y
Warning: 2048 sector(s) in last cylinder unallocated
/dev/zvol/rdsk/mypool/myvol:    102400 sectors in 17 cylinders of 48 tracks, 128
sectors
        50.0MB in 2 cyl groups (14 c/g, 42.00MB/g, 20160 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
 32, 86176,

# mkdir /data
# mount /dev/zvol/dsk/mypool/myvol /data
# df -k
Filesystem            kbytes    used   avail capacity  Mounted on
/dev/dsk/c0d0s0      14435859 3730629 10560872    27%    /
/devices                   0       0       0     0%    /devices
ctfs                       0       0       0     0%    /system/contract
proc                       0       0       0     0%    /proc
mnttab                     0       0       0     0%    /etc/mnttab
swap                 2734616     700 2733916     1%    /etc/svc/volatile
```

```
objfs                       0       0       0     0%    /system/object
/usr/lib/libc/libc_hwcap1.so.1
                     14435859 3730629 10560872    27%    /lib/libc.so.1
fd                          0       0       0     0%    /dev/fd
swap                  2733972      56 2733916     1%    /tmp
swap                  2733940      24 2733916     1%    /var/run
/vol/dev/dsk/c1t0d0/sol_10_606_x86
                       362602  362602       0   100%    /cdrom/sol_10_606_x86
mypool                 162816      25  111472     1%    /mypool
mypool/myfs            162816      24  111472     1%    /mypool/myfs
/dev/zvol/dsk/mypool/myvol
                        46111    1041   40459     3%    /data

*** myvol has used aprox 5MB.

(6) Destroying myvol
# zfs destroy mypool/myvol

(7) Additional file system
# zpool create mypool/myfs2

(8) Reservations
(to guaratee that a file system has a certain level of capacity available to it)

*** do this AFTER you have created multiple file systems but
*** BEFORE demo the quota section

# zfs set reservation=155M mypool/myfs
# zfs get reservation mypool/myfs
NAME             PROPERTY       VALUE                      SOURCE
mypool/myfs      reservation    155M                       local

# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                 155M  3.86M  27.5K  /mypool
mypool/myfs           24.5K   159M  24.5K  /mypool/myfs
mypool/myfs2          24.5K  3.86M  24.5K  /mypool/myfs2

***
*** copy data over with in the limits of the space available
***

# df -k
Filesystem            kbytes    used   avail capacity  Mounted on
/dev/dsk/c0d0s0      14435859 3729421 10562080    27%    /
. . . . .
. . . . .
mypool                 162816      27    3950     1%    /mypool
mypool/myfs            162816      24  162645     1%    /mypool/myfs
mypool/myfs2           162816      24    3950     1%    /mypool/myfs2

# ls -la /kernel/genunix
-rwxr-xr-x   1 root     sys      2235280 Apr 30 05:10 /kernel/genunix

# cp /kernel/genunix /mypool/myfs2/genunix1
```

```
# zfs list
NAME                    USED  AVAIL  REFER  MOUNTPOINT
mypool                  157M  1.60M  27.5K  /mypool
mypool/myfs            24.5K   157M  24.5K  /mypool/myfs
mypool/myfs2           2.28M  1.60M  2.28M  /mypool/myfs2

# df -k
. . . . .
mypool                 162816      27    1640     2%    /mypool
mypool/myfs            162816      24  160336     1%    /mypool/myfs
mypool/myfs2           162816    2331    1640    59%    /mypool/myfs2

# cp /kernel/genunix /mypool/myfs2/genunix2
cp: /mypool/myfs2/genunix2: No space left on device

*** ZFS actually rolls back the data that was copied cover so you will NOT
*** hit the full file system situation.

(9) Reservation -- unset
# zfs set reservation=none mypool/myfs

(10) Quotas -- maximum limit on the size a file system can be used
# zfs set quota=2m mypool/myfs2
# zfs get quota mypool/myfs2
NAME             PROPERTY       VALUE                      SOURCE
mypool/myfs2     quota          2M                         local

# zfs list
NAME                    USED  AVAIL  REFER  MOUNTPOINT
mypool                  174K   159M  27.5K  /mypool
mypool/myfs            24.5K   159M  24.5K  /mypool/myfs
mypool/myfs2           24.5K  1.98M  24.5K  /mypool/myfs2

# cp /kernel/genunix /mypool/myfs2/genunix1
cp: /kernel/genunix: Disc quota exceeded

*** Like the 'Reservation', ZFS undoes the effects of the copy as it did not
*** complete

(11) Auto NFS Sharing
# share
< no output from the share command>

# zfs set sharenfs=on mypool/myfs
# zfs get sharenfs mypool/myfs
NAME             PROPERTY       VALUE                      SOURCE
mypool/myfs      sharenfs       on                         local

# share
-               /mypool/myfs   rw   ""

----Unshare----
# zfs set sharenfs=off mypool/myfs
# zfs get sharenfs mypool/myfs
```

```
NAME             PROPERTY       VALUE                      SOURCE
mypool/myfs      sharenfs       off                        local
#
# share
< no output from the share command>

(12) Data Recovery -- Shapshots (creating Write-protected image)

----copy data to myfs----
# cp /usr/dict/words /mypool/myfs
# cp /etc/passwd /mypool/myfs
# ls -la /mypool/myfs
total 526
drwxr-xr-x   2 root     sys            4 May 17 16:07 .
drwxr-xr-x   4 root     sys            4 May 17 15:38 ..
-rw-r--r--   1 root     root         671 May 17 16:07 passwd
-r--r--r--   1 root     root      206663 May 17 16:07 words

----take a snapshot of myfs----
# zfs snapshot mypool/myfs@first
# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                 443K   159M  27.5K  /mypool
mypool/myfs            284K   159M   284K  /mypool/myfs
mypool/myfs@first         0      -   284K  -
mypool/myfs2          24.5K  1.98M  24.5K  /mypool/myfs2

----take a snapshot of myvol----
# zfs snapshot mypool/myvol@backup
# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                50.1M  13.4M  25.5K  /mypool
mypool/myfs           24.5K  13.4M  24.5K  /mypool/myfs
mypool/myvol          22.5K  63.4M  22.5K  -
mypool/myvol@backup       0      -  22.5K  -

----verify snapshot----
ls -la /mypool/myfs/.zfs/snapshot/first
total 523
drwxr-xr-x   2 root     sys            4 May 17 16:07 .
dr-xr-xr-x   3 root     root           3 May 17 14:55 ..
-rw-r--r--   1 root     root         671 May 17 16:07 passwd
-r--r--r--   1 root     root      206663 May 17 16:07 words

***    if we created mypool/myvol (zfs create -V 50m mypool/myvol), then we
*** made it UFS file system (newfs /dev/zvol/rdsk/mypool/myvol), then we
*** mounted this /dev/zvol/dsk/mypool/myvol to /data
*** we can do the snapshot for this volume as well, by using this command
*** zfs snapshot mypool/myvol@first, the /dev/zvol/dsk/mypool@first can be
*** mounted with RO (read-only)
***    if we snapshot ZFS using 'zfs snapshot mypool/myfs@backup', for example;
*** the snapshot will be located in '/mypool/myfs/.zfs/snapshot/backup'

# cd /mypool/myfs
# mkfile -n 5m junk
```

```
# ls -la
total 527
drwxr-xr-x   2 root     sys            5 May 17 16:14 .
drwxr-xr-x   4 root     sys            4 May 17 15:38 ..
-rw------T   1 root     root     5242880 May 17 16:14 junk
-rw-r--r--   1 root     root         671 May 17 16:07 passwd
-r--r--r--   1 root     root      206663 May 17 16:07 words

# df -k
Filesystem            kbytes    used   avail capacity  Mounted on
. . . . .
mypool                162816      27  162210     1%    /mypool
mypool/myfs           162816     414  162210     1%    /mypool/myfs
mypool/myfs2            2048      24    2023     2%    /mypool/myfs2

# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                 606K   158M  27.5K  /mypool
mypool/myfs            438K   158M   414K  /mypool/myfs
mypool/myfs@first     23.5K      -   284K  -
mypool/myfs2          24.5K  1.98M  24.5K  /mypool/myfs2
# rm junk

# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                 476K   159M  27.5K  /mypool
mypool/myfs            308K   159M   284K  /mypool/myfs
mypool/myfs@first     23.5K      -   284K  -
mypool/myfs2          24.5K  1.98M  24.5K  /mypool/myfs2

*** now mypool/myfs && mypool/myfs@first has the same REFER side

*** now some lines from /mypool/myfs/words, then save
# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                 608K   158M  27.5K  /mypool
mypool/myfs            440K   158M   156K  /mypool/myfs
mypool/myfs@first      284K      -   284K  -
mypool/myfs2          24.5K  1.98M  24.5K  /mypool/myfs2

*** now REFER side of mypool/myfs && mypool/myfs@first are NOT the same

#diff /mypool/myfs/words /mypool/myfs/.zfs/snapshot/first/words
<diff does some output>

---Data comparision---
# digest -a md5 /mypool/myfs/words
34a2d6e3c4851ea9a56fc5ace4ef7380 (CORRUPTED!!!)
#
# digest -a md5 /mypool/myfs/.zfs/snapshot/first/words
5dc66244a7bef7d3018538e144e4bbdc

---Roll back----
# zfs rollback mypool/myfs@first
# digest -a md5 /mypool/myfs/words
```

```
5dc66244a7bef7d3018538e144e4bbdc
# digest -a md5 /mypool/myfs/.zfs/snapshot/first/words
5dc66244a7bef7d3018538e144e4bbdc
# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                 444K   159M  27.5K  /mypool
mypool/myfs            284K   159M   284K  /mypool/myfs
mypool/myfs@first         0      -   284K  -
mypool/myfs2          24.5K  1.98M  24.5K  /mypool/myfs2

# diff /mypool/myfs/words /mypool/myfs/.zfs/snapshot/first/words
< no output here >

----Destroying the snapshot----
## zfs destroy mypool/myfs
cannot destroy 'mypool/myfs': filesystem has children
use '-r' to destroy the following datasets:
mypool/myfs@first
# zfs destroy mypool/myfs@first

(13) Import/Export
# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                 444K   159M  27.5K  /mypool
mypool/myfs            284K   159M   284K  /mypool/myfs
mypool/myfs2          24.5K  1.98M  24.5K  /mypool/myfs2
# zpool list
NAME                    SIZE    USED   AVAIL    CAP  HEALTH     ALTROOT
mypool                  191M    458K    191M     0%  ONLINE     -

# zpool export -f mypool

# zpool list
no pools available
# zfs list
no datasets available

(14) Backup

(15) Device Replacement
In case, we pretend that /zfsdisk/disk4 was corrupted, we will replace it with
/zfsdisk/disk6, by using 'zpool replace' command

# zpool status
  pool: mypool
 state: ONLINE
 scrub: none requested
config:

        NAME                STATE     READ WRITE CKSUM
        mypool              ONLINE       0     0     0
          raidz             ONLINE       0     0     0
            /zfsdisk/disk1  ONLINE       0     0     0
            /zfsdisk/disk2  ONLINE       0     0     0
            /zfsdisk/disk3  ONLINE       0     0     0
```

```
            /zfsdisk/disk4  ONLINE       0     0     0
            /zfsdisk/disk5  ONLINE       0     0     0

errors: No known data errors

# zpool replace mypool /zfsdisk/disk4 /zfsdisk/disk6

# zpool status
  pool: mypool
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function, possibly in a degraded state.
action: Wait for the resilver to complete.
 scrub: resilver in progress, 0.39% done, 0h12m to go
config:

        NAME                  STATE     READ WRITE CKSUM
        mypool                DEGRADED     0     0     0
          raidz               DEGRADED     0     0     0
            /zfsdisk/disk1    ONLINE       0     0     0
            /zfsdisk/disk2    ONLINE       0     0     0
            /zfsdisk/disk3    ONLINE       0     0     0
            replacing         DEGRADED     0     0     0
              /zfsdisk/disk4  UNAVAIL      0     0     0  corrupted data
              /zfsdisk/disk6  ONLINE       0     0     0
            /zfsdisk/disk5    ONLINE       0     0     0

errors: No known data errors

***
*** Only the actual data in use in file systems, snapshots, zvols etc will be
*** resilvered, not the entire file systems, snapshots, zvols
***



***
*** NOTE
***
*** the /zfsdisk/disk4 size is 100MB, the /zfsdisk/disk6 size is 150MB
*** the drive disk6 can replace disk4, but disk4 can not replace disk6
*** For example,
** # zpool replace mypool /zfsdisk/disk6 /zfsdisk/disk4
**cannot replace /zfsdisk/disk6 with /zfsdisk/disk4: /zfsdisk/disk4 is too small

(16) Scrubbing the pool -- to check the validity of the specified ZFS checksums
# zpool status
  pool: mypool
 state: ONLINE
 scrub: resilver completed with 0 errors on Thu May 18 14:28:38 2006
config:

        NAME                STATE     READ WRITE CKSUM
        mypool              ONLINE       0     0     0
          raidz             ONLINE       0     0     0
            /zfsdisk/disk1  ONLINE       0     0     0
```

```
            /zfsdisk/disk2  ONLINE       0     0     0
            /zfsdisk/disk3  ONLINE       0     0     0
            /zfsdisk/disk6  ONLINE       0     0     0
            /zfsdisk/disk5  ONLINE       0     0     0

errors: No known data errors
# zpool scrub mypool
# zpool status
  pool: mypool
 state: ONLINE
 scrub: scrub in progress, 18.13% done, 0h0m to go
config:

        NAME                STATE     READ WRITE CKSUM
        mypool              ONLINE       0     0     0
          raidz             ONLINE       0     0     0
            /zfsdisk/disk1  ONLINE       0     0     0
            /zfsdisk/disk2  ONLINE       0     0     0
            /zfsdisk/disk3  ONLINE       0     0     0
            /zfsdisk/disk6  ONLINE       0     0     0
            /zfsdisk/disk5  ONLINE       0     0     0

errors: No known data errors

***
*** Scrubing is done 18.13%
***

# zpool status
  pool: mypool
 state: ONLINE
 scrub: scrub completed with 0 errors on Thu May 18 14:39:15 2006
config:

        NAME                STATE     READ WRITE CKSUM
        mypool              ONLINE       0     0     0
          raidz             ONLINE       0     0     0
            /zfsdisk/disk1  ONLINE       0     0     0
            /zfsdisk/disk2  ONLINE       0     0     0
            /zfsdisk/disk3  ONLINE       0     0     0
            /zfsdisk/disk6  ONLINE       0     0     0
            /zfsdisk/disk5  ONLINE       0     0     0

errors: No known data errors

***
*** Scrubing is done with 0 errors
***

(17) ZFS Compression

*** This will compress only ZFS file system created by
*** created by 'zfs create mypool/myfs', for example
***
*** this will NOT work with UFS volume created
```

```
*** by 'zfs create -V 50M mypool/myvol',then 'newfs /dev/zvol/rdsk/mypool/myvol'
***

# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                 131M   313M    51K  /mypool
mypool/myfs             49K   313M    49K  /mypool/myfs
mypool/myvol           131M   313M   113M  -
mypool/myvol@backup   18.3M      -   126M  -
# cd /mypool/myfs
# zfs get compression mypool/myfs
NAME             PROPERTY       VALUE                      SOURCE
mypool/myfs      compression    off                        local
# zfs get compressratio mypool/myfs
NAME             PROPERTY       VALUE                      SOURCE
mypool/myfs      compressratio  1.00x                      -
# cp /kernel/genunix /mypool/myfs
# du -k /mypool/myfs/genunix
2884    /mypool/myfs/genunix

---- Set compression to ON ----
# zfs set compression=on mypool/myfs
# zfs get compression mypool/myfs
NAME             PROPERTY       VALUE                      SOURCE
mypool/myfs      compression    on                         local
# zfs get compressratio mypool/myfs
NAME             PROPERTY       VALUE                      SOURCE
mypool/myfs      compressratio  1.00x                      -

***
*** Enabling compression will ONLY effect data written after this point,
*** it is not applied retrospectively
***

# cp /kernel/genunix /mypool/myfs/genunix_compressed
# zfs get compressratio mypool/myfs
NAME             PROPERTY       VALUE                      SOURCE
mypool/myfs      compressratio  1.21x                      -

# du -k /mypool/myfs/gen*
2884    /mypool/myfs/genunix
1865    /mypool/myfs/genunix_compressed

*** if later we set the ZFS compression to OFF, the already compressed files
*** are still compressed, but the later written files to this /mypool/myfs,
*** will not be further compressed...

(18) Data Recovery -- Clones (creating of writable image)
# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                 113M   331M    51K  /mypool
mypool/myfs             49K   331M    49K  /mypool/myfs
mypool/myvol           113M   331M   113M  -

*** mypool/myfs used only 49KB
```

```
# for i in 1 2 3 4 5 6 7 8 9
> do
> cp /kernel/genunix /mypool/myfs/genunix-$i
> done
# ls -la /mypool/myfs/gen*
-rwxr-xr-x   1 root     root     2235280 May 18 15:14 /mypool/myfs/genunix-1
-rwxr-xr-x   1 root     root     2235280 May 18 15:14 /mypool/myfs/genunix-2
-rwxr-xr-x   1 root     root     2235280 May 18 15:14 /mypool/myfs/genunix-3
-rwxr-xr-x   1 root     root     2235280 May 18 15:14 /mypool/myfs/genunix-4
-rwxr-xr-x   1 root     root     2235280 May 18 15:14 /mypool/myfs/genunix-5
-rwxr-xr-x   1 root     root     2235280 May 18 15:14 /mypool/myfs/genunix-6
-rwxr-xr-x   1 root     root     2235280 May 18 15:14 /mypool/myfs/genunix-7
-rwxr-xr-x   1 root     root     2235280 May 18 15:14 /mypool/myfs/genunix-8
-rwxr-xr-x   1 root     root     2235280 May 18 15:14 /mypool/myfs/genunix-9
# digest -a md5 /mypool/myfs/gen*
(/mypool/myfs/genunix-1) = 6a594ec25150b6b84b4313c9f111bad2
(/mypool/myfs/genunix-2) = 6a594ec25150b6b84b4313c9f111bad2
(/mypool/myfs/genunix-3) = 6a594ec25150b6b84b4313c9f111bad2
(/mypool/myfs/genunix-4) = 6a594ec25150b6b84b4313c9f111bad2
(/mypool/myfs/genunix-5) = 6a594ec25150b6b84b4313c9f111bad2
(/mypool/myfs/genunix-6) = 6a594ec25150b6b84b4313c9f111bad2
(/mypool/myfs/genunix-7) = 6a594ec25150b6b84b4313c9f111bad2
(/mypool/myfs/genunix-8) = 6a594ec25150b6b84b4313c9f111bad2
(/mypool/myfs/genunix-9) = 6a594ec25150b6b84b4313c9f111bad2
#
# zfs snapshot mypool/myfs@forclone
# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                 138M   306M    51K  /mypool
mypool/myfs           25.4M   306M  25.4M  /mypool/myfs
mypool/myfs@forclone      0      -  25.4M  -
mypool/myvol           113M   306M   113M  -

*** mypool/myfs now used 25.4MB, mypool@myfs@forclone also REFERS to 25.4MB
*** but occupied 0MB.

---- Creating 2 clone images named mypool/clone1 & mypool/clone2 ----

# zfs clone mypool/myfs@forclone mypool/clone1
# zfs clone mypool/myfs@forclone mypool/clone2
# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                 138M   306M    53K  /mypool
mypool/clone1             0   306M  25.4M  /mypool/clone1
mypool/clone2             0   306M  25.4M  /mypool/clone2
mypool/myfs           25.4M   306M  25.4M  /mypool/myfs
mypool/myfs@forclone      0      -  25.4M  -
mypool/myvol           113M   306M   113M  -

# cd /mypool
# du -k
25963   ./clone2
25963   ./myfs
25963   ./clone1
```

```
77891   .

---- Make change in the original and in the first clone (clone1) ---
# mkfile 1m /mypool/myfs/clonetest
# mkfile 2m /mypool/clone1/clonetest
# zfs list
NAME                    USED  AVAIL  REFER  MOUNTPOINT
mypool                  142M   302M    55K  /mypool
mypool/clone1          2.55M   302M  27.9M  /mypool/clone1
mypool/clone2             0   302M  25.4M  /mypool/clone2
mypool/myfs            26.7M   302M  26.7M  /mypool/myfs
mypool/myfs@forclone     49K      -  25.4M  -
mypool/myvol            113M   302M   113M  -

*** mypool/myfs used up more space (from 25.4MB to 26.7MB)
*** mypool/clone1 used up more space (from 0MB to 2.55MB)

---- Recursively removing the snapshot and the clones that use it ---
---- in this case, will recursively delete mypool/myfs@forclone,
---- mypool/clone1 and mypool/clone2, since clone1 & clone2 is using
---- mypool/myfs@forclone
----
---- @ will be used with snapshot
----
# zfs list
NAME                    USED  AVAIL  REFER  MOUNTPOINT
mypool                  142M   302M    55K  /mypool
mypool/clone1          2.55M   302M  27.9M  /mypool/clone1
mypool/clone2             0   302M  25.4M  /mypool/clone2
mypool/myfs            26.7M   302M  26.7M  /mypool/myfs
mypool/myfs@forclone     49K      -  25.4M  -
mypool/myvol            113M   302M   113M  -
# zfs destroy -R mypool/myfs@forclone
# zfs list
NAME                    USED  AVAIL  REFER  MOUNTPOINT
mypool                  140M   304M    51K  /mypool
mypool/myfs            26.7M   304M  26.7M  /mypool/myfs
mypool/myvol            113M   304M   113M  -

(19) Property inheritance
to inherit properties to multiple file systems at the same time

# zfs create mypool/homedirs
# zfs create mypool/homedirs/user1
# zfs create mypool/homedirs/user2
# zfs create mypool/homedirs/user3
# zfs list
NAME                    USED  AVAIL  REFER  MOUNTPOINT
mypool                  140M   304M    53K  /mypool
mypool/homedirs         200K   304M    53K  /mypool/homedirs
mypool/homedirs/user1    49K   304M    49K  /mypool/homedirs/user1
mypool/homedirs/user2    49K   304M    49K  /mypool/homedirs/user2
mypool/homedirs/user3    49K   304M    49K  /mypool/homedirs/user3
mypool/myfs            26.7M   304M  26.7M  /mypool/myfs
mypool/myvol            113M   304M   113M  -
```

```
# zfs get compression mypool/homedirs
NAME             PROPERTY       VALUE                      SOURCE
mypool/homedirs  compression    off                        default
# zfs get compression mypool/homedirs/user1
NAME             PROPERTY       VALUE                      SOURCE
mypool/homedirs/user1  compression    off                       default
#
# zfs set compression=on mypool/homedirs/user1
# zfs get compression mypool/homedirs
NAME             PROPERTY       VALUE                      SOURCE
mypool/homedirs  compression    off                        default
# zfs get compression mypool/homedirs/user1
NAME             PROPERTY       VALUE                      SOURCE
mypool/homedirs/user1  compression    on                        local
# zfs set compression=on mypool/homedirs
# zfs get compression mypool/homedirs/user1 mypool/homedirs/user2 mypool/homedirs/
user3NAME             PROPERTY       VALUE                     SOURCE
mypool/homedirs/user1  compression    on                         local
mypool/homedirs/user2  compression    on                         inherited from
mypool/homedirs
mypool/homedirs/user3  compression    on                         inherited from
mypool/homedirs

*** mypool/homedirs/user2 & mypool/homedirs/user3 inherited compression=on
*** from mypool/homedirs, but mypool/homedirs/user1 was set compression=on
*** independently (stated local)

---- set mypool/homedirs/user1 to inherit compression property from its
---- upper mypool/homedirs file system

# zfs inherit compression mypool/homedirs/user1
#  zfs get compression mypool/homedirs/user1 mypool/homedirs/user2 mypool/homedirs/
user3
NAME             PROPERTY       VALUE                      SOURCE
mypool/homedirs/user1  compression    on                         inherited from
mypool/homedirs
mypool/homedirs/user2  compression    on                         inherited from
mypool/homedirs
mypool/homedirs/user3  compression    on                         inherited from
mypool/homedirs

---- check whether which properties are inheritable ----

# zfs get (see INHERIT)
. . . . . .
. . . . . .
        PROPERTY       EDIT  INHERIT   VALUES

        type             NO       NO   filesystem | volume | snapshot
        creation         NO       NO   <date>
        used             NO       NO   <size>
        available        NO       NO   <size>
        referenced       NO       NO   <size>
        compressratio    NO       NO   <1.00x or higher if compressed>
```

```
        mounted          NO       NO   yes | no | -
        origin           NO       NO   <snapshot>
        quota           YES       NO   <size> | none
        reservation     YES       NO   <size> | none
        volsize         YES       NO   <size>
        volblocksize     NO       NO   512 to 128k, power of 2
        recordsize      YES      YES   512 to 128k, power of 2
        mountpoint      YES      YES   <path> | legacy | none
        sharenfs        YES      YES   on | off | share(1M) options
        checksum        YES      YES   on | off | fletcher2 | fletcher4 | sha256
        compression     YES      YES   on | off | lzjb
        atime           YES      YES   on | off
        devices         YES      YES   on | off
        exec            YES      YES   on | off
        setuid          YES      YES   on | off
        readonly        YES      YES   on | off
        zoned           YES      YES   on | off
        snapdir         YES      YES   hidden | visible
        aclmode         YES      YES   discard | groupmask | passthrough
        aclinherit      YES      YES   discard | noallow | secure | passthrough

(20) Clean up -- Destroy file systems, containers, pools
# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                 140M   304M    53K  /mypool
mypool/homedirs        202K   304M    55K  /mypool/homedirs
mypool/homedirs/user1    49K   304M    49K  /mypool/homedirs/user1
mypool/homedirs/user2    49K   304M    49K  /mypool/homedirs/user2
mypool/homedirs/user3    49K   304M    49K  /mypool/homedirs/user3
mypool/myfs            26.7M   304M  26.7M  /mypool/myfs
mypool/myvol            113M   304M   113M  -

# zpool destroy mypool
cannot remove device links for 'mypool/myvol': volume is in use
cannot destroy 'mypool': pool busy
# zpool destroy -f mypool

cannot remove device links for 'mypool/myvol': volume is in use
cannot destroy 'mypool': pool busy

# df -k
Filesystem            kbytes    used   avail capacity  Mounted on
. . . .
. . . .
/dev/zvol/dsk/mypool/myvol
                       94775   11289   74009    14%    /data
# umount /data
# zpool destroy mypool
<no output here>
# zpool list
no pools available
--------------------------------------------------------------------------------
----------------------------END-OF-FILE-----------------------------------------
--------------------------------------------------------------------------------
```